НАЦІОНАЛЬНА АКАДЕМІЯ НАУК УКРАЇНИ

ІНСТИТУТ ТЕЛЕКОМУНІКАЦІЙ І ГЛОБАЛЬНОГО ІНФОРМАЦІЙНОГО ПРОСТОРУ

На правах рукопису

СЕМКО ВІКТОР ВОЛОДИМИРОВИЧ

УДК 681.5(042.3)

МЕТОДОЛОГІЯ ОПТИМАЛЬНОГО УПРАВЛІННЯ ОБ'ЄКТОМ В УМОВАХ КОНФЛІКТУ, ОБМЕЖЕНЬ ТА НЕВИЗНАЧЕНОСТЕЙ

05.13.06 – інформаційні технології

Дисертація на здобуття наукового ступеня
доктора технічних наук

Науковий консультант
Бурячок Володимир Леонідович
доктор технічних наук, професор

Київ – 2017

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ..... 6
ВСТУП..... 9
1. АНАЛІЗ СТАНУ ПРОБЛЕМИ УПРАВЛІННЯ ОБ'ЄКТОМ В УМОВАХ КОНФЛІКТУ..... 27
1.1. Аналіз стану проблеми управління об’єктом при вирішенні конфлікту..... 27
1.2. Аналіз методів та підходів до вирішення задачі управління об'єктом при розв'язанні конфлікту..... 33
1.3. Система інтелектуального управління об’єктом……. 46
1.4. Формулювання наукової проблеми та постановка задач дослідження..... 56
Висновки до першого розділу 65
2. ШЛЯХИ СТВОРЕННЯ СИСТЕМИ ІНТЕЛЕКТУАЛЬНОГО УПРАВЛІННЯ РІШЕННЯМ КОНФЛІКТУ ВЗАЄМОДІЇ ОБ'ЄКТУ УПРАВЛІННЯ З ВАРІАТИВНОЮ МНОЖИНОЮ ОБ’ЄКТІВ СПОСТЕРЕЖЕННЯ В УМОВАХ ОБМЕЖЕНЬ ТА НЕВИЗНАЧЕНОСТЕЙ……..... 68
2.1. Концепція топологічного ситуаційного аналізу та синтезу стратегій управління об'єктом в умовах конфлікту, невизначеності поведінки та варіативної множини об'єктів спостереження…..... 68
2.2. Теоретико-множинна модель взаємодії об’єкту управління з варіативною множиною ОС в умовах конфлікту, обмежень та невизначеностей……. 86
2.3. Семіотична модель взаємодії об’єкту управління з варіативною множиною ОС в просторі спостереження..... 95
2.4. Семантична модель взаємодії об’єкту управління з варіативною множиною ОС в просторі спостереження……... 104

2.5. Метод рішення конфлікту взаємодії ОУ з варіативною множиною ОС в умовах обмежень та невизначеності, як NP-повної перебірної задачі динамічної дискретної оптимізації…… 112
2.6. Модель взаємодії об’єктів кібернетичного простору……. 115
Висновки до другого розділу..... 119
3. ОСОБЛИВОСТІ ФОРМУВАННЯ ПРОСТОРУ РЕШЕНЬ КОНФЛІКТУ ВЗАЄМОДІЇ ОБ'ЄКТУ УПРАВЛІННЯ З ВАРІАТИВНОЮ МНОЖИНОЮ ОБ’ЄКТІВ СПОСТЕРЕЖЕННЯ В УМОВАХ ОБМЕЖЕНЬ ТА НЕВИЗНАЧЕНОСТЕЙ..... 122
3.1. Інформаційна множина простору комбінованого управління об'єктом..... 122
3.2. Простір рішень..... 129
3.3. Інформаційна множина небезпечних станів простору рішення..... 137
3.4. Простір гарантованого управління динамічним об’єктом..... 141
Висновки до третього розділу..... 151
4. ТЕХНОЛОГІЯ СИНТЕЗУ І ВИБОРУ СТРАТЕГІЙ ОПТИМАЛЬНОГО УПРАВЛІННЯ СКЛАДНИМИ ТЕХНІЧНИМИ ТА ТЕХНІЧНИМИ ЕРГАТИЧНИМИ СИСТЕМАМИ ПРИ РОЗВ'ЯЗАННІ КОНФЛІКТУ ЇХ ВЗАЄМОДІЇ З ВАРІАТИВНОЮ МНОЖИНОЮ ОБ’ЄКТІВ СПОСТЕРЕЖЕННЯ В УМОВАХ ОБМЕЖЕНЬ ТА НЕВИЗНАЧЕНОСТЕЙ ……..……..... 154
4.1. Метод інтегрального усікання варіантів..... 154
4.2. Алгоритм методу інтегрального усікання варіантів …….. 176
4.3. Рішення конфлікту в кібернетичному просторі ……. 183
Висновки до четвертого розділу..... 190
5. ПЕРСПЕКТИВИ ЗАСТОСУВАННЯ СИСТЕМИ ІНТЕЛЕКТУАЛЬНОГО УПРАВЛІННЯ ПРИ РІШЕННІ КОНФЛІКТУ ВЗАЄМОДІЇ ОБ’ЄКТУ УПРАВЛІННЯ З ВАРІАТИВНОЮ МНОЖИНОЮ ОБ’ЄКТІВ СПОСТЕРЕЖЕННЯ В

УМОВАХ ОБМЕЖЕНЬ ТА НЕВИЗНАЧЕНОСТЕЙ…….…….. 192
5.1. Удосконалення інформаційних технологій автоматизації процесів інтелектуального управління рішенням конфлікту…….... 192
5.1.1. Розрахунок інформаційної множини простору комбінованого управління об'єктом..... 192
5.1.2. Синтез простору рішень..... 197
5.1.3. Побудова простору гарантованого управління динамічного об’єкта..... 203
5.1.4. Імітаційна модель збурень зовнішнього середовища динамічного об’єкту управління……. 211
5.2. Імітаційні моделі вирішення задачі конфлікту взаємодії об’єктів…… 217
5.2.1. Опис та аналіз стану взаємодії об‘єкту управління з середовищем простору спостереження..... 217
5.2.2. Синтез варіантів припустимих рішень конфлікту за методом інтегрального усікання варіантів..... 226
5.2.3. Евристичний пошук рішення конфлікту..... 232
5.3. Дослідження ефективності алгоритму синтезу рішень за методом інтегрального усікання варіантів…….. 239
5.4. Аналіз характеристик функціонування системи інтелектуального управління об’єктом при рішенні конфлікту…….. 247
Висновки до п’ятого розділу 255
ВИСНОВКИ..... 259
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ..... 263
Додаток А. Акти впровадження результатів дисертаційної роботи…. 284
Додаток Б. Копії свідоцтв про авторське право.....…….... 291
Додаток В. Алгоритми і програмні компоненти розрахунку інформаційних множин опису стану об'єкту управління.....…... 298
Додаток В.1. Алгоритми програмних компонент розрахунку інформаційних множин опису стану об'єкту управління.....…... 298

Додаток В.2. Програмні засоби інформаційної технології розрахунку множин опису стану об'єкту управління..... 307
Додаток Г. Алгоритми і програмні компоненти методу інтегрального усікання варіантів та синтезу стратегій рішень..... 321
Додаток Г.1. Алгоритми програмних компонент методу інтегрального усікання варіантів та синтезу стратегій рішень..... 321
Додаток Г.2. Програмні засоби інформаційної технології методу інтегрального усікання варіантів та синтезу стратегій рішень..... 352
Додаток Д. Алгоритми і програмні компоненти семантичного аналізу і евристичного пошуку..... 384
Додаток Д.1. Алгоритми програмних компонент семантичного аналізу повідомлень системи інтелектуального управління..... 384
Додаток Д.2. Програмні засоби інформаційної технології семантичного повідомлень системи інтелектуального управління..... 396
Додаток Д.3. Програмні засоби інформаційної технології евристичного пошуку..... 418

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ДІ	-	диференційні ігри
ДО	-	динамічний об’єкт
ДС	-	динамічна система
ЕСУ	-	ергатична система управління
ЕП	-	евристичний пошук
ІТС	-	іІнформаційно-телекомунікаційна система
ІМ	-	імітаційна модель
ІМНСПР	-	інформаційна множина небезпечних станів простору рішення
ІП	-	інтелектуальний перетворювач
ІТМ	-	інтегрально-топологічні методи
КА	-	кінцевий автомат
КВ	-	контекстно-вільна
ККП	-	конфліктно-керований процес
КП	-	кібернетичний простір
КС	-	кібернетична система
КОС	-	компоненти обчислювального середовища
КУ	-	комбіноване управління
ЛА	-	лексичний аналізатор
ЛАП	-	літальний апарат
ЛАЧХ	-	логарифмічна амплітудно-частотна характеристика
ЛАФЧХ	-	логарифмічна амплітудно-фазова частотна характеристика
ЛВР	-	лінія відносного руху
ЛО	-	людина-оператор
ЛС	-	лексичний сканер
ЛШ	-	"лінійка швидкостей"
МІУВ	-	метод інтегрального усікання варіантів
МП	-	магазинна пам'ять

НСД - несанкціонований доступ

ОПР - особа, яка приймає рішення

ОС - об'єкт спостереження

ОУ - об'єкт управління

ПЗ - програмні засоби

ПІД - пропорційно-інтегрально-диференційний

ПКУО - простір комбінованого управління об’єктом

ПНП - простір неприпустимих положень

ПР - простір рішень

ПРМ - простір реалізації маневру

ПС - простір спостереження

ПСП - простір спостереження і пошуку

ПСЦР - підсистема синтезу цілі та рішень

СА - синтаксичний аналіз

СБНП - сектор безпечних напрямів переміщення

СЗІ - система захисту інформації

СІУ - система інтелектуального управління

СМ - ситуаційна модель

СМА - семантичний аналізатор

СММ - семіотична модель

СММа - семантичний аналізатор

СННП - сектор небезпечних напрямів переміщення

СО - система обмежень

СР - стабілізуючий регулятор

ССЛУО - синтез стратегій і ланцюжків управління об’єктом

СУ - система управління

ТЕС - технічна ергатична система

ТС - технічна система

ТММ - Теоретико-множинна модель

ФЧХ - фазова частотна характеристика

ШІ - штучний інтелект

ВСТУП

Актуальність теми дослідження. Сучасні системи управління і процеси, що протікають з їх допомогою в певних об'єктах, включаючи кінематичні, динамічні, нелінійні, інформаційні, тощо, структурно та функціонально є складними і багатомірними. Їх прагматична сутність обумовлюється перш за все наявністю взаємозв'язків, правил та відношень, як між власне внутрішніми компонентами, так і з компонентами зовнішнього середовища і описується, як правило, моделями, які відображають специфіку взаємодії цих компонент, елементів та підсистем з зовнішнім і внутрішнім середовищем в умовах невизначеностей, довільних обмежень та конфлікту. Невизначеності обумовлюються при цьому неоднозначністю в доступній інформації про: вектор стану і параметри системи управління (СУ), неконтрольовані завади вимірювання, зовнішні та внутрішні збурення, властивості простору спостереження (ПС), а також наявність об’єктів, які знаходяться в стані конфлікту з об’єктом управління (ОУ). Під конфліктом в цьому сенсі розуміють явище взаємодії по-різному цілеспрямованих сторін – об'єктів, технічних систем (ТС) та технічних ергатичних систем (ТЕС), які або мають складний опис, що не може бути використаним практично, або взагалі не можуть бути повною мірою описаними ні формально, ні вербально.

До проблем, що пов’язані з рішенням конфлікту, відносяться задачі оптимального управління в реальному часі активними об’єктами в умовах обмежень та невизначеностей. Рішення таких проблем пов’язане з необхідністю використання закономірностей, які визначають характеристики і властивості конфліктуючих систем і об’єктів. Характеристики об’єктів в усій області визначення мають складний характер, а самі об’єкти є неоднозначними, характеризуються наявністю статичної та динамічної невизначеності, а також наявністю особливих точок, що в значній мірі ускладнює процес аналізу і синтезу систем управління рішенням конфлікту.

Процес створення методів і алгоритмів керування конфліктуючими об’єктами і системами є складним, а існуючі методи та підходи щодо рішення конфлікту в умовах обмежень та невизначеностей можуть бути застосовані в обмежених випадках.

Новим напрямком прийняття рішень і управління в умовах конфлікту, обмежень та невизначеностей є системи інтелектуального управління (СІУ), які створюються і функціонують на основі штучного інтелекту (ШІ), що дозволяє приймати рішення і формувати гарантовано оцінене управління об’єктом на основі використання методів ШІ.

В даний час наукові дослідження спрямовані на вивчення і побудову СІУ складних, великих і слабо формалізованих ТС і ТЕС, які функціонують в умовах конфлікту. В якості наукових напрямків досліджень нової інформаційної технології можна виділити роботи з СІУ рішенням конфлікту взаємодії об’єктів в умовах обмежень та невизначеностей.

Проблеми управління, що виникають, та формування гарантовано оцінених, науково-обґрунтованих рішень визначаються при цьому шляхом застосування широкого класу лінійних, нелінійних дискретних, неперервних і неперервно-дискретних об’єктів та систем, а також методів та підходів теорії ігор (ТІ), математичного програмування, рішення динамічних задач дискретної оптимізації в класі детермінованих і недетермінованих моделей, ситуаційного управління та ШІ. Зважаючи на те, що зазначений математичний інструментарій, вирішуючи завдання зіставлення різним альтернативним варіантам множини обраних варіантів синтезу і вибору рішень, *по-перше*, не завжди теоретично обґрунтовується в логічних засадах вибору, аксіоматиці, принципах раціональності та узгодження, теоретичних і практичних аспектах дотримання принципів синтезу систем рішення конфлікту, *по-друге*, не завжди сполучається із спеціальними методами синтезу альтернативних стратегій поведінки і вибору рішень – з часом виникла нагальна потреба у розробці нових методів, моделей і процедур, які б дозволили вирішувати проблеми формування та вибору серед низки можливих альтернатив науково-обґрунтованих рішень щодо оптимального управління об’єктом у задачах

конфлікту взаємодії з варіативною множиною об’єктів спостереження (ОС) в умовах невизначеностей і довільних обмежень.

Великий теоретичний і практичний інтерес в цьому сенсі становлять методи автоматизованого і автоматичного синтезу та вибору стратегій поведінки, гарантовано оціненого (гарантованого) управління , спостереження, вивчення та навчання щодо систем вирішення конфліктів в ТС або ТЕС, які реалізовані в дискретному обчислювальному середовищі, і використовуються для забезпечення функцій управління в інформаційних, навігаційних, тренажерних, навчальних та інших системах за умов зовнішнього і/або внутрішнього конфлікту. Їх застосування передбачає, як відомо, наявність управляючого пристрою (УП) СІУ ОУ, а також ОС і середовища, в якому він знаходиться. СІУ ОУ забезпечує при цьому рішення конфліктів в середовищі існування учасників (гравців), що знаходяться в стані постійної зміни активності в середовищі конфліктуючих сторін. Стратегії управління, які синтезуються, обираються як варіанти або альтернативи можливих найкращих рішень з точки зору критерію відбору або функції, що оцінює синтезоване рішення та правила зупинки.

На дослідженні окремих аспектів цих питань зосереджені зусилля багатьох як вітчизняних, так і зарубіжних вчених. Зокрема це Чикрій А.О., Черноусько Ф.Л., Павлов В.В., Кунцевич В.М., Михалевич В.С., Шор Н.З., Понтрягін Л.С., Поспелов Д.А., Поспелов Г.С., Коваленко І.М., Красовський М.М., Скуріхін В.І, Суботін А.І., Растригін Л.О., Бурбаки Н., Касьянов В.О., Ріхтер К. та інші. Тим не менш напрацювання основних положень, які дозволили б сформувати комплексне рішення конфлікту в складних ТС та ТЕС шляхом синтезу їх цілісного цілеспрямованого управління за умов обмежень та невизначеності процесів функціонування цих систем – нині знаходиться на стадії досліджень.

Зважаючи на таке, **проблема**, процедура вирішення якої розглядається в роботі, полягає у ***забезпеченні поліноміального часу рішення задач конфлікту та стійкості функціонування СІУ об’єктами в цілому при їх взаємодії з варіативною множиною об’єктів спостереження в умовах існуючих обмежень та невизначеностей***. Виникненню цієї проблеми сприяє низка протиріч,

обумовлених як обмеженими можливостями традиційних підходів щодо рішення конфліктів в ТС і ТЕС, так і застосуванням існуючих моделей й методів синтезу оптимальних рішень і управління конфліктуючими системами та об’єктами в реальному часі. Проблема може бути вирішена за рахунок розроблення моделей і методів, які використовують упорядковану послідовність процесів синтезу траєкторій переміщення об’єкта в просторі спостереження. Головною вимогою до зазначених методів і моделей є *P*-час обчислень і сталість СІУ при рішенні конфлікту.

Враховуючи викладене, ***тема досліджень***, яка полягає у розробці методології оптимального управління об’єктом в умовах конфлікту, обмежень та невизначеностей, як основи запобігання та вирішення конфлікту взаємодії об’єктів і систем за рахунок застосування методів і підходів, що використовують упорядковану послідовність процесів представлення опису конфлікту та методу синтезу оптимальних рішень за умови гарантованого управління в СІУ ОУ – є ***актуальною***.

Зв'язок роботи з науковими програмами, планами, темами. Дослідження, результати якого викладено в дисертаційній роботі, виконувалося відповідно до державних програм та планів науково-дослідних та дослідно-конструкторських робіт Інституту телекомунікацій і глобального інформаційного простору НАН України, АНТК ім. О.К.Антонова, Державного агентства з питань електронного урядування України, Національного космічного агентства України, Міністерства оборони України, Державної служби зайнятості України; Служби безпеки України, Державної служби спеціального зв’язку та захисту інформації України, АНТК ім. О.К.Антонова: шифр «Модернізація-С» (тема «Модернізація існуючих засобів функціональних центрів, комплексів та систем подвійного призначення, які знаходяться у сфері управління НКАУ, для виконання завдань спеціального інформаційного призначення», номер держреєстрації 0103U005355); шифр «Сейсмологія» (тема «Комплекс робіт з розробки системи технічного захисту інформації для інформаційно-обчислювального центра Головного центру спеціального контролю», номер

держреєстрації 0103U008106); шифр «Моніторинг-С» (тема «Обстеження інформаційно-аналітичної системи НКАУ та розробка рекомендацій щодо забезпечення вимог технічного захисту інформації в ІАС НКАУ», номер держреєстрації 01032U008326); шифр «Мережа» (тема «Створення комп'ютерної мережі на базі технології WI-FI та дослідження питань щодо створення системи захисту інформації», номер держреєстрації 0104U004812); шифр «Аспект» (тема «Розробка бортової системи психофізіологічного моніторингу космонавтів: методичний та інженерно-технологічний аспекти», номер держреєстрації 0199U004488); шифр «TVBase70» (тема «Розробка програмного забезпечення системи реєстрації та обробки відео зображення для рішення задач вимірювання траєкторії руху стренг витяжних парашутних систем в процесі її буксирування за літаком АН-70», номер держреєстрації 0101U007888); шифр «Імунологія» (тема «Розробка нової технології функціонування на регіональному рівні державної системи проведення моніторингу і профілактики імунокерованих інфекційних захворювань – складової системи інформаційно-аналітичного забезпечення та раціонального природокористування загальнодержавних та регіональних органів влади», номер держреєстрації 0106U012520); шифр «Мережа» (тема «Розроблення технічного завдання на комплексну систему захисту інформації мультисервісної мережі Київської міської державної адміністрації», номер держреєстрації 0106U005506); шифр «Зір» (тема «Розробка технології та програмного забезпечення перетворення зображень та ведення графічних баз даних для лікувальних закладів», номер держреєстрації 0106U005505); шифр «Дозвіл» (тема «Розроблення програми для ПЕОМ щодо ведення бази даних дозволів на викиди забруднюючих речовин у атмосферне повітря та надання інформації громадськості», номер держреєстрації 0105U008998); шифр «Сховище» (тема «Розробка системи захисту інформації технології сховища даних», номер держреєстрації 0106U012519); шифр 05/07/99 (тема «Дослідження та пропозиції щодо методології оцінки екологічного ризику об'єктів навколишнього середовища в структурі дизайн-ергономічної діяльності», номер держреєстрації 0104U004812); шифр «Радіус» (тема «Створення концепції та політик технічного захисту

інформації у інформаційно-телекомунікаційних системах, що належать ВАТ "Укртелеком"», номер держреєстрації 0108U000999); EUROPEAID/114372/C/S/UA «Enhancing Border Management: Supply of Border Management. Equipment to the Sumy Border Guard Detachment Ukraine» (2004-2005р.р.); «Programme for theprevention, preparedness, and response to man-made and natural disasters in the ENPI East Region-EuropeAid/129397/C/SER/Multi», PROJECT REFERENCE: EP 105044 (2014р.); шифр «ДЦІР5-1» (тема «Послуги з доопрацювання складових компонентів інформаційно-телекомунікаційної системи електронної взаємодії органів виконавчої влади з придбанням примірнику програмної продукції перевірки/накладання електронного цифрового підпису та примірнику програмної продукції платформи віртуалізації серверів VMWare vSphere (ESX/ESXi) з системою vCenter Server», номер держреєстрації 0115U006734); шифр «НП/2-2015» (тема «Розробка вимог до архітектури Системи електронної взаємодії державних електронних інформаційних ресурсів», номер держреєстрації 0115U007171); шифр «НДР-08/123-2015» (тема «Доопрацювання Єдиного інформаційного веб-ресурсу звернень громадян до органів державної влади та органів місцевого самоврядування в частині ідентифікації особи та інтеграції до системи електронної взаємодії органів виконавчої влади», номер держреєстрації 0115U007172).

Автор дисертаційної роботи брав участь у цих науково-дослідних та дослідно-конструкторських роботах в якості наукового керівника.

Використання наукових результатів дисертаційної роботи автора підтверджуються відповідними актами впровадження, які наведені в додатку дисертаційної роботи.

Роль автора в перерахованих науково-дослідних роботах та проектах полягає в розробці наукових, методологічних та технологічних основ створення та застосування інформаційної технології системи інтелектуального управління при взаємодії об’єкту управління з варіативною множиною кінематичних, динамічних лінійних і нелінійних, а також кібернетичних об’єктів і систем в

умовах конфлікту, обмежень і невизначеностей на основі методології, моделей, методів та гарантовано оцінюванюваного значення параметрів управління.

Мета та завдання дослідження. *Мета дисертаційної роботи* полягає у підвищенні швидкодії і стійкості функціонування систем інтелектуального управління об'єктами в умовах конфлікту, обмежень та невизначеностей при взаємодії з варіативною множиною об'єктів спостереження.

Досягнення поставленої мети вимагає вирішення таких *завдань***:**

1) проведення аналізу та узагальнення наукових і методологічних засад щодо існуючих підходів і методів вирішення конфліктів в ТС та ТЕС;

2) розроблення методу рішення конфлікту взаємодії об'єкта управління з варіативною множиною об'єктів спостереження в умовах обмежень та невизначеностей, як *NP*-повної перебірної задачі динамічної дискретної оптимізації;

3) розроблення теоретико-множинної моделі взаємодії об'єкту управління з варіативною множиною об'єктів спостереження за умов конфлікту, обмежень та невизначеностей;

4) розроблення семіотичної моделі взаємодії об'єкту управління з варіативною множиною ОС в просторі спостереження;

5) розроблення семантичної моделі взаємодії об'єкту управління з варіативною множиною ОС в просторі спостереження;

6) розроблення вирішуючих правил і формальних методів визначення інформаційних множин доповнення простору рішень (ПР) та гарантованого управління ОУ;

7) розроблення методу формування функціонального віртуального ПР для варіативної множини ОС;

8) розроблення методу синтезу і вибору стратегій (траєкторій) переміщення та гарантованого управління ОУ в ПР при рішенні конфлікту (методу інтегрального усікання варіантів і *P*-алгоритму синтезу рішень та ланцюжків гарантованого управління ОУ в ПР);

9) проведення імітаційних експериментів з метою підтвердження теоретичних положень, а також дослідження стійкості і часових характеристик на імітаційній моделі функціонування СІУ при рішення конфлікту взаємодії ОУ з варіативною множиною ОС в умовах обмежень та невизначеностей.

Об’єкт дослідження – процеси оптимального управління багатомірними об’єктами в умовах конфлікту, обмежень та невизначеностей при взаємодії з варіативною множиною ОС.

Предмет дослідження – методологія синтезу стратегій оптимального управління об’єктами в умовах конфлікту, обмежень та невизначеностей при взаємодії з варіативною множиною ОС.

Методи дослідження. Для вирішення поставленої наукової проблеми в дисертаційній роботі використані методи теорії динамічних, нелінійних, конфліктних систем; теорія і обчислювальні методи дискретної динамічної оптимізації; теорії ергатичних систем; теорії автоматичного керування; теорії оптимального керування; теорії множин; теорії графів; теорії конфліктів; теорії ігор; синтезу систем ситуаційного управління; теорії диференціальних рівнянь; теорії інтегральної інваріантності та автономності нелінійних систем; основ топології; теорії алгоритмів; об’єктно-орієнтованого програмування і оптимізації (для розробки імітаційних моделей та обчислювальних експериментів).

Експериментальні дослідження здійснені шляхом імітаційного моделювання в сучасному обчислювальному середовищі з застосуванням обчислювальних методів, підходів ШІ, системного аналізу та сучасних технологій об'єктно-орієнтованого та абстрактного програмування з використанням імперативних та функціональних парадигм програмування.

Наукова новизна одержаних результатів полягає в подальшому розвитку теоретичних та практичних методів прийняття рішень з використанням методів дискретної динамічної оптимізації математичної моделі в теоретико-множинному підході (ТМП), що дозволить в подальшому отримувати більш прості процедури синтезу рішень щодо управління об’єктом при розв'язанні конфлікту взаємодії

об'єктів в ПС за умов обмежень, невизначеностей та варіативної множини об'єктів спостереження.

Новими науково-обґрунтованими результатами, які отримані в дисертаційній роботі, є:

–*вперше розроблений* метод рішення конфлікту взаємодії об’єкта управління з варіативною множиною об’єктів спостереження в умовах обмежень та невизначеностей, як *NP*-повної перебірної задачі динамічної дискретної оптимізації, **що дозволило** за рахунок реалізації основних підходів до рішення задач дискретної оптимізації: *визначити* принципи рішення конфлікту великої розмірності з високим рівнем абстракції в гарантовано оціненому просторі рішень для скінченних розривних областей параметрів математичної моделі, довільній системі обмежень та варіативній множині об’єктів спостереження; *визначити* властивості методу рішення конфлікту при множинному уявленні простору рішення, множин параметрів, які визначають переміщення об’єкта управління за синтезованою траєкторією, при умові гарантованого управління в класі поліноміальних алгоритмів; *розробити* алгоритм рішення конфлікту, як задачі динамічної дискретної оптимізації;

–*вперше розроблений* метод синтезу та вибору стратегій (траєкторій) переміщення та гарантованого управління об’єктом при рішенні конфлікту, **що дозволило** за рахунок вперше розробленого методу інтегрального усікання варіантів, адитивного критерію та правила зупинки при виборі оптимального рішення: *визначити* чисельне, а не аналітичне подання моделі рішення конфлікту і обмежень конфліктуючої системи; *визначити* оптимальну траєкторію переміщення об’єкта управління за умов його оптимального управління; *визначити* формалізовану структуру алгоритму, що спрощує процес рішення конфлікту; *забезпечити Р-час* рішення конфлікту;

– *удосконалена* теоретико-множинна модель взаємодії об’єкту управління з варіативною множиною об’єктів спостереження за умов конфлікту, обмежень та невизначеностей, **що дозволило** за рахунок системно-структурного та системно-функціонального дослідження явища: *визначити* основні взаємозв'язки конфлікту

з середовищем, в якому конфлікт розвивається; *визначити* характер і способи біфуркації елементів і підструктур конфлікту; *дослідити* властивості топологічності математичного простору представлення для теоретико-множинної моделі конфліктної системи; *сформувати* формальну модель конфлікту взаємодії об'єктів конфліктуючої системи в просторі спостереження; *визначити* концептуальний підхід і метод запобігання та рішення конфлікту в класі поліноміальних алгоритмів;

–*удосконалений* метод формування функціонального віртуального простору рішень для варіативної множини об’єктів спостереження, **що дозволило** за рахунок інтегрального формального множинного уявлення простору рішень: *врахувати* невизначеність при переміщенні об’єктів спостереження в просторі рішень; *визначити* множини гарантованого управління об’єктом управління при переміщенні за синтезованою траєкторією рішення конфлікту при переслідуванні та втіканні; *визначити* напрямки переміщення інформаційних множин небезпечних станів в просторі рішення конфлікту; *визначити* інтегральне формальне множинне уявлення простору рішень відповідно до запропонованої теоретико-множинної моделі опису конфлікту;

–*удосконалені* вирішуючі правила і формальні методи визначення інформаційних множин, доповнення простору рішень, гарантованого управління об’єктом при взаємодії з варіативною множиною об’єктів спостереження, **що дозволило** за рахунок формального представлення стану простору рішень: *визначити* множину простору комбінованого управління об’єктом; *визначити* спосіб доповнення простору рішень; *визначити* спосіб розрахунку секторів небезпечних напрямків переміщення об’єкта управління; *визначити* спосіб доповнення простору рішень; *визначити* спосіб формального опису простору рішень.

До нових науково-обґрунтованих результатів, які в дисертаційній роботі *отримали подальшого розвитку* можна віднести:

– методологію використання знакових моделей при теоретико-множинному уявленні конфлікту шляхом застосування семіотичної моделі опису взаємодії

об’єктів, **впровадження якої дозволило** за рахунок правил визначення граматик та мови опису стану об’єкта: *використати* контекстно-вільні граматики для опису мовних структур визначення ситуації взаємодії об'єктів в просторі спостереження; *забезпечити* синтез стратегій керуючих впливів в системі інтелектуального управління об’єктом при рішенні конфлікту;

– метод ситуаційного моделювання рішення конфлікту з застосуванням семантичної моделі опису взаємодії об’єктів, **впровадження якого дозволило** за рахунок удосконалення функціональної структури системи ситуаційного управління, граматик та мови опису процесів взаємодії об’єктів: *використати* методи лексичного, синтаксичного та семантичного аналізу для синтезу ланцюжків опису подій в системі інтелектуального управління об’єктом; *забезпечити* синтез ланцюжків гарантовано оцінених керуючих впливів за синтезованими стратегіями в системі інтелектуального управління при рішенні конфлікту; *визначити* набори ланцюжків керування об’єктом управління за синтезованими стратегіями рішень конфлікту за умов обмежень, невизначеності та варіативної множини об’єктів спостереження.

Практичне значення одержаних результатів у сукупності складає підґрунтя для вдосконалення методології оптимального управління об’єктом в умовах конфлікту, обмежень та невизначеностей при взаємодії з варіативною множиною об’єктів спостереження в просторі спостереження.

Теоретичні рішення, які одержані в дисертаційному дослідженні, і результати імітаційного моделювання підтверджують ефективність розроблених методів для вирішення задач синтезу стратегій ситуаційного управління конфліктом. Отримані результати та розроблена інформаційна технологія можуть широко застосовуватись для СІУ і відкривають можливість принципового вирішення нового класу задач управління об'єктом в просторі рішень для довільної варіативної множини недетермінованих об'єктів спостереження за умови невизначеності їх поведінки, зокрема, при вирішенні конфліктів взаємодії динамічних об’єктів та конфліктів в кібернетичному просторі.

Нові наукові результати, які отримані в дисертаційній роботі, ґрунтуючись на розроблених принципах і методах синтезу оптимальних стратегій поведінки складних ТС та ТЕС при попередженні або розв'язанні конфлікту їх взаємодії за умов обмежень, невизначеності та варіативної множини ОС, формального визначення і гарантованої оцінки параметрів простору рішення та управління ОУ, а також автоматизації усіх, супутніх цьому процесів, надають зацікавленим сторонам конструктивний інструмент для вирішення завдань з управління конфліктуючими складними багатомірними системами, включаючи нелінійні динамічні, в просторі спостереження, який має довільну систему обмежень.

Особливу практичну цінність мають такі отримані та апробовані результати дисертаційної роботи:

- концептуальний підхід до розробки методів рішення задачі конфлікту в класі алгоритмів з поліноміальним часом рішення;

- теоретико-множинна модель взаємодії об’єктів в умовах конфлікту, обмежень та невизначеностей, що забезпечує системну цілісність і надійність управління об'єктом в реальному масштабі часу при виникненні стану конфлікту в просторі спостереження або в функціональному стані об'єкта управління;

- метод семіотичного опису і аналізу стану взаємодії об’єктів в багатомірному просторі спостережень для СІУ ОУ;

- семантична модель взаємодії об’єкта управління з варіативною множиною об’єктів в просторі спостереження для визначення ланцюжків управління при рішенні конфлікту;

- предикативні моделі, правила, відношення та імітаційні моделі гарантованої оцінки стану простору комбінованого управління ОУ та простору рішень щодо вирішення задачі віртуального мережецентричного управління об'єктом в інваріантному просторі спостереження за умов конфлікту, обмежень і невизначеностей при варіативній множині об’єктів спостереження;

- методи, алгоритми, програмні засоби та імітаційні моделі синтезу

стратегій та ланцюжків управління щодо рішення конфлікту при інтегральному усіканні варіантів, що дозволяє використовувати властивість біфуркації керуючих впливів при гарантованому управлінні об'єктом у розширених діапазонах його станів для одержання нових можливостей вирішення конфліктних задач, у тому числі й запобігання зближення об’єкта управління з об’єктами в просторі спостереження

Промислова застосовність і значимість розроблених у дисертації концептуальних і теоретичних положень, моделей, методів та алгоритмів підтверджується їх використанням і впровадженням (акти подані у Додатку А до дисертаційної роботи):

- в Українській військово-медичній академії Міністерства оборони України при створенні інформаційно-телекомунікаційної системи гарантованого оцінювання та корекції психофізіологічного стану військовослужбовців (НДР шифр «Аспект», акт використання результатів від 02.09.2011р.);

- в Головному центрі спеціального контролю Національного центру управління та випробувань космічних засобів Національного космічного агентства України при створенні автоматизованої системи збору та обробки інформації (НДР шифр «Модернізація-С», шифр «Сейсмологія», шифр «Моніторинг-С», акт використання результатів від 15.02.2011р.);

- в Державному підприємстві “Медичний центр безпеки дорожнього руху та інформаційних технологій” Міністерства охорони здоров’я України при створенні і впровадження автоматизованої системи ведення бази даних медичних комісій та виданих ними медичних довідок (акт впровадження результатів від 06.06.2016р.);

- в Державному підприємстві “Держінформресурс” Державного агентства з питань електронного урядування при доопрацюванні та впровадженні системи електронної взаємодії органів виконавчої влади України (ДКР шифр “НП/2-2015”, НДР шифр “НДР-08/123-2015”, акт впровадження результатів від 05.03.2015р.);

- в товаристві з обмеженою відповідальністю “БМС-консалтінг” при

доопрацюванні складових частин програмних компонентів системи електронної взаємодії органів виконавчої влади та місцевого самоврядування з метою введення в промислову експлуатацію інформаційно-телекомунікаційної системи електронної взаємодії, що забезпечує автоматизацію процесів створення, відправлення, передавання, одержання, оброблення, використання, зберігання та знищення електронних електронних документів та копій паперових документів в електронному вигляді з використанням електронного цифрового підпису, які не містять інформацію з обмеженим доступом, та контролю за виконанням актів, протокольних рішень Кабінету Міністрів України та інших документів (ДКР шифр “ДЦІР5-1”, акт впровадження результатів від 11.02.2016р.) ;

- в товаристві з обмеженою відповідальністю “Об’єднання ЮГ” при створенні та впровадженні інформаційних технологій інтелектуального управління засобами захисту ресурсів інформаційно-телекомунікаційних систем від несанкціонованого доступу (акт впровадження результатів від 29.06.2016р.).

Також за результатами дисертаційних досліджень отримано:

- Свідоцтво про реєстрацію авторського права на твір №25553 від 24.09.2008. Комп'ютерна програма ”Програма для ПЕОМ щодо ведення бази даних дозволів на викиди забруднюючих речовин у атмосферне повітря та надання інформації громадськості” (”Дозвіл”) / В.В.Семко, О.О.Михайловський; заявник та власник авторських прав Товариство з обмеженою відповідальністю "Елан"; заявл. №26809 від 22.08.2008; видане Державним департаментом інтелектуальної власності Міністерства освіти і науки України; опубл. Кат. №12, Бюл. №17;

- Свідоцтво про реєстрацію авторського права на твір №26411 від 11.11.2008. Комп'ютерна программа ”Психодіагностичні методики” до тематичного блоку ”Пізнай себе” Програмно-апаратного комплексу ”Профорієнтаційний термінал” / В.В.Семко, О.О.Михайловський; заявник та власник авторських прав Товариство з обмеженою відповідальністю "Елан";

заявл. №27406 від 27.10.2008; видане Державним департаментом інтелектуальної власності Міністерства освіти і науки України; опубл. Кат. №12, Бюл. №17;

- Свідоцтво про реєстрацію авторського права на твір №39113 від 11.07.2011. Комп'ютерна программа ”Крос-платформена клієнт-серверна технологія захищеного обміну електронними даними з використанням терміналів мобільного телефонного зв’язку” (”ТЗОД”) / Семко В.В., Михайловський О.О., Гіденко О.В., Зінов'єв О.В., Мельник В.Т., Барткова В.А; заявник та власник авторських прав Товариство з обмеженою відповідальністю "Елан"; заявл. №39558 від 20.05.2011; видане Державним департаментом інтелектуальної власності Міністерства освіти і науки України; опубл. Кат. №15, Бюл. №25.

Копії свідоцтв про реєстрацію авторського права подані у Додатку Б до дисертаційної роботи.

Особистий внесок здобувача. Усі результати, які виносяться на захист, отримані автором особисто і опубліковані в одноосібно підготовлених працях [151, 155, 156, 158, 160, 161, 162, 165, 171, 172, 173, 133, 146, 168].

У наукових працях, написаних у співавторстві, автору належить: [132] – формальний опис методу інтегрального усікання варіантів; [152] – технічні вимоги до системи збирання параметрів щодо стану операторів на об'єктах соціально-економічної та інформаційної діяльності; [153] – топологія системи виявлення показників психофізіологічного стану особистості для професійної діагностики та оцінки стану оператора інформаційно-телекомунікаційної системи; [157] – постановка задачі, математична модель управління захистом інформації в інформаційно-телекомунікаційній системі, теоретичне обґрунтування, постановка імітаційного експерименту; [159] – постановка задачі, модель системи інтелектуального управління конфліктом, теоретичне обґрунтування, постановка імітаційного експерименту; [163] – формальна модель конфлікту для дискреційної моделі управління доступом в інформаційно-телекомунікаційній системі; [164] – формальна та імітаційна моделі оцінки можливості отримання *Z*-системи, як

конфліктуючої системи, при зміні стану інформаційно-телекомунікаційної системи; [166] – постановка задачі, теоретичне обґрунтування, постановка імітаційного експерименту щодо побудови областей керованих станів динамічного об’єкта; [167] – постановка задачі, обґрунтування методу вирішення; [174] – математична множинна модель конфліктуючих об’єктів в умовах невизначеності і обмежень; [139] – алгоритм вирішення конфлікту в віртуальному просторі знань при теоретичній підготовці інженерів-пілотів; [148] – топологія та концептуальна побудови модель конфлікту в інформаційно-телекомунікаційних системах санітарних та природоохоронних служб при врахуванні викидів забруднюючих речовин в атмосферне повітря; [149] – модель попередження конфлікту при ранній професійній орієнтації школярів з використанням психодіагностичних методик; [150] – модель та алгоритми ситуаційного управління системою безпеки інформації при передачі негетерогенних електронних повідомлень в системах спеціального зв'язку; [170] – структура та проблемні аспекти функціонування мережі *INTERNET* в кіберпросторі; [125] – концептуальні засади та топологія створення інформаційно-аналітичної системи; [126] – концептуальні засади та топологія створення автоматизованої системи; [144] – математична модель оцінки та прогнозування помилкових дій людини-оператора при конфлікті в інформаційно-телекомунікаційній системі; [146] – концептуальна модель конфлікту взаємодії конфліктуючих елементів кібернетичної системи; [154] – концепція, структура, топологія, основні функціональні вимоги до взаємодії об'єктів та суб'єктів захищеної операційної системи в умовах конфлікту; [168] – засади топологічного аналізу та обґрунтування теоретичних засад визначення способу рішення конфлікту.

Апробація результатів дисертації. Наукові результати та основні положення дисертаційної роботи доповідались, обговорювались на республіканських, всесоюзних і міжнародних наукових семінарах і конференціях, а саме: засіданні республіканського наукового семінару «Ергатичні системи управління» (Київ, 1987-1991р.р.); секціях п'ятої всесоюзної науково-практичної конференції по безпеці польотів «Безпека польотів і профілактика авіаційних

подій» (м.Ленінград, 1988 р.); секціях міжвідомчої науково-практичної конференції «Проблеми професійної підготовки на етапі прискорення НТП» (м.Кіровоград, 1989 р.); науково-технічному семінарі «Застосування машинної графіки в моделюванні і навчальних системах» (м.Пенза, 1989 р.); секціях всесоюзної науково-технічної конференції «Методологія створення і досвід експлуатації АСУ в цивільній авіації» (м.Рига, 1989р.); наукових семінарах кафедри безпеки польотів КІЦА (Київ, 1987-1991р.); секціях другої всесоюзної науково-практичної конференції «Проблеми застосування технічних засобів у формуванні професійних навичок при первісній підготовці льотного складу» (м.Актюбінськ, 1989р.); секціях відомчої науково-практичної конференції «Профилактическая деятельность предприятий гражданской авиации по предотвращению авиационных происшествий» (м.Київ, 1990 р.); секціях міжнародної науково-практичної конференції «Забезпечення безпеки польотів в нових економічних умовах» (м.Київ, 1997р.); секціях ІІ міжнародної науково-практичної конференції «Комп'ютерні системи та мережні технології» (м.Київ, 2009р.); секціях першого міжнародного з'їзду «Медична та біологічна інформатика і кібернетика» (м.Київ, 2010р.); секціях IV, V та IV міжнародних науково-практичних конференцій «Комп'ютерні системи та мережні технології» (м.Київ, 2011 – 2013 р.р.); наукових семінарах Інституту комп'ютерних інформаційних технологій Національного авіаційного університету (Київ, 2013р.); наукових семінарах Інституту телекомунікацій та глобального інформаційного простору НАН України (Київ, 2010-2013р.), секціях науково-технічної конференції студентів, аспірантів, викладачів та науковців «Актуальні проблеми забезпечення інформаційної безпеки держави» (м.Київ, 2014р.); наукових семінарах Державного університету телекомунікацій (Київ, 2014-2015р.); науковому семінарі кафедри системного аналізу і теорії прийняття рішень факультету кібернетики Київського національного університету імені Т.Г. Шевченка (Київ, 2012-2015р.); наукових семінарах кафедри комп'ютерних систем управління Інституту комп'ютерних інформаційних технологій Національного авіаційного університету (Київ, 2013-2014р.); наукових семінарах кафедри інформаційної та кібернетичної

безпеки Державного університету телекомунікацій (Київ, 2014-2016р.).

Публікації. Основні наукові результати дисертаційної роботи повно викладені в 33 публікаціях, що включають: 1 учбовий посібник [27], 21 наукова стаття, з яких 20 [1-17, 19-21] - в наукових спеціалізованих фахових виданнях, затверджених МОН України і 1 стаття в міжнародному науковому журналі [18]. З 11 статей [2, 5, 6, 8, 10, 11, 12, 15, 18, 19, 20], які написані без співавторів, 1 стаття [15] надрукована у виданні України, яке включено до міжнародної наукометричної бази Scopus, 2 статті [8, 11] - до міжнародної наукометричної бази eLIBRARY.RU, 3 статті [18, 19, 20] - до міжнародної наукометричної бази Google Scholar, 1 стаття [18] - до міжнародної наукометричної бази IndexCopernicus. З 10 статей [1, 3, 4, 7, 9, 13, 14, 16, 17, 21], які написані з співавторами, 3 статі [9, 13, 14] надруковані у виданнях України, які включені до міжнародної наукометричної бази IndexCopernicus, 3 статті [13, 14, 21] - до міжнародної наукометричної бази Google Scholar. Отримано 3 свідоцтва про реєстрацію авторського права на твір [24, 25, 26]. Додатково основні наукові результати відображені у 6 тезах доповідей на науково-технічних конференціях [28 - 33], 6 депонованих рукописах, 3 наукових статтях в фахових виданнях, які були опубліковані до 1992 року, наукових звітах з виконання НДР та ДКР.

Структура і обсяг дисертації. Дисертаційна робота складається зі вступу, п’яти розділів, висновків, списку використаних джерел із 232 найменувань та 5 додатків на 139 сторінках. Загальний обсяг дисертації – 422 сторінки, основний текст – 262 сторінки. Робота містить також 11 таблиць та 82 рисунки.

РОЗДІЛ 1. АНАЛІЗ. СТАНУ ПРОБЛЕМИ УПРАВЛІННЯ ОБ'ЄКТОМ В УМОВАХ КОНФЛІКТУ В ТЕХНІЧНІЙ ЕРГАТИЧНІЙ СИСТЕМІ

1.1. Аналіз стану проблеми управління об’єктом при вирішенні конфлікту

Поняття конфлікту є одним з засадних понять теорії технічних систем [102]. З поняттям конфлікту пов'язане існування напряму технічних систем (ТС) та технічних ергатичних систем (ТЕС) в кібернетиці. В цілому конфлікт розуміється як явище взаємодії по-різному спрямованих сторін - об'єктів, систем, тощо. Тобто поняття конфлікту тісно пов'язане з проблемою цілеспрямованості функціонування або переміщення сторін, що взаємодіють в фізичному або віртуальному просторі. В такому разі розглядається проблема цілі функціонування або переміщення ТС або ТЕС в цілому, як моделі майбутнього стану системи. Слід зазначити, що в загальному випадку модель цілі функціонування або переміщення об'єднує три підмоделі - попередня, поточна та майбутня, які закріплюються в структурі організації ТС або ТЕС і складають три рівні по відношенню до моделі цілі.

Для формального опису конфліктів в технічних системах застосовується математичний апарат теорії управління, а саме - теорія конфліктно-керованих процесів, яка є розділом сучасної математики, що інтенсивно розвивається.

Однією з перших робіт з теорії конфліктно-керованих процесів, слід вважати роботу Г. Штейнгауза [211], в якій він сформулював задачу переслідування.

В теорії конфліктно-керованих процесів досліджуються завдання управління динамічними процесами в умовах конфлікту, обмежень ы невизначеностей, який припускає наявність двох або більше сторін, які здатні впливати на процес з протилежними або неспівпадаючими цілями і оптимізувати задані функціонали якості процесу. Динамічні процеси можуть

описуватися диференціальними, інтегральними, різницевими, гібридними і іншими рівняннями.

В.М.Кунцевичем [75] і представниками його наукової школи досліджуються проблеми управління об’єктами та системами в умовах невизначеності. В даний час для рішення задач управління об’єктами та системами в умовах невизначеності широке поширення одержали методи, що засновані на імовірнісній інтерпретації їх властивостей. Однак ці методи вимагають для своєї реалізації великого обсягу апріорної інформації про імовірнісні властивості невизначеності, що часто відома розроблювачам систем управління (СУ) лише приблизно. Крім того, такі об’єкти і системи є критично чутливими до відмінності істинних властивостей невизначеності від їх апріорно передбачуваних властивостей.

Необхідність вирішення складних задач управління об’єктами і системами в умовах невизначеності привела у багатьох випадках до відмови від розповсюджених імовірнісних моделей на користь використання теоретико-множинних моделей.

Так в сучасній теорії управління інтенсивно розвивається підхід до створення методів управління й оцінювання, заснований на теоретико-множинних моделях невизначеності, який отримав назву гарантованого підходу. При цьому властивості невизначених величин більш природно задаються лише гарантованими інтервалами (як це широко розповсюджено в машинобудуванні та вимірювальній техніці) або компактними множинами своїх можливих значень. Як множинні оцінки використовуються опуклі багатогранники або багатомірні еліпсоїди у відповідних просторах.

Конфліктно-керовані процеси, що описуються звичайними диференціальними рівняннями, були досліджені Р.Айзексом [1] і отримали назву диференційних ігор.

Становлення теорії диференційних ігор зв'язане з такими вченими, як М.М. Красовський, Л.С. Понтрягін, Л.А. Петросян, Б.Н. Пшеничний, Р.Айзекс, У.Флеминг.

Подальший розвиток теорії диференційних ігор пов'язаний з А.А.Азамовим, А.Я.Азімовим, Е.Г.Альбрехтом, М.Барді, В.Д.Батухтіним, Т.Башаром, Ю.І.Бердишевим, А.Брайсоном, М.С.Габріеляном, М.Л.Григоренком, Р.В.Гамкрелідзе, П.Б.Гусятніковим, М.І.Гусєвим, В.Г.Гусейновим, В.І.Жуковським, В.В.Захаровим, М.І.Зелікіним, Д.Зонневендом, Р.П.Івановим, А.Ф.Клейменовим, А.В.Кряжимським, О.Б.Куржанським, А.М.Красовським, Дж.Лейтманом, В.Н.Лагуновим, Ю.С.Ледяєвим, Н.Ю Лукояновим, А.А.Мелікяном, А.В.Мезенцевим, Є.Ф.Міщенком, М.С.Нікольским, В.В.Остапенком, Ю.С.Осиповим, А.Г.Пашковим, В.С.Пацком, Н.Н.Петровим, Г.К.Пожарицьким, Є.С.Половинкиним, Б.Б.Рихсиєвим, Й.С.Раппопортом, Н.Ю.Сатимовим, А.І.Субботіним, В.Є.Третьяковим, В.М.Ушаковим, В.І.Ухоботовим, У.Флемингом, А.Фридманом, Хо-Ю-Ши, А.Г.Ченцовим, Ф.Л.Черноуськом, А.О.Чикрієм, С.В.Чистяковим, Р. Елліотом та іншими математиками і фахівцями в галузі теорії управління.

М.М.Красовским [70] і представниками його наукової школи створена теорія позиційних ігор, в основі якої лежить поняття стабільного моста і правило екстремального прицілювання. Для широких класів диференціальних ігор доведена теорема про альтернативу, обгрунтовані методи детермінованих і стохастичних програмних конструкцій. Рішення ігрової задачі зводиться до послідовного вибору екстремальних управлінь, що зберігають траєкторію конфліктно-керованого процесу на стабільному мосту і приводять траєкторію по ньому на термінальну множину. В ігровій задачі формулюються два завдання про позиційний підхід (тобто управлінні за принципом зворотного зв'язку). Перше завдання (стоїть перед першим гравцем) - завдання про зближення з цільовою множиною в просторі заданих фазових обмежень. Друге завдання (стоїть перед другим гравцем) - завдання про ухилення від цільової множини першого гравця. Сукупність цих (протилежних) завдань є диференціальною грою зближення-ухилення. Зв'язок між завданнями розкривається центральним результатом теорії позиційних диференціальних

ігор - теоремою про альтернативу, яка стверджує, що у рамках прийнятої формалізації завжди, при відповідному виборі класів стратегій гравців, вирішуваним є одне і тільки одне з вказаних завдань. Таким чином, завдання зближення-ухилення є взаємовиключаючими і взаємодоповнюючими. Звідси слідує важливий висновок про принципову відсутність покращення позиційного способу управління. Крім того, теореми про альтернативу дозволяють розглядати кожне із завдань зближення або ухилення як критерій вирішуваного протилежного завдання.

У роботах А.І.Субботіна [172, 173] умови стабільності сформульовані за допомогою похідних по напряму. В результаті отримані диференціальні нерівності, які узагальнюють основне рівняння диференціальних ігор, що було запропоновано Р. Айзексом [1] і застосовано в теорії узагальнених рішень рівнянь Гамильтона-Якобі.

Найкраще рішення позиційної диференціальної гри зближення-ухилення забезпечується шляхом створення максимальних стабільних мостів у фазовому просторі, але їх створення для дослідження реальних конфліктно-керованих процесів є дуже складним, а іноді навіть неможливим.

Початкові ідеї в цьому напрямі належать Л.С.Понтрягину [111, 112] і реалізовані ним в методі альтернованого інтеграла та досліджені Б. Н. Пшеничним [117] для лінійних та нелінійних систем для випадків фіксованого і нефіксованого часу.

В роботах Ф.Л. Черноуська [201, 202, 203] розглядалася завдання ухилення керованої точки, швидкість якої обмежена за величиною, від зустрічі з будь-яким кінцевим числом переслідуючих точок, швидкості яких обмежені при способі управління, який забезпечує ухилення від усіх переслідувачів на кінцеву відстань, причому рух точки, що ухиляється, залишається у фіксованій околиці заданого руху. Тобто запропоновано розгляд задачі групового переслідування одного убігаючого об’єкту. В подальшому задача групового переслідування була досліджена М.Л.Григоренком [38] та отримані необхідні і достатні умови ухилення від зустрічі одного об’єкту, що втікає, від

декількох переслідуючи об’єктів за умови, що той хто втікає і переслідує мають простий рух, а множина припустимих управлінь кожного з гравців - один і той же опуклий компакт.

В роботі [71] М.М.Красовським були отримані умови оптимальності часу переслідування в диференціальній грі одного об’єкту, що втікає, від декількох переслідуючи об’єктів, рух яких є простим.

Абсолютна більшість робіт, що присвячені вирішенню як задачі переслідування, так і задачі убігання розглядалися при певній перевазі однієї із сторін та для випадку простого переслідування. Слід також зазначити, що диференційні ігри за участю декількох осіб, за наявності додаткових обмежень на стан однієї із сторін та задача ухилення при рівних динамічних і інерційних можливостях гравців є практично невивченими.

При розгляді підходів до розв'язання конфліктно-керованих процесів сучасна теорія управління пропонує методи, які дозволяють вирішувати задачу запобігання та розв'язання конфліктів без використання математичного апарату теорії ігор [77, 80, 85, 101, 103, 114, 162, 166], а саме - методи ситуаційного управління.

Методи ситуаційного управління використовуються для вирішення складних задач дослідження операцій, які відносяться до класу задач упорядкування. Основною особливістю методів ситуаційного управління є те, що детальний опис множини ситуацій, які складаються в процесі функціонування реального об'єкту, за певними правилами замінюють макроописами узагальнених ситуацій, кожна з яких з достатньою вірогідністю визначає одне з можливих рішень задачі. Така заміна дозволяє істотно скоротити перебір варіантів рішень та прискорити пошук оптимального.

Дослідження за напрямком ситуаційного управління були започатковані В.Н.Пушкіним і Д.О.Поспеловым [116]. Ці дослідження є одним з нових напрямів створення систем прийняття рішень і управління в умовах конфлікту. Засадними для ситуаційного управління конфліктно-керованими процесами є інформаційні системи на основі штучного інтелекту та знань, які

дозволяють синтезувати стратегії, приймати рішення та здійснювати управління об'єктом на основі предметних знань за умов невизначеності похідних даних та обмежень.

Системи ситуаційного управління конфліктно-керованими процесами знайшли широке використання при управлінні ТС, ТЕС, включаючи автоматичні системи (ергамати) [159, 160, 162], які функціонують за певними алгоритмами і за сутністю копіюють поведінку людини [146, 151, 156, 158, 162].

Синтез та вибір стратегій управління конфліктно-керованими процесами в технічних системах на основі методів та підходів штучного інтелекту знайшли широке використання в управлінні системами, для яких носії інтелекту традиційно знаходили застосування, включаючи ручне і напівавтоматичне.

В роботі Д.О.Поспелова [114] ситуаційне управління на відміну від традиційного управління характеризується рядом властивостей, що відрізняють його від управління, яке вивчається в традиційній теорії управління, а саме - наявністю елементу, що реалізує функції відтворення штучного інтелекту. Таким елементом є система формалізованого опису знань предметної області (формальна мова, правила, граматики, обмеження, машина виводу, тощо).

Застосування методів ситуаційного управління конфліктно-керованими процесами враховує систему обмежень при синтезі та виборі рішень, а саме: технологічних, які визначаються кінематичними [138, 158], та динамічними [165, 166] властивостями об’єктів, що є учасниками конфлікту; функціональний та професійний стан особи, що приймає рішення (ОПР) (людський фактор) [100, 102, 103, 104, 131, 142, 149, 152, 153]; обмеження на цілі і способи реалізації стратегій управління об'єктом, які обумовлені фактичним характером поточної ситуації конфліктно-керованого процесу [106, 132, 134, 137, 138, 149, 151, 155, 158, 160, 162, 167].

Поточна ситуація управління може бути визначена як суб'єктивна оцінка конкретних характеристик об'єкта управління, зовнішнього середовища (ситуаційних змінних) і зв'язків між ними, що мають місце на поточний момент часу, але залежних від стану середовища спостереження конфліктно-керованого процесу, в якому функціонує об'єкт управління в часі і просторі.

Формальною основою опису процесів ситуаційного управління складними ТС і ТЕС є логіко-лінгвістичні моделі опису поточних ситуацій за станом об'єкту управління.

1.2. Аналіз методів та підходів до вирішення задачі управління об'єктом при розв'язанні конфлікту

За результатами аналізу методів та підходів до рішення конфлікту взаємодії ОУ з варіативною множиною об’єктів спостереження (ОС) в умовах обмежень і невизначеностей визначено проблеми, що притаманні методам дослідження операцій (рис. 1.1), а саме: методи стандартних схем, оптимізаційні методи, ймовірносно-статистичні методи, полеві методи.

Для методів стандартних схем визначальною є оцінка термінальної множини (точки) зустрічі ОУ і ОС при переведенні з початкової в кінцеву множину термінальних позицій шляхом вибору рішення та управління щодо рішення конфлікту. Основною проблемою, яка виникає при рішенні конфлікту за методами стандартних схем є ефект «доміно», який призводить до неконтрольованого зростання часу обчислень.

Оптимізаційним методам притаманні проблеми складності математичної моделі і, як наслідок, складність алгоритмів функціонування регулятора, неконтрольоване зростання часу рішення конфлікту при збільшенні кількості учасників конфлікту (ефект «прокляття розмірності»).

Ймовірносно-статистичні методи мають проблему нестачі похідних даних для математичної моделі, що призводить до невизначеності конфлікту.

Полеві методи методологічно використовують складні алгоритми визначення локальних мінімумів множини параметрів, які описують властивості переміщення об’єктів за усіма можливими траєкторіями в просторі конфлікту. При збільшенні кількості учасників та ускладненні системи обмежень призводить до унеможливлення рішення конфлікту.

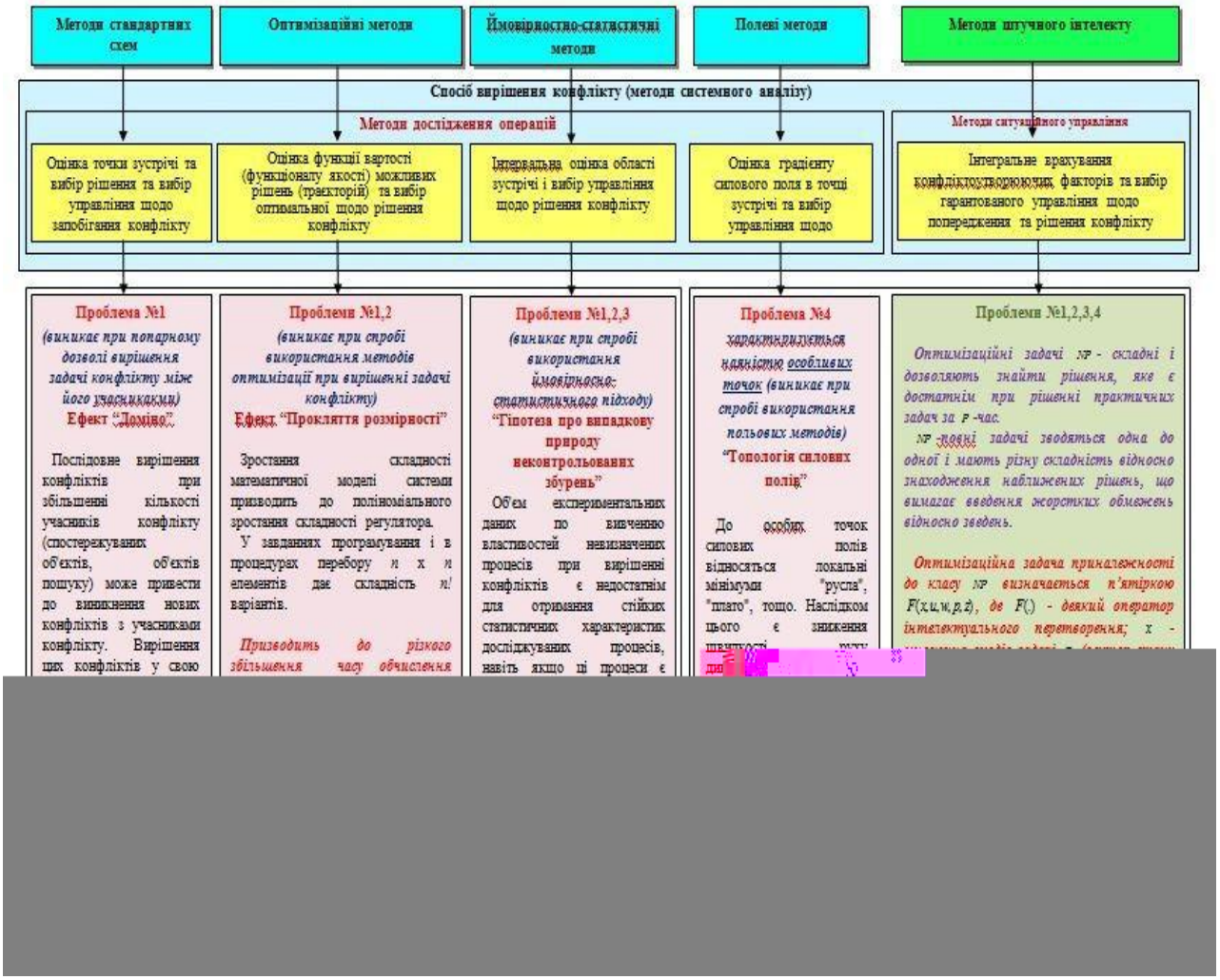


Рис. 1.1. Проблеми методів дослідження операцій при рішенні конфлікту

Застосування методів штучного інтелекту дозволяє вирішити задачу конфлікту як *NP*-складну за рахунок інтегрального врахування конфліктоутворюючих факторів та вибору гарантованого управління ОУ і отримати алгоритм рішення, який відноситься до класу поліноміальних. В такому разі час розрахунку рішення конфлікту матиме поліноміальну залежність від кількості конфліктуючих сторін.

В якості ІП [115, 116, 162], який використовує системи ШІ, застосовуються експертні системи, технології ситуаційного управління, управління структурною динамікою складних технологічних та інших інтелектуальних системи та їх елементів.

При використанні в СІУ ІП, який використовує експертну систему або систему, яка створена за методологією оптимального управління об’єктом в умовах конфлікту, обмежень і невизначеностей, до складу СІУ включено два функціональні набори модулів, які об'єднані в блоки синтезу та реалізації мети (рис. 1.4).

Рис. 1.4. Структурна схема системи інтелектуального управління системою з динамічною маніпулювання знаннями

В такому разі математична модель СІУ складається з трьох частин: ІП (експертна система або система ситуаційного управління); ОУ; пристрої управління СУ (обчислювальні, перетворюючі та виконавчі пристрої).

ІП є логіко-перетворюючим пристроєм, який перетворює інформацію про стан зовнішнього середовища і параметри функціонування ОУ, трансформує в сигнали для пристрою управління СУ [65].

Визначимо математичну модель ІП в операторній формі

$$Y = F(x,u,w,p,z), \tag{1.1}$$

де $F(.)$ - деякий оператор інтелектуального перетворення, який характеризує структуру та роботу ІП; x - вектор стану СУ; u - вектор управління; w - вектор впливу зовнішнього середовища; p - вектор сигналів мети (цілі); z - вектор параметрів ОУ.

В загальному випадку ОУ можна описати системою рівнянь

$$\begin{cases} x = f(x,u,w,t) \\ y = C(x) \\ x(t_0) = x_0 \\ t \geq t_0 \end{cases}, \tag{1.2}$$

де $f(.)$ - вектор-функція, яка описує властивості ОУ; $C(.)$ - задана функція вихідних сигналів; t - координата часу: y - вихідний вектор (вектор вимірів).

В такому разі обчислювальні та перетворюючи пристрої формують вектор управління u для ОУ з множини його можливих значень згідно задачі управління для досягнення мети, яка сформована ІП на підставі похідних даних згідно співвідношень (1.1) та (1.2).

В разі побудови ІП на основі методів ситуаційного управління використовуються семіотичні моделі, які грунтуються на формальній моделі, що задається четвіркою

$$M = \langle T, P, A, \Pi \rangle, \tag{1.3}$$

де T - множина базових елементів; P - синтаксичні правила; A - система аксіом; Π - семантичні правила.

Крім четвірки M відповідно співвідношенню (1.3**)** задається формальна модель, що інтерпретується,

$$L = \langle Z, D, H, V \rangle, \tag{1.4}$$

де Z - множина значень, які інтерпретуються, D - правила відображення, які надають відображення $T \to Z$ та зворотнє $Z \to T$, тобто приписує кожному відображенню T деяке відображення, що його інтерпретує; H - правила

відображення; V - правила інтерпретації, які дозволяють приписувати деяке інтерпретуюче значення до будь-якої синтаксично правильної сукупності базових елементів.

Тоді семіотична модель $C(.)$ з врахуванням (1.3) і (1.4) можна описати четвіркою

$$C = \langle M, \chi_T, \chi_P, \chi_A, \chi_П \rangle, \tag{1.5}$$

де $\chi_T, \chi_P, \chi_A, \chi_П$ відповідно є правилами зміни $T, P, A.П$.

На відміну від формальних моделей використання семіотичної моделі $C(.)$ відповідно співвідношення (1.5) дозволяє змінювати усі елементи формальної моделі M, яка визначена співвідношенням (1.3), в процесі ситуаційного управління, що дозволяє будувати моделі, які відповідають діяльності людини, ТС та ТЕС, для поточних ситуацій.

Використання мови знаків робить ситуаційне управління ефективним. Оскільки людина оточена знаковими системами, вона дістає можливість домовлятися зі своїми партнерами про синтаксис, семантику і прагматику знаків.

Ситуаційне управління дозволяє описувати ряд конфліктних ситуацій, в яких вимагається приймати конкретні рішення. В той же час великий клас реальних задач по управлінню в умовах конфлікту використовують методи дослідження операцій на основі теорії ігор, які припускають використання формальних моделей. Тому для успішного застосування методів ситуаційного управління при синтезі та прийнятті рішень в умовах конфлікту доцільним є також використання інструментарію методів теорії ігор та функціонального аналізу.

Найбільш загальною моделлю прийняття рішення в умовах конфлікту є модель динамічної системи інтелектуального управління на основі моделей теорії ігор [28] і сучасного апарату функціонального аналізу [64]. Теорія ігор як розділ математики нині став теорією математичних моделей [28, 42, 45, 72, 79, 84, 94, 100, 101, 102, 117, 120, 204].

При ігровому підході для опису конфлікту вводяться поняття:

- коаліцій дії $R_Д$, яка об'єднує множину гравців (учасників конфлікту) за їх діями;

- коаліцій інтересів $R_И$ яка об'єднує гравців за інтересами (спільністю мети);

- стратегій, що характеризують рішення коаліцій K (дій $R_Д$ і коаліцій інтересів $R_И$, які залежно від типу гри можуть мати одних і тих же гравців або утворюватися з різних гравців);

- стосунків переваги G, як абстрактного бінарного відношення на множині всіх стратегій (часто стосунки переваги задаються функцією виграшу W_K і тоді коаліція K, якщо вона віддає перевагу ситуації x до ситуації y, означають через відношення переваги у вигляді $xG_K y$ або при використанні функції виграшу за умови визначення $W_K(x)$ і $W_K(y)$ у вигляді $xW_K y$).

Тоді формальний опис конфлікту полягає в завданні системи

$$\Gamma = \left\langle R_Д, S_{R_Д}, S, R_И, G_{R_И} \right\rangle, \qquad (1.6)$$

де $R_Д$ - множина, яка об'єднує гравців (учасників конфлікту) за їх діями; $S_{R_Д}$ - множина стратегій коаліції дій; S - множина стратегій поведінки (дій) об'єктів; $R_И$ - множина, яка об'єднує гравців (учасників конфлікту) за інтересами (цілями); $G_{R_И}$ - множина відношень коаліції інтересів. Тобто множина Γ згідно співвідношенню (1.6) є формальним описом конфлікту в залежності від його змісту.

Динамічна системи управління складаються з об'єкту, що характеризується деякою множиною станів, регулятора, під яким розуміється математична модель, що складається з елементу, який забезпечує оцінку стану об'єкту, і елементу, що формує управління. Для опису динамічного об'єкту або системи використовується математична модель сукупності елементів (рис.1.5), які задовольняють основним аксіомам.

Аксіома 1. Для заданої множини моментів часу T та множини станів системи X множина вхідних впливів U є непорожньою множиною їх припустимих значень.

Нехай $\Omega = \{\omega : T \to U\}$, тоді множина значень вихідних величин Y та множина їх припустимих значень

$$Y = \{\gamma : T \times X \to Y\}.$$

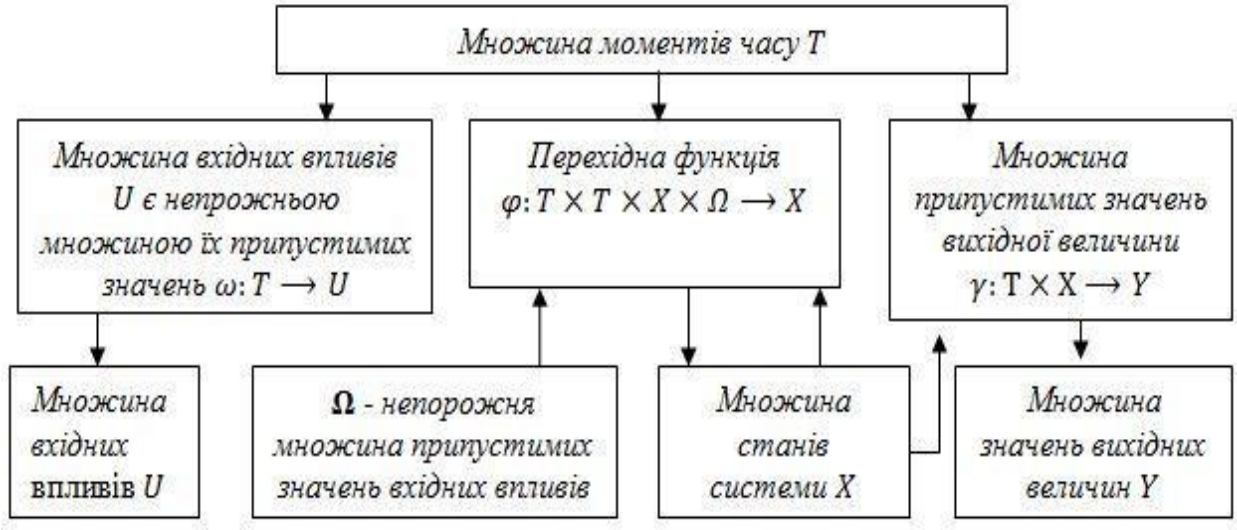


Рис. 1.5. Структурна схема математичної моделі сукупності елементів динамічного об'єкту або системи

Аксіома 2. Множина T є деякою упорядкованою підмножиною множини дійсних чисел (напрямок часу).

Аксіома 3. Існує перехідна функція

$$\Phi = \{\varphi : T \times T \times X \times \Omega \to X\},$$

де стан системи $x(t) = \varphi[t, t_0, x(t_0), \omega] \in X$.

Аксіома 4. Задане вхідне відображення $Y = \{\gamma : T \times X \to Y\}$.

В такому разі стан системи $x(t)$ в момент часу t або пари елементів множини $T \times X$ буде являти подію (фазу) динамічної системи. Множина $T \times X$ є простором подій (фазовим простором, простором станів). Управляючі дії переводять систему з одного стану в деякий інший. При переводі системи з одного стану в інший за допомогою керуючих дій остання знаходиться в стані

руху, описуючи в просторі стану траєкторію, а динамічна система являє собою об'єкт управління (рис. 1.6).

Законом управління є відображення $k: T \times X \to U$, де значення $u(t) = k[t, x(t)]$ належить множині U.

Для реалізації управління необхідно мати значення змінних стану системи $x(t)$, що потребує здійснення операції зворотнього відображення $\gamma^1 : Y \to X$, що дозволяє визначити координати системи з умови $x(t) = \gamma^1[y(t)]$.

Для оцінки стану системи оцінюється точність визначення $x(t)$, тобто оцінюється стан $x_0(t)$.

Для врахування факторів взаємодії ОУ з зовнішнім середовищем та іншими учасниками конфлікту зійснюється аналіз їх взаємодії. При цьому чинники зовнішнього середовища і умов конфлікту розглядються з позицій динамічної системи.

Рис. 1.6. Структурно-логічна схема основного блоку управління системи

При прийнятті рішення в СІУ має сенс здійснити об'єднання ІП, який вміщує блок прийняття рішення, з динамічною СУ.

Стратегія вирішення конфлікту залежить від ресурсів ОУ та факторів, які утворюють ситуацію прийняття рішень в умовах конфлікту. Тим самим умови конфлікту породжують фактори (визначені та невизначені), які впливають на стратегію вирішення конфлікту, а саме на синтез та вибір керуючих впливів на ОУ, які здійснюються інтелектуальною системою.

Математична модель синтезу та прийняття рішень формується з урахуванням всіх чинників і наявної інформації про них. Спрощена модель прийняття рішення в цьому випадку може бути описана у вигляді

$$D_0 = \langle Y, G, U, J, \Omega \rangle, \qquad (1.7)$$

де Y - множина результатів, G - модель переваг результатів (рішень, що приймаються); U - множина стратегій прийняття рішень; L - множина можливих значень невизначених чинників; J - функція, що визначає взаємозв'язок невизначеного чинника і результат прийнятого рішення (функція ціни, критерій); Ω - вся інша інформація про рішення, що приймається, у формалізованому виді відомості про конфлікт (переваги інших учасників, які беруть участь в конфлікті, тощо).

Зручність використання моделі (1.7) в умовах конфлікту визначається тим положенням, що вона дозволяє просто і наочно зв'язати значення невизначених чинників і стратегій з управлінням, що реалізовується інтелектуальною системою.

Множини Y, G, U, L, Ω і функція J формально задають компоненти рішення, що приймається, і визначають зв'язок з системою управління через поняття функції ціни і показників ефективності системи (показників якості і критеріїв). Показником якості або ефективності системи управління W є міра відповідності реального результату управління Y щодо необхідного Y^{mp} для досягнення мети управління ОУ при вирішенні конфлікту і отриманні оцінок або вимірів інтенсивності результатів рішення. Критерієм K [32, 42, 67, 102,

159-162] є правило, введене на основі певної концепції раціональної поведінки інтелектуальної системи (придатності, оптимізації, адаптивності, витрат ресурсів, надійності, безпеки, тощо).

В такому разі рішення задачі конфлікту доцільно розглядати в постановці задачі дискретної оптимізації [120] для множини усіх значень цільової функції

$$f(x)=\min\{f(x): x\in X\}\,, \tag{1.8}$$

де X - припустима область значень параметрів; f - цільова функція; кожен елемент $x\in X$ - припустиме рішення задачі дискретної оптимізації (J, X) за умови скінченності множини X.

Оптимальному рішення $x^*\in X$ відповідає значення цільової функції

$$f(x^*)=\min\{f(x): x\in X\}\,. \tag{1.9}$$

Тоді дискретний процес для рівняння стану можна уявити у вигляді

$$y_t = P(y_{t-1}, x_{t-s},..., x_t), t=1,2,..., T\,, \tag{1.10}$$

де $x_{t-s}, x_{t-s+1},..., x_t$ - управління при перетворенні стану y_{t-1} етапу $t-1$ в стан y_t етапу t. Стан y_t обирається виходячи з множин станів Y_t, областей управління X_t і множин Z_t. Множини X_t скінченні, а цільова функція J для кожного t дозволяє визначити мінімум сумарної оцінки $f(\bullet,\bullet,...,\bullet)$ на $Y_{t-1}\times X_{t-s}\times...\times X_t$.

Визначимо немарківську динамічну задачу дискретної оптимізації у вигляді

$$\sum_{t=1}^{T} f_t(y_{t-1}, x_{t-s},..., x_t)+ f_0(y_T)\to\min\,, \tag{1.11}$$

де $y_t = P_t(y_{t-1}, x_{t-s},..., x_t)$, $x_t\in X_t$, $y_t\in Y_t$, $(y_{t-1}, x_{t-s},..., x_t)\in Z_t$, $t=1,2,..., T$, $x_t = x_t^0$, $t=1-s,...,0$, $y_0 = y^0$.

Виходячи з співвідношень (1.8) – (1.11) модель алгоритму рішення конфлікту, як задачі дискретної динамічної оптимізації, можна представити наступним чином.

Визначимо скінченні множини $\overset{\circ}{Y}_t$, що вміщують стани, в яких може знаходитись процес, і для яких є справедливим включення:

$$\left\{ y_t \in Y_t : y_t = P_t(y_{t-1}, x_t), x_t \in X_t, (y_{t-1}, x_t) \in Z_t, y_{t-1} \in \overset{\circ}{Y}_{t-1}, t = 1,2,...,T, Y_0 = \overset{\circ}{\{y^0\}} \right\}.$$

1. Для усіх $y \in \overset{\circ}{Y}_T$ визначаємо $F_{T+1}(y) = f_0(y)$.

2. Для усіх $y \in \overset{\circ}{Y}_{t-1}$ і $t = T, T-1,...,1$ обчислюємо $F_t(y) = \min\{ f_t(y, x + F_{t+1}(P_t(y,x)) : x \in X_t, P_t(y,x) \in Y_t, (y,x) \in Z_t) \}$.

Якщо задача $F_t(y)$ не може бути вирішена, припускаємо, що $F_t(y) : +\infty$. В іншому випадку визначаємо оптимальне рішення, як $x_t(y)$.

3. $F_1(y^0)$ є мінімальним значенням функції для немарківської динамічної задачі.

4. Визначаємо $x_1 : x_1(y^0)$ та $y_1 := P_1(y^0, x_1)$.

5. Для усіх $t = 2,3,...,T$ припускаємо $x_t := x_t(y_{t-1})$, $y_t := P_t(y_{t-1}, x_t)$.

При $F_1(y^0) < +\infty$ немарківська динамічна задача має оптимальне рішення $\{x_t, y_{t-1}\}_{t=1}^{T}$. Кроки 4 і 5 алгоритму породжують оптимальне рішення оскільки $x_t(y)$ визначає відповідне оптимальне управління на першому етапі процесу, який складається з етапів $t, t+1,..., T$ і починається з стану y. Тим самим здійснюється перехід до нового стану $P(y, x_t(y))$, для якого відомо оптимальне управління $x_{t+1}(P(y, x_t(y)))$ першого етапу процесу, який вміщує етапи $t, t+1,..., T$.

Для моделі рішення задачі конфлікту (1.3) запропоновано еквівалентну модель (1.4), що дозволяє визначити поліноміальну еквівалентну задачу оптимізації про оптимальну траєкторію переміщення ОУ за методом дискретної динамічної оптимізації і неявного перебору. Саме моделі дискретної оптимізації при рішенні конфлікту є доцільними при використанні методів декомпозиції [120].

1.3. Система інтелектуального управління об'єктом

ТС або ТЕС, яка знаходиться в стані конфлікту, в процесі функціонування має забезпечувати вимоги цілісності поведінки, що обумовлено варіантами рішення задачі конфлікту, які синтезуються. Тобто рішення конфлікту, які синтезуються та обираються, повинні бути такими, що можуть бути здійснені відповідно технічних властивостей ОУ.

Таким чином, при вирішенні конфлікту взаємодії об'єктів в ПС стратегії управління повинні бути можливими до реалізації в ПР, який враховує можливість здійснення обраного рішення із синтезованої множини стратегій за умов забезпечення цілісності і цілеспрямованості ОУ.

СІУ ОУ, структурна схема якої наведена на рис. 1.7, забезпечує реалізацію підходи щодо синтезу та вибору рішень конфлікту, а саме підсистему синтезу цілі та рішень.

На структурній схемі СІУ ОУ всі похідні дані про ОС, ОУ та ПС надходять в модуль приймання та перетворення похідних даних, а саме до функціональних складових апаратно-програмних засобів введення похідних даних. Зазначені апаратно-програмні засоби включають до свого складу: датчики сигналів щодо стану зовнішнього середовища, датчики сприймання даних про ОС, датчики характеристик переміщення ОУ, датчики контролю стану ОУ, датчики та засоби контролю за функціональним станом ОУ та систем СІУ, перетворювачі похідних даних в цифрові (бінарні) значення похідних даних та параметрів, засоби передачі похідних даних.

Обчислювальне середовище попередньої обробки похідних даних модуля приймання та перетворення похідних даних фактично є апаратно-програмним комплексом, який, як правило, є окремою функціональною одиницею в складі СІУ. Програмні засоби модуля призначені для розпізнавання та виділення ОС, супроводження переміщення об'єкту при наявності або відсутності даних про ОС, фільтрацію похідних сигналів (даних) та формування даних про місце знаходження ОС в ПС.

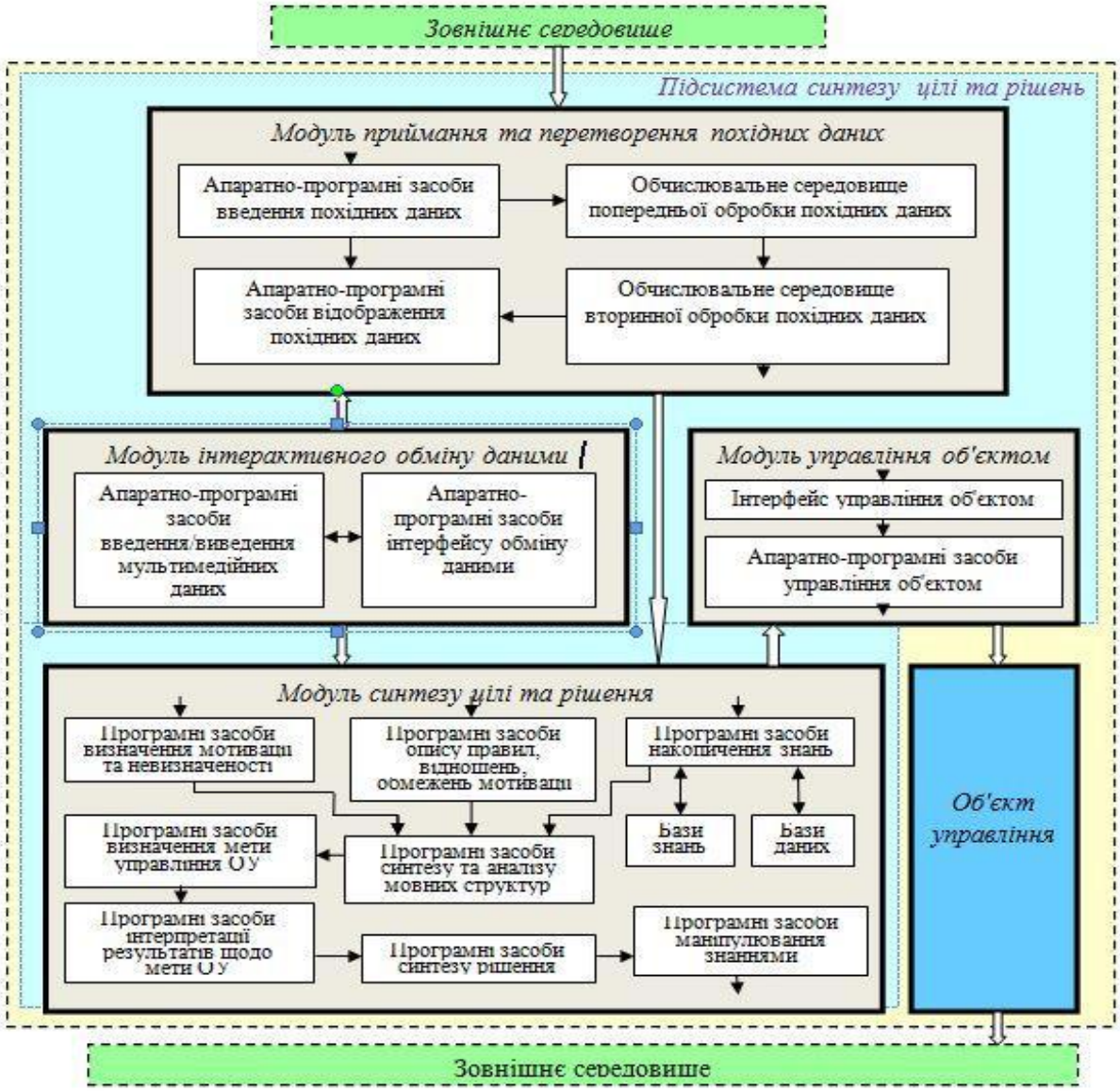


Рис. 1.7. Структурна схема СІУ ОУ

Обчислювальне середовище вторинної обробки похідних даних модуля приймання та перетворення похідних даних призначено для розрахунку векторів швидкості переміщення та прискорення ОУ та ОС, невизначеності при супроводженні переміщення ОС, розрахунок прогнозної траєкторії переміщення ОС.

Апаратно-програмні засоби відображення похідних даних модуля приймання та перетворення похідних даних здійснюють підготовку вихідних даних для відображення засобами модуля інтерактивного обміну даними.

До складу апаратно-програмних засобів модуля інтерактивного обміну даними входить модуль апаратно-програмних засобів введення/виведення мультимедійних даних та апаратно-програмних засобів інтерфейсу обміну даними. Модулі призначені для забезпечення мультимедійних функцій інтерактивного обміну похідною та проміжною інформацією в процесі синтезу та вибору рішення, визначення мети переміщення ОУ, уточнення характеристик взаємодії конфліктуючих об'єктів, обмежень ПС, а також інтерфейсу взаємодії з СІУ.

Для технічної ергатичної системи модуль інтерактивного обміну даними забезпечує виконання вимог до інтерфейсу взаємодії ОПР з ОУ [49, 50, 102, 103, 104, 149, 151, 152].

Модуль синтезу цілі та рішення ІП СІУ ОУ (рис. 1.8) забезпечує синтез стратегій оптимального управління ОУ в умовах конфлікту, обмежень та невизначеностей.

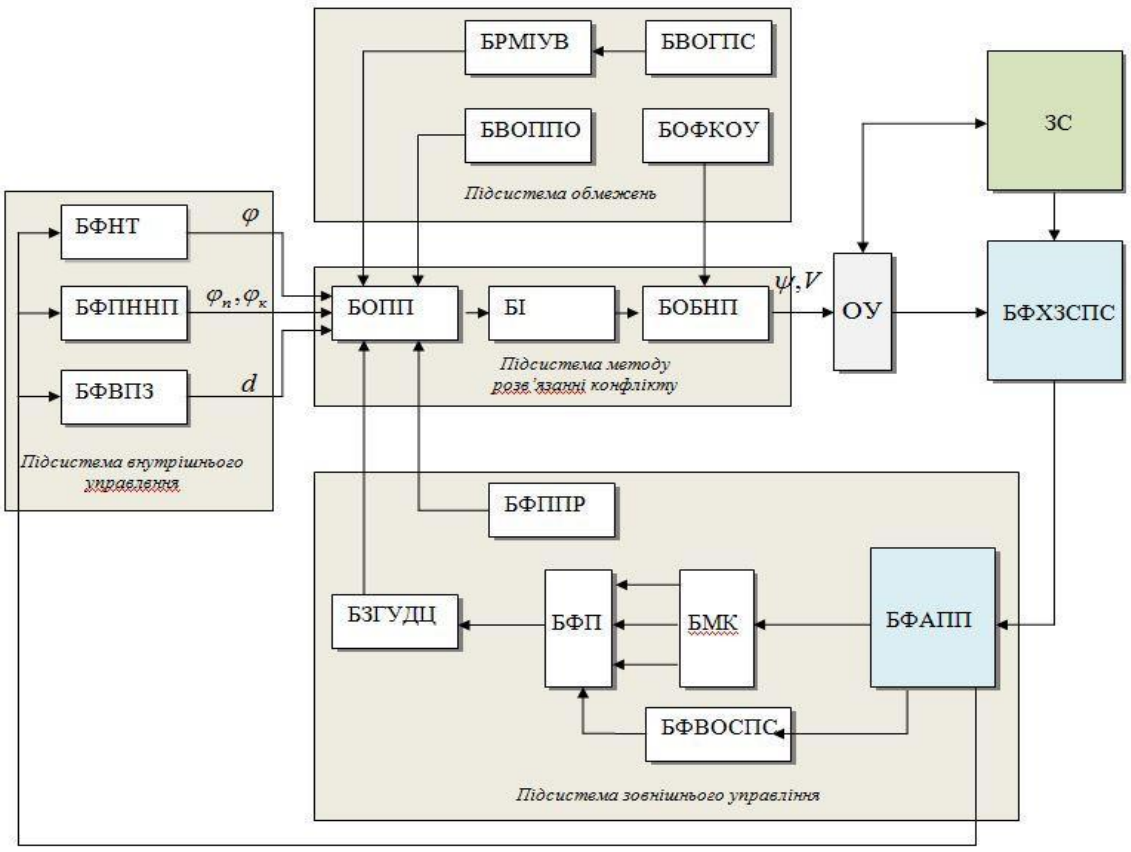


Рис. 1.8. Структурна схема функціонування інформаційної технології модуля синтезу цілі та рішення ІП СІУ ОУ

Структурна схема влючає блоки: БФНП – блок формування напрямку траєкторії; БФПННП – блок формування параметрів напрямнів небезпечного переміщення; БФВПЗ – блок формування відстані припустимого зближення; БОПП – блок обчислення параметрів переміщення; БІ – блок інтегрування; БОБНП – блок обчислення безпечного напрямку переміщення; ЗС – зовнішнє середовище; БРМІУВ – блок реалізації методу інтегрального усікання варіантів; БВОГПС – блок введення даних про обмеження і границі простору спостереження; БВОППО – блок введення обмежень параметрів переміщення об’єктів; БОФКОУ – блок обмежень фазових координат об’єкту управління; БФППР – блок формування параметрів простору рішення; БЗГУДЦ – блок забезпечення гарантовано оціненого управління досягнення цілі переміщення; БФП - блок формування пріоритетів при переміщенні ОУ; БФВОСПС – блок формування відомостей про характеристики переміщення об’єктів в просторі спостереження; БМК – блок моделювання конфлікту; БФАПП – блок формування і аналізу повідомлень про події; БФХПСЗС – блок формування відомостей про стан зовнішнього середовища і простору спостереження.

Структурна схема визначає основні потоки інформації та управляючих впливів, які забезпечують процес функціонування інформаційної технології.

Модуль синтезу цілі та рішення вміщує програмні компоненти, бази даних та бази знань, які реалізовують алгоритми синтезу стратегій управління та вибору рішень СІУ. Похідні дані для функціонування модуля формують програмні засоби (ПЗ) визначення мотивації та невизначеності, ПЗ опису правил, відношень, обмежень мотивації та ПЗ накопичення знань.

ПЗ визначення мотивації на похідних даних від модуля інтерактивного обміну даними у відповідності до заданої цілі переміщення ОУ, прогнозу переміщення ОС, заданої системи обмежень ПС та невизначеності переміщення ОС дозволяють розпізнати стан конфлікту взаємодії об'єктів в ПС. Відповідно до моделі опису конфлікту для ОС розраховуються значення припустимого зближення ОУ з ОС, які є засадними при синтезі ланцюжків стратегій вирішення конфлікту та вибору оптимального рішення.

ПЗ опису правил, відношень, обмежень мотивації фактично основні засадні підходи щодо розрахунку інформаційних множин та ознак приналежності елементу траєкторії напрямку переміщення ОУ, ПР, а також синтезу та аналізу структури та змісту описів та ланцюжків синтезованих стратегій описів взаємодії ОУ з варіативною множиною ОС за умов обмежень та невизначеностей в ПР. Правила, відношення і обмеження мотивації в такому сенсі поділяються на дві основні групи: сталі і такі, що керуються даними. Останні є сутністю функціонування системи синтезу стратегій управління з використанням мовних абстракцій та синтезованих ланцюжків альтернативних стратегій для використання в алгоритмах евристичного пошуку.

ПЗ накопичення знань з відповідними базами даних (БД) та структурованими базами знань (БЗ) дозволяють здійснювати синтез стратегій за інформаційним образом ПС. В такому разі можна оцінювати стратегію рішення як можливу з певною ймовірністю отримання позитивного результату. Слід зауважити, що ПЗ накопичення знань не дозволяють здійснювати продукцію нових знань. Отримані в процесі функціонування СІУ знання в вигляді правил, відношень та словників додаються в процесі використання в відповідні бази даних та бази знань, що дозволяє узагальнювати досвід використання СІУ в окремих галузях діяльності та при вирішенні конфліктів в технічних системах.

За результатами функціонування ПЗ визначення мотивації та невизначеності, ПЗ опису правил, відношень, обмежень мотивації та ПЗ накопичення знань в автоматичному режимі формують повідомлення на внутрішній мові СІУ. Ці повідомлення є похідними для ПЗ синтезу та аналізу мовних структур.

ПЗ синтезу та аналізу мовних структур вміщують засоби побудови таблиці лексем, лексичний аналізатор та синтаксичний аналізатор. Семантичний аналізатор для аналізу та синтезу мовних ланцюжків СІУ не

потрібен. Структура запису лексем, ідентифікаторів та утворення речень опису взаємодії об'єктів в ПС визначається за формою Бекуса-Наура.

Визначення мети управління здійснюють ПЗ синтезу ланцюжків цілеспрямованого управління ОУ, як альтернативних рішень конфлікту переміщення ОУ з врахуванням прогнозу та динаміки переміщення ОС в ПР.

ПЗ інтерпретації результатів щодо мети (цілі) переміщення ОУ дозволяють відібрати тільки ті синтезовані стратегії управління ОУ, за якими може бути досягнута мета (ціль) його переміщення в ПР, виходячи з наявних можливостей гарантованого управління згідно критерію оптимальності та правила зупинки.

Обрати та оцінити обране рішення по управлінню ОУ серед множини альтернативних забезпечує ПЗ синтезу рішення згідно критерію (функції ціни) та правила зупинки. Для ТЕС ОПР може прийняти рішення, яке не відповідає критерію оптимальності.

ПЗ визначення мети управління, інтерпретації результатів щодо мети (цілі) переміщення ОУ та синтезу рішення є частиною алгоритму методу синтезу стратегій щодо управління ОУ в умовах конфлікту, обмежень та невизначеностей.

Врахуванням досвіду, який накопичився впродовж використання СІУ при синтезі рішення щодо вирішення конфлікту, забезпечується ПЗ маніпулювання знаннями.

За результатами функціонування системи синтезу цілі та рішень за допомогою апаратно-програмних засобів модуля управління об'єктом здійснюються функції управління переміщенням ОУ по траєкторії в ПР згідно обраній стратегії рішення конфлікту.

В загальному випадку до складу СІУ ТС входить ПСЦР та ОУ, які утворюють систему з квазілінійною моделлю управління (рис. 1.9).

Для ТС сигнал управління СІУ передається до ОУ. Збурення $D(s)$, які виникають в ПС та середовищі функціонування ТС, впливають на процес управління ОУ.

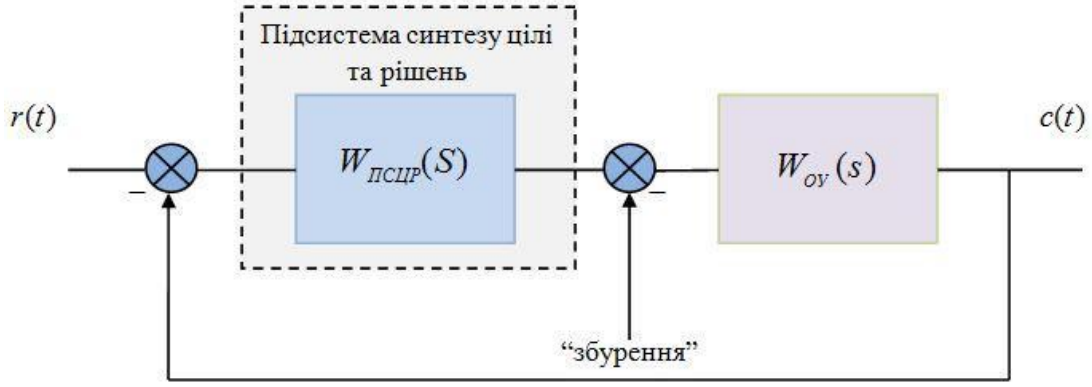


Рис. 1.9. Система з квазілінійною моделлю інтелектуального управління об'єктом

Слід зазначити, що не кожен регулятор $W_{ПСЦР}(S)$ може стабілізувати СІУ. Проте для забезпечення стійкості регулятор $W_{ПСЦР}(S)$ має бути стабілізуючим.

Для СІУ визначимо передатну функцію

$$W(S)=\frac{W_{ПСЦР}(S)W_{ОУ}(S)}{1+W_{ПСЦР}(S)W_{ОУ}(S)}. \qquad (1.12)$$

В співвідношенні (1.12) підсистема синтезу цілі та рішення (ПСЦП) входить нелінійно, що значно ускладнює аналіз та синтез СІУ. Представимо співвідношення (1.12) у вигляді

$$W(S)=Q(S)W_{ОУ}(S), \qquad (1.13)$$

де

$$Q(S)=\frac{W_{ПСЦР}(S)}{1+W_{ПСЦР}(S)W_{ОУ}(S)}. \qquad (1.14)$$

В такому разі співвідношення (1.13) виглядає як передатна функція послідовного з'єднання ОУ $W_{ОУ}(S)$ та «регулятора» $Q(S)$.

За співвідношенням (1.14) функція $Q(S)$ має бути стійкою, що забезпечить стійкість передатної функції (1.13).

В такому разі множина параметризуючих стабілізуючих регуляторів за параметризацією Юла має вигляд

$$W_{ПСЦР} = \frac{Q(S)}{1 - Q(S)W_{ОУ}(S)}. \quad (1.15)$$

За наявності людини-оператора [104] СІУ перетворюється в технічну ергатичну систему управління (рис. 1.10).

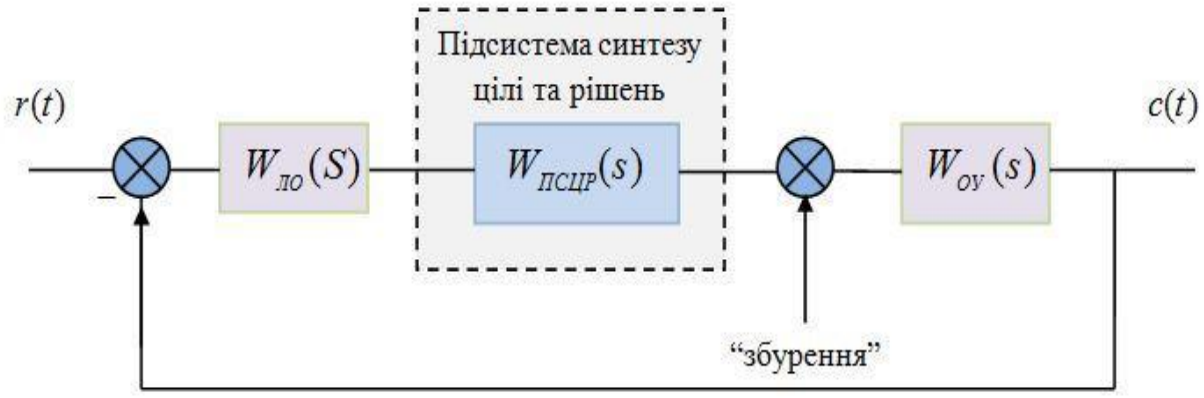


Рис. 1.10. Технічна ергатична система управління

Моделі (1.12) – (1.15) добре функціонують в часовій і частотній областях.

Для ТЕС

$$W(S) = \frac{W_{ЛО}W_{ПСЦР}(S)W_{ОУ}(S)}{1 + W_{ЛО}W_{ПСЦР}(S)W_{ОУ}(S)}. \quad (1.16)$$

З врахуванням співвідношення (1.13) та (1.16)

$$Q(S) = \frac{W_{ЛО}W_{ПСЦР}(S)}{1 + W_{ЛО}W_{ПСЦР}(S)W_{ОУ}(S)}. \quad (1.17)$$

Таким чином множина параметризуючих стабілізуючих регуляторів за параметризацією Юла для ЕСУ має вигляд

$$W_{ПСЦР} = \frac{Q(S)}{(1 - Q(S)W_{ОУ}(S))W_{ЛО}}. \quad (1.18)$$

Для моделей (1.17) та (1.18) припускається, що ОПР діє згідно певних критеріїв оптимальності в умовах, які накладаються обмеженнями.

Модель (1.17) відтворює залежність вихідного сигналу від вхідного в припущенні, що ОПР має власні правила щодо оптимальності управління ОУ.

В такому разі алгоритм функціонування ПСЦР має забезпечувати вимоги функціонального гомеостазису ОПР при синтезі та реалізації стратегій гарантованого управління ОУ в просторі функціональних станів, які можуть бути здійснені.

СІУ ТС (рис. 1.9, рис. 1.10) здійснює управління за принципом відхилення змінної, яка є керуємою. Від'ємний гнучкий зворотній зв'язок створює замкнений контур керування. Таким чином на ОУ подається керуючий вплив, який є пропорційним різниці між значенням вхідної змінної та її заданим значенням.

В такому разі СІУ є повністю керованою, якщо вона може бути переведена з будь-якого початкового стану $r_0(t)$ в стан $r_1(t)$ в довільний момент часу шляхом здійснення кусково-безперервного впливу $U(t) \in [t_0, t_1]$.

Повна спостережність СІУ забезпечується тим, що всі змінні стану $r(t)$ можна визначити за вихідним впливом $c(t)$.

В даному випадку замкнена СІУ представлена ОУ та регулятором. Слід зазначити, що замкнення СІУ призводить до зміни знаменника передатної функції в співвідношеннях (1.12) і (1.16), а саме полюсів системи.

Для аналізу стійкості СІУ доцільним є використання діаграми Ніколса, яка дозволяє представити вихідний сигнал лінійної стаціонарної динамічної системи в частотній області (частотний відгук) як реакцію на вхідний сигнал, що розрізняється по частоті і постійний по амплітуді у вигляді графіку в декартових координатах. На такому графіку частота виступає параметром кривої, фаза і амплітуда системи на заданій частоті представляється абсцисою і ординатою кожної точки характеристики. По суті такий графік об'єднує на одній площині логарифмічну амплітудно-частотну характеристику (ЛАЧХ) і фазову частотну характеристику (ФЧХ) СІУ.

В основі визначення стійкості СІУ розглядається модель у вигляді ланки, яка охоплена від'ємним зворотним зв'язком. При цьому припускається можливість входження СІУ в автоколивання (коливальна межа стійкості).

Умовою автоколивань є наявність додатнього зворотного зв'язку. В такому разі коефіцієнт підсилення в прямому напрямку проходження сигналу має бути не нижчий одиниці. Фаза вихідного сигналу (визначена ФЧХ) через ланку від'ємного зворотного зв'язку подається назад на вхід, при цьому "запасом по фазі" називається додаткове зрушення по фазі, яке має бути на виході СІУ, для забезпечення додатнього зворотного зв'язку. Коефіцієнт передачі в прямій гілці описується АЧХ. При цьому частота, якій відповідає одиничне посилення, визначається як "частота зрізу". На ЛАЧХ частота зрізу визначається як точка перетину характеристики з віссю абсцис. Графічно запас по фазі визначається як різницю між фазою рівної π радіан (180°) і фазою на частоті зрізу (умова одиничного коефіцієнта в прямій гілці).

Аналіз стійкості з використанням діаграми Боде є простим і зручним. Діаграма Боде є ЛАФЧХ представлення частотного відгуку лінійної стаціонарної системи в логарифмічному масштабі, яка будується у вигляді двох графіків - ЛАЧХ і ФЧХ, які зазвичай розташовуються один під одним.

Для СІУ важливим є також показник запасу стійкості по модулю, фазі та полоса пропускання. Визначення цих покажчиків можливе за діаграмою Боде. Слід зазначити, що вплив запізнення за часом є істотним щодо стійкості та якості СІУ.

Для уточнення параметрів регулятора СІУ та отримання перехідних характеристик по заданих каналах регулювання необхідно скласти алгоритмічну структуру системи регулювання враховуючи використання в системі ПІД-регулятора, який працює за певним законом, і має власну перехідну характеристику. При цьому регулятор повинен формувати ПІД-закон регулювання для отримання необхідного за завданням характеру перехідного процесу. Настройки регулятора, яким є алгоритм синтезу стратегій управління ОУ, визначається за результатами аналізу діаграм Ніколса та Боде, а також характером перехідного процесу СІУ.

1.4. Формулювання наукової проблеми та постановка задач дослідження

Виходячи з аналізу стану проблеми та методів і підходів до вирішення задачі управління об'єктом при розв'язанні конфлікту можна зробити висновки щодо постановки задачі дослідження:

- методи системного аналізу є засадними при логічному та послідовному підході до проблеми синтезу і прийняття рішень за умов конфлікту та невизначеності;

- ефективність вирішення задач конфлікту за допомогою методів системного аналізу визначається структурою вирішуваних задач;

- усі задачі конфлікту, як і інші проблеми, що вирішуються методами та підходами системного аналізу, поділяються на три класи: добре структуровані, неструктуровані та слабо структуровані);

- для вирішення добре структурованих кількісно визначених проблем системного аналізу використовується методологія дослідження операцій, що виходить з наявності детермінованої математичної моделі досліджуваного процесу (наприклад, методів математичного програмування, теорії масового обслуговування, теорії ігор, тощо) та застосуванні методів синтезу та вибору оптимальної стратегії управління цілеспрямованими діями щодо вирішення задачі конфлікту;

- для вирішення слабо структурованих проблем використовується методологія системного аналізу, системи підтримки прийняття рішень, що за процедурою включає такі етапи, як формулювання проблемної ситуації, визначення цілей, визначення критеріїв досягнення цілей, побудова моделей досліджуваного процесу для синтезу та вибору стратегії вирішення проблеми, вибір оптимального припустимого варіанту рішення, прийняття рішення, підготовка рішення до реалізації, управління ходом реалізації рішення, перевірка ефективності рішення;

- системний підхід є не стільки методом вирішення задачі конфлікту, скільки методом постановки завдань і є вищим за предметний методом пізнання проблеми та відповідає таким основним принципам як цілісність (дозволяє розглядати одночасно систему як єдине ціле і в той же час як підсистему для вищестоящих рівнів), ієрархічність будови (наявність менше двох елементів, розташованих на основі підпорядкування елементів нижчого рівня елементам вищого рівня, наприклад, керуюча та керована підсистема), структуризація (дозволяє аналізувати елементи системи і їх взаємозв'язок у рамках конкретної організаційної структури), множинність (дозволяє використати множину кібернетичних, економічних і математичних моделей для опису окремих елементів і системи в цілому), системність (властивість об'єкту мати усі ознаки системи);

- основою методології дослідження операцій є системний підхід до поставленої проблеми та методу її вирішення за умов дотримання принципу оптимальності та включає теорію графів, теорію ігор та мережеве планування і управління;

- як розділ дослідження операцій теорія ігор розглядає взаємодію математичних об'єктів (гравців) з конфліктуючими цілями, кожний з яких має набір стратегій та показників виграшів щодо кожної комбінації стратегій рішень і має недоліки (наявність декількох принципів оптимальності при рішенні однієї і тієї ж задачі, що викликає множинність варіантів концепції оптимальності; недостовірність деяких допущень, прийнятих при описі гри, яка пов'язана із складністю оцінки і перевірки цих допущень; ігрове моделювання вимагає значних витрат часу на побудову моделі, збір інформації про стан зовнішнього середовища) та достоїнства (наочність наслідків рішень, що приймаються; можливість повторення досвіду (ігри) зі зміною початкових даних; можливість зміни масштабу часу, що дозволяє отримувати результати гри за короткий проміжок часу; можливість застосування теорії ігор при дослідженні систем управління різного рівня ієрархії);

- практичне вирішення задач вибору з метою запобігання конфлікту використовує моделі і процедури, які не завжди теоретично обґрунтовується в логічних засадах вибору, аксіоматиці, принципах раціональності, узгодженні, теоретичних і практичних аспектів дотримання принципів синтезу ергатичних систем керування і запобігання конфлікту;

- процес прийняття рішень має на меті досягнення усвідомлюваного результату раціонального або ірраціонального вибору альтернатив на основі нормативної теорії (описує раціональний процес прийняття рішення) та дескриптивної теорії (описує досвід прийняття рішень);

- прийняття рішень в умовах невизначеностей (стохастичної, поведінкової, природної, апріорної) зводиться до звуження початкової множини альтернатив рішень, а вірогідність є тільки однією з альтернатив прийняття рішення;

- методи ситуаційного управління прийнятні для управління функціонуванням ТС або ТЕС на основі результатів інтерпретації ситуацій, прогнозування, планування, синтезу і вибору рішень, щодо управління для процесів функціонування об'єктів, які не можуть бути повністю формально описані з використанням різних математичних моделей;

- при ситуаційному управлінні макромодель управління виникає в результаті функціонування семіотичної мови імітації за допомогою апарату формальних граматик трансформаційного типу, в процесі якої формуються структурні моделі стосунків між елементами поточних ситуацій, а також здійснюється узагальнення структурних моделей, а також на основі узагальнень синтезуються, вибираються та оцінюються рішення по управлінню об'єктом;

- метод ситуаційного управління дозволяє ефективно вирішувати завдання управління складними системами та об'єктами, відносно яких застосування класичних методів теорії управління є складним;

- формальна семіотична мова ситуаційного управління не вміщує морфології у зв'язку з використанням слів, які вміщують одну літеру, а

сутність синтаксису визначається поняттями правильно складеної фрази на цій мові, що обумовлює те, що синтаксично правильно складена фраза на семіотичній мові семантично може бути помилковою;

- методи синтезу та вибору варіантів рішення задачі конфлікту за умов невизначеностей дозволяють знаходити рішення в умовах опуклої системи обмежень в евклідовому безперервному просторі, що обмежує коло проблем, які можуть бути вирішені з використанням існуючих методів та підходів до розв'язання задач конфлікту.

Проведений аналіз показав, що існують методи і моделі вирішення задачі конфлікту взаємодії об'єктів в ПС за умов невизначеностей, але вони мають ряд недоліків, а саме:

- побудова систем управління об'єктом в стані конфлікту та невизначеностей базується на методах системного аналізу, зокрема дослідження операцій та ситуаційного управління за умов опуклості та нерозривності ПС при обмеженій кількості об'єктів спостереження;

- методи оптимального рішення розглядаються для малої розмірності (не більше трьох);

- евристичні методи з комп'ютерним моделюванням, як правило використовують нейронні моделі для синтезу рішень, але побудова самих нейронних моделей є предметом окремих досліджень і на поточний час автоматизація їх синтезу за технологіями самонавчання не вирішені;

- методи проб і помилок, методи гілок і границь, методи перебору варіантів, методи теорії ігор призводять до значних об'ємів обчислювальних процедур і можуть призвести до прояву "ефекту доміно", що є наслідком "прокляття розмірності".

Тобто аналіз і синтез систем управління та рішення конфлікту здійснюється для систем з низькою розмірністю, з опуклими обмеженнями, в замкненому евклідовому просторі.

Для систем великої розмірності, довільними обмеженнями, розривністю значення змінних і параметрів математичної моделі, невизначеністю поведінки

(переміщення) конфліктуючих об’єктів доцільно використовувати методи комбінаторного типу, якими є методи та підходи динамічної дискретної оптимізації., які дозволяють отримати алгоритми поліноміального класу.

В такому разі при ситуаційному управлінні:

- виникає макромодель управління для ІП СІУ при вирішенні еквівалентної задачі дискретної динамічної оптимізації;

- в результаті застосування семіотичної мови та моделі імітації за допомогою апарату формальних граматик формуються структурні моделі правил та відношень між конфліктуючими елементами ТС або ТЕС;

- здійснюється узагальнення структурних моделей, декомпозиція і виділення кінцевих рахуємих множин припустимих значень параметрів управління ОУ;

- простір множин припустимих значень параметрів управління може бути довільним і розривним при забезпеченні гарантованого управління ОУ при рішенні конфлікту;

- забезпечує рішення конфлікту для систем великої розмірності в умовах обмежень, невизначеностей і варіативної множини ОС в класі поліноміальних алгоритмів, що дозволяє отримати поліноміальний час рішення задачі дискретної динамічної оптимізації.

Модель дозволяє спростити вирішення проблем використання інженерії знань при побудові моделей, призначених для прийняття ситуаційних рішень з управління станом ТС і ТЕС в штатних ситуаціях та в стані конфлікту.

Вирішення конфлікту в умовах невизначеності, довільних обмежень ПС та варіативної множини ОС пропонується досягти шляхом визначення потрібних базисів уявлення процесів, систем та їх складових в межах загальносистемного розгляду властивостей предметної області. Таким чином, при створенні СІУ необхідно представити загальносистемні вимоги складу, змісту та структури логічних взаємопов'язаних задач, які вирішуються в процесі функціонування СУ з максимальним ступенем декомпозиції та деталізації.

Вирішення цієї задачі дозволяє конкретизувати властивості атрибутів ситуаційної моделі (СМ), які використовуються на різних етапах процесу управління станом ОУ, а саме мети, цілей, ситуацій, станів, рішень, їх структури та семантичних відношень між ними.

Властивості СМ, які використовуються в СУ, відображають властивості функціональних складових та структури СІУ ОУ. Для опису властивостей процесів прийняття ситуаційних рішень пропонується виявити та описати властивості загальносистемних закономірностей причинно-наслідкових процесів керуємого функціонування ОУ. При цьому мають бути визначені правила представлення атрибутів СМ в множині значень параметрів, які відображують матеріальні процеси та які відбуваються під час функціонування ОУ, а також існуючі відношення між вжитими семантичними поняттями. Тобто мають бути відображені взаємозв'язки між СМ та моделями процесів управління об'єктом в штатних і позаштатних ситуаціях та в умовах конфлікту.

При аналізі та синтезі СІУ ОУ в системному плані враховується те, що принципово СІУ є динамічною системою, при тому як чисто автоматичною так і ергатичною.

Найважливішою є проблема створення методології, методів і алгоритмів рішення та високоякісного управління ОУ в стані конфлікту за умов обмежень, невизначеностей при взаємодії з варіативною множиною ОС для складних процесів, включаючи динамічні нелінійні та збурюємі.

Аналіз показав, що для вирішення проблеми аналізу та синтезу систем рішення конфлікту за умов обмежень, невизначеностей і варіативної множини ОС необхідно: розробити методи модельного представлення систем з різним рівнем інтегрованості та розподіленості; розробити єдину системну методологію аналізу та синтезу стратегій оптимального рішення конфлікту для таких систем; провести дослідження методів побудови простору рішення конфлікту; провести дослідження властивостей розробленої методології, методів і алгоритмів оптимального рішення конфлікту та їх ефективності.

Для досягнення мети дисертаційного дослідження запропонована структурно-логічна схема (рис. 1.11).

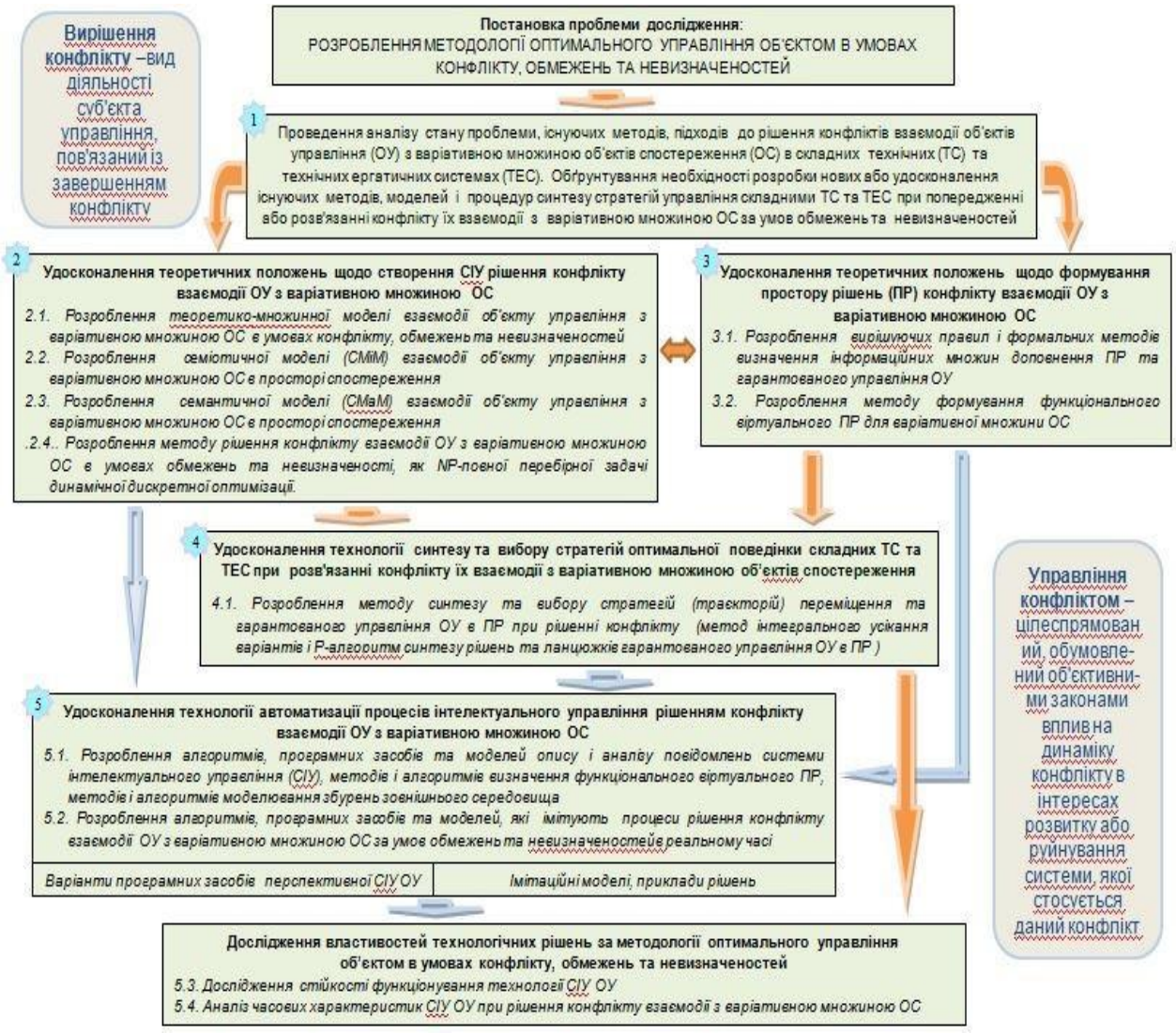


Рис. 1.11. Структурно-логічна схема досліджень

В дисертації запропоновано проводити дослідження систем рішення задач конфлікту на функціональному та структурному рівнях.

На функціональному рівні запропоновано розглядати динамічний процес рішення задачі конфлікту і всі його характеристики безвідносно до структури управління на рівні опису самого процесу в цілому, не розділяючи його на компоненти.

При цьому на структурному рівні функціональні рішення розглядаються з точки зору їх реалізації в СІУ ОУ.

На функціональному рівні запропоновано досліджувати функціональну модель конфлікту методами ситуаційного управління з використанням автоматних граматик.

В такому разі всі підсистеми ОУ можна описати з позицій єдиного математичного модельного уявлення, врахувавши їх основні характеристики, а саме - багатозв'язність, неопуклість, розривність характеристик, множинність, невизначеність, тощо.

При вирішенні задачі управління об’єктом в умовах конфлікту на структурному рівні ставиться задача синтезу стратегій та вибору оптимального управління при переміщенні ОУ за синтезованою траєкторією в просторі рішень за умови його динамічної змінюваності та наявності гарантованого управління.

Мета дослідження: розробити методологію, моделі та теоретичні засади формального опису та розрахунку простору рішень, методів синтезу стратегій оптимального управління об’єктом і вибору рішень по гарантованому безконфліктному управлінню цілеспрямованим переміщенням ОУ в багатозв'язному, неопуклому, небезперевному ПС, що містить жорсткі, гнучкі, статичні та динамічні обмеження для варіативної варіативної множини ОС в реальному масштабі часу за умов обмежень і невизначеностей.

Для досягнення поставленої мети слід розв'язати таки завдання:

1. Розробити концепцію і методи аналізу та синтезу управління об’єктами, включаючи складні нелінійні динамічні та віртуальні, у стратегіях прямого керування, що реалізуються засобами комп'ютерних систем, які максимально використовують весь потенціал функціональних і структурних можливостей управління складними процесами конфлікту взаємодії ОУ з варіативною множиною ОС в умовах обмежень та невизначеностей, як NP-повної перебірної задачі динамічної дискретної оптимізації.

2. Удосконалити теоретичні положень щодо створення СІУ рішення конфлікту взаємодії ОУ з варіативною множиною ОС за рахунок:

- розроблення теоретико-множинної моделі взаємодії об’єкту управління

з варіативною множиною ОС в умовах конфлікту, обмежень та невизначеностей;

- розроблення семіотичної моделі взаємодії ОУ з варіативною множиною ОС в ПС;

- розроблення семантичної моделі взаємодії ОУ з варіативною множиною ОС в ПС;

- розроблення методу рішення конфлікту взаємодії ОУ з варіативною множиною ОС в умовах обмежень та невизначеності, як *NP*-повної перебірної задачі динамічної дискретної оптимізації.

3. Удосконалити теоретичні положення щодо формування простору рішень (ПР) конфлікту взаємодії ОУ з варіативною множиною ОС за рахунок:

- розроблення вирішуючих правил і формальних методів визначення інформаційних множин доповнення ПР та гарантованого управління ОУ;

- розроблення методу формування функціонального віртуального ПР для варіативної множини ОС

4. Удосконалити технологію синтезу та вибору стратегій оптимальної поведінки складних ТС та ТЕС при розв'язанні конфлікту їх взаємодії з варіативною множиною об’єктів спостереження за рахунок розроблення методу синтезу та вибору стратегій (траєкторій) переміщення та гарантованого управління ОУ в ПР при рішенні конфлікту (метод інтегрального усікання варіантів і Р-алгоритм синтезу рішень та ланцюжків гарантованого управління ОУ в ПР).

5. Удосконалити технологію автоматизації процесів інтелектуального управління рішенням конфлікту взаємодії ОУ з варіативною множиною ОС за рахунок:

- розроблення алгоритмів, програмних засобів та моделей опису і аналізу повідомлень системи інтелектуального управління (СІУ), методів і алгоритмів визначення функціонального віртуального ПР, методів і алгоритмів моделювання збурень зовнішнього середовища;

- розроблення алгоритмів, програмних засобів та моделей, які імітують

процеси рішення конфлікту взаємодії ОУ з варіативною множиною ОС за умов обмежень та невизначеностей в реальному часі.

6, Дослідити властивості технологічних рішень за методології оптимального управління об’єктом в умовах конфлікту, обмежень та невизначеностей, а саме:

- дослідити стійкість функціонування технології СІУ ОУ;

- проаналізувати часові характеристики СІУ ОУ при рішення конфлікту взаємодії з варіативною множиною ОС.

7. Розробити математичне та алгоритмічне забезпечення:

- аналізу та опису властивостей простору спостереження динамічних та віртуальних систем, включаючи аналіз можливості реалізації управління конфліктом при максимальному використанні всього наявного потенціалу управління об'єктом в умовах невизначеності поведінки учасників конфлікту та варіативної множини ОС;

- методу синтезу управління складним об'єктом з використанням мінімально-перебірного методу синтезу стратегій рішення конфлікту взаємодії об'єктів в просторі спостереження в умовах невизначеностей їх поведінки для відкритої та варіативної множини ОС в реальному масштабі часу.

8. Виконати імітаційні експерименти щодо підтвердження достовірності, працездатності та ефективності розроблених методів на практичних задачах.

Висновки до першого розділу

Дисертаційна робота виконана за спеціальністю 05.13.06 - інформаційні технології.

Із формули спеціальності використовуються: охоплює теоретичні та методологічні основи й інструментальні засоби створення та використання інформаційних технологій у різних галузях людської діяльності; розроблення принципів оптимізації та моделей і методів прийняття рішень за умов

невизначеності при створенні автоматизованих систем різноманітного призначення; дослідження закономірностей побудови інформаційних комунікацій і розроблення теоретичних і прикладних засад побудови та впровадження інтелектуальних інформаційних технологій для створення новітніх систем накопичування, переробки, збереження інформації та систем управління.

Із напрямів досліджень використовуються:

- розроблення наукових і методологічних засад створення та застосування інформаційних технологій та інформаційних систем для автоматизованої переробки інформації й управління;

- розроблення інформаційних технологій для аналізу та синтезу структурних, інформаційних і функціональних моделей об'єктів і процесів, що автоматизуються;

- дослідження та побудова інформаційних технологій для розроблення та впровадження баз і сховищ даних, баз знань і систем комп'ютерної підтримки рішень в автоматизованих системах і мережах;

- розроблення теоретичних і прикладних основ побудови інформаційних технологій для автоматизації функціональних завдань керування, аналізу й оцінювання ефективності автоматизованих систем переробки інформації й управління;

- розроблення інформаційно-пошукових і експертних систем обробки інформації для прийняття рішень, а також знання орієнтованих систем підтримки рішень в умовах ризику та невизначеності як інтелектуальних інформаційних технологій;

Виходячи з викладеного вище, основною метою дисертаційної роботи є розробка та обґрунтування методу вирішення задачі конфлікту взаємодії об’єкту управління з варіативною множиною об'єктів спостереження в інтегрованому просторі спостережень, станів та рішень за умов невизначеностей їх поведінки в реальному масштабі часу, що забезпечує

підвищення швидкодії і стійкості функціонування систем інтелектуального управління об’єктом.

В дисертаційній роботі має бути:

- розроблена, теоретично обгрутована та досліджена модель конфлікту взаємодії об’єктів в умовах обмежень, невизначеності та відкритої варіативної множини учасників;

- розроблені методи формального опису та побудови інформаційних множин простору рішення конфлікту для об’єкту управління;

- розроблений, обґрунтований та досліджений методу синтезу стратегій управління об’єктом при вирішенні конфлікту взаємодії з відкритою варіативною множиною об'єктів (суб'єктів) за умов забезпечення гарантованого результату в максимально розширеній області стану учасників конфлікту з врахуванням невизначеності та обмежень;

- розроблені алгоритми та імітаційні моделі для дослідження методів побудови інформаційних множин простору рішення конфлікту;

- розроблені алгоритми та програмні засоби для дослідження методу синтезу стратегій рішення конфлікту взаємодії об’єктів в умовах обмежень, невизначеності та відкритої варіативної множини учасників;

- проведені імітаційні експерименти щодо підтвердження ефективності розроблених методів та алгоритмів на практичних задачах рішення конфлікту.

Основні результати розділу опубліковані у роботах [132, 133, 137, 146, 151, 172].

РОЗДІЛ 2. ШЛЯХИ СТВОРЕННЯ СИСТЕМИ ІНТЕЛЕКТУАЛЬНОГО УПРАВЛІННЯ РІШЕННЯМ КОНФЛІКТУ ВЗАЄМОДІЇ ОБ'ЄКТУ УПРАВЛІННЯ З ВАРІАТИВНОЮ МНОЖИНОЮ ОБ'ЄКТІВ СПОСТЕРЕЖЕННЯ В УМОВАХ ОБМЕЖЕНЬ ТА НЕВИЗНАЧЕНОСТЕЙ

2.1. Концепція топологічного ситуаційного аналізу та синтезу стратегій управління об'єктом в умовах конфлікту, невизначеності поведінки та варіативної множини об'єктів спостереження

В загальному випадку під конфліктом розуміють ситуацію, в якій кожна зі сторін намагається зайняти позицію несумісну з інтересами іншої сторони.

При аналізі конфліктів в технічних системах [12, 58, 102, 104, 132, 133, 137, 142, 143, 146, 151, 159, 160, 162, 164, 165] конфлікт визначається як взаємодія об'єктів, ТС або ТЕС, що мають несумісні цілі і способи їх досягнення, діяльність яких так чи інакше пов'язана з постановкою і рішенням завдань управління, з прогнозуванням і прийняттям рішень, а також з плануванням цілеспрямованих дій.

Дослідження та аналіз об'єктів (учасників) конфлікту здійснюється за методологією системного аналізу, як напряму методології наукового пізнання, в основі якого лежить розгляд об'єктів як систем. При дослідженні конфлікт розглядається як цілісне явище з певним розмаїттям типів зв'язків між його елементами, об'єктами, системами, зовнішнім середовищем, тощо, які мають бути зведені в єдину теоретичну концепцію, що різнобічно і глибоко відбиває реальні особливості явища конфлікту, яке досліджується.

При дослідженні конфліктів в технічних системах доцільним є застосування:

- системно-структурного аналізу конфлікту, як дослідження явища в цілому, яке складається з системи підструктур, які, у свою чергу, складаються з елементів, і в якості підсистем входить в систему більш високого рівня;

- системно-функціонального аналізу конфлікту, як дослідження з метою визначення всіх основних взаємозв'язків конфлікту із зовнішнім середовищем, в якому конфлікт розвивається, у виявленні характеру і способів впливу одних елементів і підструктур конфлікту на інші (біфуркації).

Для досліджень доцільним є використання інтегрально-топологічних методів аналізу складних нелінійних систем, які базуються на використанні математичних моделей опису властивостей динамічних процесів, пов'язаних з об'єктами як системами. При цьому під інтегрально-топологічними методами розуміють сукупність методів, які базуються на аналітико-комп'ютерних методах, методах нелінійної інтегральної інваріантності, що призначені для одержання топологічних структур просторів спостереження, поведінки та рішення для синтезу та вибору управлінь ОУ.

Надалі при дослідженні методів синтезу та вибору рішень по управлінню об'єктом в умовах невизначеності будемо розглядати оптимізаційні методи та структури ОУ, в яких ці методи управління конфліктом можуть бути реалізовані.

В якості методів дослідження застосуємо методи функціонально-структурного аналізу, методи опису топологічних простору пошуку, станів та рішень, методи ситуаційного управління щодо опису моделі взаємодії об'єктів в просторі спостереження та синтезу стратегій (рішень), а також вибору рішення в просторі рішень за умов конфлікту та невизначеності.

Інтегрально-топологічні методи дослідження та аналізу складних нелінійних систем в ПС дозволяють виявити характеристики взаємодії моделі ОУ та ОС, які включають параметри безпечного розв'язання конфлікту при синтезі та виборі рішень з врахуванням невизначеності ([131], [132], [133], [134], [135], [136], [137], [140, [141], [142], [143], [146], [151], [155], [156], [157], [158]) , а саме:

- значення припустимої відстані зближення ОУ з ОС;
- координати ОС в просторі спостереження;
- інтервали часу спостереження ОС в просторі спостереження;

- опис границі простору спостереження в просторі існування конфлікту (координати вершин ламаних ліній границь простору спостереження) ;

- координати цільової точки траєкторії переміщення ОУ в просторі спостереження;

- обмеження кінематичних характеристик ОУ та ОС;

- обмеження динамічних характеристик ОУ та ОС;

- цілі переміщення ОС.

Зазвичай вирішення задачі взаємодії ОУ з множиною ОС в ПС породжує збільшення розмірності моделі процесу управління і, як наслідок, поліноміальне збільшення складності адекватного до моделі процесу управління, тобто $N = \frac{(m+1)^2}{2}$, де N - кількість часткових задач управління, m - мірність вектору фазових характеристик ОС; n - кількість ОС. В такому разі збільшується час розв'язання задачі конфлікту, що може призвести до ефекту "доміно" та неконтрольованому зростанню часу обчислень. Виходячи з наведених міркувань, можна зробити висновок про те, що інтегральне врахування складності моделі взаємодії ОУ з множиною ОС в ПС приводить до лінійного збільшення складності, а саме отримаємо $N = (m+1)n$. Ці міркування складають сутність підходу до розрахунку та формування простору рішень, як підпростору ПС, при вирішенні задачі конфлікту, що дозволяє достатньо просто врахувати топологічну складність конфлікту. Такий підхід отримав назву метод інтегрального усікання варіантів [131, 132, 133, 135, 140, 142, 143, 150, 151, 155, 156, 158, 159, 160, 162, 165].

Слід зазначити, що ПС є топологічним, має складну структуру і за сутністю може бути ототожнений з простором пошуку для задач пошуку та переслідування. До множини ПС простори станів та рішень (рис. 2.1.) входять як підмножини.

Для подальших міркувань введемо певну аксіоматику.

Аксіома 1. ПС є незамкненою, неопуклою довільною множиною позицій ОС та ОУ.

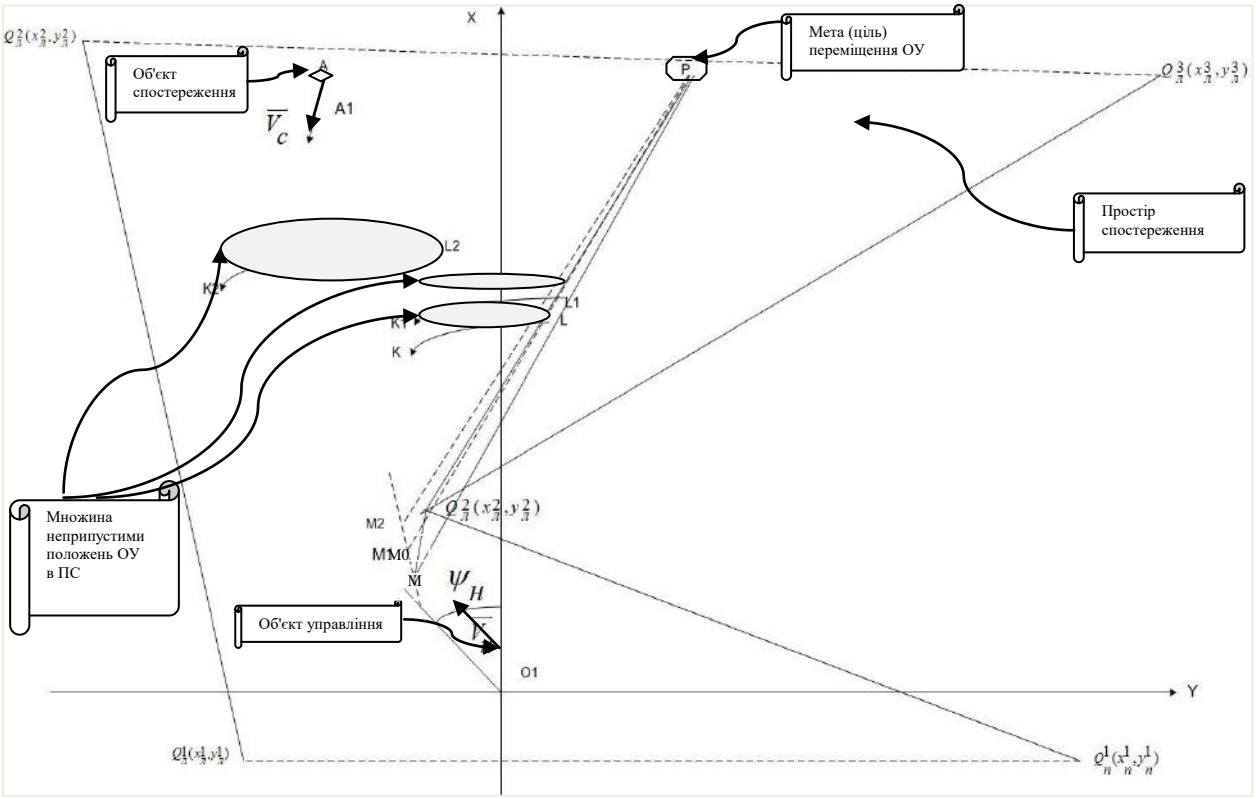


Рис 2.1. Простір спостереження

Аксіома 2. Множина ОС є відкритою.

Аксіома 3. З позицій класичної механіки ПР та замкнена множина ОУ і ОС являє собою повну замкнену динамічну систему об'єктів.

Аксіома 4. Кожний об'єкт в ПС являє собою матеріальну точку з власною масою.

Аксіома 5. Система, яка утворена відкритою множиною ОС в ПС є неоднорідною. ОС в ПС мають різні пріоритети, що враховується в дистанції припустимого зближення з ОУ та невизначеністю при переміщенні ОС.

Аксіома 6. Система, яка утворена відкритою множиною ОС в замкненому ПР, є однорідною. ОС в ПР мають рівні пріоритети.

Аксіома 7. Обмеження ПС розглядаються як щільна сукупність ламаних ліній без маси.

Аксіома 8. Початкові позиції об'єктів в ПР не мають маси, а самі об'єкти є кінематичними.

Аксіома 9. Цільові позиції ОС в ПС є невизначеними і їх позиції в ПС визначаються прогнозованими траєкторіями.

Аксіома 10. Цільова позиція ОУ в ПС початково визначається і є незмінною при синтезі стратегій поведінки.

Аксіома 11. Кінематичні характеристики динамічних ОС визначаються параметричними характеристиками реальних об'єктів ПС, а також фізико-механічними властивостями реального ПС.

Аксіома 12. Еволюція функціонального стану ОС залежить від його цільового функціонування, характеру топології траєкторій переміщення і визначає інтервали життєвого циклу об'єкта в ПР.

Слід зазначити, що множини неприпустимих положень ОУ в ПС мають властивості переміщення, що послідовно вказано за напрямами $\overrightarrow{LK}$, $\overrightarrow{L_1K_1}$, $\overrightarrow{L_2K_2}$. Тобто простір неприпустимих положень ОУ в ПС має власні динамічні та кінематичні властивості, що дозволяє розглядати процес рішення задачі конфлікту, як динамічний процес.

Переміщення ОУ можна розуміти як екзогенні прояви у вигляді переміщення внаслідок перебудови переваг, які в свою чергу залежать від того, що діється з ОУ в екзогенному просторі.

Мета (ціль) переміщення ОУ позначена точкою P, яка фактично є центром мас деякої множини, що за аксіомою 8 не має маси.

При переміщенні ОУ та ОС створюють в ПС підпростори (множини) неприпустимих положень (ПНП) ОУ. За своєю сутністю ці множини є геометричними місцями точок в ПС, в яких при незмінній стратегії переміщення ОУ буде здійснене зближення з ОС на відстань меншу припустимої. Слід зазначити, що ПНП в процесі переміщення ОУ змінюються в розмірах, а також переміщуються в ПС. Тобто ПНП мають змінні розміри і кінематичні характеристики. Маса ПНП визначається масою ОУ та ОС, які створюють ПНП.

При довільній кількості ОС ПНП створюються кожною парою ОУ та ОС в ПС. Позначимо для *i*-го ОС в ПС ПНП як σ_i.

Тоді для довільної кількості n ОС маємо інтегральний опис ПНП $\Omega = \bigcup_{i=1}^{n} \sigma_i$.

ПНП формується як множина в ПС ОУ шляхом виділення деякої його частини, як множини точок простору H, що обмежена замкненою границею G Тобто $\Theta = H \bigcap \Omega$, де Θ - ПР, H - ПС.

Таким чином синтез та вибір рішення не залежить від кількості ОС.

При переміщенні ОУ в цільову точку P траєкторія переміщення може мати довільну кількість елементів.

Виходячи з динамічних і кінематичних властивостей ОУ, а також стратегії управління, для ОУ обирається кількість елементів траєкторії переміщення. На рис. 2.1 траєкторія визначає стратегію управління ОУ, що має шість елементів, для кожного з яких обираються гарантовані значення параметрів управління (модуль швидкості переміщення та напрямок переміщення).

На першому елементі траєкторії здійснюється синтез стратегій та вибір рішення щодо управління ОУ. В точці M траєкторії розраховується ПНП, який позначено кінцевими точками сектору небезпечних напрямків переміщення (СННП) $\overrightarrow{LK}$, образ якого має власний напрямок переміщення в ПС. При переміщенні ОУ в точку M_1 перерахований СННП $\overrightarrow{L_1K_1}$ має власний напрямок переміщення образу в ПС. Тобто при лінійному прогнозі переміщення ОУ, незмінній траєкторії та параметрах переміщення ОС (початково знаходиться в точці A) СННП зміщується в напрямку безпечного переміщення ОУ в кінцеву точку траєкторії переміщення. При цьому динаміка переміщення ПНП є додатною (зміщується в бік відкриття напрямку переміщення ОУ в кінцеву точку елемента траєкторії).

Зміна напрямку переміщення ОУ в точці M_1 дозволяє вирішити задачу конфлікту взаємодії ОУ з ОС при переміщенні в цільову точку P.

В разі зміни напрямку переміщення в точці M_2 розраховується ПНП, який позначений кінцевими точками СННП (L_2K_2), а напрямок переміщення його образу визначається вектором $\overrightarrow{L_2K_2}$. Для незмінної траєкторії переміщення ОС динаміка переміщення ПНП зберігається. В разі зміни напряму переміщення ОУ в точці M_2 отримуємо траєкторію безконфліктного переміщення ОУ, яка не є провокуючою.

Стратегія рішення конфлікту взаємодії ОУ з ОС в ПС є багатоелементною. Крім зміни напрямків за траєкторією переміщення ОУ необхідно обрати швидкість переміщення з діапазону припустимих для ОУ. Таким чином в двомірному евклідовому просторі маємо стратегію, яка забезпечує синтез багатоелементної траєкторії переміщення ОУ при рішенні задачі конфлікту. В даному випадку маємо шестиелементний варіант рішення, який на кожному елементі траєкторії переміщення має діапазони значень параметрів швидкості та напряму переміщення, які є гарантовано припустимими і враховують динамічні властивості ОУ.

В загальному випадку ПНП має масові, динамічні та кінематичні характеристики. В такому разі при управлінні ОУ маємо можливість впливу на переміщення ПНП, що означає зміну стану ПР для ОУ (керування станом ПР).

Слід зазначити, що в випадку переміщення ОУ в двомірному евклідовому топологічному просторі, маємо екзогенні координати x, y ОУ для випадку незв'язаної системи координат (зазвичай при вирішенні задач пошуку чи переслідування задача конфлікту вирішується в зв'язаній ортогональній системі координат, одна вісь якої співпадає з будівельною чи іншою характерною віссю ОУ, а початок координат співпадає з центром мас ОУ - точковим об'єктом). Слід мати на увазі, що закони механіки в екзогенному просторі виконуються повністю, а координати x, y ОУ є миттєвими координатими. При управлінні ОУ маємо можливості обирати управління

довільним способом в залежності від обраної стратегії управління в просторі комбінованого управління об’єктом (ПКУО) .

При використанні методів ситуаційного управління для ТЕС канонічні переваги формують щодо ОПР певні рішення з запропонованої множини стратегій (рис. 2.2) що співпадає з функціональним складом модулів СІУ конфліктом (рис. 1.7).

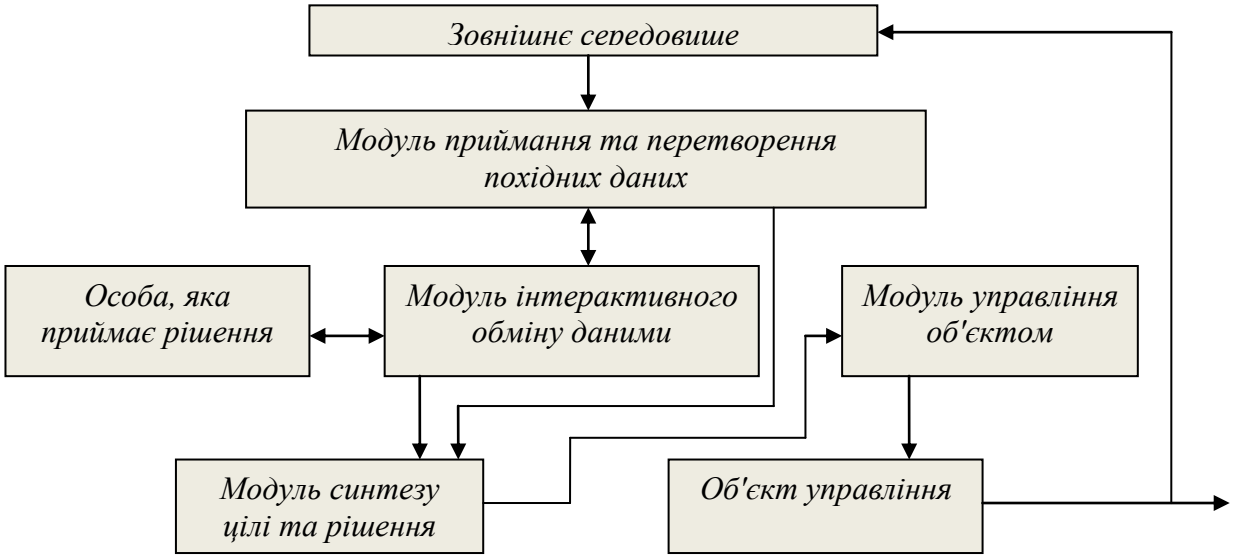


Рис 2.2. Структурна схема системи ситуаційного управління конфліктом

ОПР задає цілі (мету) переміщення ОУ та обирає стратегію управління. В такому разі через модуль інтерактивного обміну даними та інтелектуальний модуль синтезу цілі управління ОУ засобами модуля управління об'єктом реалізується стратегія рішення конфлікту. Таким чином СІУ забезпечує безконфліктне переміщення ОУ в ПР засобом найбільш прийнятним до поставленої ОПР цілі.

Система обмежень топологічного ПС являє собою n-мірну гіперповерхню. Ребра гіперповерхні утворюються при перетині площин, що формують її грані, як систему обмежень топологічного ПС. Тобто при відображенні в довільно взятій площині ортогональних координат обмеженого простору маємо ламані лінії його обмежень.

Так, для двомірного евклідового лінійного ПС (рис. 2.3) на площині визначені ламані лінії обмежень. З метою забезпечення топологічності ПС

визначається правило замикання. В даному випадку правило замикання визначає з’єднання відповідних начальних та кінцевих точок ламаних ліній обмежень. Отриманий таким чином ПС в загальному випадку не є опуклим.

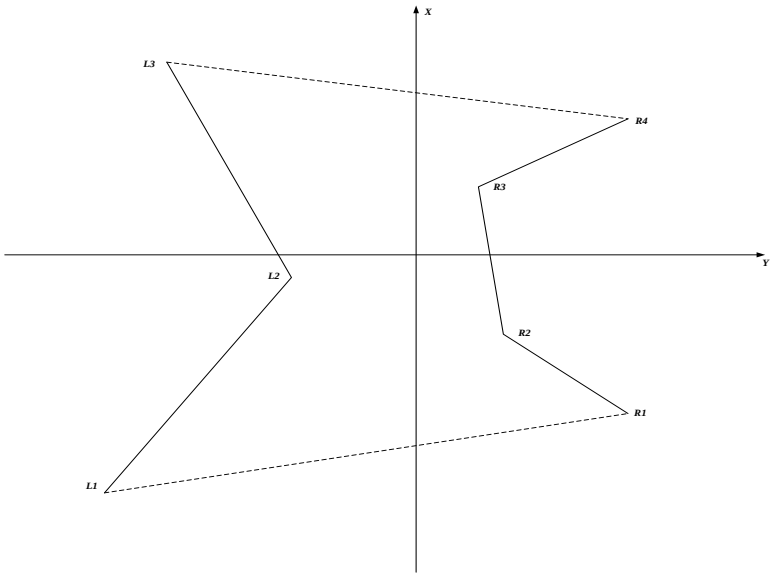


Рис. 2.3. Двомірний простір спостереження

Для гіперпростору спостережень маємо кількість двомірних відображень $N_{ПС} = C_k^2 = \frac{A_k^2}{P_2} = \frac{k!}{2!(k-2)!}$, де k – мірність ПС. Тобто кількість відображень в перерізах ПС дорівнює значенню кількості сполучень з мірності ПС по 2 елементи (двомірний простір відображення).

Для спрощення подальших міркувань будемо розглядати двомірний ПС, як один з перерізів гіперпростору спостережень. При цьому треба враховувати те, що при топологічності перерізів гіперпростору спостережень і сам гіперпростір буде топологічним.

Велика кількість даних гіперпростору із специфічними взаємними зв'язками вимагає топологічного опису об'єктів і груп об'єктів, яке залежить від "зв'язаності" (простої або складної), яка визначає сукупність топологічних моделей. Слід враховувати, що топологічні властивості фігур не змінюються при будь-яких деформаціях, які здійснюються без розривів або з'єднань. Топологічно спорідненими фігурами є чотирикутник, замкнутий контур

довільної форми без петель, коло, трикутник. Для гіперпростору топологічна модель визначається наявністю і зберіганням сукупностей взаємозв'язків, таких, як сполученість дуг на перетинах, впорядкований набір ланок, взаємозв'язки суміжності між ареалами і тому подібне.

На відміну від класичної мережевої моделі в цій моделі довжина ребер може не нести інформативного навантаження. Топологічні характеристики ареальних об'єктів можуть бути представлені за допомогою графів покриття і суміжності.

Топологічне векторне представлення даних в ПС відрізняється від нетопологічного можливістю отримання вичерпного списку взаємовідносин між пов'язаними геометричним примітивами без зміни координат просторових об'єктів, що зберігаються.

В такому разі векторно-топологічне представлення є різновидом векторного представлення лінійних і полігональних просторових об'єктів, що описує не лише їх геометрію, але і топологічні відношення між полігонами, дугами і вузлами.

О.Д.Олександров ввів загальне поняття кінематики як впорядкованого топологічного простору де відношення порядку відповідним чином погоджено з топологією [9].

Разом з тим слід зазначити, що в подальшому слід враховувати те, що для топологічного простору мають місце твердження.

Твердження 1. *Для довільного геометричного замкненого топологічного простору його перерізи є топологічними двомірними підпросторами з границями довільної форми.*

Виходячи з положень загальної топології [17, 18], приведемо твердження, які підтверджують справедливість наведеного твердження.

Твердження 2. *Нехай* X_1 *та* X_2 - *топологічні простори, в яких* $\forall \alpha_1 \in X_1$ *відображення* $x_2 \mapsto (\alpha_1, x_2)$ *є гомеоморфізм простору* X_2 *на підпростір* $\{\alpha_1\} \times X_2$ *простору* $X_1 \times X_2$.

Відображення $x_2 \mapsto (\alpha_1, x_2)$ є безперервним перетином за відношенням еквівалентності $pr_2 z = pr_2 z'$ в $X_1 \times X_2$. Таким чином факторпростір простору $X_1 \times X_2$ за цим відношенням еквівалентності гомеоморфно X_2.

В даному випадку точка $z \in Z$, де Z - простір наділений топологією $\mathcal{F}$ і нерозривний в точці z. Тобто $r_2 = X_2 / R$ є фактор множиною, яка наділена фактор топологією X_2 щодо відношення еквівалентності R. В такому разі pr_2 є безперервною проекцією r_2.

Наслідок. Перетин (переріз) $A(x_1)$ відкритої замкненої підмножини A простору $X_1 \times X_2$ за довільною точкою $x_1 \in X_1$ є відкритою замкненою множиною в X_2.

Твердження 3. *Проекція довільної множини U з $X_1 \times X_2$ на будь-яке з просторів-співмножників є відкритою множиною.*

Справедливість твердження витікає з наслідку попереднього твердження, наприклад, $pr_2 U = U_{x_1 \in X_1} U(x_1)$.

Зауваження. Проекції замкненої множини з $X_1 \times X_2$ на простори-співмножники можуть бути незамкненими множинами. Так, наприклад, для раціональної площини множина Q^2 є гіперболою, яка визначається рівнянням $x_1 x_2 = 1$, яка є замкненою множиною, але обидві її проєкції співпадають з доповненням до точки 0 в Q, але ця множина є незамкненою.

Твердження 4. *Нехай X_1, X_2, Y - топологічні простори і f є відображенням простору $X_1 \times X_2$ в Y. Якщо відображення f безперервно в точці (α_1, α_2), то часткове відображення $x_2 \mapsto f(\alpha_1, x_2)$ простору X_2 в Y безперервно в точці α_2.*

Зазначене відображення є композицією f та відображення $x_2 \mapsto f(\alpha_1, x_2)$. Таким чином, справедливість твердження є наслідком твердження 2.

Твердження 4 можна висловити інакше, кажучи, що безперервна функція двох аргументів є безперервною по кожному аргументу окремо.

Зауваження. Може статися так, що всі часткові відображення, які визначаються відображенням $f: X_1 \times X_2 \to Y$, безперервні, а відображення f не є небезперервним на множині $X_1 \times X_2$. Наприклад, у відображенні f числової площини R^2 в R, визначеного умовами $f(x,y) = \dfrac{xy}{x^2+y^2}$ для $(x,y) \neq (0,0)$ та $f(0,0) = 0$. Тобто всі часткові відображення f є безперевними, але відображення f не небезперервне в точці $(0,0)$, оскільки $f(x,x) = \dfrac{1}{2}$ при $x \neq 0$.

Виходячи з наведених тверджень можна зазначити, що за умови якщо g є відображенням простору X_1 в Y, яке безперервне в точці α_1, то відображення $(x_1, x_2) \mapsto g(x_1)$ простору $X_1 \times X_2$ в Y безперервне в будь-якій точці (α_1, x_2), як композиція відображення g та проєкції на X_1.

Результати для цього топологічного відкритого простору n° безпосередньо розповсюджується на довільний добуток $\prod_{i \in I} X_i$ топологічних просторів, якщо помітити, що останній є гомеоморфним добутку $\left(\prod_{i \in J} X_i\right) \times \left(\prod_{i \in K} X_i\right)$ для кожного розбиття (J, K) множини I.

Тим самим доведено правильність твердження 1.

Розглянемо спосіб визначення приналежності ОС як точки ПСП та визначимо загальний вигляд предиката приналежності точки ПС на прикладі двомірного евклідового простору (рис. 2.4), який в загальному випадку може бути перерізом гіперпростору ПС.

Маємо дві ламані лінії $L_1L_2L_3L_4$ та $R_1R_2R_3R_4$, які визначають ліву та праву границі ПС.

Згідно правила замикання з'єднаємо точки L_1 з R_1 та L_4 з R_4. Отримаємо многокутник $L_1L_2L_3L_4R_4R_3R_2R_1$, який обмежує частину ПС і утворює ПС, який визначимо як множину Q.

Визначимо правило приналежності будь-якої точки топологічній множині Q у вигляді предикату.

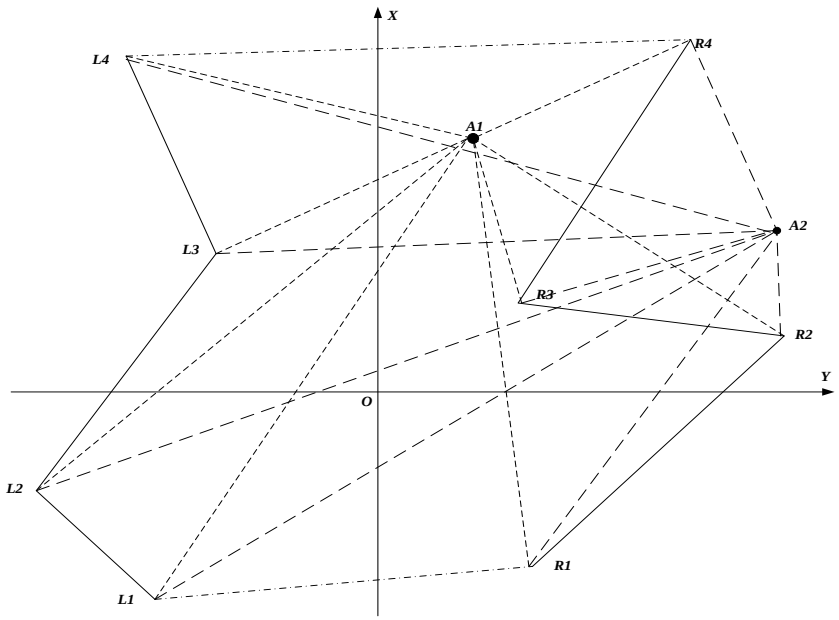


Рис 2.4. Розміщення об'єкта спостереження в ПС

З'єднаємо довільно обрані по відношенню до площини пошуку внутрішню точку A_1 та зовнішню A_2 з вершинами многокутника $L_1L_2L_3L_4R_4R_3R_2R_1$.

Для точки A_1 маємо:

$$\angle L_1A_1L_3 + \angle L_2A_1L_4 + \angle L_3A_1L_4 + \angle L_4A_1R_1 + \angle R_1A_1R_2 + \angle R_2A_1R_3 + \angle R_3A_1R_4 + \angle R_4A_1L_4 = 2\pi \quad . \quad (2.1)$$

Для точки A_2 маємо:

$$\angle L_1A_2L_2 + \angle L_2A_2L_3 + \angle L_3A_2L_4 + L_4A_2R_1 + \angle R_1A_2R_2 + \angle R_2A_2R_3 + \angle R_3A_2R_4 + \angle R_4A_2L_1 = 0 \quad . \quad (2.2)$$

Виходячи з співвідношень (2.1) і (2.2), визначимо правило приналежності будь-якої точки ПС V_j топологічній множині Q у вигляді

$$V_j = \begin{cases} \Theta_j = 0, \forall \sum_{i=1}^{m} \xi_j^i = 0 \\ \Theta_j = 1, \forall \sum_{i=1}^{m} \xi_j^i = 2\pi \end{cases}, \quad (2.3)$$

де Θ_j - значення, яке приймає ознака приналежності точки V_j простору Q за умови обмеження зазначеного простору m-кутником; ξ_j^i - кут повороту вектору-вказівника при його переміщенні від початку в кінець j-ї границі Q за годинниковою стрілкою. При $\Theta_j = 1$ точка V_j належить топологічній множині простору Q, а при $\Theta_j = 0$ точка V_j не належить топологічній множині простору Q.

В загальному вигляді границя простору Q задається неявною функцією

$$\Gamma_{гр}(Q) = 0, \quad (2.4)$$

яка є безперервною та неопуклою.

Співвідношення (2.4) є загальним для довільного перерізу багатомірного топологічного неопуклого ПС з границею, що визначена співвідношенням (2.4).

Для структурованої інформації в просторі Q виникають інформаційні співвідношення невизначеності аналогічні принципу невизначеності Гейзенберга в квантовій механіці. При цьому не порушуючи принципи фізичного квантованого світу, слід виходити з посилки "квантової структури інформації" [201], що випливає з принципу структуризації інформації [58] в системах ситуаційного управління.

Таким чином, можна стверджувати, що крім "фізичної нерозрізненості" об'єктивно існує і інформаційна нерозрізненість (нерозрізненість в межах кроку зміни структури) при взаємодії ОУ з ОС в просторі Q. У такому разі співвідношення невизначеності є наслідком дії апарату представлення і не залежить від точності представлення інформації.

Існує точна кількісна аналогія між співвідношеннями невизначеності Гейзенберга і властивостями хвиль або сигналів, які математично аналогічні координаті і (квантово-механічному) імпульсу частки (об'єкту) в просторі Q. За аналогією з квантовою механікою отримуємо те, що просторова частота $p_x = \hbar k_x$ вздовж відповідної координати x суть аналогічна імпульсу в квантовій механіці, а значення k_x є частотою коливання вздовж відповідної координати. Величина $\hbar$ є постійною Планка, значення якої надзвичайно мале і тому співвідношення невизначеностей накладають слабкі обмеження на похибки вимірювання, які свідомо непомітні на тлі реальних практичних похибок вимірювальних приладів і систем перетворення інформації (даних).

Принцип невизначеності не відноситься тільки до координати та імпульсу. У своїй загальній формі, він прйнятний до кожної пари сполучених змінних. У загальному випадку, на відміну від випадку координати і імпульсу, нижня межа множення значень “невизначеностей” двох сполучених змінних залежить від стану дсліджуваної системи. В такому випадку принцип невизначеності може бути сформульований як теорема в теорії операторів [201].

Виходячи з співвідношення невизначеності Гейзенберга та поняття ентропії [33, 58-60, 214], можна зв’язати визначення положення ОС в ПС з причиною існуючої в природі різноманітністю, що можна розглядати як неоднорідність в розподілі енергії або речовини у просторі та часі.

Щодо інформації, В.М.Глушков характеризує її як міру цієї неоднорідності [214]. Таким чином, інформація існує постільки, поскільки існують самі матеріальні тіла і, отже, створені ними неоднорідності. В даному випадку поняття інформації стає більш містким і "дорастає" до рівня філософських категорій, таких як матерія і енергія.

В такому сенсі семіотична теорія та семіотичні моделі дозволяють оцінити кількісний і змістовний бік інформації, що відображено в структурній схемі інтелектуального управління ОУ (рис. 1.7).

Слід зазначити, що в загальному випадку ПС є неопуклим і небезперевним, що є принциповим.

Дослідимо топологічність та лінійність простору Q [156].

Теорема. Для довільних самосполучених операторів $A: H \to H$ та $B: H \to H$ і будь-якого елементу x з H такого, що ABx та BAx обидва визначені (тобто Ax і Bx також визначені), маємо:

$$\langle x|AB|\rangle\langle x|BA|\rangle = |\langle Bx \mid Ax\rangle|^2 \le |\langle Ax \mid Ax\rangle\langle Bx \mid Bx\rangle| = \|\langle Ax\rangle\|^2 \|\langle Bx\rangle\|^2. \qquad (2.5)$$

Ця теорема випливає з нерівності Коші - Буняковського, яке пов'язує норму і скалярний добуток векторів в евклідовому просторі. Нерівність (2.5) еквівалентна нерівності трикутника для норми.

Слід зазначити, що оператор A в комплексному чи дійсному гільбертовому просторі $\mathfrak{H}$ є ермітовим, симетрическим, якщо він задовольняє рівності $(Ax, y) = (xA, y)$ для всіх (x, y) з області визначення A. В даному випадку припускається, що (x, y) — скалярний добуток в $\mathfrak{H}$. Оператор в $\mathfrak{H}$ є самосполученим або гіпермаксимальним ермітовим якщо він співпадає зі своїм сполученим.

В даному випадку поняття сполученості пов'язане з топологічністю лінійного простору Q.

Нехай E, L — лінійні простори, а E^*, L^*, — сполучені лінійні простори (простори лінійних функціоналів, які визначені на E, L). Тоді для будь-якого лінійного оператора $A: E \to L$ та будь-якого лінійного функціоналу $g \in L^*$ є визначеним лінійний функціонал $f \in E^*$, який є суперпозицією g. Тобто $f(x) = g(A(x))$.

В такому разі сполучений лінійний оператор має вигляд відображення $g \to f$, а $(A^* g, x) = (gAx)$, для якого в загальному випадку (Bx) є дією функціоналу B на вектор x.

Тоді відображення оператора $g \to f$ лінійно та безперервно, а сполучений оператор можна записати у вигляді $A^* : L^* \mapsto E^*$.

В загальному випадку форма, безперервність, лінійність, топологічність простору Q є невизначеною. Тому слід враховувати деякі характеризуючі ознаки, які дозволяють узагальнити погляд на характеристики існування ПС.

Слід зазначити, що поняття сполученості існує для топологічного лінійного, банахова та гільбертова просторів.

Задамо для банахова простору безперевний лінійний оператор $A: X \to Y$, який діє з банахова простору X в банаховий простір Y.

Тоді для сполучених просторів X^*, Y^* визначимо, що $\forall x \in X,\ f \in Y^*, [Ax, f] = f(Ax)$.

Якщо f — фіксовано, то $[Ax, f]$ — лінійний безперервний функціонал в просторі $X, [Ax, f] \in X^*$. Таким чином, для $\forall f \in Y^*$ є визначеним лінійний безперервний функціонал з X^* такий, що $[Ax, f] = [x, A^* f]$. В такому разі оператор A^* є сполученим.

Аналогічно можна визначити сполучений оператор до лінійного необмеженого оператора. При цьому слід враховувати те, що такий сполучений оператор буде визначеним на всьому просторі Q.

Слід зазначити, що для сполученого оператора A^* в банаховому просторі справедливими є наступні властивості:

- оператор A^* — лінійний;

- оператор C — лінійний безперевний;

- якщо оператор A є лінійним безперервним, то оператор A^* є також лінійним безперервним;

- нехай O — нульовий оператор, а E — одиничний оператор, тоді справедливими є вислови:

$$\begin{cases} O^* = O, E^* = E \\ (A+B)^* = A_* + B^* \\ \forall \alpha \in C, (\alpha A)^* = \overline{\alpha} A^*. \\ (AB)^* = B^* A^* \\ \left(A^{-1}\right)^* = \left(A^*\right)^{-1} \end{cases} \quad (2.6)$$

Згідно співвідношення (2.6) для відображення в гільбертовому просторі H, як і для відображення в топологічному лінійному та банаховому просторах, норма яких породжена додатно визначеним скалярним множенням, отримаємо ототожнення простору зі своїм сполученим. Тобто для оператора $A: H \to H$ рівність $(Ax, y) = \left(x, A^* y\right)$ визначає сполучений оператор $A^*: H \to H$. В цьому випадку (x, y) є скалярним результатом множення в просторі H.

Таким чином, для будь-яких сполучених операторів вірною є загальна форма принципу невизначеності, описувана співвідношенням Робертсона-Шредингера [114]

$$\frac{1}{4}\left|\langle x|AB - BA|x\rangle\right|^2 \le \left\|\langle Ax\rangle\right\|^2 \left\|Bx\right\|^2. \quad (2.7)$$

Для операторів A і B в виразі (2.7) комутатор $[A, B] = AB - BA$ і визначений для тих x, для яких визначені обидва ABx та BAx.

В такому разі співвідношення невизначеності Гейзенберга слідує з співвідношення Робертсона — Шредингера.

Тоді для двох фізичних величин A і B, пов'язаних з самосполученими операторами, для яких визначені ABx та BAx згідно співвідношенню (2.7), буде справедливим вираз

$$\Delta_\psi A \Delta_\psi B \ge \frac{1}{2}\left|\langle [A, B] \rangle_\psi\right|. \quad (2.8)$$

Виходячи з співвідношення (2.8), оператор стандартного відхилення величини X в стані ψ системи можна записати в вигляді

$$\Delta_\psi X = \sqrt{\left\langle X^2 \right\rangle_\psi - \left\langle X \right\rangle_\psi^2}. \quad (2.9)$$

В співвідношенні (2.8) середнє значення оператора величини X для ψ визначено і має вид $\langle X \rangle_{\psi} = \langle \psi | X | \psi \rangle$.

Сказане вище відноситься до теорії операторів і може бути узагальнено на будь-яку пару ермітових операторів. В загальному випадку співвідношення невизначеності можна застосувати до віртуальних явищ фізичного та кібернетичного простору. В такому разі некомутуючі самоспопучені оператори A і B мають один і той же власний вектор ψ , який представляє собою стан, який є одночасно вимірюємим для A і B.

Наведені міркування поширюються на кінематичні і динамічні об'єкти, включаючи поля, а також і на об'єкти кібернетичного простору (КП), в якому можна ввести аналог координат частинки і аналог компонент імпульсу частинки, що є канонічними імпульсами, пов'язаними зі зміною середовища у часі.

Разом з тим КП складається з взаємозв'язаної сукупності інформаційних структур, включаючи глобальну комп'ютерну мережу Інтернет, інформаційно-телекомунікаційних систем, обчислювальних систем, а також процесорів і контролерів, вбудованих в технічні засоби.

В такому сенсі сукупність структур, систем і засобів в КП можна розглядати як сукупність кібернетичних організмів (кіберорганізмів). З іншого боку взаємопов'язана сукупність кіберорганізмов складає сутність КП. При цьому кіберорганізми можуть бути як суб’єктами та об’єктами КП.

2.2. Теоретико-множинна модель взаємодії об’єкту управління з варіативною множиною ОС в умовах конфлікту, обмежень та невизначеностей

Виходячи з топологічності ПС, синтезуємо модель конфлікту взаємодії ОУ з варіативною множиною ОС в ПС за умов обмежень та невизначеностей. В такому разі задачу синтезу управління можна сформулювати як задачу

розв'язання конфлікту взаємодії ОУ з ОС в ПС згідно принципу оптимальності та правилу зупинки, як динамічну задачу динамічної дискретної оптимізації.

Взаємодія різних за своєю фізичною сутністю об'єктів в просторі Q, як в просторі спостережень, станів та поведінки, пошуку щодо ОУ може бути описана як модель конфлікту [114, 132, 137, 146, 151, 155, 156, 158].

Виходячи з особливостей функціонування ОУ в ПС, можна перейти до методів ситуаційного управління ОУ [114]. При цьому потрібно враховувати те, що при синтезі рішень може змінюватись структура управління ОУ та інформація щодо функціонування та переміщення ОС в ПС.

Виходячи з топологічності простору Q розглянемо метод синтезу рішень для задачі ситуаційного управління конфліктом для ОУ з використанням семіотичних моделей. В основу методу покладено спосіб формування простору рішень як множини, в якій ОУ зберігає властивості керування при запобіганні розвитку ситуації конфлікту взаємодії з відкритою множиною ОС при відкритій системі обмежень.

Розглянемо модель M взаємодії об'єктів в ПСП за умови замкненості простору Q з урахуванням співвідношення (2.4)

$$M = \bigcup_{i=0}^{N} M^i, \tag{2.10}$$

де M^i - часткова модель i-го ОС, яка може бути представлена у вигляді

$$M^i = \left\langle Б^i, F^i, \Gamma^i_{пр} \right\rangle, \tag{2.11}$$

а модель ОУ має вигляд

$$M^0 = \left\langle Б^0, F^0, \Gamma^0_{пр} \right\rangle. \tag{2.12}$$

В співвідношеннях (2.11) та (2.12) базис $Б^i$ визначає потенційні можливості взаємодії *і-го* ОС та ОУ в просторі Q

$$Б^i = \left(X^i, Y^i, A^i\right), \tag{2.13}$$

де X^i - множини потенційно можливого місця знаходження *і*-го ОС в просторі Q, які визначаються як множини (області) керуємих та

напівкеруємих станів в просторі параметрів у відповідності з припущенням невизначеності та прогнозу переміщення *i*-го ОС; множина Y^i визначається характеристиками переміщення *i*-го ОС в просторі керованих і напівкерованихих станів та враховує наявні ресурси управління щодо зміни динамічних та кінематичних характеристик *i*-го ОС у відповідності з припущенням A^i, яке враховує прогноз, невизначеність, динаміку та небезпечність переміщення *i*-го ОС щодо ОУ для співвідношення F^i, враховує прогноз, динаміку та безпечність переміщення ОС в просторі Q з базисом $Б^i$.

Значення F^i в співвідношенні (2.11) визначає властивості *i-го* ОС щодо переміщення в просторі Q

$$F^i = \left(f_x^i, f_c^i, d^i \right), \tag{2.14}$$

де f_x^i - згладжені значення координат для *i-го* ОС в кожен момент спостереження в просторі Q; f_c^i - згладжені значення першої похідної (вектору швидкості зміни координат); d^i - припустиме зближення ОУ з *i*-м ОС.

В загальному вигляді для *i*-го ОС значення d^i може бути визначено як

$$d^i = d_{зад}^i + \Delta d^i, \tag{2.15}$$

де $d_{зад}^i$ - припустиме значення зближення ОУ з *i*-м ОС; Δd^i - невизначеність, яка враховує динамічність *i*-го ОС.

В загальному вигляді значення Δd^i для *k* - мірного простору Q визначається співвідношенням

$$\Delta d^i = \sqrt{\sum_{j=1}^{k} \left(\ddot{x}_j^i \right)^2 \frac{\Delta t^2}{2}}, \tag{2.16}$$

де $\ddot{x}_j^i$ - друга похідна відповідної *j*-ї координати *i*-го вектору швидкості f_c^i *i*-го ОС; Δt - інтервал часу вимірювання.

З метою врахування невизначеності для різних за фізичною сутністю ПС (фізичний, віртуальний, тощо) доцільним є врахування невизначеності у вигляді

$$\Delta x_i^z = \frac{p}{2m\Delta \dot{x}_i^z}, \tag{2.17}$$

де $\dot{x}_i^z$ - перша похідна відповідної *z*-ї координати *i*-го вектору швидкості f_c^i *i*-го ОС, x_i^z - відповідна координата *z* вектору f_x^i *i*-го ОС; m - маса *i*-го ОС; p - коефіцієнт, який визначається сутністю ПС та *i*-го ОС.

В квантовій механіці за ознакою невизначеності Гейзенберга $p = \frac{\hbar}{2}$, де $\hbar$ - стала Планка.

З співвідношення (2.17) видно, що чим менше маса ОС в просторі Q, тим менш визначеними стають координати і швидкість. Тим самим для макроскопічних тіл практична застосовність класичного способу опису руху не викликає сумнівів.

Таким чином, співвідношення (2.15) - (2.17) дозволяють визначити вектор f_x^i для *i*-го ОС.

Для кібернетичного простору значення сталої p має бути визначена окремо. При цьому слід враховувати, що в теорії систем [83, 84, 205] та сінергетиці [58] інформація за своєю сутністю має енергетичні та польові властивості і не має маси. Тобто значення Δx_i^z в співвідношенні (2.17) співпадає за сутністю з поняттям ентропії.

Таким чином для різних за сутністю ПС значення сталої p і невизначеності Δx_i^z має визначатись окремо.

Слід зазначити, що Рейнхардом Р. Кехером [231] було введено нове співвідношення, яке на відмінність від поняття еквівалентності “інформація-маса” використовує відношення “інформація-маса-енергія”. В такому разі з

закону збереження первинно витікає закон збереження енергії, а вторинно – закон збереження маси.

Ввівши в закон Ейнштейна $E = mc^2$ поняття універсальної інформаційної одиниці [231]

$$I_S = \frac{m}{c^2}, \tag{2.18}$$

отримаємо інформаційний продукт

$$I_S E = \frac{mc^2 m}{c^2} = m^2 \ . \tag{2.19}$$

Співвідношення (2.19) відображає саморезонуючу масу з власною внутрішньою вібрацією.

Цілісність універсальної інформаційної одиниці (2.18) забезпечується інформаційно-енергетичним множником $\frac{c^2}{c^2}$.

Тоді маємо

$$\frac{I_S}{E} = \frac{m}{c^2}\frac{1}{mc^2} = \frac{1}{c^4}. \tag{2.20}$$

Співвідношення (2.19) визначає взаємодію двох граничних множин, які визначаються фактором c^2.

За формулою Планка $E = \nu\hbar$, де ν точно визначає частоту коливань.

Тоді значення універсальної інформаційної одиниці визначимо співвідношенням

$$I_S = \frac{\nu}{\hbar}. \tag{2.21}$$

З врахуванням співвідношень (2.21) маємо значення інформаційно-енергетичного продукту

$$I_S E = \frac{\nu\hbar\nu}{\hbar} = \nu^2. \tag{2.22}$$

Тоді з співвідношення (2.20) отримаємо знаменник зв’язку, який визначає інформаційно-енергетичну одиницю

$$\frac{I_S}{E} = \frac{\nu}{\hbar}\frac{1}{\hbar\nu} = \frac{1}{\hbar^2}. \qquad (2.23)$$

У співвідношенні (2.23) параметр $\hbar^2$ визначає інформаційну цілісність, яка взаємодіє з значенням c^2.

З врахуванням (2.21) та (2.22) для співвідношення (2.13) маємо

$$I_S E = \nu m. \qquad (2.24)$$

Однією з найбільш специфічних ознак інформації є “несиловий” характер її дії [231]. Несилова дія виявляється в тому, що при незначній величині енергії сигналу, що несе інформацію, енергія реакції сприймаючої системи є незрівнянно більшою. Достатньо очевидною є та обставина, що міра перетворень в сприймаючій системі залежить, насамперед, від “змісту” сигналу, а не від його енергії. Це дає підстави, з одного боку, говорити про нематеріальність інформації, а з іншого - про те, що інформація матеріальна, але при цьому вона не є власне матерією, а лише її особливою властивістю.

В співвідношенні (2.11) $\Gamma^i_{пр}$ визначає граматику та правила утворення співвідношень при взаємодії ОУ та *i*-го ОС, а також взаємодії *i*-го ОС з іншими ОС в просторі Q.

Тобто в загальному випадку за співвідношенням (2.11) вводиться поняття мови для формального опису процесів, пов'язаних з вирішенням задачі пошуку і переслідування в просторі Q.

В загальному випадку мова є нескінченною множиною, а нескінченні об'єкти важко задати, наприклад, простим перерахуванням елементів [29, 33]. Будь-який кінцевий механізм задавання мови є граматикою. В співвідношенні (2.11) це $\Gamma^i_{пр}$.

Для ОУ в співвідношенні (2.12) визначимо базис у вигляді

$$Б^0 = \left(X^0, Y^0, A^0\right), \qquad (2.25)$$

де X^0- множина потенційно можливого знаходження ОУ в просторі Q, множина Y^0 визначається характеристиками переміщення ОУ в просторі

керуємих і напівкеруємих станів за наявних ресурсів управління щодо зміни динамічних та кінематичних характеристик у відповідності з припущенням A^0 , яке враховує прогноз, динаміку та безпечність переміщення ОУ.

Властивості ОУ щодо переміщення в просторі Q визначає співвідношення $F^0 = \left(f_x^0, f_c^0, d^0\right),$ (2.26)

де f_x^0 - згладжені значення координат ОУ в кожен момент спостереження в просторі Q; f_c^0 - згладжені значення першої похідної (вектору швидкості зміни координат ОУ); d^0 - припустиме зближення ОУ з ОС.

Слід зазначити, що для співвідношень (2.11), (2.12), (2.15) та (2.17) суттєвим є вибір системи координат для моделі (2.11).

В співвідношенні (2.26) значення d^0 є в загальному випадку критичною величиною найменшого зближення, при якому не порушується цілісність конфлікту взаємодії ОУ з ОС в просторі Q. Таким чином для простору Q визначимо узагальнене значення припустимого зближення об’єктів

$$d = \bigcup_{i=0}^{N} d^i \,. \qquad (2.27)$$

В моделі (2.11) за умов невизначеності до границі $\Gamma_{гр}$ простору рішень Q ставиться вимога замкненості.

Простір Q формується з урахуванням усіх характеристик поведінки об'єктів: невизначеності їх координат в кожен момент спостереження, значення вектору швидкості зміни координат, припустимого зближення. За вимог замкненості ПР границю цього простору представимо в вигляді (2.4).

Границя $\Gamma_{гр}$ в загальному випадку є безперервною, кусково-гладкою неопуклою і замкненою, що не дозволяє використовувати традиційні підходи щодо опису простору обмежень $G_{обм}$, і не дозволяє використовувати традиційні методи рішення задач опуклого програмування в топологічному просторі. Слід зауважити, що підпростір $G_{обм}$ має задовольняти вимогам

зв'язності та безперервності хоча б впродовж траєкторії керуємого переміщення ОУ в ПС.

Для співвідношення (2.10) додаткові обмеження підпростору $G_{обм}$ з границею $Г_{гр}$ визначаються співвідношенням

$$A = \bigcup_{i=0}^{N} A^i , \tag{2.28}$$

яке враховує невизначеність та прогноз переміщення об'єктів в ПС з врахуванням системи обмежень. Тобто для ОУ маємо

$$F = \bigcup_{i=0}^{N} F^i, \forall f_x^i \subset f_x, \forall f_c^i \subset f_c, \forall d^i \subset d . \tag{2.29}$$

Таким чином, співвідношення (2.28) узагальнює всі характеристики об’єктів в просторі Q з врахуванням співвідношень (2.14), (2.2.15) і (2.17).

Відображення співвідношення F^i для i – го ОС з врахуванням $\left(A^i, A^0\right)$ та базису $\left(Б.Г_{пр}\right)$ на підмножину $G_{обм}$ породжує підпростори $G_{обм}^i$, які є неприпустимими для позицій X^0 ОУ та параметрів його переміщення Y^0, що дозволяє використати принцип невизначеності при дослідженні динамічних характеристик поведінки об'єктів в ПС.

Слід зауважити, що простір Q є топологічним і, при необхідності, може бути декомпозований на класи еквівалентності [172].

Визначимо множину обмежень для i-го ОС

$$G_{обм}^i = W\left(X^0, Y^0, f_c^0, f_c^i, X^i, Y^i, A^0\right). \tag{2.30}$$

В такому випадку інтегральна множина ПР з врахуванням (2.23) матиме вигляд

$$G_{ріш} = G_{обм} \bigcap_{i=0}^{N} G_{обм}^i . \tag{2.31}$$

Співвідношення (2.25) дозволяє визначити геометричне місце точок простору Q, як лінійного евклідового простору, в якому значення параметру взаємного розміщення об'єктів (відстань, напрямок на об'єкт, тощо) не менший

за значення співвідношення порівняння d^i (відстані, напряму на об'єкт, тощо) для ОУ.

Виходячи з співвідношень, які є формальним описом стану i-го ОС, визначимо інформаційну множину ПКУО

$$S = \bigcup_{N,\Psi_{прип}} S^i\left(X^0, Y^0, f_c^0, f_c^i, X^i, Y^i, d^i, A^i, \Delta t\right), i = \overline{0,N}, \tag{2.32}$$

де S^i - множина припустимих значень параметрів, які визначають характеристики переміщення та позицію i-го ОС в просторі Q при його переміщенні за напрямком $\psi_j \in \Psi_{прип}$, який належить множині припустимих напрямків переміщення ОУ за умови дотримання значення d^i, Δt - інтервал часу вимірювань та розрахунків.

В загальному випадку інформаційні множини і підпростори простору $G_{обм}$ не є опуклими та безперервними але є топологічними лінійними. Тоді задачу синтезу стратегії переміщення ОУ можна вирішити в банановому або гільбертовому просторі за умов виконання відповідних перетворень з врахуванням відповідних співвідношень відображення і невизначеності.

На підставі отриманого формального опису інформаційних множин (2.10) - (2.17) та (2.25) – (2.32) синтез стратегій управління ОУ здійснюється за мінімально-переборною процедурою методу інтегрального усікання варіантів [132, 151], згідно критерію оптимальності Φ.

Критерій Φ фактично є адитивним критерієм вибору

$$\begin{cases} \Phi = \sum_{i=1}^{k} C_i \lambda_i \\ \lambda_i = \dfrac{\Delta u_i}{\sup|u_i|}, \forall \lambda_i \in \lambda \\ \sum_{i=1}^{k} C_i = 1 \end{cases}. \tag{2.33}$$

Значення коефіцієнтів C_i в співвідношенні (2.33) визначається для кожного виду конфліктів окремо. λ_i визначає "витрати на управління"

параметром або функцією управління u_i для i-го елемента траєкторії переміщення ОУ в просторі $G_{ріш}$.

В такому разі задачу синтезу управління можна сформулювати як конфлікт взаємодії об'єктів в просторі $G_{ріш}$

$$K = \langle M, A, S, \Gamma_{пр}, G_{ріш}, \mu \rangle, \tag{2.34}$$

а процедуру синтезу стратегій рішення конфлікту μ згідно принципу оптимальності χ, який реалізовує вимоги критерію Φ, можна представити у вигляді

$$\begin{cases} K = \langle M, A, S, \Gamma_{пр}, G_{ріш}, \mu \rangle \\ \lambda K = \mu \end{cases}. \tag{2.35}$$

Вибір оптимальної стратегії рішення конфлікту μ^* з врахуванням правила зупинки $\Gamma_{зуп}$ формулюється в вигляді

$$\mu^* = \inf_{\chi, \Gamma_{зуп}} K. \tag{2.36}$$

Співвідношення (2.34) - (2.36) фактично є постановкою задачі опису конфлікту взаємодії об'єктів та синтезу оптимального рішення в просторі Q.

Для рішення задачі конфлікту доцільно використати методи ситуаційного управління ОУ.

2.3. Семіотична модель взаємодії об’єкту управління з варіативною множиною ОС в просторі спостереження

Згідно співвідношенням (1.1) - (1.6) семіотична модель $\Im(.)$ ІП СІУ описується четвіркою $\Im = \langle M, \chi_Б, \chi_F, \chi_Г, \chi_Q \rangle$, де $\chi_Б, \chi_F, \chi_Г, \chi_Q$ відповідно є правилами зміни $Б, F, Г, Q$ для інтерпретуємої формальної моделі $M = \langle Б, F, \Gamma_{пр} \rangle$, яка узагальнює уявлення про теоретико-множинну модель

(ТММ) взаємодії ОУ з варіативною множиною ОС в умовах конфлікту, обмежень і невизначеностей (2.10) – (2.12).

Виходячи з моделі M взаємодії об'єктів в ПР, розглянемо семіотичний підхід до синтезу стратегій управління ОУ в умовах конфлікту та невизначеностей.

В моделі M^i за співвідношенням (2.11) $\Gamma_{пр}^i$ є граматикою та правилами утворення співвідношень при взаємодії ОУ та *i*-го ОС, а $\Gamma_{пр}^0$ в моделі M^0 за співвідношенням (2.12) є граматикою та правилами утворення співвідношень при взаємодії ОУ з простором $G_{ріш}$.

В загальному випадку за співвідношенням (2.11) можна ввести поняття мови для формального опису процесів, які пов'язані з вирішенням задачі пошуку та переслідування в просторі Q.

В загальному випадку мова є нескінченною множиною, а нескінченні об'єкти важко задати, наприклад, простим перерахуванням *елементів* [29]. Будь-який кінцевий механізм задавання мови є граматикою $\Gamma_{пр}^i$.

Формальна мова є множиною ланцюжків в деякому кінцевому алфавіті. За моделлю M^i кінцевим алфавітом за співвідношенням (2.14) є F^i.

Для завдання опису формальної мови використовується алфавіт F^i, в якому сукупністю об'єктів, що називаються символами, є множини $\left(f_x^i, f_c^i, d^i\right)$, які можна відтворювати в необмеженій кількості екземплярів. В такому разі для *i*-го ОС можна задати формальну граматику мови, а саме правила P^i, за якими з символів мови будуються їх послідовності, що належать визначеній мові. За правилами P^i утворюються так звані правильні ланцюжки. Такими ланцюжками є описані в термінах мови рішення, які є алгоритмами для цифрового автомату [33] СІУ ОУ.

Правила формальної граматики P^i можна розглядати як продукції (правила виведення), тобто елементарні операції.

В такому разі для *i*-го ОС визначимо $\Gamma_{пр}^{i}$ як граматику загального виду

$$\Gamma_{пр}^{i} = \left(F^{i}, T^{i}, P^{i}, C^{i} \subset Q\right), \tag{2.37}$$

де F^{i} - множина нетермінальних позицій, T^{i} - множина термінальних (кінцевих) позицій, P^{i} - кінцева множина правил (продукцій), яка вміщує хоча б одну нетермінальну позицію, C^{i} - початкова нетермінальна позиція *i*-го ОС.

Множина термінальних позицій має належати простору Q, тобто

$$T^{i} \bigcap Q \neq \varnothing. \tag{2.38}$$

Таким чином, згідно (2.38) граматика $\Gamma_{пр}^{i}$ регламентує індивідуальну стратегію переміщення *i*-го ОС, виходячи, наприклад, з вимог політики безпеки (для ІТС), нормативних документів (на транспорті), цілі переміщення в просторі Q, тощо.

Для множини початкових нетермінальних позицій *i*-го ОС обов'язковою є вимога приналежності його цілі переміщення C^{i} простору Q, тобто $C^{i} \subset Q$.

При вирішенні задач конфлікту слід враховувати модель M^{0} щодо ОУ, для якої задано співвідношення (2.33) для F^{0}, а також термінальні позиції для *i*-го ОС згідно співвідношенню (4.2).

Співвідношення (2.26) для ОУ є аналогічним співвідношенню (2.14) для *i*-го ОС. Координати f_{x}^{0} зв'язані з центром мас, а сама система координат є зв'язаною з ОУ. Тобто вісь ординат зв'язаної системи координат ОУ співпадає з будівельною віссю ОУ, що дозволяє зробити узагальнення щодо фізичних і кібернетичних об'єктів в фізичному та віртуальному просторі Q. Вектор швидкості f_{c}^{0} має мірність простору Q, а його елементами є значення згладжених координат вектору швидкості (першої похідної) ОУ в просторі Q. Значення d^{i} відображає припустиме значення відстані зближення ОУ з *i*-м ОС в просторі Q.

В моделі M^0 множина X^0 вироджена в одну точку внаслідок того, що система координат є зв'язаною з ОУ. Множина Y^0 визначається характеристиками переміщення в просторі керуємих і напівкеруємих станів у відповідності з припущенням A^0 та враховує динамічні можливості ОУ.

Припустима відстань d^0 визначається з огляду на вимоги безпеки, нормативних та керівних документів, експертних знань, тощо.

В такому сенсі ОУ є керованою системою в просторі Q.

Граматику $\Gamma^0_{пр}$ в моделі M^0 визначимо для ОУ як регулярну.

$$\Gamma^0_{пр} = \left(F^0, T^0, P^0, C^0 \subset Q\right), \tag{2.39}$$

де F^0 - множина нетермінальних позицій; T^0 - множина термінальних (кінцевих) позицій; P^0 - кінцева множина правил (продукцій), яка вміщує хоча б одну нетермінальну позицію; C^0 - початкова нетермінальна позиція ОУ.

Тоді для переміщення ОУ в базисі $Б^0$, який визначено співвідношенням (2.25), можна задати n ланцюжків $U^0_{j_k}, k = \overline{1,n}$, які характеризуються переміщенням ОУ в просторі Q. Координати точки $X^0_{j_k} \in Q$, в яку переміщується ОУ, і значення характеристик переміщення $Y^0_{j_k}$ визначаються динамічними та фізичними обмеженнями щодо можливих управлінь ОУ.

Тобто

$$\begin{cases} Y^0_{j_k} \subset Y^0_{j_0} \\ f^0_{c_{j_k}} \subset Y^0_j \\ f^0_{x_{j_k}} \subset X^0_j \end{cases}, \tag{2.40}$$

для кожного j-го ланцюжка k-го рішення.

Таким чином, система (2.40) визначає приналежність характеристик переміщення ОУ множині припустимих значень при вирішенні задачі конфлікту для всіх ОС в просторі Q.

Стратегія переміщення ОУ в просторі Q є конкатенацією ланцюжків переміщення, кількість яких обумовлена властивостями і динамічними характеристиками ОУ, обмеженнями простору Q, характеристиками ОС, обмеженнями політики безпеки для інформаційних об'єктів, а також іншими вимогами, які є обов'язковими для всіх об'єктів в ПС.

Слід зазначити, що формальна граматика $\Gamma_{пр}^{0}$ в співвідношенні (2.36) є породжуючою і використовується для побудови "правильних" ланцюжків. При цьому послідовність правил P^0 в процесі породження ланцюжка $U_{j_k}^{0}$ є його виводом.

Тобто

$$U_j^0 = \bigcup_{k=1}^{n} U_{j_k}^0 . \qquad (2.41)$$

Критерій відбору при розрахунку і конкатенації ланцюжків U_j^0 за співвідношенням (2.41) може входити до правил P^0 ОУ.

Граматики (2.37) та (2.39) за Хомським [29] відносяться до третього типу і є автоматними. Кінцевий автомат (алгоритм), який побудовано за такою граматикою, має кінцеву кількість станів. Тобто мова зазначених граматик є регулярною, а синтезований за ними цифровий автомат має бути кінцевим автоматом.

Таким чином, виходячи з можливостей граматик (2.37) і (2.39), можна запропонувати узагальнені автоматні моделі побудови ланцюжків при синтезі стратегій управління ОУ для прийняття рішень в разі довільної поведінки всіх ОС в просторі Q.

Процес синтезу стратегій поведінки щодо цілеспрямованого керування ОУ в просторі Q використовує основні об'єкти моделі M^0, які є базовими елементами непустої множини $Б^0$, і можуть бути визначені як алфавіт.

Ланцюжки $U_{j_k}^{0}$, які характеризують стратегію переміщення ОУ в просторі Q, будемо вважати орієнтованими зліва направо. Довжиною

ланцюжка буде кількість базових елементів множини $Б^0$, які входять в цей ланцюжок.

Нехай

$$Б^0 = \{b_1^0,...,b_r^0\}, \tag{2.42}$$

де b_l^0 - l-та підмножина алфавіту, яка не є пустою.

При безперервному управлінні ОУ позначимо ланцюжок

$$R_{j_k}^0 = U_{j_k}^0 \circ U_{j_{k+1}}^0, \tag{2.43}$$

де $\circ$ операція конкатенації символів (базової складової мови).

Виходячи з співвідношень (2.42) і (2.43), для операції конкатенації (об'єднання ланцюжків) будуть справедливими властивості замкненості

$$\circ : \left(U_j^0\right)^* \times \left(U_j^0\right)^* \to \left(U_j^0\right) \tag{2.44}$$

та асоціативності

$$\left[\left(U_{j_k}^0 \circ U_{j_{k+1}}^0\right) \circ V_j^0 = U_{j_k}^0 \circ \left(U_{j_{k+1}}^0 \circ V_j^0\right)\right] \forall \left[U_{j_k}^0 \subset \left(U_j^0\right), U_{j_{k+1}}^0 \subset \left(U_j^0\right), R_{j_k}^0 \subset \left(U_j^0\right)\right], \tag{2.45}$$

де $\left(U_j^0\right)^*$ - множина усіх можливих ланцюжків, складених з кінцевої множини базових елементів $U_{j_k}^0$, які включають пустий ланцюжок ε.

Тоді з урахуванням співвідношень (2.44) і (2.45) можна зазначити, що множина $R_{j_k}^0$, яка визначена співвідношенням (2.43), є формальною мовою, що визначена на алфавіті базових елементів $Б^0$ згідно співвідношенню (2.42).

Зазначимо, що мову R_j^0 для множин $R_{j_k}^0$, які притаманні ОУ, засновано на обмеженнях, які накладаються на систему напіввідношень (продукції) Туе [29]. Тобто граматика $Г_{пр}^0$ визначена як механізм породження ланцюжків $U_{j_k}^0$, які визначають переміщення ОУ в просторі Q.

Визначення формальної граматики $Г_{пр}^0$ вимагає наявності ще одного алфавіту

$$R^0 = \bigcup_{m=1}^{N} R_m^0 , \qquad (2.46)$$

який за своєю сутністю є алфавітом опису простору $G_{ріш}$ для ОУ, що обумовлює зручність алфавітного способу завдання інформації для СІУ.

З точки зору об'єкту, який сприймає інформацію, інформація є дискретною і алфавітною, а алфавітний спосіб надання (завдання) інформації в ТС або ТЕС за співвідношенням (2.46) можна вважати універсальним. Таким чином СІУ та ОУ виконують функції перетворення інформації.

Семіотичні моделі є засадними для методу ситуаційного управління при створенні СІУ ОУ. Саме використання семіотичних моделей становить головну відмінність СІУ від СУ, які використовують традиційні методи теорії управління, в основі яких лежать деякі формальні (у логічному сенсі) моделі.

Для моделі M^i за співвідношенням (2.37) $\Gamma_{пр}^i$ є граматикою та правилами утворення співвідношень при взаємодії ОУ та *i*-го ОС, до якої входить кінцева множина правил (продукцій) P^i, яка вміщує хоча б одну нетермінальну позицію, а для моделі M^0 за співвідношенням (2.39) граматика $\Gamma_{пр}^0$ є граматикою та правилами утворення співвідношень при взаємодії ОУ з ПР.

В такому сенсі граматика, яка пов'язана з мовою опису процесу взаємодії об'єктів в ПР, може бути визначена в загальному вигляді як

$$\Gamma_{пр} = \bigcup_{i=1}^{N} \Gamma_{пр}^i , \qquad (2.47)$$

де *N* - загальна кількість ОС в ПР.

Таким чином, мовою, яка описує ланцюжки синтезованих рішень буде $L(\Gamma_{пр})$.

Узагальнюючи співвідношення (2.37) - (2.39) можна визначити граматику $\Gamma_{пр}$, як граматику загального виду

$$\Gamma_{пр} = (F, T, P, C \subset Q), \qquad (2.48)$$

де $F = \bigcup_{i=0}^{N} F^i$ - множина нетермінальних позицій; $T = \bigcup_{i=0}^{N} T^i$ - множина термінальних (кінцевих) позицій, для якої має виконуватись співвідношення (2.38); $P = \bigcup_{i=0}^{N} P^i$ - кінцева множина правил (продукцій), яка вміщує хоча б одну нетермінальну позицію; $C = \bigcup_{i=0}^{N} C^i$ - множина початкова нетермінальна позиція об'єктів; N - загальна кількість ОС в ПР, а множина термінальних позицій має належати ПР.

В такому разі для завдання опису формальної мови $L(\Gamma_{пр})$ згідно співвідношень (2.13), (2.25) - (2.29) використовується алфавіт

$$F(f_x, f_y, d), \tag{2.49}$$

в якому сукупністю об'єктів, що називаються символами, є множини, кожну з яких можна відтворювати в необмеженій кількості екземплярів. В такому разі для об'єктів простору Q можна задати формальну граматику мови. Тобто правила P, за якими з символів будуються їх послідовності ланцюжків управління ОУ згідно співвідношенню (2.41) належать визначеній мові R^0. В такому сенсі правила P формальної граматики $\Gamma_{пр}$ можна розглядати як продукції (правила виведення), тобто елементарні операції.

Множини T і P з співвідношення (2.48) породжують мову семіотичної системи $L(\Gamma_{пр})$, якою може бути будь-яка виразна мова. Така мова має задовольняти вимогам Вудса [232], а саме:

- мова має дозволяти описувати усі істотні з точки зору наступних задач об'єкти, їх властивості, ознаки і зв'язки;

- мова має дозволяти зневажувати неістотними властивостями об'єктів та ПС;

- мова має надавати можливість створювати нові об'єкти або розглядати сукупності об'єктів як такі, створювати ідентифікатор об'єкту для подальшого посилання на нього;

- опис, при якому будь-які об'єкти є відсутніми або мають невизначені властивості, повинен розглядатись як природній, а якість функціонування СІУ повинна монотонно залежати від повноти опису.

Елементами мови $L(\Gamma_{пр})$ є лексеми та лексичні позначення даних про об'єкти в просторі Q.

Виходячи з співвідношень (2.47) - (2.49), введемо правило формування лексем за формою Бекуса-Наура

$$\langle лексема \rangle \rightarrow \langle A|B|C|D|E|FX|FY|FZ|FU|FW|FV|FT|H|I|J|K|L|M|N \rangle$$

$$\langle : \rangle \langle цифра \rangle \mid \langle лексема \rangle \langle цифра \rangle \mid \langle лексема \rangle \langle . \rangle \mid \langle лексема \rangle \langle цифра \rangle. \quad (2.50)$$

Лексичні позначення для співвідношення (2.50) наведені в Таблиці 2.1.

Таблиця 2.1.

Лексичні позначення даних про об'єкти в просторі Q

Дані про об'єкт	Позначення
Позначення об'єкту	A
Координати (в метриці простору)	B
Відстань	C
Швидкість зміни координат (в мірності простору) - вектор швидкості за перерізами	D
Зміна напрямку переміщення (кутові значення) за перерізами	E
Просторовий СННП (в перерізах за сполученням координат): - кут початку; - кут кінця; - відстань до середини: - координати кінців; - вектор швидкості зміни координат кінців; - ознака збільшення/зменшення розміру; - ознака перетину з напрямком переміщення ОУ.	 FX FY FZ FU FW FV FT
Дистанція зближення (встановлена початкова)	H
Дистанція зближення (обрахована за результатами спостереження) за перерізами	I
Час до входу в простір рішення за перерізами	J
Час до входу в СННП за перерізами	K
Ознака можливості перетину СННП за перерізами	L
Ознака можливості зміни параметрів переміщення ОУ за перерізами	M
Ознака когнітивності зміни параметрів переміщення ОУ за перерізами	N

Змістовні одиниці поєднуються в речення, які відокремлюються одне від іншого символом "&".

Речення опису поточної ситуації ПР об'єднуються в змістовний запис, який розміщається в квадратних дужках "[ ]".

Символи "&", "[", "]" є метасимволами мови для граматики $\Gamma_{пр}$, що являє відступ від форми Бекуса-Наура, але, при необхідності, дозволяє виключити рекурсію. Використання метасимволів при запису правил є найбільш вживаною формою для регулярної $\Gamma_{пр}$.

Таким чином, для опису поточної ситуації взаємодії об’єктів в просторі Q формується набір речень з лексем похідної мови за похідними даними, які поступають від відповідних датчиків модуля приймання та перетворення похідних даних (рис. 2.2). В ІП СІУ здійснюється обробка похідної формалізованої інформації, синтезується відповідне дерево опису в мовних конструкціях R^0 і синтезується відповідний ланцюжок управління для переміщення ОУ за траєкторією рішення.

Обробка мовних структур опису подій в СІУ забезпечується шляхом використання методів семантичного аналізу повідомлень.

2.4. Семантична модель взаємодії об’єкту управління з варіативною множиною ОС в просторі спостереження

Першим етапом опрацювання згенерованого формалізованого опису ПР є його лексичний аналіз.

На початковому етапі лексичного аналізу будується таблиця ідентифікаторів.

При аналізі запису у формі (2.50) згідно лексичним позначенням даних про об'єкти в просторі Q будується бінарне дерево для послідовності ідентифікаторів (рис. 2.5).

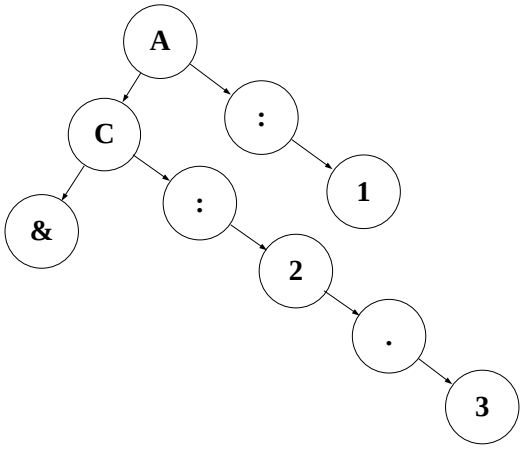


Рис. 2.5. Заповнення бінарного дерева для послідовності ідентифікаторів

Слід зазначити, що кількість об'єктів в просторі Q є довільною та статистично невпорядкованою, а побудоване бінарне дерево є невиродженим. В такому разі можна оцінити середнє значення часу формування дерева $T_ф$ та пошуку ідентифікатора $T_п$ у вигляді

$$\begin{cases} T_ф = m \times O(\log_2 m) \\ T_п = O(\log_2 m) \end{cases}, \tag{2.51}$$

де m- кількість елементів в таблиці лексем; $O(\log_2 m)$- функція, яка визначається алгоритмом бінарного пошуку з використанням хеш-таблиць.

Метод бінарного дерева є досить вдалим механізмом організації таблиць ідентифікаторів [40, 93]. Пошук елемента в дереві здійснюється за алгоритмом, який є схожим з алгоритмом формування дерева з часовими характеристиками, які визначаються співвідношеннями (2.50) і (2.51).

Маючи таблицю ідентифікаторів, проводиться лексичний аналіз (ЛА) поточної ситуації взаємодії об'єктів в просторі Q.

ЛА забезпечується лексичним сканером, що здійснює зчитування повідомлення СІУ та будує таблицю лексем з повідомлення формального опису поточного стану взаємодії об’єктів в просторі Q, опис якого синтезується в автоматичному або автоматизованому режимі на мові $L(\Gamma_{пр})$ в ІП.

Засоби ЛА, які входять до складі модулів СІУ, дозволяють спростити процес обробки похідного формалізованого опису стану простору Q, виконати простий та ефективний аналіз похідного опису, забезпечити взаємодію з формальними описами, які мають мовні особливості. Тобто ЛА спрощує обчислювальні процеси, які пов'язані з функціонуванням синтаксичного аналізатора (СА).

При створенні ЛА для системи синтезу стратегій управління ОУ мова $L(\Gamma_{пр})$ передбачає визначення границі лексем в явному вигляді. В такому разі ЛА функціонує послідовно з СА.

Формальна граматика $\Gamma_{пр}$, яка визначена співвідношенням (2.48), є контекстно-вільною і за Хомським [40] відноситься до типу 2. В граматиці $\Gamma_{пр}$ ліві частини всіх продукцій є одиночними нетерміналами. Сенс терміну "контекстно-вільна" полягає в тому, що на відміну від загального випадку необмеженої граматики Хомського, існує можливість застосувати продукцію до нетерминалу незалежно від контексту цього нетерміналу.

Вхідна мова $L(\Gamma_{пр})$ для синтезу рішень вміщує набір умовних операторів умов ***if ... then ... else*** та ***if ... then***, які розділені символом ";" (точка з комою). Ці оператори вміщують логічні вирази, побудовані з допомогою операцій ***or***, ***xor*** та ***and*** операндами яких є ідентифікатори та цілі десяткові константи без знаку. В такому разі ідентіфікатори позначають операції над описами згідно співвідношенню (2.50), враховуючи те, що з символів будуються послідовності ланцюжків управління ОУ згідно співвідношенню (2.41), які належать визначеній мові R^0. В виконавчій частині оператори опису вміщують оператор привласнення значення (**:=**) або інший умовний оператор.

Таким чином вхідна мова може бути описана КВ-граматикою:

$$\Gamma_{пр}\ (\{if,then,else,a,:=,xor,and,(,),;\},\{S,J,E,D,C\},P,S) \qquad (2.52)$$

з правилами *P*:

S->F;

$$
\begin{aligned}
&J\text{->}if\ E\ then\ T\ else\ J|if\ E\ then\ J|a:=E\\
&T\text{->}if\ E\ then\ T\ else\ T|\ a:=E\\
&E\text{->}E\ or\ D\ |\ E\ xor\ D|\ D\\
&D\text{->}D\ and\ C|C\\
&C\text{->}a|(E).
\end{aligned}
\qquad (2.53)
$$

Для співвідношення (2.48) множина нетермінальних символів визначається

$$F=(if\ ,k,then,else,a,:=,xor,and,(,),;)\ . \qquad (2.54)$$

Задача ЛА для мови $L(\Gamma_{пр})$ полягає у тому, щоб розпізнавати і виділяти у вихідному тексті повідомлення ІП СІУ усі лексеми цієї мови.

Лексемами даної мови для ІП СІУ є :

- шість ключових слів мови (*if,then,else,or,xor,and*);
- роздільники: відкриваюча та закриваюча круглі дужки , точка з комою;
- знак операції присвоєння;
- ідентифікатори;
- десяткові константи зі знаком.

Для визначеного типу лексем маємо регулярну граматику та створений на її основі КА. Це дозволяє побудувати КА безпосередньо по опису лексем.

Визначимо за формою Бекуса-Наура правила формування ідентифікатора та десяткового числа без знаку:

⟨ідентицікатор⟩ → ⟨буква⟩|⟨ідентицікатор⟩⟨буква⟩|

⟨ідентицікатор⟩⟨цифра⟩ (2.55)

та десяткового числа без знаку:

⟨ціле десятичне число без знаку⟩ →

⟨цифра⟩|⟨ціле десятичне число без знаку⟩⟨цифра⟩| . (2.56)

Межами лексем для ЛА є символи: “пробіл”, “табуляції” , “повернення рядка”, “переведення каретки”, “круглі дужки”, “відкриваюча фігурна дужка”, “крапка з комою”, “двокрапка”. Слід зазначити, що символи “круглі дужки” та

“крапка з комою” самі по собі є лексемами, а “двокрапка”, будучи роздільним символом в лексемі, є ознакою початку іншої лексеми - операції присвоювання, що визначено правилами згідно співвідношенням (2.52) – (2.54).

У даній мові ЛА завжди може однозначно визначити межі лексеми, що обумовлює відсутність необхідності у його взаємодії із СА і іншими елементами компілятора опису з вхідної мови, що включає елементи, які визначені співвідношеннями (2.55) і (2.56).

На рис. 2.6 неведено фрагмент графу переходів кінцевого автомату (КА), на якому показано розпізнавання розділяючих символів, знаку присвоєння, змінних та констант (усіх лексем вхідної мови, крім ключових слів). Ключовими словами є нетермінальні символи *if* та *then*.

В наведеному фрагменті графу КА мають місце наступні позначення:

А - будь-який алфавітно-цифровий символ;

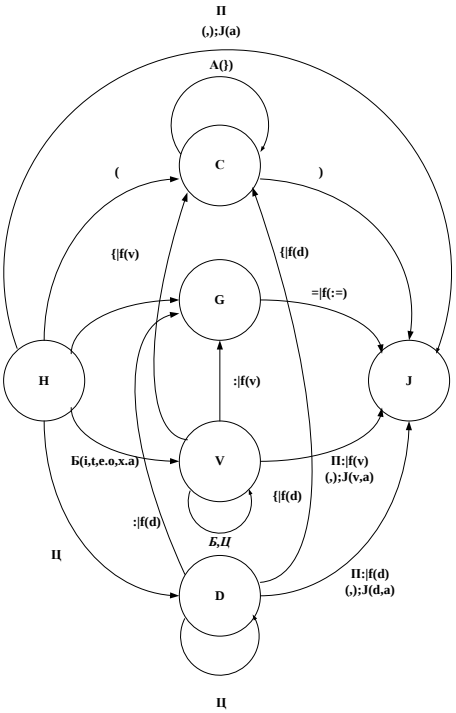


Рис. 2.6. Фрагмент графу переходів КА розпізнавання розділяючи символів

П - будь-який незначущий символ (пробіл, знак табуляції, переведення рядка, повернення каретки);

Б - будь-яка буква англійського алфавіту (прописна або рядкова) чи символ підкреслення “_”;

Б(*) - будь-яка буква англійського алфавіту (прописна або рядкова) або символ підкреслення(_) , крім перерахованих у дужках;

Ц - будь-яка цифра від 0 до 9;

J - функція обробки таблиці лексем , яка викликається при переході КА із одного стану в інший. Позначення її аргументів :

υ - змінна , яка запам'ятована при роботі КА;

d - константа , запам'ятована при роботі КА ;

a - поточний вхідний символ КА.

Фрагмент графу переходів КА для лексем (ключових слів) *if* та *then* мови $L(\Gamma_{пр})$ згідно вхідної мови R^0, що описана КВ-граматикою $\Gamma_{пр}$ з правилами P, які визначені співвідношеннями (2.52) та (2.53), наведено на рис. 2.7.

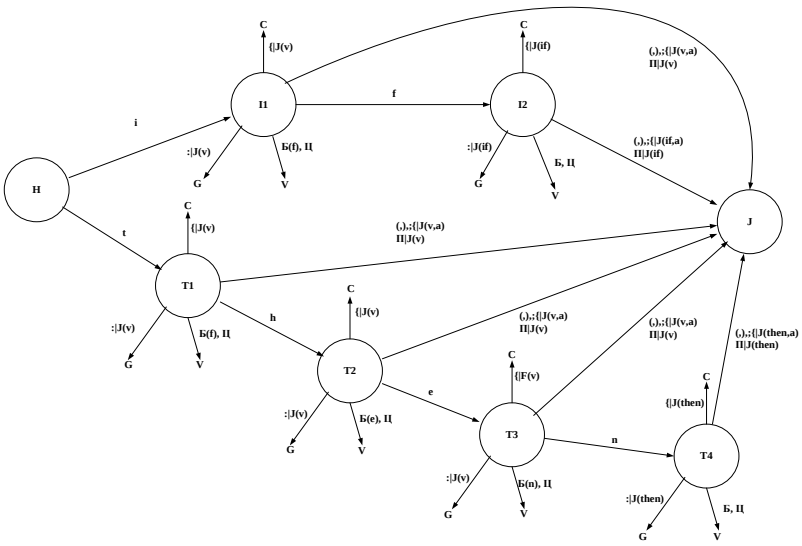


Рис. 2.7. Фрагмент графу переходів КА для ключових слів *if* та *then*

Позначення в графі переходів мають наступний зміст:

$M(Q, E, \delta, q_0, J)$:

Q={*H*,*C*,*G*,*V*,*D*,*I1*,*I2*,*T1*,*T2*,*T3*,*T4*,*E1*,*E2*,*E3*,*E4*,*O1*,*02*,*X1*,*X2*,*X3*,*A1*,*A2*,*A3*,*F*);

E = *A* (усі припустимі алфавітно-цифрові символи);

q_0 = *H*;

δ - функція переходів КА;

$J = \{J\}$.

Із початкового стану КА літери "*i*" , "*t*" , "*e*" ,"*o*" , "*x*" , "*a*" ведуть у початок ланцюжка станів, кожна із яких відповідає ключовому слову:

стани *I1* , *I2* - ключовому слову *if* ;

стани *T1,T2,T3,T4*,- ключовому слову *then*;

стани *E1,E2,E3,E4*, - ключовому слову *else*;

стани *O1,O2*, - ключовому слову *or*;

стани *X1,X2,X3*, - ключовому слову *xor*;

стани *A1,A2,A3*, - ключовому слову *and*.

Інші літери ведуть до стану $\Im$, який є відповідним певній змінній (ідентифікатору). Якщо у будь-якому з ланцюжків зустрічається літера або цифра, які не відповідають ключовому слову, то КА також переходить у стан $\Im$, а якщо зустрічається межа лексеми - запам'ятовує вже прочитану частину ключового слова як змінну (щоб правильно виділяти такі ідентифікатори , як "*i*" чи "*els*" , які співпадають з початком ключових слів).

Цифри ведуть у стан *D*, який відповідає вхідній константі. Відкриваюча фігурна дужка веде у стан *C*, який відповідає виявленню коментаря (подібно до ЛА, які функціонують в компіляторах мов програмування), КА виходить з цього стану при отриманні на вхід закриваючої фігурної дужки. Стан *G* відповідає лексемі "знак присвоєння". КА переходить в цей стан після отримання на вхід символу “двокрапка”, і очікує в цьому стані символу "рівність" або числове значення без знаку у відповідності до форми лексеми згідно співвідношень (2.50), (2.52) – (2.54).

Стан *H* для КА є початковим, а стан *J* - кінцевим. Після досягнення стану *J* КА переходить в стан *H* для продовження обслуговування потоку лексем.

Функція *J*, якою помічені дуги КА на графі і переходи у функції переходів, забезпечує запис даних у таблицю лексем. Аргументи функції залежать від поточного стану КА. В загальному випадку функції *J*

відповідають функціям, які викликаються КА в залежності від його поточного стану та значенню вхідного символу.

Треба зазначити, що для коректного запису змінних і констант у таблицю лексем КА повинен запам'ятати відповідні їм ланцюжки символів, які забезпечується шляхом збереження позиції у вхідному потоці символів при знаходженні КА в стані *H* для продовження обслуговування потоку лексем.

За результатами обробки ЛА вхідного потоку даних здійснюється синтез варіантів ланцюжків стратегій рішення конфлікту. Для вирішення цієї задачі застосовується СА. З таблиці лексем СА використовує лише тип лексем згідно співвідношенням (2.52) – (2.54), а також опрацьовує виділені ключові слова для подальшого опрацювання семантичним аналізатором (СМА).

В загальному випадку СА виконує перевірку правильності конструкцій похідного опису та перетворення результатів функціонування ЛА у вигляді, який є прийнятним для СМА при подальшій семантичній (смисловій) обробці.

Для мови $L(\Gamma_{пр})$ функції СА не визначають приналежність ланцюжка заданій мові. Синтаксичні конструкції є похідними для подальшої генерації ланцюжків синтезуємих рішень. Тобто СА має вихідну мову, за допомогою якої ІП СІУ формує рішення, як знайдені та розібрані синтаксичні структури похідної мови. В такому випадку СА не є різновидом автомату з магазинною пам'яттю, а перетворювачем з магазинною пам'яттю [93].

Слід зауважити, що при створенні СА маємо конкретну граматику $\Gamma_{пр}$ похідної мови $L(\Gamma_{пр})$, що дозволяє побудувати дерево зв’язків та відношень лексичних структур за результатами ЛА. При цьому КВ-граматика для СА задовольняє вимогам перетворення та еквівалентності згідно співвідношенням (2.5) і (2.6).

При створенні СА, виходячи з складності синтезованого за допомогою мови $L(\Gamma_{пр})$ тексту опису ситуації взаємодії ОУ з ОС в просторі $G_{ріш}$, оберемо алгоритм правобічного аналізу тексту. Граматика $\Gamma_{пр}$ не потребує додаткових

перетворень і відноситься до класу КВ-граматик операторного передування, для яких існує відомий алгоритм лінійного розпізнавача. Тому створення СА істотно спрощується.

Для обраного класу КВ-граматик операторного передування щодо синтаксичного розбору ланцюжків доцільним є використання алгоритму "зсув-згортка".

Для кожної впорядкованої пари символів у граматиці $\Gamma_{пр}$ встановлені відношення, які є відношенням передування.

В такому разі в процесі розбору автомат з магазинною пам’ятью (МП-автомат) порівнює поточний символ вхідного ланцюжка з одним з символів, що знаходяться на верхівці стека автомата. У процесі порівняння перевіряється ознака того, яке з можливих відносин передування існує між символами, що порівнюються. Залежно від знайденого значення відносини символів виконується операція зсуву або згортки. В разі відсутності відносини передування між символами формується ознака помилки функціонування МП-автомату.

Таким чином реалізується функціональність СА, яка полягає в забезпеченні можливості несуперечливим чином визначити відносини передування між символами граматики та контролювати достовірність перетворення похідної інформації ІП СІУ.

2.5. Метод рішення конфлікту взаємодії ОУ з варіативною множиною ОС в умовах обмежень та невизначеності, як *NP*-повної перебірної задачі динамічної дискретної оптимізації

Виходячи з наведених міркувань, припустимо, що є деяке явище α_j з деякого класу явищ U_j^0, що визначені співвідношенням (2.41), призводить до появи деякого визначеного явища β_j з класу явищ U_j^0. В такому разі маємо справу з перетворенням інформації

$$\alpha_j \to \beta_j = \varphi_j\left(U_j^0\right). \tag{2.57}$$

З абстрактної точки зору згідно співвідношення (2.57) можна здійснити відображення $\varphi_j\left(U_j^0\right)$ явища α_j в явище β_j певним ІП СІУ ОУ, який є перетворювачем інформації в термінах мови $R_j^0 \subset R^0$ для ланцюжка U_j^0 траєкторії його переміщення в просторі Q.

Перетворення φ_j є детермінованим та однозначним. В цьому разі вихідна інформація повністю визначає похідну інформацію. При цьому еквівалентність перетворення φ_j не є обов'язковою. Тобто припустимим є твердження $\varphi_j = \varnothing$, що відповідає випадку $G_{ріш} = \varnothing$.

Тим самим, синтез ланцюжків U_j^0 є перетворенням похідної інформації з використанням двійкового кодування.

В такому разі синтез множини R_j^0 є результатом функціонування детермінованого КА, який в загальному вигляді згідно співвідношенню (2.46) визначається п'ятіркою

$$R^0 = \left(Б^0, S^0, U^0, F^0, C^0 \subset Q\right), \tag{2.58}$$

де $Б^0$ - базис, який визначено співвідношенням (2.32), що вміщує базові елементи непорожньої множини $Б^0$ та визначається як алфавіт; S^0 - алфавіт стану автомату; U^0 - функція переходів для ОУ; F^0 - початковий стан автомату; $C^0 \subset Q$ - множина кінцевих станів автомату.

Функція переходів U^0 в співвідношенні (2.58) є множиною кінцевих точок станів КА, яким є ІП СІУ при переміщенні ОУ в просторі Q.

Попередньо при рішенні конфлікту було визначено простір Q, в якому аналогічно співвідношенню (2.3) ОУ має ознаку приналежності V_0 та Θ_0.

Визначимо алфавіт стану КА

$$S^0 = Q \bigcap_{N,\psi^0} S^i\left(Б^0, Б^i, F^0 F^i, t\right), \tag{2.59}$$

де t - час; N - кількість ОС в просторі Q; ψ^0 - припустимі для ОУ напрямки переміщення в просторі Q.

Множина $S^i\left(Б^0, Б^i, F^0, F^i, t\right)$ в співвідношенні (2.59) є автоматним відображенням підмножини станів *i*-го ОС на множину простору Q, S^0 є множиною автоматного відображення, що задовольняє умовам повноти для ОУ, а множина ψ^0 визначає просторове автоматне відображення множини S^i в множину простору Q, включаючи пусті відображення ε.

Синтез ланцюжків управління для ОУ здійснюється в просторі S^0, який фактично є простором рішень $G_{ріш}$. Синтезовані ланцюжки можуть бути побудовані тільки в просторі гарантованого існування управлінь ОУ при вирішенні задачі цілеспрямованості $C^0 \subset Q$.

Враховуючи граматику $\Gamma^0_{пр}$, як формальну граматику, та правила утворення співвідношень при взаємодії ОУ з ПР, в термінах мови R^0_j ланцюжки стратегії рішення конфлікту взаємодії ОУ з варіативною множиною ОС визначаються за будь-яким методом пошуку (евристичним, нейронних мереж, тощо) за певними правилами, відношеннями, критерієм відбору та правилами зупинки.

Тим самим, задача синтезу ланцюжків стратегій рішень щодо управління ОУ перестає бути ігровою, а пошук відображення похідного алфавіту стає мінімально-переборною процедурою для ІП СІУ ОУ в просторі рішень S^0. Щодо визначення правил формування інформаційних множин $S^i\left(Б^0, Б^i, F^0, F^i, t\right)$, зазначимо, що вони є словами мови L класу P (класу предикатів) і входять в граматику $\Gamma^0_{пр}$ згідно співвідношенню (2.59). Слід зазначити, що при відображенні ланцюжків рішень для ОПР в ТЕС реалізовується гарантоване управління ОУ за умови первинного аналізу мовних конструкцій похідного алфавіту. Тобто лексичний та семантичний

аналіз мовних структур дозволяє визначити множину керованих та напівкерованих станів ОУ.

2.6. Модель взаємодії об’єктів кібернетичного простору

Забезпечення безпеки функціонування кібернетичних систем (КС), як рішення задачі конфлікту, пов’язане з забезпеченням захисту від несанкціонованого доступу (НСД) об’єктів і суб’єктів інформаційно-телекомунікаційних систем (ІТС). Відповідно до обраної політики безпеки та моделі доступу до ресурсів КС необхідно визначити технічні та технологічні рішення і засоби щодо унеможливлення реалізації спроб НСД.

Вирішення такої задачі за своєю сутністю є вирішенням задачі конфлікту взаємодії об’єктів та суб’єктів КС [148 - 153, 156, 157, 163, 164, 167] з об’єктами КП, які можуть бути ініційовані будь-яким суб’єктом або об’єктом.

В такому разі задача вирішення конфлікту розглядається як варіаційна, а екстремальний елемент за критерієм вибору стратегій рішення є припустимим, незважаючи на можливу стрибкоподібну зміну вектора стану формального середовища опису конфлікту згідно співвідношенням (2.10) – (2.32).

При попередженні та вирішенні конфлікту, який пов’язаний з НСД до ресурсів КС, оберемо модель дискреційного доступу (DAC). Виходячи з моделі управління доступом Харрісона-Руссо-Ульмана (HRU), що є однією з найбільш поширених моделей забезпечення безпеки сучасних інформаційних і інформаційно-телекомунікаційних систем, визначимо множину елементів таких систем

$$\begin{cases} O = O^O \bigcup O^S \\ O^O = \bigcup\limits_{i=0}^{N_O} O_i^O \\ O^S = \bigcup\limits_{j=0}^{N_S} O_j^S \end{cases}, \qquad (2.60)$$

де O - множина усіх об'єктів O^O та суб'єктів O^S КС, множина яких визначається їх кількістю N_O та N_S для об'єктів та суб'єктів відповідно.

Визначимо множину прав доступу суб'єктів до об'єктів КС

$$R = \{r_r, r_w, r_o\}, \tag{2.61}$$

де r_r - право читання, r_w - право запису, r_o - право володіння.

Для визначення прав доступу суб'єктів O^S до об'єктів O^O КС сформуємо матрицю доступу $\Omega(O^S, O^O)$. Таким чином, в процесі функціонування КС здійснюються зміни у матриці доступу, що є відображенням переходу системи з одного стану в інший.

Усі зміни стану здійснюються множиною, яка складається з шести операторів

$$\mathrm{A}\{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6\}, \tag{2.62}$$

де α_1 - „внести" право $r_i \in R, l = \{1,2,3\}$ в $\Omega[O^S, O^O]$ (додавання суб'єкту O_m^S права доступу r_i до об'єкта O_n^O - до елементу матриці доступу $\Omega[O^S, O^O]$ додається елемент r_i; α_2 - „видалити" право $r_i \in R, l = \{1,2,3\}$ в $\Omega[O^S, O^O]$ (видалення у суб'єкта O_m^S права доступу r_i до об'єкта O_n^O - з матриці $\Omega[O^S, O^O]$ видаляється елемент r_i; α_3 - „створити" суб'єкт O_m^S (додавання в систему нового суб'єкта O_m^S - в матрицю доступів $\Omega[O^S, O^O]$ додається новий стовпчик та рядок); α_4 - „створити" об'єкт O_n^O (додавання в систему нового об'єкта O_n^O - в матрицю доступів $\Omega(O^S, O^O)$ додається новий стовпчик); α_5 - „знищити" суб'єкт O_m^S (видалення з системи суб'єкта O_m^S - з матриці доступів $\Omega(O^S, O^O)$ видаляються відповідні стовпчик і рядок); α_6 - „знищити" об'єкт O_n^O (видалення з системи суб'єкта O_n^O - з матриці доступів $\Omega(O^S, O^O)$ видаляються відповідний стовпчик).

В результаті виконання оператора α_i з множини A, яка визначена співвідношеннями (2.61) і (2.62), формально опишемо перехід КС $Q \mapsto \alpha_i Q'$ із стану $Q = \left(O^S, O^O, \Omega\left(O^S, O^O\right)\right)$ до стану $Q' = \left(O^S{}', O^O{}', \Omega\left(O^S{}', O^O{}'\right)\right)$.

Таким чином, значення F^i в співвідношенні (2.14) дорівнює α_i з множини A і визначає властивості *i-го* ОС в КП, який є простором $G_{ріш}$.

Співвідношення (2.60) - (2.62) визначають властивості моделі взаємодії КС з варіативною множиною ОС за умов невизначеностей їх поведінки та обмежень простору $G_{ріш}$.

В моделі (2.10) – (2.12) за умов невизначеності границі ПР $\Gamma_{гр}(Q)$ ставиться вимога замкненості. Властивості ПР є детермінованими за умови переміщення ОУ в КП у відповідності до співвідношення (2.62).

Простір $G_{ріш}$ формується з врахуванням усіх характеристик поведінки об'єктів в цьому просторі, а саме: невизначеності їх координат в кожен момент спостереження, значення вектору швидкості зміни координат, припустимого зближення об’єктів та замкненості ПР.

В такому разі координатами ОС в КП при зв’язаній з КС системі координат можуть бути такі характеристики, як загрузка процесорів, об’єм пам’яті, яка використовується процесом взаємодії з i-м ОС в КП, характеристики інформаційних потоків в КС, тощо.

Границя $\Gamma_{гр}(Q)$ в загальному випадку є безперервною, кусково-гладкою неопуклою і замкненою, що не дозволяє використовувати традиційні підходи щодо опису простору $G_{ріш}$, і не дозволяє використовувати традиційні методи рішення задач опуклого програмування в топологічному просторі. Слід зауважити, що простір $G_{ріш}$ має задовольняти вимогам зв'язності та безперервності хоча б впродовж траєкторії керованого переміщення ОУ в КП.

Для співвідношення (2.10) додаткові обмеження простору $G_{ріш}$ з границею $\Gamma_{гр}(Q)$ визначаються співвідношенням (2.62), яке враховує невизначеності та прогнози переміщення об'єктів в КП з врахуванням системи обмежень. Тобто для ОУ в КП задовольняється умова співвідношення (2.29), яке узагальнює всі характеристики КС в ПС з врахуванням співвідношень (2.60) - (2.62), і можуть бути обчислені множини ПКУО та сформовано простір $G^{*}_{ріш}$. Слід зауважити, що КП є топологічним і, при необхідності, може бути декомпозований на класи еквівалентності [155].

Виходячи з співвідношень (2.60) – (2.62) можна зазначити, що перехід КС із попереднього по наступного стану $Q \mapsto \alpha_i Q'$ є ланцюжком рішення по управлінню ОУ в ІП СІУ, функції якого виконує система захисту інформації (СЗІ). Синтез ланцюжка рішень здійснюється за вербальним описом, який автоматично генерується за даними матриці доступу $\Omega(O^S, O^O)$, множини прав доступу та множини операторів зміни вмісту матриці доступу засобами СЗІ.

Таким чином, при зміні змісту матриці доступу автоматично генерується інформаційне повідомлення про подію на внутрішній мові СЗІ КС щодо синтезу управління ОУ, яким є об’єкт доступу або будь-який елемент КС.

В інформаційному повідомленні СІУ КС права доступу визначаються згідно співвідношенню (2.61), а послідовність дій визначається ОПР (операторами СЗІ) згідно співвідношення (2.62). Слід зазначити, що ідентифікатори, які визначають ключові слова, є елементами мови КС, що відповідають моделі (2.10) – (2.12). Після проведення лексичного та семантичного аналізу інформаційне повідомлення перетворюється в ланцюжок дій по переведенню КС в новий стан, що означає переміщення ОУ в просторі $G_{ріш}$ при виникненні конфлікту взаємодії з відкритою множиною ОС.

Тим самим СІУ ОУ, якою є СЗІ КС, має бути топологічно та функціонально побудована як СІУ [159, 163].

Висновки до другого розділу

1. При дослідженні конфліктів в технічних системах запропоновано застосовувати методологію: системно-структурного аналізу конфлікту, як дослідження явища в цілому, яке складається з системи підструктур, які, у свою чергу, складаються з елементів, і в якості підсистем входить в систему більш високого рівня; системно-функціонального аналізу конфлікту, як дослідження з метою визначення усіх основних взаємозв'язків конфлікту із зовнішнім середовищем, в якому конфлікт розвивається, у виявленні характеру і способів впливу одних елементів і підструктур конфлікту на інші (біфуркації).

2. Для досліджень доцільним є використання інтегрально-топологічних методів аналізу складних нелінійних систем, які базуються на використанні математичних моделей опису властивостей процесів, пов'язаних з об'єктами як системами, що дозволяє отримати топологічні структури просторів спостереження і пошуку, поведінки та рішення для синтезу та вибору управлінь об'єктом.

3. Інтегрально-топологічні методи дослідження та аналізу складних нелінійних систем в просторі спостереження і пошуку дозволяють виявити характеристики взаємодії кінематичної моделі ОУ та ОС, які включають параметри безпечного розв'язання конфлікту при синтезі та виборі рішень з врахуванням невизначеностей.

4. Запропоновано аксіоматику, яка є похідною при вирішенні задач синтезу та вибору рішень при розв'язанні задачі конфлікту для відкритої множини ОС.

5. Запропоновано метод формального опису ПС при довільній системі обмежень.

6. Запропоновано концептуальну структурну схему системи ситуаційного управління ОУ при вирішенні задачі конфлікту.

7. Розглянута система обмежень топологічного ПС, як n-мірної довільної неопуклої поверхні простору, та визначена топологічність його перерізів.

8. Виходячи з топологічності ПС, зпропоновано теоретико-множинну модель взаємодії об'єктів, на підставі якої отримано модель конфлікту їх взаємодії в умовах обмежень і невизначеностей.

9. Визначено доцільність застосування семіотичної моделі опису взаємодії ОУ з варіативною множиною ОС в ПС за умов обмежень і невизначеностей.

10. Виходячи з семіотичної моделі опису взаємодії ОУ з варіативною множиною ОС в ПС за умов обмежень і невизначеностей запропонована семантична модель аналізу повідомлень СІУ об’єктом.

11. На основі результатів семантичного аналізу повідомлень СІУ ОУ запропоновано семантичний підхід щодо синтезу рішень конфлікту в ІП СІУ.

12. Задачу синтезу управління сформульовано як задачу розв'язання конфлікту взаємодії ОУ з ОС в ПС відповідно принципу оптимальності та правилу зупинки.

13. Запропоновано адитивну форму критерію оптимальності при синтезі рішень конфлікту.

14. Визначено можливість використання запропонованої теоретико-множинної моделі конфлікту взаємодії ОУ з варіативною множиною ОС в умовах обмежень та невизначеностей для рішення конфлікту взаємодії КС в КП.

Отримані результати відрізняються від відомих здатністю визначати:

- *NP*-повну модель рішення задачі дискретної динамічної оптимізації, що дозволяє отримати поліноміальний алгоритм синтезу рішень конфлікту (траєкторій переміщення) ОУ;

- модель рішення конфлікту в класі P та *NP*-алгоритмів в термінах машин Тюрінга.

За рахунок вперше розробленої концепції рішення конфлікту на основі моделей і методів дискретної оптимізації при виділенні скінченних областей значень параметрів, які можуть бути розривними, та кінцевої кількості варіантів рішення з врахуванням критерію оптимальності отримані результати дозволяють визначити:

- принципи рішення конфлікту великої розмірності з високим рівнем абстракції в гарантовано оціненому простору рішень для скінченних розривних областей параметрів математичної моделі, довільній системі обмежень та варіативній множині об’єктів спостереження;

- властивості методу рішення конфлікту при множинному уявленні простору рішення, множин параметрів, які визначають переміщення об’єкта управління за синтезованою траєкторією, при умові гарантованого управління в класі поліноміальних алгоритмів;

- метод рішення конфлікту, як задачі динамічної дискретної оптимізації.

Основні результати розділу опубліковані у роботах [132, 133, 135, 137, 140, 146, 151, 156, 158, 168, 169, 172].

РОЗДІЛ 3. ОСОБЛИВОСТІ ФОРМУВАННЯ ПРОСТОРУ РЕШЕНЬ КОНФЛІКТУ ВЗАЄМОДІЇ ОБ'ЄКТУ УПРАВЛІННЯ З ВАРІАТИВНОЮ МНОЖИНОЮ ОБ’ЄКТІВ СПОСТЕРЕЖЕННЯ В УМОВАХ ОБМЕЖЕНЬ ТА НЕВИЗНАЧЕНОСТЕЙ

3.1. Інформаційна множина простору комбінованого управління об'єктом

Надалі будемо мати на увазі, що використовується зв'язана система координат, центр якої розміщено в центрі мас ОУ, а усі конфліктуючі об'єкти розглядаються як точкові.

Розрахунок та побудова ПКУО має місце щодо об'єктів, множина положень або станів яких утворює деяку кінцеву або нескінчену область на площині або в перерізі простору Q.

Розглянемо спосіб розрахунку інформаційного образу ПКУО при синтезі стратегій комбінованого управління ОУ за напрямком переміщення та швидкістю згідно співвідношенню (2.32) для перерізу площиною або двомірного простору Q.

При розрахунку ПКУО для кожного ОС здійснимо декомпозицію задачі на два рівні: обчислення множин (діапазонів) припустимих швидкостей при обраному (заданому) напрямку переміщення ОУ в просторі Q; об'єднання множин припустимих швидкостей на множині обраних (припустимих) напрямків переміщення ОУ.

При здійсненні розрахунків введемо значення часу затримки τ. Значення τ враховує час виконання обчислювальних процедур, прийняття рішення щодо обраного напрямку руху та швидкості в діапазоні припустимих швидкостей ОУ $V^i = \left[V^i_{\min}, V^i_{\max}\right]$. Координати i-го ОС $A_i = \left(x^i_a, y^i_a\right)$ визначені в зв'язаній з ОУ системі декартових координат для випадку відносного руху (рис. 3.1).

Для *i*-го ОС визначимо координати вектору швидкості $\overrightarrow{V_c^i} = \left(x_c^i, y_c^i\right)$ в режимі істинного руху.

Формально ПКУО можна визначити як

$$\aleph = \bigcup_{N, \Psi_{прип}} V_{\psi_j}^i, \forall \psi_j \in \Psi_{прип}, i = \overline{1, N}, \forall V_{\psi_j}^i \subset V_e, \qquad (3.1)$$

де $\aleph$ - ПКУО; $\Psi_{прип}$ - множина напрямків переміщення; $V_{\psi_j}^i$ - множина припустимих швидкостей переміщення для *i*-го ОС за напрямком $\psi_j \in \Psi_{прип}$, яка визначається як “лінійка швидкостей” (ЛШ); N - кількість ОС на час спостереження; V_e - множина припустимих швидкостей переміщення ОУ в просторі Q.

Під терміном “лінійка швидкостей” розуміють об’єднання діапазонів припустимих швидкостей переміщення ОУ в ПР за обраним напрямком в межах діапазону припустимих швидкостей об’єкту, при яких не виникне неприпустиме зближення при взаємодії з ОС.

Згідно співвідношень (2.32) і (3.1) ПКУО являє множину, яка утворена в результаті перетину множин припустимих швидкостей переміщення ОУ по відношенню до *i*-го ОС за напрямками припустимого переміщення ОУ.

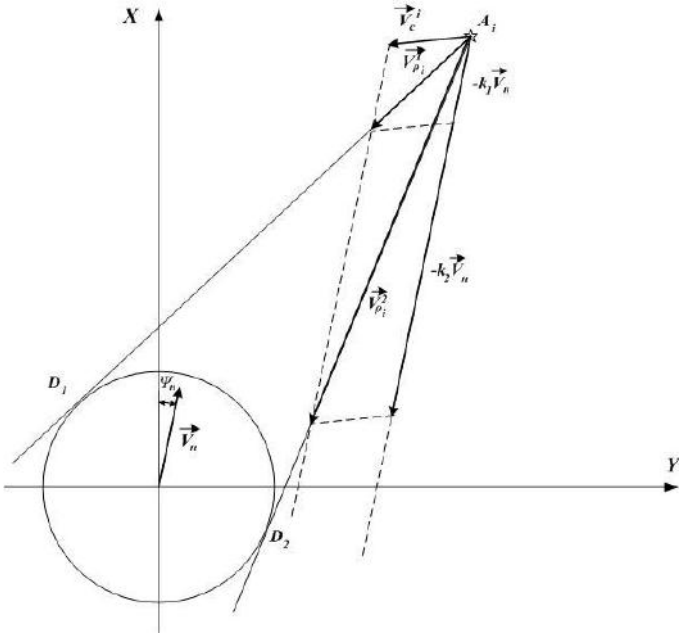


Рис. 3.1. Визначення діапазону припустимих швидкостей ОУ при взаємодії з ОС

Тим самим ПКУО буде являти об'єднання за напрямками переміщення окремих множин припустимих швидкостей ОУ відносно всіх ОС на поточний момент часу .

Виходячи з геометричної інтерпретації способу визначення ПКУО, отримаємо евристичне вирішуюче правило у вигляді предикативної структури.

Побудуємо з початку координат, як із центру, коло радіусу d^i, який дорівнює припустимій відстані зближення ОУ з *i*-м ОС. Відобразимо точкою згладжені координати *i*-го ОС. Проведемо з точки A^i дотичні (D_1A_i) та (D_2A_i) до побудованого кола, які за своїм геометричним змістом є гранично припустимими лініями відносного руху (ЛВР) і забезпечують переміщення ОУ відносно *i*-го ОС на відстані, яка не є меншою від d^i. З початку координат будуємо вектор істинної швидкості ОУ. З точки A_i будуємо вектор власної швидкості *i*-го ОС. Через точку A_i проводимо лінію, яка є паралельною до вектора $\overrightarrow{V_н}$. В точці будуємо трикутник швидкостей для отримання вектору відносної швидкості, спрямованого по ЛВР1 та ЛВР2, змінивши модуль, а при необхідності і напрямок вектору $\overrightarrow{V_н}$ на протилежний. Таким чином, отримуємо значення множників при векторі $\overrightarrow{V_н}$, які задовольняють умовам

$$\begin{cases} \left(\overrightarrow{V_c^i} - n_1\overrightarrow{V_н}\right)\overrightarrow{A_iD_1} = 0 \\ \left(\overrightarrow{V_c^i} - n_2\overrightarrow{V_н}\right)\overrightarrow{A_iD_2} = 0 \end{cases}, \tag{3.2}$$

де n_1 та n_2 - коефіцієнти, які визначають зміну значення вектору $\overrightarrow{V_н}$ для побудови вектору відносної швидкості $\overrightarrow{V_{\rho_i}^1}$ та $\overrightarrow{V_{\rho_i}^2}$, які відповідно забезпечують переміщення вздовж ліній ЛВР1 та ЛВР2. Тобто

$$\begin{cases} \overrightarrow{V_{\rho_i}^1} = \overrightarrow{V_c^i} - n_1\overrightarrow{V_н} \\ \overrightarrow{V_{\rho_i}^2} = \overrightarrow{V_c^i} - n_2\overrightarrow{V_н} \end{cases}. \tag{3.3}$$

Виходячи з співвідношення (3.2) визначимо коефіцієнти n_1 та n_2

$$\begin{cases} n_1 = \dfrac{\overrightarrow{A_i D_1 V_c^i}}{\overrightarrow{A_i D_2 V_н}} \\ n_2 = \dfrac{\overrightarrow{A_i D_2 V_c^i}}{\overrightarrow{A_i D_2 V_н}} \end{cases}. \tag{3.4}$$

Аналіз можливостей визначення вектору відносної швидкості переміщення *i*-го ОС вздовж ліній ЛВР1 та ЛВР2 дозволяє зробити висновок про те, що в окремих випадках співпадіння векторів $\overrightarrow{A_i D_1}$ та $\overrightarrow{A_i D_2}$ з векторами $\overrightarrow{V_{\rho_i}^1}$ та $\overrightarrow{V_{\rho_i}^2}$ можна отримати по два вектора $\overrightarrow{V_{\rho_i}^1}$ та $\overrightarrow{V_{\rho_i}^2}$ відповідно для кожної ЛВР. що є характерним для випадків зустрічного зближення та зближення при переслідуванні ОС. Тим самим коефіцієнти n_1 та n_2 визначаються множинами.

$$\begin{cases} n_1 = \left\{ n_1^1, n_1^2 \right\} \\ n_2 = \left\{ n_2^1, n_2^2 \right\} \end{cases}. \tag{3.5}$$

Значення коефіцієнтів n_1^1, n_1^2, n_2^1, n_2^2 в співвідношенні (3.5) визначаються згідно співвідношенню (3.4) для випадків зустрічного зближення та зближення при переслідуванні.

В такому разі маємо

$$\begin{cases} \overrightarrow{V_{н_1}^j} = n_1^j \overrightarrow{V_н} \\ \overrightarrow{V_{н_2}^j} = n_2^j \overrightarrow{V_н} \end{cases}, j = \overline{1,2}. \tag{3.6}$$

З врахуванням (3.6) співвідношення (3.3) для *i*-го ОС прийме вигляд

$$\begin{cases} \overrightarrow{V_{\rho_i^j}^1} = \overrightarrow{V_c^i} - n_1 \overrightarrow{V_н} \\ \overrightarrow{V_{\rho_i^j}^2} = \overrightarrow{V_c^i} - n_2 \overrightarrow{V_н} \end{cases}, j = \overline{1,2}. \tag{3.7}$$

Визначимо коефіцієнт, який визначає взаємне положення векторів, які співпадають з ЛВР і напрямком відносної швидкості переміщення *i*-го ОС

$$\begin{cases} k_1^j = 1 \times Sign(\overrightarrow{A_i D_1} \times \overrightarrow{V_{\rho_i^j}^1}) \\ k_2^j = 1 \times Sign(\overrightarrow{A_i D_2} \times \overrightarrow{V_{\rho_i^j}^2}) \end{cases}, j = \overline{1,2}. \tag{3.8}$$

Виходячи з співвідношень (3.2) - (3.8), визначимо евристичне вирішуюче правило формування інтервалів безпечних швидкостей $V_e^i \subset [0, V_{\max}]$ переміщення ОУ щодо *i*-го ОС згідно співвідношення (3.6).

Інтервали безпечних швидкостей для ОУ визначаються набором предикатів, кожен з яких надає однозначну відповідь про припустимі швидкості переміщення.

$$\left\{\left\{\left(n_1^j > 0\right) \wedge \left(k_1^j > 0\right) \wedge \left(n_2^j > 0\right) \wedge \left(k_2^j > 0\right) \wedge \left(\left|\overrightarrow{V_{н_1}^j}\right| < \left|\overrightarrow{V_{н_2}^j}\right|\right)\right\} \wedge \left(\left|\overrightarrow{V_{н_1}^j}\right| < V_{\max}\right)\right\} \mapsto$$
$$\left\langle V_e \subset \left\{\left[0, \left|\overrightarrow{V_{н_1}^j}\right|\right] \cup \left[\left|\overrightarrow{V_{н_2}^j}\right|, V_{\max}\right]\right\}\right\rangle, j = \overline{1,2} \quad , \quad 3.9)$$

$$\left\{\left\{\left(n_1^j > 0\right) \wedge \left(k_1^j > 0\right) \wedge \left(n_2^j > 0\right) \wedge \left(k_2^j > 0\right) \wedge \left(\left|\overrightarrow{V_{н_1}^j}\right| \geq \left|\overrightarrow{V_{н_2}^j}\right|\right)\right\} \wedge \left(\left|\overrightarrow{V_{н_1}^j}\right| < V_{\max}\right)\right\} \mapsto$$
$$\left\langle V_e \subset \left\{\left[0, \left|\overrightarrow{V_{н_2}^j}\right|\right] \cup \left[\left|\overrightarrow{V_{н_1}^j}\right|, V_{\max}\right]\right\}\right\rangle, j = \overline{1,2} \quad , \quad 3.10)$$

$$\left\{\left\{\left(n_1^j > 0\right) \wedge \left(k_1^j > 0\right) \wedge \left(n_2^j > 0\right) \wedge \left(k_2^j > 0\right) \wedge \left(\left|\overrightarrow{V_{н_1}^j}\right| < \left|\overrightarrow{V_{н_2}^j}\right|\right)\right\} \wedge \left(\left|\overrightarrow{V_{н_1}^j}\right| \geq V_{\max}\right)\right\} \mapsto$$
$$\left\langle V_e \subset [0, V_{\max}]\right\rangle, j = \overline{1,2} \quad , (3.11)$$

$$\left\{\left\{\left(n_1^j > 0\right) \wedge \left(k_1^j > 0\right) \wedge \left(n_2^j > 0\right) \wedge \left(k_2^j > 0\right) \wedge \left(\left|\overrightarrow{V_{н_1}^j}\right| < \left|\overrightarrow{V_{н_2}^j}\right|\right)\right\} \wedge \left(\left|\overrightarrow{V_{н_2}^j}\right| > V_{\max}\right)\right\} \mapsto$$
$$\left\langle V_e \subset [0, V_{\max}]\right\rangle, j = \overline{1,2} \quad , \quad (3.12)$$

$$\left\{\left\{\left(n_1^j > 0\right) \wedge \left(k_1^j > 0\right) \wedge \left(n_2^j > 0\right) \wedge \left(k_2^j > 0\right) \wedge \left(\left|\overrightarrow{V_{н_1}^j}\right| \leq \left|\overrightarrow{V_{н_2}^j}\right|\right)\right\} \wedge \left(\left|\overrightarrow{V_{н_1}^j}\right| < V_{\max} < \left|\overrightarrow{V_{н_2}^j}\right|\right)\right\} \mapsto$$
$$\left\langle V_e \subset \left\{\left[0, \left|\overrightarrow{V_{н_1}^j}\right|\right] \cup \left[\left|\overrightarrow{V_{н_2}^j}\right|, V_{\max}\right]\right\}\right\rangle, j = \overline{1,2} \quad , \quad (3.13)$$

$$\left\{\left\{\left(n_1^j > 0\right) \wedge \left(k_1^j > 0\right) \wedge \left(n_2^j > 0\right) \wedge \left(k_2^j > 0\right) \wedge \left(\left|\overrightarrow{V_{н_1}^j}\right| > \left|\overrightarrow{V_{н_2}^j}\right|\right)\right\} \wedge \left(\left|\overrightarrow{V_{н_2}^j}\right| < V_{\max} < \left|\overrightarrow{V_{н_1}^j}\right|\right)\right\} \mapsto$$
$$\left\langle V_e \subset \left\{\left[0, \left|\overrightarrow{V_{н_2}^j}\right|\right] \cup \left[\left|\overrightarrow{V_{н_1}^j}\right|, V_{\max}\right]\right\}\right\rangle, j = \overline{1,2} \quad , \quad (3.14)$$

$$\left\{\left\{\left(n_1^j > 0\right) \wedge \left(k_1^j > 0\right) \wedge \left(n_2^j > 0\right) \wedge \left(k_2^j \leq 0\right) \wedge \left(\left|\overrightarrow{V_{н_1}^j}\right| < \left|\overrightarrow{V_{н_2}^j}\right|\right)\right\} \wedge \left(\left|\overrightarrow{V_{н_2}^j}\right| < V_{\max}\right)\right\} \mapsto$$
$$\left\langle V_e \subset \left[\left|\overrightarrow{V_{н_1}^j}\right|, V_{\max}\right]\right\rangle, j = \overline{1,2} \quad ,(3.15)$$

$$\left\{\left\{\left(n_1^j>0\right)\wedge\left(k_1^j\le 0\right)\wedge\left(n_2^j>0\right)\wedge\left(k_2^j>0\right)\wedge\left(\left|\overrightarrow{V_{H_1}^j}\right|<\left|\overrightarrow{V_{H_2}^j}\right|\right)\right\}\wedge\left(\left|\overrightarrow{V_{H_2}^j}\right|>V_{\max}\right)\right\}\mapsto \langle V_e\subset\varnothing\rangle,\ j=\overline{1,2} \quad (3.16)$$

$$\left\{\left\{\left(n_1^j>0\right)\wedge\left(k_1^j>0\right)\wedge\left(n_2^j>0\right)\wedge\left(k_2^j\le 0\right)\wedge\left(\left|\overrightarrow{V_{H_1}^j}\right|>\left|\overrightarrow{V_{H_2}^j}\right|\right)\right\}\wedge\left(\left|\overrightarrow{V_{H_1}^j}\right|<V_{\max}\right)\right\}\mapsto \left\langle V_e\subset\left[\left|\overrightarrow{V_{H_1}^j}\right|,V_{\max}\right]\right\rangle,\ j=\overline{1,2} \quad (3.17)$$

$$\left\{\left\{\left(n_1^j>0\right)\wedge\left(k_1^j\le 0\right)\wedge\left(n_2^j>0\right)\wedge\left(k_2^j>0\right)\wedge\left(\left|\overrightarrow{V_{H_1}^j}\right|>\left|\overrightarrow{V_{H_2}^j}\right|\right)\right\}\wedge\left(\left|\overrightarrow{V_{H_1}^j}\right|>V_{\max}\right)\right\}\mapsto \langle V_e\subset\varnothing\rangle,\ j=\overline{1,2} \quad (3.18)$$

$$\left\{\begin{array}{l}\left[\left(n_1^j>0\right)\wedge\left(n_2^j>0\right)\wedge\left(k_1^j\le 0\right)\wedge\left(k_2^j<0\right)\right]\vee\\ \left[\left(n_1^j<0\right)\wedge\left(n_2^j<0\right)\wedge\left(k_1^j\le 0\right)\wedge\left(k_2^j\le 0\right)\right]\vee\\ \left[\left(n_1^j<0\right)\wedge\left(n_2^j>0\right)\wedge\left(k_1^j\le 0\right)\wedge\left(k_2^j<0\right)\right]\vee\\ \left[\left(n_1^j<0\right)\wedge\left(n_2^j<0\right)\wedge\left(k_1^j>0\right)\wedge\left(k_2^j<0\right)\right]\vee\\ \left[\left(n_1^j>0\right)\wedge\left(n_2^j<0\right)\wedge\left(k_1^j<0\right)\wedge\left(k_2^j>0\right)\right]\end{array}\right\}\mapsto\langle V_e\subset[0,V_{\max}]\rangle,\ j=\overline{1,2}, \quad (3.19)$$

$$\left\{\begin{array}{l}\left\langle\begin{array}{l}\left[\left(n_1^j>0\right)\wedge\left(n_2^j<0\right)\wedge\left(k_1^j\ge 0\right)\wedge\left(k_2^j\ge 0\right)\right]\vee\\ \left[\left(n_1^j<0\right)\wedge\left(n_2^j>0\right)\wedge\left(k_1^j>0\right)\wedge\left(k_2^j\ge 0\right)\right]\end{array}\right\rangle\wedge\\ \left(\left|\overrightarrow{V_{H_1}^j}\right|\le V_{\max}\right)\end{array}\right\}\mapsto\left\langle V_e\subset\left[\left|\overrightarrow{V_{H_1}^j}\right|,V_{\max}\right]\right\rangle,\ j=\overline{1,2}, \quad (3.20)$$

$$\left\{\begin{array}{l}\left\langle\begin{array}{l}\left[\left(n_1^j>0\right)\wedge\left(n_2^j<0\right)\wedge\left(k_1^j\ge 0\right)\wedge\left(k_2^j\ge 0\right)\right]\vee\\ \left[\left(n_1^j<0\right)\wedge\left(n_2^j>0\right)\wedge\left(k_1^j>0\right)\wedge\left(k_2^j\ge 0\right)\right]\end{array}\right\rangle\wedge\\ \left[\left(\left|\overrightarrow{V_{H_2}^j}\right|<\left|\overrightarrow{V_{H_1}^j}\right|\right)\wedge\left(\left|\overrightarrow{V_{H_2}^j}\right|\le V_{\max}\right)\right]\end{array}\right\}\mapsto\left\langle V_e\subset\left[\left|\overrightarrow{V_{H_2}^j}\right|,V_{\max}\right]\right\rangle,\ j=\overline{1,2}, \quad (3.21)$$

$$\left\{\begin{array}{l}\left\langle\begin{array}{l}\left[\left(n_1^j>0\right)\wedge\left(n_2^j<0\right)\wedge\left(k_1^j\ge 0\right)\wedge\left(k_2^j>0\right)\right]\vee\\ \left[\left(n_1^j<0\right)\wedge\left(n_2^j>0\right)\wedge\left(k_1^j>0\right)\wedge\left(k_2^j\ge 0\right)\right]\end{array}\right\rangle\wedge\\ \left[\left(\left|\overrightarrow{V_{H_2}^j}\right|>\left|\overrightarrow{V_{H_1}^j}\right|\right)\wedge\left(\left|\overrightarrow{V_{H_1}^j}\right|\le V_{\max}\right)\right]\end{array}\right\}\mapsto\langle V_e\subset\varnothing\rangle,\ j=\overline{1,2}, \quad (3.22)$$

$$\left\{\begin{array}{l}\left\langle\begin{array}{l}\left[\left(n_1^j>0\right)\wedge\left(n_2^j<0\right)\wedge\left(k_1^j\geq 0\right)\right]\vee\\ \left[\left(n_1^j<0\right)\wedge\left(k_1^j>0\right)\wedge\left(k_2^j\geq 0\right)\right]\end{array}\right\rangle\wedge\\ \left[\left(\left|\overrightarrow{V_{H_2}^j}\right|<\left|\overrightarrow{V_{H_1}^j}\right|\right)\wedge\left(\left|\overrightarrow{V_{H_2}^j}\right|\leq V_{\max}\right)\right]\end{array}\right\}\mapsto\left\langle V_e\subset\varnothing\right\rangle,\ j=\overline{1,2},\tag{3.23}$$

$$\left\{\begin{array}{l}\left\langle\begin{array}{l}\left[\left(n_1^j>0\right)\wedge\left(n_2^j<0\right)\wedge\left(k_1^j>0\right)\wedge\left(k_2^j\geq 0\right)\right]\vee\\ \left[\left(n_1^j<0\right)\wedge\left(n_2^j>0\right)\wedge\left(k_1^j\geq 0\right)\wedge\left(k_2^j>0\right)\right]\end{array}\right\rangle\wedge\\ \left[\wedge\left(\left|\overrightarrow{V_{H_1}^j}\right|<\left|\overrightarrow{V_{H_2}^j}\right|<V_{\max}\right)\right]\end{array}\right\}\mapsto\left\langle V_e\subset\left[0,\left|\overrightarrow{V_{H_1}^j}\right|\right]\right\rangle,\ j=\overline{1,2},\tag{3.24}$$

$$\left\{\begin{array}{l}\left\langle\begin{array}{l}\left[\left(n_1^j>0\right)\wedge\left(n_2^j<0\right)\wedge\left(k_1^j>0\right)\wedge\left(k_2^j\geq 0\right)\right]\vee\\ \left[\left(n_1^j<0\right)\wedge\left(n_2^j>0\right)\wedge\left(k_1^j\geq 0\right)\wedge\left(k_2^j>0\right)\right]\end{array}\right\rangle\wedge\\ \left[\left(\left|\overrightarrow{V_{H_2}^j}\right|<\left|\overrightarrow{V_{H_1}^j}\right|\right)\wedge\left(\left|\overrightarrow{V_{H_1}^j}\right|\leq V_{\max}\right)\right]\end{array}\right\}\mapsto\left\langle V_e\subset\left[0,\left|\overrightarrow{V_{H_1}^j}\right|\right]\right\rangle,\ j=\overline{1,2},\tag{3.25}$$

$$\left\{\begin{array}{l}\left\langle\begin{array}{l}\left[\left(n_1^j>0\right)\wedge\left(n_2^j<0\right)\wedge\left(k_1^j>0\right)\wedge\left(k_2^j\geq 0\right)\right]\vee\\ \left[\left(n_1^j<0\right)\wedge\left(n_2^j>0\right)\wedge\left(k_1^j\geq 0\right)\wedge\left(k_2^j>0\right)\right]\end{array}\right\rangle\wedge\\ \left[\left(\left|\overrightarrow{V_{H_1}^j}\right|<\left|\overrightarrow{V_{H_2}^j}\right|\right)\wedge\left(\left|\overrightarrow{V_{H_2}^j}\right|>V_{\max}\right)\right]\end{array}\right\}\mapsto\left\langle V_e\subset\left[0,V_{\max}\right]\right\rangle,\ j=\overline{1,2},\tag{3.26}$$

$$\left\{\begin{array}{l}\left\langle\begin{array}{l}\left[\left(n_1^j>0\right)\wedge\left(n_2^j<0\right)\wedge\left(k_1^j>0\right)\wedge\left(k_2^j\geq 0\right)\right]\vee\\ \left[\left(n_1^j<0\right)\wedge\left(n_2^j>0\right)\wedge\left(k_1^j\geq 0\right)\wedge\left(k_2^j>0\right)\right]\end{array}\right\rangle\wedge\\ \left[\left|\overrightarrow{V_{H_2}^j}\right|<\left|\overrightarrow{V_{H_1}^j}\right|\leq V_{\max}\right]\end{array}\right\}\mapsto\left\langle V_e\subset\left[0,\left|\overrightarrow{V_{H_1}^j}\right|\right]\right\rangle,\ j=\overline{1,2},\tag{3.27}$$

$$\left\{\begin{array}{l}\left\langle\begin{array}{l}\left[\left(n_1^j>0\right)\wedge\left(n_2^j<0\right)\wedge\left(k_1^j\leq 0\right)\wedge\left(k_2^j<0\right)\right]\vee\\ \left[\left(n_1^j<0\right)\wedge\left(n_2^j>0\right)\wedge\left(k_1^j<0\right)\wedge\left(k_2^j\leq 0\right)\right]\end{array}\right\rangle\wedge\\ \left[\left(\left|\overrightarrow{V_{H_2}^j}\right|>\left|\overrightarrow{V_{H_1}^j}\right|\right)\wedge\left(\left|\overrightarrow{V_{H_2}^j}\right|\leq V_{\max}\right)\right]\end{array}\right\}\mapsto\left\langle V_e\subset\varnothing\right\rangle,\ j=\overline{1,2},\tag{3.28}$$

$$\left\{\begin{array}{l}\left\langle\begin{array}{l}\left[\left(n_1^j>0\right)\wedge\left(n_2^j<0\right)\wedge\left(k_1^j>0\right)\wedge\left(k_2^j<0\right)\right]\vee\\ \left[\left(n_1^j<0\right)\wedge\left(n_2^j>0\right)\wedge\left(k_1^j<0\right)\wedge\left(k_2^j>0\right)\right]\end{array}\right\rangle\wedge\\ \left[\left|\overrightarrow{V_{H_1}^j}\right|<\left|\overrightarrow{V_{H_2}^j}\right|<V_{\max}\right]\end{array}\right\}\mapsto\left\langle V_e\subset\left[0,\left|\overrightarrow{V_{H_2}^j}\right|\right]\right\rangle,\ j=\overline{1,2},\tag{3.29}$$

$$\left\{\begin{array}{l}\left\langle\begin{array}{l}\left[\left(n_1^j>0\right)\wedge\left(n_2^j<0\right)\wedge\left(k_1^j>0\right)\wedge\left(k_2^j<0\right)\right]\vee\\ \left[\left(n_1^j<0\right)\wedge\left(n_2^j>0\right)\wedge\left(k_1^j<0\right)\wedge\left(k_2^j>0\right)\right]\end{array}\right\rangle\wedge\\ \left[\left(\left|\overrightarrow{V_{н1}^j}\right|<\left|\overrightarrow{V_{н_2}^j}\right|\right)\wedge\left(\left|\overrightarrow{V_{н_2}^j}\right|>V_{\max}\right)\right]\end{array}\right\}\mapsto\langle V_e\subset[0,V_{\max}]\rangle,\ j=\overline{1,2},\qquad(3.30)$$

$$\left\{\begin{array}{l}\left\langle\begin{array}{l}\left[\left(n_1^j>0\right)\wedge\left(n_2^j<0\right)\wedge\left(k_1^j>0\right)\wedge\left(k_2^j<0\right)\right]\vee\\ \left[\left(n_1^j<0\right)\wedge\left(n_2^j>0\right)\wedge\left(k_1^j<0\right)\wedge\left(k_2^j>0\right)\right]\end{array}\right\rangle\wedge\\ \left[\left|\overrightarrow{V_{н_2}^j}\right|<\left|\overrightarrow{V_{н_1}^j}\right|<V_{\max}\right]\end{array}\right\}\mapsto\left\langle V_e\subset\left[0,\left|\overrightarrow{V_{н_1}^j}\right|\right]\right\rangle,\ j=\overline{1,2},\qquad(3.31)$$

$$\left\{\begin{array}{l}\left\langle\begin{array}{l}\left[\left(n_1^j>0\right)\wedge\left(n_2^j<0\right)\wedge\left(k_1^j>0\right)\wedge\left(k_2^j<0\right)\right]\vee\\ \left[\left(n_1^j<0\right)\wedge\left(n_2^j>0\right)\wedge\left(k_1^j<0\right)\wedge\left(k_2^j>0\right)\right]\end{array}\right\rangle\wedge\\ \left[\left(\left|\overrightarrow{V_{н_2}^j}\right|<\left|\overrightarrow{V_{н_1}^j}\right|\right)\wedge\left(\left|\overrightarrow{V_{н_1}^j}\right|>V_{\max}\right)\right]\end{array}\right\}\mapsto\langle V_e\subset[0,V_{\max}]\rangle,\ j=\overline{1,2},\qquad(3.32)$$

Вирішуюче правило, яке визначено співвідношеннями (3.9) - (3.32) описує всі можливі стани взаємодії ОУ з ОС для зустрічного зближення та зближення при переслідуванні (задач пошуку і переслідування), а також синтезу рішень при розв'язанні конфлікту. Означене вирішуюче правило є прийнятним для динамічних, кінематичних і віртуальних (інформаційних) об'єктів в фізичному та інформаційному (віртуальному) просторі. Це досягається завдяки врахуванню можливості зближення ОУ з *i*-м ОС при перетині границі кола з радіусом d^i для *i*-го ОС (рис. 3.1). При вході в межі та виході за межі кола при його перетині ЛВР точка входу визначає вирішення для задачі пошуку, а точка виходу визначає вирішення для задачі переслідування.

Слід зазначити, що в перерізах простору Q площинами параметрів маємо просторове відображення множини фазових траєкторій управління ОУ.

3.2. Простір рішень

Вирішення задачі конфлікту взаємодії об'єктів в просторі Q здійснюється в ПР - $G_{ріш}$.

Для спрощення представлення та опису ПР, як багатомірного топологічного простору, здійснимо декомпозицію ПР, використовуючи перерізи площинами параметрів для визначення особливостей (рис. 3.2) процесу взаємодії ОУ з ОС при пошуку та переслідуванні.

При формальному описі простору $G_{ріш}$ скористуємося інформацією щодо параметрів, які визначають стан взаємодії ОУ з довільним ОС, а саме: відстань до об'єкта, кут спостереження, припустиме зближення з ОС, екстрапольований час найкоротшого зближення, істинний напрямок переміщення, істинна швидкість переміщення. Зазначені параметри переміщення ОС є похідними для синтезу стратегій та вибору рішення конфлікту взаємодії з ОУ. Параметри переміщення ОС визначаються з геометричних співвідношень в топологічному просторі Q або його перерізі площиною параметрів.

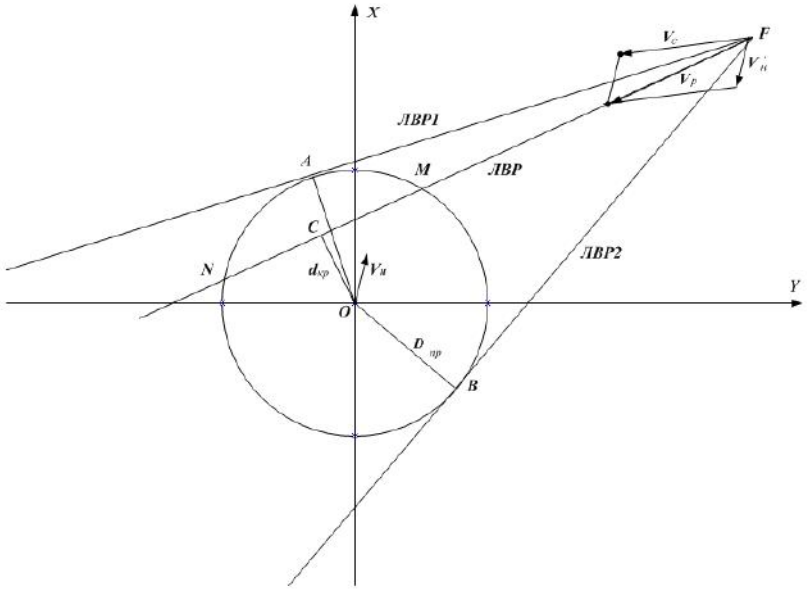


Рис. 3.2. Представлення ситуації зближення ОУ з ОС

Побудуємо ЛВР, яка співпадає з вектором $\overrightarrow{V_\rho}$ відносної швидкості при зближенні з ОС. З трикутника швидкостей отримуємо значення вектора істинної швидкості ОС

$$\overrightarrow{V_c} = \overrightarrow{V_\rho} - \overrightarrow{V_н}. \qquad (3.33)$$

Співвідношення (3.33) є векторним відображенням трикутника швидкостей. Приймемо $\overrightarrow{V_н'} = -\overrightarrow{V_н}$. Довжина перпендикуляру $|OC|$, який опущено з початку координат на ЛВР, є відстанню найкоротшого зближення $d_{кр}$ ОУ з ОС. Час зближення визначається як частка від ділення модуля вектору, проведеного від ОС - $\left|\overrightarrow{FC}\right|$, на $\left|\overrightarrow{V_\rho}\right|$, тобто

$$t_{збл} = \frac{\left|\overrightarrow{FC}\right|}{\left|\overrightarrow{V_\rho}\right|}. \qquad (3.34)$$

Співвідношення (3.34) визначає параметр часу для розрахунку сектору небезпечних напрямків переміщень (СННП) ОУ.

Побудуємо з початку координат, як з центру, коло радіусу, який дорівнює припустимій відстані зближення ОУ з ОС $D_{пр}$. Тоді рішення задачі конфлікту взаємодії при переміщенні ОУ і ОС в просторі Q буде зводитися до визначення такого відхилення від початкового напрямку траєкторії переміщення ОУ, при якому ЛВР можна буде у самому гіршому випадку направити вздовж дотичних ЛВР1 та ЛВР2, що проведені з точки $F(x_f, y_f)$ до кола радіусу $D_{пр}$. В такому разі стратегія запобігання небезпечного зближення ОУ з ОС визначається параметрами зближення при зміні напрямку переміщення і значенням швидкості ОУ. При цьому очевидним є ускладнення рішення задачі для варіативної множини ОС.

Додаткові можливості для вирішення конфлікту надають відомості про СННП, який за своєю сутністю є множиною напрямів переміщення ОУ, що призводять до зближення з ОС на відстань меншу $D_{пр}$ у припущенні рівномірного і прямолінійного руху ОС.

Виходячи з геометричних уявлень, визначимо спосіб розрахунку СННП.

Визначення СННП зводиться до розгляду таких змін напрямку чи модуля вектору швидкості $\overrightarrow{V_н}$ ОУ, при яких ЛВР розвернеться на кут $\alpha = [\angle AFN, \angle BFN]$, який забезпечує додержання значення $D_{пр}$ при взаємному

переміщенні ОУ та ОС. При цьому визначається напрямок повороту ЛВР та значення кута α зміни напрямку траєкторії переміщення ОУ в просторі Q.

Обидва крайніх положення вектору швидкості ОУ, які визначають потрібну зміну напряму ЛВР, розділяють коло радіусу $\left|\overrightarrow{V_н}\right|$ відрізком прямої лінії (MN) на два сектори.

Виходячи з геометричного уявлення щодо СННП визначимо аналітичний вираз для розрахунку його параметрів (рис. 3.3).

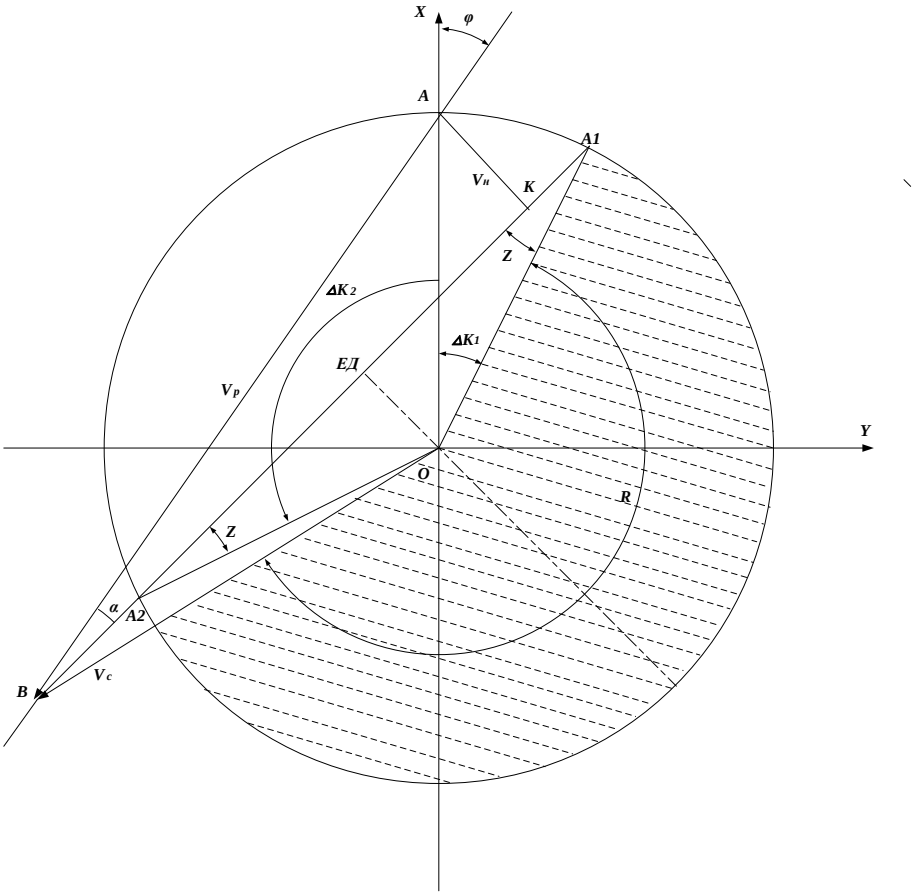


Рис. 3.3. Геометричні уявлення СННП

У першому секторі (повністю заштрихований) усі проміжні значення нових напрямків переміщення визвуть розворот ЛВР на кут, який є більшим кута α. Таким чином отримуємо СННП. Розвертаючи вектор $\overrightarrow{V_н}$ у другий сектор бажаний результат не буде досягнуто.

Знайдемо аналітичний вираз для визначення СННП. Виходячи з рис. 3.3, маємо кут $\frac{R}{2}$, який утворено віссю симетрії сектору і лінією нового граничного напрямку переміщення ОУ (OA_1), який дорівнює

$$\frac{R}{2} = 180^\circ - \left(90^\circ - Z\right) = 90^\circ + Z. \tag{3.35}$$

Для спрощення введемо коефіцієнт

$$P = \frac{|OA_1|}{|OB|} = \frac{\sqrt{\left(x_{a_1} - x_o\right)^2 + \left(y_{a_1} - y_o\right)^2}}{\sqrt{\left(x_b - x_o\right)^2 + \left(y_b - y_o\right)^2}}. \tag{3.36}$$

З врахуванням співвідношень (3.35) і (3.36) кут Z визначається співвідношенням

$$\begin{cases} \sin Z = \sin\left(\varphi + \alpha\right) - P\sin\alpha \\ \Delta K_j = \varphi + \alpha - Z_j,\ j = \overline{1,2} \end{cases}. \tag{3.37}$$

З співвідношень (3.35) та (3.37) маємо

$$\cos\left(\frac{R}{2}\right) = P\sin\alpha - \sin\left(\varphi + \alpha\right). \tag{3.38}$$

Очевидним є те, що напрямок вісі симетрії СННП, якщо розраховувати його від лінії напрямку руху ОУ, буде визначатися співвідношенням

$$\psi = \varphi + \alpha + 90^\circ. \tag{3.39}$$

Межі СННП (мінімальні кути, які потрібні для зміни напрямку переміщення ОУ в просторі Q від початкового напрямку переміщення) будуть дорівнювати

$$\begin{cases} \Delta K_1 = \psi - \frac{Q}{2} \\ \Delta K_2 = \psi + \frac{Q}{2} \end{cases}. \tag{3.40}$$

Виходячи з співвідношень (3.35) - (3.39), можна визначити СННП як множину напрямків переміщення ОУ, які знаходяться в інтервалі, який визначено його межами у відповідності до співвідношення (3.40).

Аналітично можна визначити кутові швидкості, які є необхідними для зміни напрямку переміщення ОУ з метою унеможливлення зближення з ОС на припустиму безпечну відстань $d_{кр}$ в просторі Q.

Таким чином, для N ОС в просторі Q можна визначити множину значень $\Delta d^i = \left[\Delta d_1^i, \Delta d_2^2\right]$ для i-го ОС і тим самим визначити $\Delta K_j^i, j = \overline{1,2}, i = \overline{1,N}$ для співвідношенню (3.37) і (3.40).

При вирішенні задачі конфлікту взаємодії ОУ з варіативною множиною ОС в просторі Q, значний вплив на вибір стратегії рішення мають динамічні та кінематичні характеристики об'єктів-учасників конфлікту.

Виходячи з геометричної інтерпретації процесу взаємодії ОУ з відкритою множиною ОС та співвідношень (3.2) - (3.8), можна визначити просторові множини напрямків переміщення ОУ в просторі Q. На рис. 3.2 визначені точки перетину ЛВР з границею кола – точки M та N. Відрізок $|MN|$ визначає небезпечний елемент траєкторії переміщення ОУ в просторі Q.

В такому разі стає зрозумілим геометричний зміст ПР - $G_{ріш}^*$ в сенсі співвідношень (3.5) - (3.8) за умови забезпечення вимог співвідношення (3.1).

Рішення конфлікту взаємодії ОУ з варіативною множиною ОС в просторі $G_{ріш}^*$ є синтезованою траєкторією переміщення об’єкту з початкової в кінцеву термінальну позицію .

Синтез траєкторій переміщення і управління переміщенням ОУ при вирішенні конфлікту (2.34) - (2.36) пов'заний з визначенням її приналежності синтезованої траєкторії простору Q. Тобто при формальному аналізі необхідно синтезувати правило приналежності синтезованої траєкторії μ переміщення ОУ простору $G_{ріш}$.

Визначимо ознаку приналежності f_μ i-го елемента ланцюжка траєкторії переміщення ОУ j-ї стратегій синтезованих рішень (траєкторій переміщення ОУ) простору $G_{ріш}$.

$$\begin{cases} f_\mu = 1 \forall \mu_j \subset G_{piш} \\ f_\mu = 0 \forall \mu_j \not\subset G_{piш}, \\ \mu_j = \bigcup_{i=1}^{n_j} \mu_j^i, \; j = \overline{1,m} \end{cases} \tag{3.41}$$

де n_j - кількість елементів лінії траєкторії; m - кількість напрямків зміни лінії траєкторії.

Визначення ознаки приналежності у відповідності з виразом (3.41) здійснюється за правилом Π_μ, яке визначає значення ознаки у відповідності з рекурсією перебору для рівнянь лінії межі $\Gamma_{гр}(Q)$ простору $G_{piш}$. Спосіб формування ознаки приналежності f_μ у відповідності з співвідношеннями (2.4) і (3.41) на основі правила Π_μ використовує властивості векторного добутку в топологічному (евклідовому) просторі (рис. 3.4).

Визначимо межу $\Gamma_{гр}(Q)$ співвідношенням, яке описує межу як об'єднання відповідних ламаних ліній лівої та правої границі простору Q або його перетину площиною параметрів (рис. 3.4).

$$\begin{cases} \Gamma_{гр}(Q) = \Gamma_л \bigcup_{\Pi_{зам}} \Gamma_n, \forall \Gamma_л^i \subset \Gamma_л, \forall \Gamma_n^j \subset \Gamma_n, i = \overline{1, N_л}, \; j = 1, N_n \\ \Gamma_л = \bigcup_{i=1}^{N_л} \Gamma_л^i = \bigcup_{i=1}^{N_л} \left(\overline{L_i L_{i+1}} \right) \\ \Gamma_n = \bigcup_{j=1}^{N_n} \Gamma_n^j = \bigcup_{j=1}^{N_n} \left(\overline{R_j R_{j+1}} \right) \end{cases}, \tag{3.42}$$

де $N_л$ – кількість вершин лінії лівої межі; N_n - кількість вершин лінії правої межі; $\Gamma_л^i$ – i -й елемент лінії лівої межі; Γ_n^j - j -й елемент лінії правої межі; L_i - i -та вершина лінії $\Gamma_л$; R_j - j -та вершина лінії Γ_n.

Правило замикання для границі лінії обмежень в співвідношенні (3.42) визначено як “з’єднання точок початку та точок кінців ламаної лінії обмежень

простору Q або його перерізу площиною параметрів за напрямом переміщення ОУ".

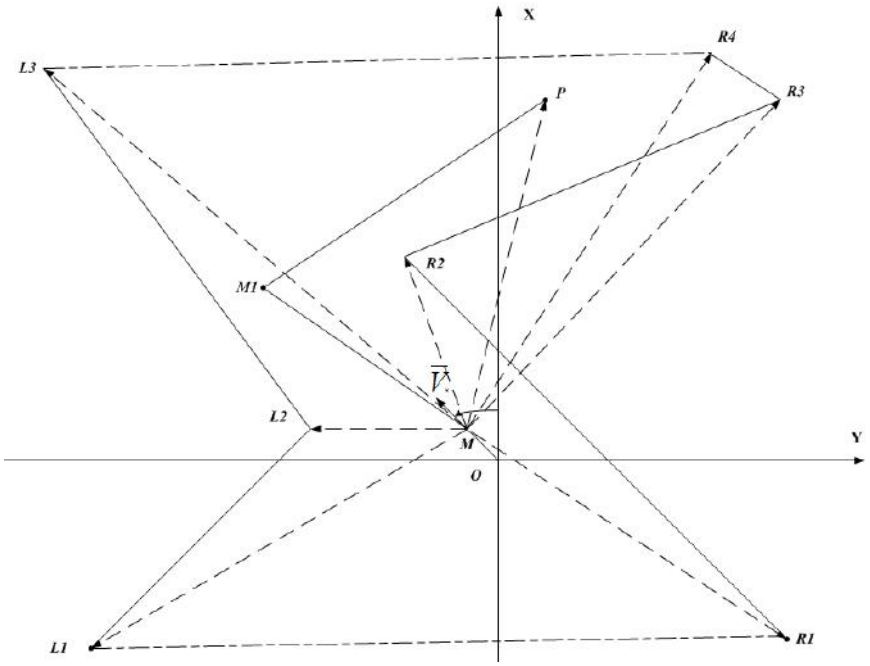


Рис. 3.4. Геометрична інтерпретація способу формування ознаки приналежності траєкторії ПР

Співвідношення (2.4) і (3.42) дозволяють визначити згідно співвідношенню (3.41) ознаку приналежності f_{μ} щодо приналежності траєкторії переміщення ОУ простору $G_{ріш}$ з межею $\Gamma_{гр}(Q)$.

Нехай ОУ знаходиться в точці M.

В такому разі для правої трійки векторів маємо значення дискретної функції Хевісайда для дійсного аргументу, яка визначає предикат приналежності Π_{μ} елемента лінії траєкторії переміщення ОУ або її проекції на переріз простору Q площиною параметрів

$$\begin{cases} r_{л}=1, \forall r_{л}^{i}=\overrightarrow{MP}\times\overrightarrow{ML_1}\geq 0, i=\overline{1,N_{л}} \\ r_{n}=1, \forall r_{n}^{j}=\overrightarrow{MR_1}\times\overrightarrow{MP}\geq 0, \ j=\overline{1,N_{n}} \\ r_{л}=0, \forall r_{л}^{i}=\overrightarrow{MP}\times\overrightarrow{ML_1}<0, i=\overline{1,N_{л}} \\ r_{n}=0, \forall r_{n}^{j}=\overrightarrow{MR_1}\times\overrightarrow{MP}<0, \ j=\overline{1,N_{n}} \\ \Pi_{\mu}=r_{л}\times r_{n} \end{cases} . \qquad (3.43)$$

Тоді ознака належноті синтезованої траєкторії простору $G_{ріш}$ можна визначити

$$f_{\mu} = \begin{bmatrix} 1, \Pi_{\mu} \geq 0 \\ 0, \Pi_{\mu} < 0 \end{bmatrix}. \tag{3.44}$$

Значення $\Pi_{\mu} > 0$ за співвідношенням (3.3/44) означає те, що синтезована траєкторія належить простору $G_{ріш}$, значення $\Pi_{\mu} = 0$ відповідає випадку примикання траєкторії границі простору $\Gamma_{гр}(Q)$, значення $\Pi_{\mu} < 0$ визначає те, що траєкторія переміщення ОУ не належить простору $G_{ріш}$.

3.3. Інформаційна множина небезпечних станів простору рішення

Розглянемо спосіб розрахунку інформаційної множини небезпечних станів простору рішення (ІМНСПР) для *i*-го ОС в просторі Q або його перерізі довільною площиною параметрів (рис.3.5).

ІМНСПР є властивістю взаємодії ОУ з *i*-м ОС і відображає множину геометричне місць точок в просторі $G_{ріш}$, в яких можливе зближення ОУ з *i*-м ОС на відстань не більшу d^i.

Визначимо згладжені координати *i*-го ОС в просторі Q або його перерізі площиною параметрів $A_i = \left(x_a^i, y_a^i\right)$. Побудуємо з початку координат, як із центру, коло радіусу d^i. З точки A_i проведемо дотичні $\left(A_i D_1\right)$ і $\left(A_i D_2\right)$ до кола з центром в точці O, які співпадають з гранично припустимою ЛВР.

Для знаходження координат точок D_1 та D_2 в перерізі ПС вирішимо систему рівнянь

$$\begin{cases} \left(x - x_a^i\right)x + \left(y - y_a^i\right)y = 0 \\ x^2 + y^2 = \left(d^i\right)^2 \end{cases}. \tag{3.45}$$

Подвійність отриманого рішення системи (3.45) дозволяє інтерпретувати його як знайдені координати точок $D_1 = \left(x_d^1, y_d^1\right)$ та $D_2 = \left(x_d^2, y_d^2\right)$.

Переміщення з точок D_1 та D_2 вздовж векторів $\overrightarrow{A_i D_1}$ та $\overrightarrow{A_i D_2}$ є можливим при повертанні вектора $\overrightarrow{V_н}$ навкруги власного початку, що відповідає зміні напрямку переміщення ОУ, який визначає переміщення в точки D_1 та D_2 відповідно.

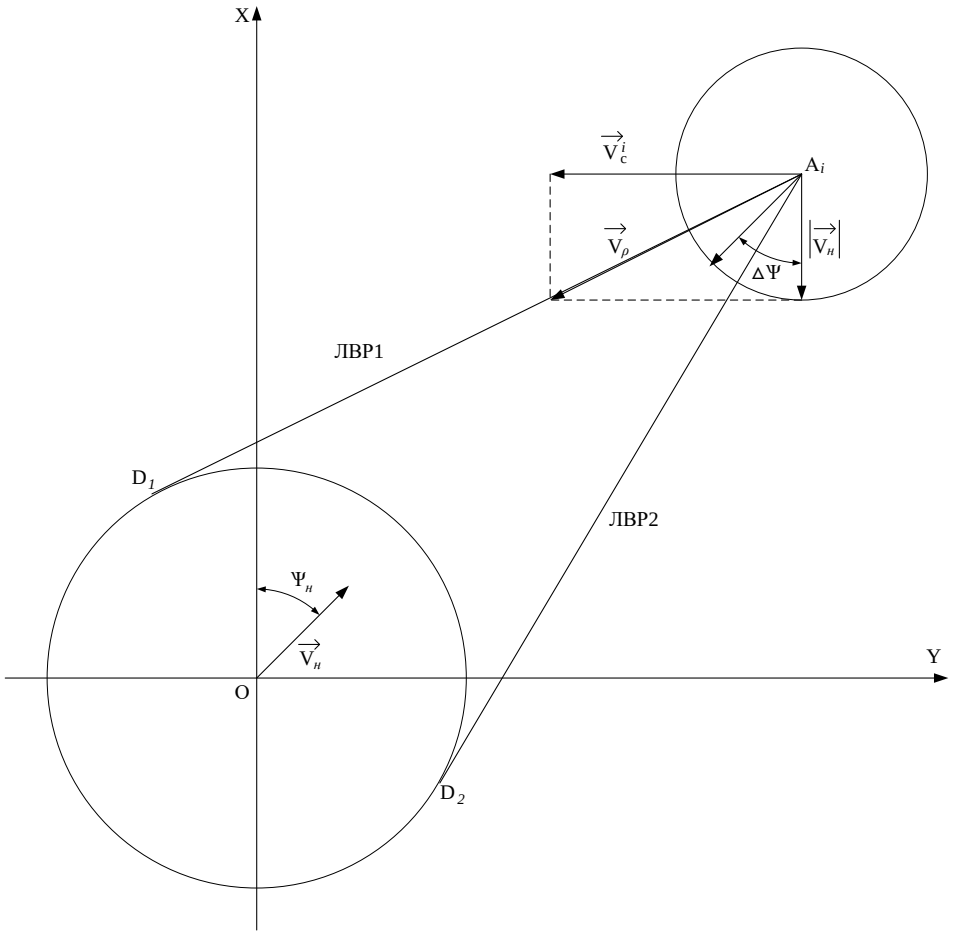


Рис.3.5. Відображення зближення з *i*-м ОС в просторі Q

Визначення необхідної зміни напрямку переміщення ОУ здійснюється наступним чином: маючи дані про ЛВР1 та ЛВР2, які співпадають з векторами $\overrightarrow{A_i D_1}$ та $\overrightarrow{A_i D_2}$, з трикутника швидкостей знаходимо необхідні напрямки переміщення ψ_1 та ψ_2, які визначають зміну напрямку руху, що відповідно забезпечує переміщення ОУ за побудованими ЛВР1 та ЛВР2.

Побудуємо ІМНСПР у вигляді дуги $\widetilde{D_1 D_2}$, яка визначає границі інформаційної множини в просторі Q (рис. 3.6).

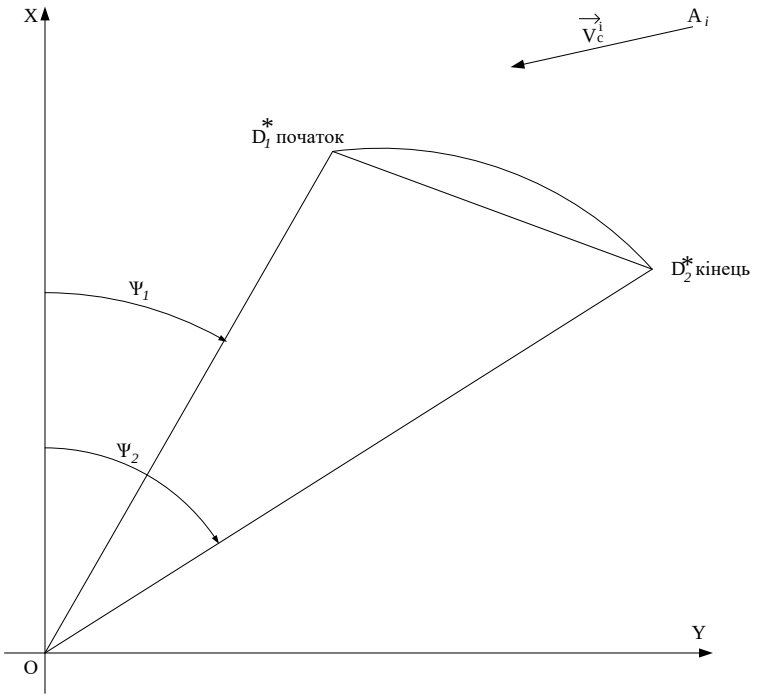


Рис. 3.6. Відображення множини небезпечного стану простору рішення

Визначивши напрямки ψ_1 та ψ_2, знаходимо координати точок D_1^* та D_2^*, що відповідає перетворенню простору при переміщенні на вектори $\overrightarrow{OD_1^*}$ та $\overrightarrow{OD_2^*}$. Кут $\widehat{D_1^*OD_2^*}$ є образом СННП в перерізі простору Q для ОУ. Слід зазначити, що форма ІМНСПР не має значення. Побудований образ ІМНСПР $\left(D_1^*D_2^*\right)$ можна "зафарбувати" в залежності від характеру зближення з *i*-м ОС. Якщо час переміщення ОУ до точки перетину по границі ІМНСПР додатній, то ситуація відповідає зближенню з передньою частиною *i*-го ОС за напрямком переміщення при лінійному прогнозі його переміщення. Якщо час переміщення до точки перетину по границі ІМНСПР від'ємний, то ситуація відповідає зближенню з кінцевою частиною *i*-го ОС за напрямком переміщення при лінійному прогнозі його переміщення. Аналіз побудованих таким чином образів ІМНСПР при зближенні з *i*-м ОС показав, що можливі п'ять варіантів його "зафарбування", а саме: "початок-кінець", "кінець-початок", "початок-початок", кінець-кінець", "байдуже". Слід зауважити, що при зближенні ОУ з *i*-м ОС ІМНСПР може бути відсутня або її кількість може

змінюватись від однієї до двох. Відсутність ІМНСПР відповідає ситуації відсутності конфлікту при взаємодії ОУ з *i*-м ОС.

Виходячи з геометричного змісту ІМНСПР, при зближенні ОУ з *i*-м ОС в просторі $G_{piш}$ маємо

$$\begin{cases} W_i = \bigcup_{k=1}^{K_i} W_i^k \\ W_i^k = W_i^k\left(\overrightarrow{w_i^k}, r_i^k, \varphi_{1i}^k, \varphi_{2i}^k, \Pi_i^k\right), \\ K_i = \overline{0,2} \end{cases} \tag{3.46}$$

де W_i - множина ІМНСПР для *i*-го ОС;

W_i^k - множина k-го ІМНСПР для *i*-го ОС;

$\overrightarrow{w_i^k}$ - вектор в просторі Q, який з'єднує кінець і початок для k-го ІМНСПР для *i*-го ОС;

r_i^k - відстань до середини $\overrightarrow{w_i^k}$;

φ_{1i}^k - напрямок (кут) початкової точки k-го ІМНСПР для *i*-го ОС при відліку від осі ординат проти годинникової стрілки;

φ_{2i}^k - напрямок (кут) кінцевої точки k-го ІМНСПР для *i*-го ОС при відліку від осі ординат проти годинникової стрілки;

Π_i^k - тип "окрасу" k-го ІМНСПР для *i*-го ОС ("початок-кінець", "кінець-початок", "початок-початок", "кінець-кінець", "байдуже");

K_i - кількість ІМНСПР для *i*-го ОС.

Враховуючи співвідношення (3.46), визначимо простір $G_{piш}$, в якому враховані геометричні місця точок можливого конфлікту взаємодії ОУ з усіма ОС в просторі Q

$$G_{piш}^* = G_{piш} \setminus \bigcup_{i=1}^{N} W_i , \tag{3.47}$$

де N - кількість ОС в просторі Q.

Тоді предикат приналежності ІМНСПР для *i*-го ОС простору $G_{ріш}$ можна визначити як

$$\begin{cases} f_{w_i} = 1 \forall w_i \bigcap G_{ріш} = \varnothing \\ f_{w_i} = 0 \forall w_i \bigcap G_{ріш} \neq \varnothing \end{cases}. \quad (3.48)$$

Співвідношення (3.47) і (3.48) визначають поточний стан простору Q, а також враховують систему обмежень при взаємодії ОУ з варіативною множиною ОС.

3.4. Простір гарантованого управління динамічним об'єктом

В процесі синтезу рішень конфлікту взаємодії динамічних об'єктів у просторі Q необхідно визначити послідовність дій по управлінню ОУ, які мають забезпечити максимум або мінімум заданої сукупності критеріїв, що забезпечують якість функціонування об'єкту [58, 101, 102, 104, 165].

У загальному випадку простір Q має довільну систему обмежень при варіативній множині ОС. В такому разі простір Q перетворюється у простір $G_{ріш}$, в якому необхідно синтезувати ланцюжок дій, що за умов цілеспрямованості забезпечить можливість гарантованого управління ОУ [137, 138, 142, 143, 158]. При цьому СІУ ОУ може функціонувати в автоматичному і/або автоматизованому режимах управління, що значно ускладнює математичну модель , яка описує переміщення ОУ в просторі Q за умов гарантованого управління.

При управлінні ДО виникає питання щодо можливості переведення ОУ з одного стану до іншого шляхом виконання послідовності дій. В такому разі необхідно визначити множини досяжності управління за умов забезпечення повної керованості ОУ. Проблема дослідження множин керованості та досяжності управління ОУ за умов цілеспрямованості та забезпечення якості процесу управління є однією з головних в теорії управління ДС.

Розглянемо метод розрахунку та побудови множин гарантованої керованості ДО за умови обмежень та забезпечення якості процесу управління ОУ. Для цього проаналізуємо властивості СІУ, враховуючи існуючі внутрішні зв'язки фазових змінних, які не є істотними при розгляді динаміки зміни окремих параметрів просторового руху ДО [71, 165, 166].

В загальному випадку динаміку просторового руху ДО та керуючі впливи на нього можна описати системою звичайних диференційних та алгебраїчних рівнянь, які визначаються вимогами міцності динамічного об'єкта та складністю управління просторовим рухом ОУ

$$\begin{cases} \dfrac{dx_1}{dt} = f_1(x_1,...,x_n,t,u_1,...,u_{n-r}) \\ \qquad\qquad ... \\ \dfrac{dx_r}{dt} = f_r(x_1,...,x_n,...,u_1,...,u_{n-r}) \\ x_{r+1} = u_{r+1}(x_1,...,x_n,t) \\ \qquad\qquad ... \\ x_n = u_n(x_1,...,x_n,t) \end{cases}. \tag{3.49}$$

Визначимо множину кінцевомірних векторів стану $X(t)$ та управляючих впливів $U(t)$ на ОУ:

$$\begin{cases} X(t) = \{x_1,...,x_n,t\} \\ U(t) = \{u_1,...u_{n-r},t\} \end{cases}, \tag{3.50}$$

де $n-$ кількість змінніх, які описують поточний стан ОУ, $r-$ кількість впливів управління на ОУ.

Структура СІУ ДО, синтез та вибір рішення щодо управління при вирішенні задачі конфлікту для ОУ залежить від значень коефіцієнтів рівнянь системи (3.49) з урахуванням співвідношення (3.50). Рішення системи (3.49) не завжди може бути знайдено в аналітичній формі, а коефіцієнти рівнянь в загальному випадку є суттєво нелінійними і враховують особливості динаміки переміщення ОУ в фізичному середовищі або віртуальному середовищі.

Зазвичай задача про управління ДО має не одне рішення навіть за умов відповідності певним вимогам якості процесу управління та оптимізації

значення інтенсивності управління $\aleph[U]$, яка визначає витрати ресурсів на здійснення процесу управління [71] і задовольняє основним принципам побудови ТС і ТЕС [101, 102].

На вектор рішення системи (3.49) впливають також невизначені величини, відносно яких була складена система рівнянь, як то збурення середовища переміщення ОУ [161]. В такому разі маємо розкид значень отриманих рішень системи (3.49) внаслідок зміни не тільки початкових значень параметрів, а й самої математичної моделі опису просторового рішення задачі управління ОУ.

Шляхом певних математичних перетворень системи (3.49) маємо отримати нову систему рівнянь, яка є інваріантною до вектору зовнішніх збурень, лінеаризувати отриману систему рівнянь та за умов певних припущень звести її до системи алгебраїчних рівнянь. Вектор рішень такої системи алгебраїчних рівнянь дозволить приблизно визначати поведінку ДО під управлінням СІУ в тих чи інших умовах її функціонування.

Зазначений підхід до рішення задачі управління ОУ на поточний час не використовується розробниками СІУ ДО. Поширеними є методи та критерії, за якими здійснюється аналітичне конструювання регуляторів систем автоматичного та автоматизованого (ергатичного) управління ДО.

Синтезуємо математичну модель управління ДО, яка є інваріантною до зовнішніх збурень середовища переміщення ОУ.

Згідно із співвідношенням (3.49), існує похідна система диференційних рівнянь відносно множини гарантованого управління ДО Φ_u:

$$\dot{X} = \Phi_u\left(X, t, F(t)\right), \tag{3.51}$$

де $\dot{X} = \begin{vmatrix} \dot{x}_1 \\ \dots \\ \dot{x}_n \end{vmatrix}$, $X = (x_1, \dots, x_n)$, $F(t) = \left(f_1(t), \dots, f_n(t)\right)$, $\Phi_u = \left(\Phi_u^1, \dots, \Phi_u^n\right)$.

Виходячи з методу нелінійної інтегральної інваріантності [101], похідну систему рівнянь (3.49) з урахуванням співвідношення (3.51) перетворимо таким чином, щоб теоретично повністю виконувались умови інваріантності.

Таким чином, отримаємо систему рівнянь:

$$\begin{cases} \dfrac{dx_1}{dt} = \Phi_u^1\left(x_1, t, f_1(t), ..., f_n(t)\right) \\ \qquad\qquad ... \qquad\qquad . \\ \dfrac{dx_n}{dt} = \Phi_u^n\left(x_1, ..., x_n, t, f_1(t), ..., f_n(t)\right) \end{cases} \tag{3.52}$$

У зв'язку з тим, що для системи рівнянь (3.52) складно досягти вимоги щодо точності виконання умов інваріантності, отримуємо систему рівнянь:

$$\begin{cases} \dfrac{dx_1}{dt} = \overline{\Phi}_u^1\left(x_1, t, f_1(t), ..., f_n(t)\right) \\ \qquad\qquad ... \qquad\qquad , \\ \dfrac{dx_n}{dt} = \overline{\Phi}_u^n\left(x_1, ..., x_n, t, f_1(t), ..., f_n(t)\right) \end{cases} \tag{3.53}$$

де $\overline{\Phi}_u = \left(\overline{\Phi}_u^1, ..., \overline{\Phi}_u^n\right)$ – множина гарантованого управління ДО $\overline{\Phi}_u$, яка враховує ε-окіл припустимих значень рішення системи (3.51).

Слід зазначити, що в системі рівнянь (3.53) вектор X не є інваріантним до вектора зовнішніх збурень $F(t)$. Разом з тим при цьому існують припустимі відхилення на величину ε поведінки отриманої системи (3.53) від похідної системи (3.49).

Якщо компоненти вектора X є обмеженими функціями, для оцінювання рішення системи (3.45) в ε-околі припустимих значень рішення системи (3.49), згідно із співвідношенням (3.50), можна визначити вектор управляючих впливів $U(t)$, кожен елемент якого задовольняє умові

$$u_i(t) \geq \left| x_i^a - x_i \right| \forall u_i(t) \in U(t), \tag{3.54}$$

де x_i^a – рішення системи (3), x_i – справжнє значення рішення системи (3.51).

Співвідношення (3.54) має задовольнятись на всьому діапазоні значень часу управління ДО

$$t \in \left[\overline{t_n, t_к}\right], \tag{3.55}$$

де t_n та $t_к$ – відповідно час початку та закінчення управляючих впливів на ДО.

У такому разі на множині (3.55) маємо множину управляючих впливів (3.54), які є рішеннями системи (3.53), що мають знаходитись у множині простору обмеження параметрів систем (3.49) та (3.51).

Границі простору обмежень на значення істотних параметрів управління ДО визначаються рішеннями сукупностей систем диференційних рівнянь для кожної змінної $x_i(t) \in X_i$:

$$\left[\begin{array}{l} \left\{\begin{array}{l} \dfrac{dx_1^a}{dt} = \Phi_u^1\left(x_1^a, t, f_1(t),..., f_n(t)\right) \\ \dfrac{dx_1}{dt} = \overline{\Phi}_u^1\left(x_1^a, x_2,..., x_n, t, f_1(t),..., f_n(t)\right) \end{array}\right. \\ \qquad ... \qquad . \\ \left\{\begin{array}{l} \dfrac{dx_n^a}{dt} = \Phi_u^n\left(x_n^a, t, f_1(t),..., f_n(t)\right) \\ \dfrac{dx_n}{dt} = \overline{\Phi}_u^n\left(x_1,..., x_n^a, t, f_1(t),..., f_n(t)\right) \end{array}\right. \end{array}\right. \tag{3.56}$$

Виходячи з результатів рішення сукупностей систем диференційних рівнянь (3.56), згідно із співвідношенням (3.54), визначимо сукупності управляючих впливів

$$\left[\begin{array}{l} u_i^{\max} = \sup\left(u_i(t)\right) \\ u_i^{\min} = \inf\left(u_i(t)\right) \end{array}\right. . \tag{3.57}$$

У такому разі X_i визначає область фазового потоку параметра $x_i(t)$, а співвідношення (3.57) визначає границі фазового потоку управління $u_i(t)$ ОУ. Слід зазначити, що фазовий простір параметрів для системи (3.49) є безперервним, а точки перетину з осями координат фазового простору або його перерізів за обраними парами параметрів не є точками біфуркації параметрів і можуть розглядатись як сідлові точки.

Для кожної змінної $x_i(t)$ визначимо першу похідну функції управління ДО:

$$\frac{du_i(t)}{dt} = L_i\left(u_i(t),t\right) + \varphi_i(t). \qquad (3.58)$$

Вид функції $L_i\left(u_i(t),t\right)$ обирається з умов інваріантності управління ОУ. Функція $\varphi_i(t)$ визначається за параметрами похідної системи і враховує особливості динаміки ОУ.

Слід зазначити, що в загальному випадку задача Коші для співвідношення (3.58) може бути задана з ненульовими початковими умовами.

У загальному випадку функції $L_i\left(u_i(t),t\right)$ та $\varphi_i(t)$ визначаються відповідно до системи (3.51) згідно з умовою

$$L_i\left(u_i(t),t\right) = 0, \{\forall i \in [1,n] \,|\, u_i(t) \geq 0\}. \qquad (3.59)$$

Враховуючи співвідношення (3.58) і (3.59), функції $L_i\left(u_i(t),t\right)$ та $\varphi_i(t)$ можна визначити таким чином:

$$\begin{cases} K_i^L = sign\left(x_i^a - x_i\right) \\ L_i\left(u_i(t),t\right) \geq K_i^L[\Phi_u^i(x_i^a, t, f_1(t),..., f_{i-1}(t), f_{i+1}(t),..., f_n(t)) - \\ \Phi_u^i(x_i, t, f_1(t),..., f_{i-1}(t), f_{i+1}(t),..., f_n(t))] \end{cases}, \qquad (3.60)$$

$$\varphi_i(t) \geq \left|\Phi_i^a(x_i^a, t, f_1(t),..., f_{i-1}(t), f_{i+1}(t),..., f_n(t)) - \overline{\Phi}_i^n(x_1,..., x_{i-1}, x_i^a, x_{i+1},..., x_n, t, f_1(t),..., f_n(t))\right|. \qquad (3.61)$$

Співвідношення (3.60) і (3.61) характеризують властивості отриманої інваріантності та ступінь наближення рішень системи (3.49) до теоретично можливих рішень у класі функціональних просторів, що визначають множину значень параметрів, при яких можливе рішення системи (3.49).

Метод рішення системи (3.53) є практично можливим за умови її дифеоморфізму системі (3.61), що дозволяє розглядати інваріантність області або поля рішення похідної системи (3.49) з рішеннями інваріантної системи (3.53). В такому разі рішення системи (3.49) з урахуванням співвідношень (3.60) і (3.61) можна замінити рішеннями інваріантної системи (3.53). В

такому разі області обмежень параметрів залишаться такими, як і для системи (3.49).

Таким чином виникають дві задачі:

– розробити методи розрахунку області значень параметрів системи (3.49) з метою визначення області керованих станів ДО в евклідовому просторі Q;

– при даному дифеоморфізмі системи (3.49) знайти область її рішення у вигляді лінійних алгебраїчних рівнянь з метою дослідження таких динамічних властивостей ОУ, як керованість, стійкість тощо.

Перша задача досить просто вирішується в аналітичній формі для систем, які вміщують на більше двох рівнянь. Для систем, які вміщують більше двох рівнянь, застосовують ітераційні обчислювальні процедури, які дозволяють визначити області активних керованих станів ДО. При цьому необхідно визначити збіжність процесу обчислень, час обчислень, можливості та засоби визначення точності, умову завершення ітераційного процесу, що є притаманним “ефекту доміно” та “прокляттю розмірності”. Слід відмітити, що зазначені проблеми пов’язані між собою і вимагають комплексного підходу при їх вирішенні.

За своєю сутністю рівняння, які входять до систем (3.49), (3.51)–(3.53) , є Гамільтоновими функціями, а куруючі впливи $U(t)$ змінюють стан фазового простору існування ДО. У такому разі співвідношення (3.60) і (3.61) визначають принцип найменшої (стаціонарної) дії рівняння Гамільтона. Співвідношення (3.48), (3.58) і (3.59) описують гамільтоніан, який при застосуванні дужок Пуассона дозволяє використовувати дужки Лі в алгебрі Пуассона, що вирішує другу задачу, а саме - можливості використання алгебраїчних рівнянь замість диференційних, що значно спрощує знаходження рішення системи (3.49). Слід зазначити, що рівняння Гамільтона у класичній механіці є аналогом рівняння Гейзенберга у квантовій механіці і дозволяє враховувати ймовірну невизначеність обмежень і властивостей простору Q, в якому здійснюється переміщення ДО.

Виходячи з початкових умов для рішення системи (3.49), можна визначити функції для обчислення границь перерізів фазового простору ОУ площинами в ортогональній евклідовій n-мірній системі координат [159, 160]:

$$z_i^j = Ax_i + Bx_j, i \in [1,..., n], j \in [1,..., n], i \neq j. \quad (3.62)$$

Границю фазового простору в перерізі площиною (14) визначимо параметрично заданою функцією

$$\begin{cases} x_i = u_i(x_1,..., x_n, t) \\ x_j = u_j(x_1,..., x_n, t) \end{cases}, \quad (3.63)$$

яка визначає граничний цикл або атрактор перерізу фазового простору:

$$x_i = x_i(x_j). \quad (3.64)$$

Атрактор за співвідношеннями (3.63) та (3.64) є замкненою кривою на площині, до якої асимптотично притягаються усі навколишні траєкторії, що виходять з різних початкових точок всередині і зовні граничного циклу.

Слід зазначити, що позитивним аспектом є те, що більшість диференційних рівнянь системи (3.49), які виходять з різних початкових умов, показують схожу поведінку рішень. Зокрема, при $t \to \infty$ цілі сімейства рішень асимптотично наближаються до атракторів, які є стаціонарними точками фазового простору або до певних замкнених кривих - граничних циклів.

Для визначення параметрів системи (3.45), які мають взаємний істотний вплив на рішення, скористаємося канонічними рівняннями Гамільтона:

$$\begin{cases} \dot{p}_j = \dfrac{dp_j}{dt} = -\dfrac{\partial H}{\partial q_{jq}} \\ \dot{q}_j = \dfrac{dq_j}{dt} = \dfrac{\partial H}{\partial p_j} \end{cases}. \quad (3.65)$$

У системі (3.65) p_j є значенням узагальнених координат параметрів, які визначають точки фазового простору, а q_j є значенням узагальнених координат керуючих впливів на ОУ.

Тобто маємо функцію Гамільтона:

$$H(p,q,t)=\sum_i \dot{q}_i p_i - L\left(q,\dot{q},t\right). \tag{3.66}$$

У співвідношенні (3.66) лагранжиан $L\left(q,\dot{q},t\right)$ є функцією узагальнених координат та відповідних швидкостей, що відповідає базису моделі, які визначені співвідношеннями (2.3 – (2.5).

За рівняннями Ейлера-Лагранжа, згідно з теоремою Бобильова [165], маємо

$$\frac{d}{dt}\frac{\partial L}{d\dot{q}}-\frac{dL}{dq}=0. \tag{3.67}$$

Для співвідношення (3.67) визначимо узагальнені керуючі впливи як

$$p=\frac{\partial L}{d\dot{q}}. \tag{3.68}$$

З урахуванням співвідношення (3.68) визначимо узагальнені сили, які впливають на переміщення ДО

$$F=\frac{\partial L}{dq}. \tag{3.69}$$

Таким чином, отримуємо рівняння Ейлера-Лагранжа:

$$\frac{dp}{dt}=F. \tag{3.70}$$

У такому разі співвідношення (3.70) визначає для співвідношення (3.67) узагальнену форму другого закону Ньютона з точністю до повної похідної у часі від довільної функції координат ДО в просторі Q. Додавання такої функції у співвідношення (3.67) не впливає на рівняння переміщення динамічного ОУ.

Слід зазначити, що, згідно з гамільтоновою механікою, переміщення ДО пов’язане з забезпеченням принципу найменшої дії:

$$S[p,q]=\int\sum_i\left(p_i\dot{q}_i-\dot{H}(p,q,t)\right)dt. \qquad (3.71)$$

Для першої варіації дії згідно співвідношення (3.71) за умов стаціонарності можна визначити умову найменшої дії

$$S[p,q]=0. \qquad (3.72)$$

Згідно із співвідношенням (3.72), визначаємо функцію Гамільтона

$$H(p,q,t)=\int\sum_i\left(p_i,\dot{q}_i\right)dt. \qquad (3.73)$$

Таким чином, співвідношення (3.73) дозволяє кількісно визначити запас енергії при початку переміщення ДО за обраною траєкторією. При цьому слід зауважити, що термінальне управління для точки простору стану ДО визначається за співвідношенням (3.57).

Отже, можна визначити область керованості власне нестійкого ДО за умови опису його просторового переміщення у евклідовому просторі Q. В такому разі управління переміщенням ДО розглядається в ортогональних перерізах простору гарантованого управління динамічним об’єктом у зв’язаній системі координат в просторі Q.

Для похідних параметрів x_i та x_j, згідно із співвідношенням (3.57), можна визначити лінії обмежень $u_i^{\max}$, $u_i^{\min}$, $u_j^{\max}$ та $u_j^{\min}$, які визначають область c_i^j керованості ДО у площинах змінних в просторі Q. Здійснивши зворотне перетворення області c_i^j з площини змінних $\frac{dx_i}{dt}$ і $\frac{dx_j}{dt}$ у площину перерізу простору фізичних параметрів x_i та x_j, отримаємо переріз простору параметрів, які відображають переріз простору гарантованого управління ДО.

Слід зазначити, що ОУ перебуває під впливом не лише СІУ, а й зовнішнього середовища, на яке певною мірою впливає об’єкт**.** Через наявність взаємозв’язків, що пов’язані з взаємодією ОУ з варіативною

множиною ОС в умовах конфлікту, обмежень і невизначеностей, переміщення об’єкту в просторі рішень $G^{*}_{ріш}$ відбувається під впливом як керуючих впливів системи управління, так і зовнішнього середовища. Ці впливи можуть відхиляти рух ОУ від обраної траєкторії. Природно, що чим докладніше вивчено реакцію ОУ (ТС, ТЕС) на зовнішні впливи і чим повнішою є інформація про сутність зовнішніх впливів, тим ефективнішими можуть бути вибрані керуючі впливи. Відсутність інформації про суттєві параметри, які дозволяють визначити вплив зовнішнього середовища на функціонування СІУ ОУ, знижують показники якості та стійкості управління функціонуванням ТС або ТЕС.

Виходячи з принципу необхідної різноманітності Ешбі [84, 114], СІУ ОУ можна розглянути в якості системи передавання інформації, яка має обмеження на траєкторію переміщення в просторі $G^{*}_{ріш}$ та гарантоване управління, на яке має вплив зовнішнє середовище.

В якості зовнішнього середовища для суттєво нелінійного ОУ можна розглянути атмосферне повітря, збурення якого визначає велика кількість факторів, які можна врахувати в статистичній моделі.

В такому разі модель збурень може бути врахована в передатковій функції ІП СІУ при рішенні конфлікту взаємодії ОУ з варіативною множиною ОС в умовах обмежень і невизначеностей.

В такому разі сутність управління полягає в зменшенні різноманітності стратегій рішення конфлікту та вибору оптимального за критерієм (функцією ціни) гарантованого управління ОУ за умов впливу середовища. Тоді ІП СІУ зменшуватиме невизначеність вихідних станів ОУ, мінімізуючи умовну ентропію керуючих впливів.

Висновки до третього розділу

За результатами проведених досліджень:

1. Отримані результати відрізняються від відомих здатністю формально визначити:

-вирішуюче правило визначення інформаційної множини простору комбінованого управління об’єктом;

-інформаційні множини простору рішення, секторів небезпечних напрямків переміщення ОУ, простору комбінованого управління ОУ, правила та відношеннями, що дозволяють визначити формальний образ множинного уявлення ПР;

-простір параметрів гарантованого управління динамічним об’єктом;

-обмеження та правило замикання простору рішень;

-правило та формальну ознаку приналежності елемента траєкторії переміщення ОУ простору рішень;

-інформаційні множини небезпечних станів простору рішення при вирішенні задачі ухиляння та переслідування з варіативною множиною ОС;

-правило ранжування ОС за критерієм небезпечності їх переміщення;

-представляти простір рішень з врахуванням невизначеності та взаємного переміщення варіативної множини ОС.

2. Формально визначені вирішуючі правила та інформаційні множини, які є елементами теоретико-множинної моделі взаємодії ОУ з варіативною множиною ОС в умовах обмежень, невизначеності і конфлікту дозволяють:

-формувати множину простору комбінованого управління ОУ (ПКУО);

-обирати спосіб формального опису простору рішень та його доповнення;

-обирати спосіб розрахунку секторів небезпечних напрямків переміщення (СННП) ОУ.

3. Формально визначені характеристики простору рішення в умовах конфлікту взаємодії ОУ з варіативною множиною ОС, обмежень та невизначеності дозволяють:

-враховувати невизначеності переміщення ОС в просторі рішень;

-визначати множини гарантованого управління ОУ при переміщенні за синтезованою траєкторією рішення конфлікту при переслідуванні та збіганні;

-визначати напрямки переміщення інформаційних множин небезпечних станів в просторі рішення конфлікту;

-визначати інтегральне формальне множинне уявлення простору рішень конфлікту відповідно до запропонованої теоретико-множинної моделі опису конфлікту.

Основні результати розділу опубліковані у роботах [133, 134, 138, 139, 155, 156, 157, 161, 163, 164, 165, 166, 167, 168, 172, 175].

РОЗДІЛ 4. ТЕХНОЛОГІЯ СИНТЕЗУ І ВИБОРУ СТРАТЕГІЙ ОПТИМАЛЬНОГО УПРАВЛІННЯ СКЛАДНИМИ ТЕХНІЧНИМИ ТА ТЕХНІЧНИМИ ЕРГАТИЧНИМИ СИСТЕМАМИ ПРИ РОЗВ'ЯЗАННІ КОНФЛІКТУ ЇХ ВЗАЄМОДІЇ З ВАРІАТИВНОЮ МНОЖИНОЮ ОБ'ЄКТІВ СПОСТЕРЕЖЕННЯ В УМОВАХ ОБМЕЖЕНЬ ТА НЕВИЗНАЧЕНОСТЕЙ

4.1. Метод інтегрального усікання варіантів

Методи вибору оптимальних рішень дозволяють у результаті математичного моделювання ситуації конфлікту отримати множину допустимих з точки зору реалізації альтернатив рішення задачі, що дозволяє вибрати стратегію управління ОУ [7, 8, 42, 58? 98? 102, 114, 120, 203, 204]. В процесі синтезу та вибору стратегії управління ОУ при рішенні задачі конфлікту метод рішення має зменшувати множину припустимих альтернатив рішень, а також враховувати релевантні фактори, які не були враховані в математичній моделі конфлікту взаємодії об'єктів в просторі $G_{ріш}$.

Рішення задачі синтезу та вибору стратегій вирішення конфліктів взаємодії об'єктів в умовах обмежень та невизначеностей вміщує етапи: побудова моделі конфлікту, розробка та обґрунтування методу вирішення задачі конфлікту, синтез алгоритмів синтезу рішень, дослідження властивостей алгоритмів та рішень.

Синтез стратегій при вирішенні задачі конфлікту здійснюється в просторі Q з врахуванням системи обмежень згідно співвідношенню (2.4), ПКУО відповідно до співвідношення (3.1), СННП відповідно до співвідношення (3.40), згідно моделі M, яка визначена співвідношенням (2.10), для формального опису конфлікту за співвідношенням (2.35) з врахуванням принципу оптимальності та правила зупинки (2.36), представлення простору $G_{ріш}$ згідно співвідношенню (2.31) та можливостей, що надані мовою $L(\Gamma_{пр})$,

яка визначена формальною граматикою $\Gamma_{пр}$ та правилами P згідно співвідношенням (2.52) і (2.53).

При розгляді можливих управлінь за перерізами в евклідовому ортогональному просторі на ОУ накладаються додаткові обмеження щодо можливості реалізації управління, а саме - обмеження на швидкість переміщення ОУ V_e^0, напрямки переміщення $\psi_{прип}$, час запізнювання τ на виконання розрахунків і аналіз результатів щодо альтернативних рішень конфлікту, координати цільової точки переміщення ОУ, припустимі дистанції зближення d^i з i-м ОС, обмеження $\Gamma_{гр}$ простору Q.

Визначимо множину припустимих значень модуля швидкостей переміщення ОУ в перерізі простору Q площиною параметрів

$$V_e^0 = [0, V_{\max}], \tag{4.1}$$

де $V_{\max}$ - найбільше значення модуля вектору припустимої швидкості переміщення ОУ, наприклад, найбільша вертикальна швидкість набору висоти чи зниження для повітряного судна, кібернетичної системи чи її елемента, тощо.

Визначимо множину припустимих напрямків переміщення ОУ в перерізі простору Q площиною параметрів

$$\psi_{прип} = [\psi^n_{прип}, \psi^к_{прип}], \forall \psi^n_{прип} \in [0, 2\pi], \forall \psi^к_{прип} \in [0, 2\pi], \tag{4.2}$$

де $\psi^n_{прип}$- початкове значення припустимих напрямків переміщення ОУ, $\psi^к_{прип}$- кінцеве значення припустимих напрямків переміщення ОУ.

Початкове та кінцеве значення множини припустимих напрямків переміщення ОУ в співвідношенні (4.2) визначається за напрямком повороту за годинниковою стрілкою.

В загальному вигляді траєкторії, що визначають стратегії рішення, які синтезуються, визначають геометричне місце точок проекції центру мас при переміщенні ОУ в площині перерізу простору Q. Тобто отримуємо

траєкторію переміщення ОУ в просторі Q та її проекції на площини перерізів при синтезі та реалізації стратегії управління ОУ.

Синтезовані траєкторії переміщення ОУ в просторі Q мають належати простору $G_{ріш}$. Найбільш незручним для аналізу приналежності траєкторії простору Q є наявність обмежень $\Gamma_{гр}(Q)$, які задаються у вигляді послідовності точок, що визначають вершини ламаних ліній межі простору Q або його перерізу. Враховуючи те, що досліджувана траєкторія і обмеження $\Gamma_{гр}(Q)$ є кусково-лінійними функціями, для аналізу приналежності синтезованих рішень (траєкторій) простору Q доцільним є використання евристичних методів аналізу, які базуються на поняттях і співвідношеннях, що отримані з геометричних побудов та понять векторної алгебри.

При синтезі стратегій управління ОУ щодо рішення конфлікту взаємодії ОУ з ОС при переміщенні в кінцеву термінальну позицію (цільову точку переміщення) простору Q задача вирішується шляхом синтезу траєкторій переміщення з врахуванням параметрів довгострокового і оперативного гарантованого управління. Тобто при синтезі стратегій управління ОУ фактично формується множина альтернативних траєкторій переміщення ОУ та визначається множина швидкостей і напрямків переміщення ОУ у відповідності до опису стану простору $G_{ріш}$, ПКУО та гарантованого управління.

Траєкторії переміщення ОУ в просторі Q мають вигляд безперервних ламаних ліній, які складаються з декількох елементів. Тобто управління здійснюється за напрямком переміщення та швидкістю на кожному елементі синтезованої траєкторії за умови наявності гарантованого управління. Вибір траєкторії переміщення та стратегії управління ОУ при переміщенні в просторі Q здійснюється за критерієм, який визначено співвідношенням (2.33).

Синтез стратегій оперативного управління ОУ здійснюється безперервно і може застосовуватись для корекції обраної стратегії (траєкторії) переміщення ОУ в просторі Q. В такому випадку синтезована стратегія може обиратись з врахуванням динамічних та кінематичних характеристик ОУ. При синтезі траєкторії переміщення та виборі стратегії управління динамічні і конструктивні властивості ОУ можна певним чином врахувати в адитивному критерії Φ за співвідношенням (2.33). Таким чином визначається для кожного моменту часу траєкторія переміщення

Слід зазначити, що траєкторія переміщення і стратегія гарантованого управління при рішенні конфлікту визначається кінематичними і динамічними характеристиками ОУ і ОС та стратегією і тактикою взаємодії об’єктів в просторі Q.

Синтез та вибір стратегії рішення конфлікту взаємодії ОУ з ОС є складним багатозв'язним процесом, що не обов'язково передбачає використання єдиної альтернативи рішення, яке задовольняє принципу оптимальності. При синтезі та виборі стратегії повинна розглядатись множина альтернатив рішення задачі конфлікту. Тому в процесі рішення можливі два випадки: існування альтернативних рішень, які задовольняють системі обмежень СІУ або стратегії поведінки ОПР з точки зору практичної реалізації; альтернативні рішення відсутні чи їх неможливо реалізувати з технічних, технологічних, нормативних, правових чи інших причин.

В основі методу синтезу стратегій рішення конфлікту лежить теоретико-множинна модель M, яка визначена співвідношенням (2.10). В моделі M виділимо елементи множини відносин, які формують правила P і визначені співвідношеннями (2.37), (2.39), (2.48). У відповідності з прийнятою термінологією [88] та враховуючи співвідношення (2.29) визначимо інформаційні множини:

$$f_x = \bigcup_{i=0}^{N} f_x^i \qquad (4.3)$$

та

$$f_c = \bigcup_{i=0}^{N} f_c^i , \tag{4.4}$$

де N - кількість ОС в просторі Q.

Співвідношення (4.3) описує інформаційну множину, яка визначає взаємне розташування ОС відносно ОУ в просторі Q.

Співвідношення (4.4) описує інформаційну множину , яка містить дані про вектори істинної швидкості ОС та ОУ. Якщо множина f_x порожня $(f_x = \varnothing)$, то така ситуація є простішою і рішення задачі находження траєкторії переміщення ОУ в просторі Q зводиться до чисто геометричних побудов, які задовольняють вимогам функції ціни *J*, безперервності і реалізованості синтезованих альтернативних стратегій рішення конфлікту

$$\begin{cases} J = \sum_{i=1}^{3} k_i \times J_i \\ J_i = \sum_{j=1}^{2} \overline{k_j} \times \left\| \left(\xi_j\right)^2 \right\| \end{cases}, \tag{4.5}$$

де k_i - коефіцієнт , який враховує вагу *i*-го квадратичного критерію;

J_i – частковий квадратичний критерій оцінки елементів стратегії (ланцюжка U_i^0 управління ОУ), рішення, яке синтезовано, траєкторії;

$\overline{k_j}$ - коефіцієнт, який враховує осереднені значення динамічних характеристик ОУ;

$\left\| \left(\xi_j\right)^2 \right\|$ - нормоване значення зміни параметрів стратегії управління ОУ $U_{j_i}^0$ для *j*-го елементу *i*-ї синтезованої траєкторії при переході з одного елемента лінії переміщення на інший.

Додатковими інформаційними множинами, які дозволяють спростити процесс пошуку альтернативних рішень серед припустимих стратегій у відповідності до обмежень (4.1) і (4.2) на основі співвідношення (2.29) і (2.32) є СННП і ПКУО.

СННП дозволяють визначити напрями переміщення ОУ в просторі Q, при яких можлива ситуація зближення з ОС на відстані, які не менші від d^i. СННП визначають множину точок простору Q, в яких можлива ситуація небезпечного зближення ОУ з i-м ОС (область можливих небезпечних зближень). Істотним доповненням інформації про СННП є визначення «забарвлення», що формує ознаку зближення по передній чи задній частині з i-го ОС.

ЛШ є множиною припустимих швидкостей ОУ V_e^0 в просторі Q або його перерізі площиною параметрів при русі за напрямком $\psi_j \in \psi_{прип}$, за яким не виникає ситуація небезпечного зближення з множиною ОС в просторі Q.

ПКУО є інформаційною множиною, яка вміщує відомості про можливі значення швидкостей та напрямків переміщення ОУ в просторі Q або його перерізі площиною параметрів, при яких значення відстані зближення з множиною ОС може приймати задане значення.

Рішення конфлікту зводиться до знаходження множини альтернативних стратегій рішень у вигляді траєкторій переміщення ОУ в просторі Q за умови обмежень, варіативної множини ОС, невизначеності їх переміщення та неприпустимості небезпечного зближення з жодним з ОС. Суттєвим є можливість реалізації іх альтернативних рішень на множинах, які визначаються ЛШ, ПКУО і СННП.

Таким чином, при синтезі стратегій рішення конфлікту в першу чергу необхідно визначити нерозривність простору $G_{ріш}$ при переміщенні ОУ в кінцеву точку траєкторії за напрямком переміщення, який належить множині $\psi_{прип}$, що забезпечує безконфліктне переміщення ОУ в цільову точку з множини кінцевих термінальних позицій. При цьому необхідно забезпечити приналежність модуля вектору швидкості $|\overrightarrow{V_н}|$ переміщення ОУ розрахованим множинам безпечних швидкостей V_e^0 для простору Q.

Розривність простору Q при взаємодії ОУ з i-м ОС визначається за ознакою

$$\begin{cases} R_n^i = \left|\overrightarrow{V_н}\right| \in V_e^i \\ \left[\begin{array}{l} \Theta_R^i = 0, \forall R_n^i = \varnothing, \\ \Theta_R^i = 1, \forall R_n^i \neq \varnothing \end{array}\right. \end{cases} \tag{4.6}$$

де V_e^i - множина припустимих швидкостей переміщення i-го ОС в просторі Q .

$$V_e^i = \left[0, V_{\max}^i\right], \tag{4.7}$$

де $V_{\max}^i$ - найбільша припустима швидкість переміщення i-го ОС в просторі Q; R_n^i - правило приналежності, яке визначене співвідношенням (4.6).

Узагальнюючи отримані значення для ознак нерозривності простору $G_{piш}$ при взаємодії ОУ з кожним з N ОС в просторі Q, визначимо ознаку безперевності простору $G_{piш}$ при вирішенні конфлікту взаємодії ОУ з ОС

$$\left[\begin{array}{l} \Theta_R = 1, \forall \Theta_R^i = 1, i = \overline{1, N} \\ \Theta_R = 0, \exists \Theta_R^i = 0, i = \overline{1, N} \end{array}\right. . \tag{4.8}$$

За ознакою Θ_R, яка визначена співвідношеннями (4.6) – (4.8), оцінюється нерозривність простору $G_{piш}$. Якщо ознака приймає значення $\Theta_R = 1$, простір $G_{piш}$ є нерозривним для швидкості $\left|\overrightarrow{V_н}\right|$ переміщення ОУ в просторі Q.

Якщо ознака $\Theta_R = 0$, простір $G_{piш}$ є нерозривним з точки зору можливості безконфліктного переміщення ОУ в просторі Q з врахування обмежень простору $\Gamma_{гр}(Q)$.

Слід зазначити, що співвідношення (4.6) - (4.8), за якими визначається ознака безперервності Θ_R простору $G_{piш}$, співпадають з умовами вирішення задач пошуку та переслідування з використанням апарату диференційних ігор [12, 13, 28, 38, 42, 70-72, 201-204]. В такому разі при вирішенні конфлікту управління ОУ є безперервним і цілеспрямованим.

Таким чином можна сформулювати умови, за якими можуть бути реалізовані альтернативні стратегії управління ОУ при синтезі траєкторій переміщення і ланцюжків управління щодо рішення конфлікту в просторі Q:

1. Наявність траєкторій переміщення ОУ та ланцюжків управління ОУ в просторі $G_{piш}$ при зміні тільки напрямку переміщення можливо при $\Theta_R = 1$.

2. Наявність траєкторій переміщення та ланцюжків управління ОУ при рішенні конфлікту в просторі $G_{piш}$ можливе при зміні напрямку переміщення та швидкості при $\Theta_R = 0 \exists \Theta_R^i = 1, i = \overline{1, N}$.

3. Справедливість умов 1 і 2 у відповідності з співвідношеннями (4.6) і (4.8) визначається дискретністю процесу спостереження та пошуку при синтезі стратегій рішення конфлікту, а також дискретністю перегляду напрямків переміщення ОУ в просторі Q.

Оптимальність χ в співвідношенні (2.36) полягає в критерії вибору (2.33) і функції ціни J для конфлікту (2.35), рішення якого здійснюється шляхом реалізації деякої множини стратегій рішення (альтернатив, траєкторій, ланцюжків), що обираються як оптимальні. Представлення оптимальності набуває вигляд відображення χ, яке у моделі конфлікту K за співвідношенням (2.35) ставить у відповідність множину його результатів μ. В такому разі маємо відображення $\chi \mapsto \lambda$, що дозволяє визначити конфлікт співвідношенням

$$\begin{cases} K = \langle M, A, S, \Gamma_{пр}, G_{piш}, \mu \rangle \\ \chi K = \mu \end{cases}. \tag{4.9}$$

В співвідношенні (4.9) χ є принципом оптимальності для даного типу конфлікту, а множина результатів χK є реалізацією цього принципу для конфлікту K або рішенням конфлікту в сенсі принципу оптимальності χ. В загальному випадку принцип оптимальності χ враховує можливість неоднозначності і множинності рішення , яке приймається.

Найбільш слабкою логічною залежністю між конфліктом K і множиною його оптимальних рішень μ можна вважати проблему можливості реалізації

принципу оптимальності χ для кожного класу конфлікту та питання про порожність множини результатів χK. Тобто для оптимального за принципом χ рішення маємо

$$\chi K \neq \varnothing \Rightarrow \mu \neq \varnothing. \qquad (4.10)$$

При рішенні конфлікту у відповідності до співвідношення (4.10) у сенсі оптимальності χ принциповим є додержання вимог гомеостатичності функціональних поведінок ТС, ТЕС, до складу яких входить СІУ ОУ [45, 50, 103, 104]. Таким чином слід звернути увагу не тільки на збереження стереотипу поведінки ОУ в просторі Q, але і на збереження свободи вибору і здійснення дій щодо управління ОУ незалежно від поведінки учасників конфлікту, а саме - їх активності. В такому разі доцільним є використання організмічної теорії синтезу оптимальних ТС і ТЕС для рішення конфлікту [100, 102, 104].

Задамо оптимальність χ в вигляді функціоналу у відповідності з співвідношеннями (2.33) і (4.5)

$$J = J\left(\bar{k}, S, \Delta\psi, \Delta V\right), \qquad (4.11)$$

де $\bar{\bar{k}} = \left\{\overline{k_1}, \overline{k_2}, \overline{k_3}, \overline{k_4}\right\}$ - коефіцієнти часткових критеріїв відбору;

$S = \left\{S_1, S_2, S_3, S_{пр}\right\}$ - відстані , які характеризують переміщення ОУ згідно стратегії (траєкторії, що підлягає аналізу);

$\Delta\psi = \left\{\Delta\psi_{зм}^1, \Delta\psi_{зм}^2\right\}$ - кути відвороту при переході з однієї на іншу ланку ланцюжка траєкторії переміщення ОУ;

$\Delta V = \left\{\Delta V_1, \Delta V_2\right\}$ - зміна швидкості , яка необхідна для безпечного переміщення по відповідній ланці траєкторії.

Функціонал ціни J згідно співвідношенню (4.11) забезпечує:

- збереження певної стратегії управління ОУ в просторі Q при вирішенні конфлікту;

- можливість вибору стратегії зміни напрямку переміщення;

- можливість зміни швидкості переміщення;

- можливість одночасної зміни напрямку і швидкості, що є комбінованим управлінням (КУ) об’єктом.

Геометрична інтерпретація ланцюжка управління ОУ для забезпечення переміщення за траєкторією в перерізі простору Q при управлінні за напрямком та швидкістю наведена на рис. 4.1.

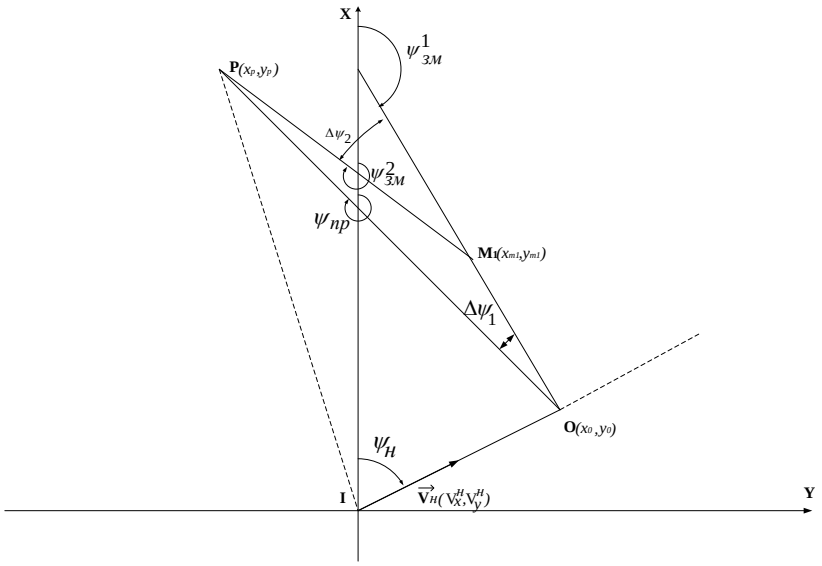


Рис. 4.1. Геометрична інтерпретація ланцюжка управління при переміщенні по траєкторії в просторі спостереження або його перерізі

Ланцюжок траєкторії переміщення ОУ в просторі Q є трьох ланковою ламаною лінією IOM_1P. Точка O є фактичним початком траєкторії переміщення ОУ.

Визначимо координати точки O

$$O = (x_0, y_0) = \tau\left(V_x^н, V_y^н\right), \tag{4.12}$$

де τ - час зміщення ОУ в перерізі простору Q до початку вирішення конфлікту.

Напрямок переміщення ОУ за елементом траєкторії (OM_1) визначається СННП. Кут $\Delta\psi_1$ визначає відхиленням від напрямку прямого переміщення ОУ з точки O у точку P при переміщенні по ланці (OM_1), а значення $\Delta\psi_2$ визначає відхиленням від напрямку $\psi^1_{зм}$ при переміщенні з точки M_1 в точку

P. При переміщенні по елементу ланцюжка траєкторії (IO) ОУ зберігає швидкість $\left|\overrightarrow{V_н^1}\right|$, при переміщенні по лінії (OM_1) ОУ зберігає швидкість $\left|\overrightarrow{V_н^1}\right|$, а при переміщенні по лінії (M_1P) ОУ зберігає швидкість $\left|\overrightarrow{V_н^2}\right|$.

У відповідності з співвідношеннями (2.26), (4.5) і (4.11) запишемо систему рівнянь, які дозволяють знайти числове значення функціоналу ціни

$$\begin{cases} J = k_2 J_1 + (1 - k_3 k_4) J_2 + k_4 J_3 \\ J_1 = k_1 \left(\dfrac{\Delta V_1}{\left|\overrightarrow{V_н}\right|} \right)^2 + (1 - k_1) \left(\dfrac{\Delta \psi_1}{\dfrac{\pi}{2}} \right)^2 \\ J_2 = k_2 \left(\dfrac{\Delta V_2}{\left|\overrightarrow{V_н}\right|} \right)^2 + (1 - k_1) |OM_1| \left(\dfrac{\Delta \psi_2}{\dfrac{\pi}{2}} \right)^2 \\ J_3 = \left| 1 - \dfrac{|IP|}{|IO| + |OM_1| + |M_1P|} \right| \\ \Delta\psi_1 = \begin{bmatrix} (\Delta\psi_{зм}^1 - \psi_{зм}^1) \forall (\psi_{зм}^1 - \psi_{пр}) \in [0, 2\pi] \\ (2\pi - \psi_{зм}^1 + \psi_{пр}) \forall (\psi_{зм}^1 - \psi_{пр}) < 0 \end{bmatrix} \\ \Delta\psi_2 = \begin{bmatrix} (\psi_{зм}^2 - \psi_{пр}) \forall (\psi_{зм}^2 - \psi_{зм}^1) \in [0, 2\pi] \\ (2\pi - \psi_{зм}^2 + \psi_{зм}^1) \forall (\psi_{зм}^2 - \psi_{зм}^1) < 0 \end{bmatrix} \\ \Delta V_1 = \left|\overrightarrow{V_н^1}\right| - \left|\overrightarrow{V_н}\right| \\ \Delta V_2 = \left|\overrightarrow{V_н^2}\right| - \left|\overrightarrow{V_н^1}\right| \\ k_3 + k_4 \le 1 \\ k_1 \in [0,1] \\ k_2 \in [0,1] \end{cases}, \qquad (4.13)$$

де k_1, k_2, k_3, k_4 - коефіцієнти, які визначають переваги при синтезі стратегій управління ОУ.

Відповідно до співвідношення (4.13), функціонал J, який визначає стратегію управління ОУ, буде відповідати екстремальному числовому значенню виразу

$$\begin{cases} J^* = \inf\ J\left(k_1, k_2, k_3, k_4, \Delta V_1, \Delta V_2, \Delta\psi_1, \Delta\psi_2, S_{пр}, S_1, S_2, S_3\right) \\ \Delta\psi \in [0, 2\pi] \\ \Delta V \in V_e^0 \end{cases}, \qquad (4.14)$$

де $S_{пр} = |IP|$; $S_1 = |IO|$; $S_2 = |OM_1|$; $S_3 = |M_1P|$; V_e^0 – припустимі швидкості при переміщенні ОУ в просторі $G_{ріш}$.

Зважаючи на те, що всі коефіцієнти, що входять у співвідношення (4.13) і (4.14) квадратичного функціоналу ціни J, додатні, а в співвідношенні (4.13) відсутні члени, які містять взаємні множення змінних, то можна зробити висновок про те, що мінімальне значення виразу (4.14) буде відповідати випадку траєкторії переміщення ОУ з точки O безпосереднього в точку P. Виходячи з співвідношення (4.13) можна зробити висновок про гладкість, монотонність і наявність мінімуму функціоналу J у точці із значенням параметрів $(\Delta V_1 = 0, \Delta V_2 = 0, \Delta\psi_1 = 0, \Delta\psi_2 = 0)$ незалежно від значень коефіцієнтів k_1, k_2, k_3, k_4, що визначають характер монотонності зміни значення функціоналу J. В залежності від значення змінних функціоналу J визначається характер синтезованого рішення у вигляді переваги при виборі управління напрямком переміщення, швидкістю чи напрямком переміщення і швидкістю за елементами ланцюжка траєкторії та управління ОУ.

Значення коефіцієнтів k_1, k_2, k_3, k_4 задаються на етапі налаштування СІУ щодо динамічних та кінематичних характеристик, що забезпечує адаптивність синтезованих стратегій управління ОУ при рішенні конфлікту. Така особливість є суттєвою при синтезі стратегій управління в ТС, ТЕС та ОУ.

За своєю сутністю співвідношення (2.36), (4.11), (4.13) і (4.14) відображують принцип оптимальності Беллмана [10, 11] та правило зупинки при здійсненні процедури перебору (синтезі та вибору стратегії рішення). Сутність правила зупинки полягає в досягненні сталості значення критерію оптимальності при здійсненні процесу перебору

$$\Gamma_{зуп}=\begin{cases}0\forall\dfrac{J_{m-2}+J_{m-1}+J_m}{3J_m}>\varepsilon\\ 1\forall\dfrac{J_{m-2}+J_{m-1}+J_m}{3J_m}\le\varepsilon\end{cases},\qquad(4.15)$$

де m- номер шагу процесу перебору; J_m- значення критерію оптимальності для шагу перебору з номером m; ε- припустиме відносне відхилення значення функції ціни.

Приклад зміни значень функціоналу ціни при рішенні задачі конфлікту з використанням імітаційної моделі наведено на рис.4.2.

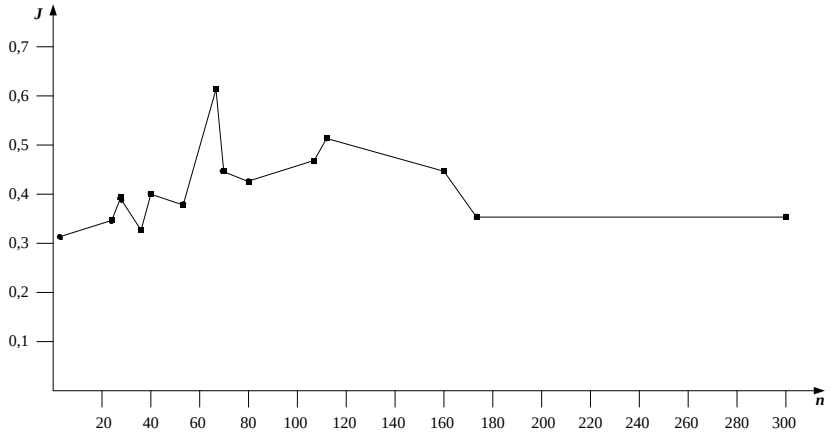


Рис. 4.2. Зміна значення функціоналу ціни

За результатами імітаційного експерименту визначено, що вирішення задачі конфлікту згідно співвідношенню (4.9) потребує незначної кількості процедур перебору при варіативній кількості ОС та довільній границі ПС $\Gamma_{гр}(Q)$.

Виходячи з співвідношень (2.36), (4.5), (4.9), (4.11), (4.13) - (4.15) оптимальне рішення конфлікту має вигляд

$$\mu^*=\inf_{J,\Gamma_{зуп}}K\left(M,A,S,\Gamma_{пр},G_{ріш},\mu\right).\qquad(4.16)$$

Рішення конфлікту K у відповідності з співвідношеннями (4.10) і (4.16) знаходиться на основі принципу оптимальності χ при заданому правилі зупинки $\Gamma_{зуп}$.

Таким чином можна визначити загальний алгоритм синтезу стратегій рішення конфлікту взаємодії об'єктів в умовах обмежень та невизначеностей для траєкторії переміщення ОУ:

- перший елемент ланцюжка траєкторії переміщення ОУ визначається з затримкою на час виконання обчислювальних процедур визначення простору Q, $G_{ріш}$, синтезу та прийняття рішення;

- другий і наступні елементи ланцюжка синтезується у відповідності з інформаційною множиною простору $G_{ріш}$, СННП та ПКУО.

При синтезі стратегії рішення конфлікту враховується наступне:

- усі ОС при вирішенні задачі синтезу стратегій рішення конфлікту знаходяться в стані прямолінійного рівномірного переміщення в просторі Q;

- невизначеність поведінки *i*-го ОС враховується в значенні d^i у відповідності з співвідношеннями (2.15) - (2.17);

- останній елемент ланцюжка траєкторії переміщення ОУ в просторі Q закінчується у заданій цільовій точці;

- при синтезі альтернативних рішень інформаційні множини перераховуються у відповідності з співвідношеннями (2.10) – (2.12), (2.15), (2.16), (2.18) - (2.24) на основі співвідношення (4.9);

- оптимальне рішення (4.16) визначається співвідношеннями (2.33), (4.5), (4.11), (4.13), (4.14) та правилом зупинки (4.15).

Таким чином можна зробити висновок про те, що запропонований метод синтезу ланцюжків управління з врахуванням обмежень при переміщенні ОУ за обраною траєкторією в просторі Q по своїй суті є підмножиною у загальній схемі рішення задач на основі методу послідовного аналізу варіантів.

Тобто, виходячи з можливостей мови $L(\Gamma_{пр})$, запропоновано алгоритм методу синтезу та вибору рішень щодо безконфліктної взаємодії ОУ з ОС за умов обмежень та невизначеностей при переміщенні ОУ в цільову точку простору Q.

Синтез рішень конфлікту при відображенні переміщення ОУ на площину перерізу здійснюється на основі множинного підходу до опису простору Q, в якому формуються значення параметрів управління переміщенням за елементами синтезованої траєкторії.

Альтернативні рішення синтезуються як траєкторії переміщення та ланцюжки управління ОУ в просторі Q. Траєкторія переміщення представляється в вигляді ламаної лінії з визначенням координат кінців її відрізків, кутів напрямків руху за елементами траєкторії та значеннями швидкості переміщення ОУ, що є сутністю ланцюжків рішень щодо управління ОУ та відповідає умовам безперервності.

Значення координат кінців відрізків траєкторії переміщення ОУ, кутів напрямків руху за елементами траєкторії та значеннями швидкості визначаються за умов того, що ланцюжок управління має бути можливим для реалізації, траєкторія повинна вкладатись в обмеження простору Q, а синтезовані рішення щодо напрямків переміщення та швидкості забезпечувати безконфліктне переміщення ОУ по відношення до всіх ОС.

Виходячи з моделі опису взаємодії об'єктів в просторі Q (2.10) - (2.12), (2.34), (2.35), можна виділити такі задачі:

- створення інформаційної множини образу простору $G_{ріш}$;
- синтез інформаційних множин образу ПКУО і СННП;
- розпізнавання образу зовнішньої ситуації з точки зору синтезу траєкторії переміщення ОУ в просторі Q;
- синтез траєкторії переміщення ОУ, як варіанту альтернативного рішення конфлікту у відповідності з критерієм вибору Φ (2.33), функціоналом ціни J (4.13) і (4.14) та правилом зупинки $\Gamma_{зуп}$ (4.15).

Виходячи з визначених інформаційних множин СННП і параметрів переміщення ОС в просторі Q, можна задати рекурсію їх перегляду на основі правила “найбільш небезпечним є ОС, час і відстань зближення з яким є

найменша, за умови наявності СННП, який пересікає синтезована траєктірія переміщення ОУ".

При синтезі стратегій рішення конфлікту слід враховувати діаграму зон передчасного попередження про ризик зближення з ОС (рис. 4.3).

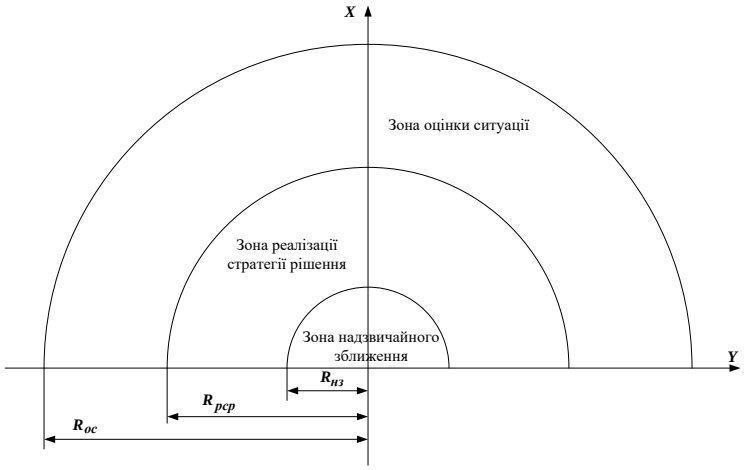


Рис. 4.3. Діаграма зон передчасного попередження про ризик зближення

Значення R_{oc}, R_{pcp} та $R_{нз}$ формуються на підставі вимог нормативних та керівних документів (на авіаційному та морському транспорті), політикою безпеки (для кібернетичного простору) або тактико-технічними вимогами щодо рішення задачі забезпечення безпеки в автоматизованих та автоматичних системах. В такому разі для ОС, які знаходяться в зоні R_{oc}, здійснюється лише спостереження, в зоні R_{pcp} синтезується стратегія, обирається рішення та виконуються дії щодо реалізації стратегії рішення конфлікту. В зоні $R_{нз}$ здійснюється реалізація стратегії управління, яка визначена нормативною та керівною документацією або (в разі її відсутності) управління ОУ з метою забезпечення його безпеки при вирішенні конфлікту. Для співвідношення співвідношенні (2.15) для i-го ОС визначається $d^i_{зад} = R_{pcp}$, а для ОУ визначається $d^0 = R_{нз}$.

Слід також зазначити, що представлення про простір рішення задачі синтезу траєкторії переміщення ОУ виявляється далеко неповним з точки зору формування інформаційного образу зовнішньої ситуації у вигляді ПКУО.

Розглянемо метод синтезу стратегій управління ОУ на основі інформаційних множин.

Нехай задано простір $G^{*}_{ріш}$ (рис. 4.4), для якого визначені:

- правила Θ_R безперервності простору $G_{ріш}$;

- правила визначення ознаки Π_μ приналежноті синтезованої траєкторії ПР;

- ознака приналежності f_μ траєкторії переміщення ОУ простору Q з обмеженнями $\Gamma_{гр}(Q)$ (2.4) у відповідності з співвідношеннями (3.46) - (3.48);

- ознака Θ_j приналежності точки замкнутій множині у відповідності з співвідношенням (2.3);

- ознака приналежності СННП простору рішення конфлікту f_{w_i};

- формальні правила, які регламентують взаємодію ОУ з ОС.

Цільовою точкою маршруту переміщення ОУ в просторі Q є $P=(x_p, y_p)$, яка задана в ортогональному базисі рішення задачі конфлікту.

Для точки P за правилом приналежності (2.3) визначимо

$$(\Theta_P = 1) \rightarrow (P \in G_{ріш}). \qquad (4.17)$$

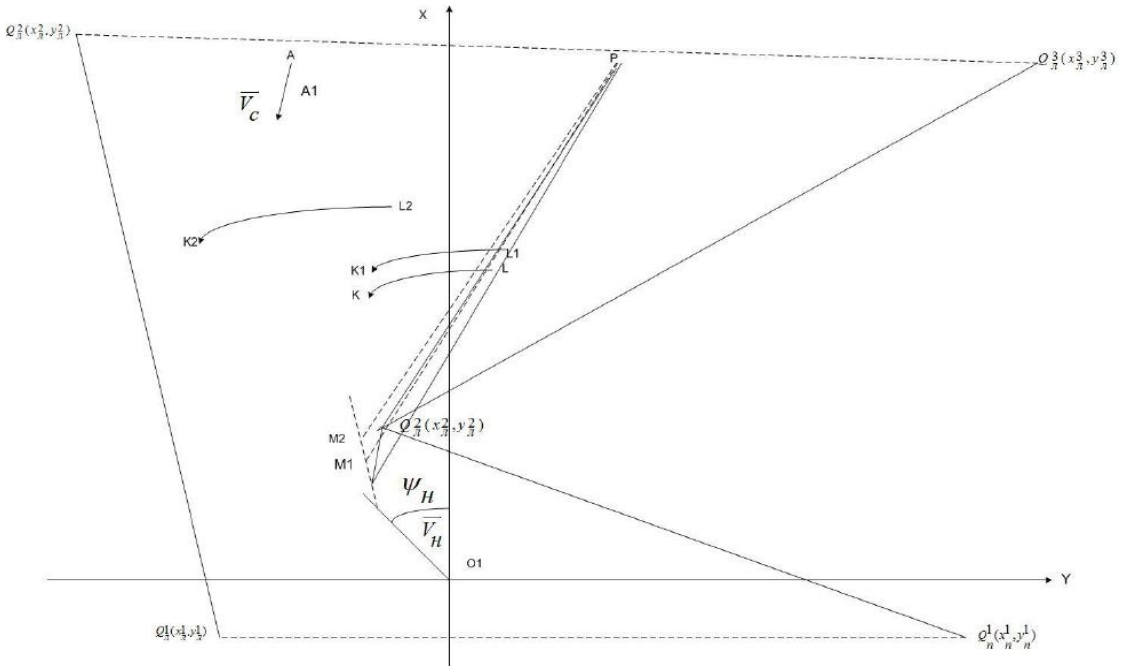


Рис. 4.4. Синтез траєкторії переміщення ОУ в умовах неопуклих обмежень

Визначимо напрямок переміщення ОУ φ_p з точки $O=(x_o,y_o)$ в кінцеву точку $P=(x_p,y_p)$.

При заданому базисі $Б^0$ точка O визначається як перетвореннння простору, що обумовлене переміщенням ОУ за час τ з точки O_1 в точку O. Тобто

$$O=(x_0,y_0)=\tau\left(V_н^x,V_н^y\right), \tag{4.18}$$

де $\overrightarrow{V_н}=\left(V_н^x,V_н^y\right)$ – вектор швидкості ОУ в базисі $Б$ моделі (2.10).

Таким чином напрямок переміщення ОУ з точки O в точку P можна визначити як

$$\varphi_p=\arcsin\left(\frac{x_p-x_0}{\sqrt{\left(x_p-x_0\right)^2+\left(y_p-y_0\right)^2}}\right). \tag{4.19}$$

Значення φ_p знаходиться за умови (4.17) через трансцедентну функцію (4.19), зважаючи на її нерозривність на всій множині області визначення.

Слід зазначити, що СННП для *i*-го ОС в загальному випадку є множиною, яка включає дві підмножини, що зрозуміло з співвідношень (3.33) - (3.40), (3.42), (3.43).

В такому разі визначимо ознаку можливості переміщення ОУ в просторі $G^*_{ріш}$ з точки O в точку P з врахуванням ІМНСПР згідно співвідношенням (4.19), (3.46), (3.47)

$$f_\varphi=\begin{bmatrix}1,\varphi_p\bigcap\limits_{G_{ріш}}W_i^l=\varnothing,i=\overline{1,N},l=[1,2]\\0,\varphi_p\bigcap\limits_{G_{ріш}}W_i^l\neq\varnothing,i=\overline{1,N},l=[1,2]\end{bmatrix}. \tag{4.20}$$

Справедливість значення $f_\varphi=1$ відповідно правила $П_W$ згідно співвідношення (4.20) свідчить про наявність вільного переміщення ОУ в цільову точку руху при дотриманні правил $П_W$ і $П_R$, що відповідає ситуації коли траєкторія переміщення ОУ не перетинається з множиною СННП в

просторі $G^{*}_{ріш}$. У такому випадку синтезована траєкторія переміщення і управління ОУ представляє собою дволанкову ламану лінію, переміщення за елементами якої можна здійснити без зміни швидкості, тобто синтезована траєкторія переміщення ОУ вміщує два елементи відповідно множини ПКУО, що визначена співвідношенням (3.1).

Якщо відповідно співвідношення (4.20) ознака $f_{\varphi}=0$, ОУ не має можливості забезпечити безконфліктне переміщення та управління в просторі $G^{*}_{ріш}$ за синтезованою траєкторією в цільову точку $P=(x_p,y_p)$ в базисі *Б*. В такому разі здійснюється рекурсія при переборі синтезованих рішень з врахуванням розрахованих значень СННП і границь множин W_i^l по кожному ОС в просторі $G^{*}_{ріш}$. При цьому напрямок переміщення ОУ φ_p в цільову точку $P=(x_p,y_p)$ простору $G^{*}_{ріш}$ визначається мінімально-перебірною процедурою, починаючи з напрямків на границі множин W_i^l, якими є ІМНСПР. Переміщення множин W_i^l визначає “забарвлення” ІМНСПР при забезпеченні можливості безконфліктного переміщення ОУ в цільову точку $P=(x_p,y_p)$ простору $G^{*}_{ріш}$. Таким чином здійснюється процедура синтезу траєкторії переміщення та управління ОУ для рішення конфлікту (2.35) і вибору оптимальної стратегії за співвідношенням (2.36).

Проаналізуємо виконання умови пересічення відрізку (OP) з межею $\Gamma_{гр}(Q)$ простору Q при синтезі траєкторії переміщення ОУ в цільову точку $P=(x_p,y_p)$ і визначимо правило та ознаку пересікання елементів синтезованої траєкторії переміщення ОУ з границею простору $G_{ріш}$

$$f_r=\begin{bmatrix}1,\exists(OP)\subset G_{ріш},\left(\Gamma_{гр}(Q)\bigcap(OP)=\varnothing\right)\\ 0,(OP)\not\subset G_{ріш},\left((OP)\bigcap\Gamma_{гр}(Q)\neq\varnothing\right)\end{bmatrix}. \qquad (4.21)$$

Відповідно до (4.21) значення $f_r = 0$ відповідає можливості безконфліктного переміщення ОУ в точку $P = (x_p, y_p)$. Найбільш складним є випадок $f_r = 1$ пересічення синтезованої траєкторії переміщення ОУ з межею $\Gamma_{гр}(Q)$. В такому разі для зменшення числа переборів і скорочення часу синтезу стратегії рішення визначають елемент лінії границі $\Gamma_{гр}(Q)$, який є найближчим до точки $O = (x_o, y_o)$. За визначеним елементу лінії границі $\Gamma_{гр}(Q)$ будується образ СННП з характеристичним вектором

$$W_i^0 = W_i^0\left(r_i^0, \varphi_i^н, \varphi_i^к\right), \tag{4.22}$$

де r_i^0 - відстань до середини i-го елемента лінії $\Gamma_{гр}(Q)$; $\varphi_i^н$ - напрямок (кут) початкової точки i-го елемента лінії $\Gamma_{гр}(Q)$ при відліку за напрямком від осі ординат проти годинникової стрілки; $\varphi_i^к$ - напрямок (кут) кінцевої точки i-го елемента лінії $\Gamma_{гр}(Q)$ при відліку від осі ординат за напрямком проти годинникової стрілки.

Співвідношення (4.22) і (3.46) визначають просторовий образ СННП, як для ОС так і для лінії $\Gamma_{гр}(Q)$.

Маючи дані про неприпустимі напрямки переміщення ОУ, визначається процедура перебору можливих варіантів рішення конфлікту шляхом зміни напрямку переміщення на кут, який відповідає переміщенню на кінець СННП з послідовним аналізом можливості безпечного переходу у цільову точку переміщення ОУ (рис.4.4).

За синтезованою траєкторією ОУ переміщується зі швидкістю $\left|\overrightarrow{V_н}\right|$ за напрямком $\psi_н$. Тоді через час τ ОС, який позначено точкою $A = (x_a, y_a)$, що має вектор істинної швидкості $\overrightarrow{V_c} = \left(V_c^x, V_c^y\right)$ зміститься у точку

$$A_1 = \left(x_a^1, y_a^1\right) = (x_a, y_a) + \tau\left(V_c^x, V_c^y\right). \tag{4.23}$$

Виходячи з опису простору $G^*_{ріш}$, розраховується траєкторія переміщення ОУ, яка оминає СННП, що має опис у вигляді (3.46) для *i*-го ОС та (4.22) для лінії межі $\Gamma_{гр}(Q)$.

Для $\Gamma_{гр}(Q)$, характеристичний вектор СННП має вигляд відрізка $\left(Q_n^1, Q_n^2\right)$.

В такому разі маємо інформаційний образ СННП

$$\begin{cases} W_1^0 = W_1^0\left(r_1^0, \varphi_1^н, \varphi_1^к\right) \\ r_1^0 = \dfrac{\left|\overrightarrow{OQ_n^1}\right| + \left|\overrightarrow{OQ_n^2}\right|}{2} \\ \varphi_1^н = \arcsin\left(\dfrac{x_n^2 - x_0}{\sqrt{\left(x_n^2 - x_0\right)^2 + \left(y_n^2 - y_0\right)^2}}\right). \\ \varphi_1^к = \arcsin\left(\dfrac{x_n^1 - x_0}{\sqrt{\left(x_n^1 - x_0\right)^2 + \left(y_n^1 - y_0\right)^2}}\right) \end{cases} \tag{4.24}$$

Отримавши інформаційний образ СННП, аналізуємо можливість прямого переміщення ОУ з точки Q_n^2 в точку P, тобто оцінюємо можливість переміщення ОУ за елементом траєкторії $\left(Q_n^2 P\right)$. Для обраного елемента траєкторії переміщення ОУ в просторі $G^*_{ріш}$ обчислимо інформаційний образ ПКУО для синтезу ланцюжка управління, який може бути здійсненим в діапазоні припустимих швидкостей ОУ на основі даних, які отримані в результаті розрахунку ЛШ. Відсутність припустимих швидкостей ОУ при аналізі можливості переміщення з точки Q_n^2 в точку P свідчить про розривність простору $G^*_{ріш}$ і відсутність можливості синтезу стратегії рішення конфлікту, що було визначено в співвідношеннях (4.6) - (4.8), за якими визначається ознака Θ_R безперервності простору. Про це свідчить наявність обчислених значень СННП, який визначено дугою $\overset{\frown}{L_1 K_1}$, яка перетинає відрізок синтезованої траєкторії $\left(Q_n^2 P\right)$. В разі неможливості реалізації

стратегії управління ОУ обирається інший напрямок його переміщення з врахуванням інформаційного образу СННП (4.24)

$$\varphi_{\partial} = \varphi_1^{н} - \Delta\psi(n-1), \tag{4.25}$$

де $\Delta\psi$ – інтервал зміни напрямків (кутів) траєкторії переміщення ОУ; n – номер напрямку переміщення, який аналізується.

При переміщення за напрямом φ_{∂} у відповідності з співвідношенням (4.25) отримуємо точки $M_i = \left(x_m^i, y_m^i\right)$, координати яких визначаються з умов можливості реалізації управління ОУ за ПКУО. В кожній точці M_i здійснюється перерахунок множин, які характеризують стан простору $G_{piш}^{*}$ та ПКУО при аналізі переміщення за елементом траєкторії $(M_i P)$ у цільову точку за умов лінійного прогнозу переміщення ОС.

Для кожної стратегії управління ОУ, яка може бути реалізована, обчислюється значення функціоналу ціни згідно співвідношень (4.13) і (4.14).

Аналіз значення функціоналу ціни для кожної синтезованої траєкторії переміщення ОУ за умови наявності стратегії управління, яка може бути реалізована, дозволяє вирішити задачу неопуклого програмування, тобто знайти оптимальне рішення

$$\mu^{*} = \arg\min_{\mu} J\,. \tag{4.26}$$

Знайдене рішення (4.26) відповідає визначенню конфлікту (4.9) та моделі його оптимального рішення (4.16).

Слід зазначити, що при синтезі траєкторій переміщення та стратегій управління ОУ, згідно співвідношенню (4.26) оптимальним є застосування евристичних методів пошуку згідно прогнозній моделі зміни стану простору $G_{piш}^{*}$ та ПКУО, які для евристичного методу пошуку визначаються ознаками, предикатами та правилами, які розглянуті вище.

Правило зупинки (4.15) є кількісною ознакою сталості варіантів синтезуємих рішень конфлікту для ОУ за критерієм (2.33), (4.5), (4.11), (4.13)

та (4.14). Процес пошуку результату може бути також перервано ОПР в ТЕС або відповідним правилом для СІУ ОУ і ТС.

4.2. Алгоритм методу інтегрального усікання варіантів

Алгоритм методу інтегрального усікання варіантів вміщує наступні кроки:

Крок 1. Визначається ознака приналежність цільової точки простору рішень.

Крок 2. Визначається напрямок переміщення ОУ в кінцеву точку траєкторії.

Крок 3. Початкова точка траєкторії визначається як перетворення простору, яке обумовлене переміщенням ОУ за час τ з початкової точки в точку можливої зміни напрямку переміщення.

Крок 4. Визначається напрямок переміщення ОУ з точки зміни напрямку в кінцеву точку та визначається кут зміни напрямку переміщення ОУ в кінцеву точку.

Крок 5. Визначається ознака можливості переміщення ОУ в кінцеву точку для перерахованих характеристик простору рішень з врахуванням ІМНСПР.

Крок 6. Визначається ознака можливості вільного переміщення ОУ в кінцеву точку траєкторії.

Крок 7. При неможливості вільного переміщення в кінцеву точку траєкторії застосовується рекурсія вибору рішень з врахуванням розрахованих значень СННП і ІМНСПР по кожному ОС в просторі рішень. Напрямок переміщення визначається за мінімально-переборною процедурою в просторі з границі ІМНСПР, яка зміщується в напрямку забезпечення можливості безконфліктного переміщення ОУ в кінцеву точку траєкторії.

Крок 8. Визначається ознака перетину оцінюваного елементу траєкторії переміщення ОУ в кінцеву точку межею простору рішень.

Крок 9. При перетині оцінюваного елементу траєкторії переміщення ОУ в кінцеву точку з межею простору рішень для зменшення числа переборів і скорочення часу синтезу стратегії рішення визначають елемент лінії границі простру рішень, який є найближчим до точки переходу на наступний елемент траєкторії. За визначеним елементу лінії границі простору рішень будується образ СННП з характеристичним вектором.

Крок 10. Визначається стратегія рішення у вигляді обходу СННП.

Крок 11. Відповідно інформаційного образу СННП, аналізується можливість переміщення ОУ в кінцеву точку траєкторії.

Крок 12. Для обраного елемента траєкторії обчислюється інформаційний образ ПКУО для синтезу ланцюжка гарантованого управління ОУ.

Крок 13. Визначається ознака розривності простору рішень. За ознакою розривності простору рішень визначається інший напрямок переміщення з врахуванням інформаційного образу СННП.

Крок 14. Здійснюється рекурсія розрахунку елементу траєкторії переміщення до закінчення діапазону напрямків чи досягнення рішення з урахуванням правила зупинки.

Розроблений метод інтегрального усікання варіантів і *P*-алгоритм синтезу рішень та ланцюжків гарантованого управління ОУ в ПР враховує релевантні фактори, які не були враховані в математичній моделі конфлікту взаємодії об'єктів в просторі $G_{ріш}$ та дозволяє зменшити множину припустимих альтернатив рішень.

Алгоритм методу інтегрального усікання варіантів та синтезу стратегій рішень наведено на рис. 4.5.

Алгоритм передбачає наявність інформації щодо ОС, ОУ, простору Q, мети переміщення ОУ, границі простору $\Gamma_{гр}(Q)$, значень припустимого

зближення ОУ з ОС, тощо. Усі похідні дані мають вводитись модулем приймання та перетворення похідних даних СІУ (рис. 2.2).

Робота алгоритму методу інтегрального усікання варіантів та синтезу стратегій рішень щодо управління ОУ починається з визначення змінних та параметрів (блок 2), які необхідні для функціонування алгоритму.

Функціонування алгоритму рішення конфлікту згідно співвідношень (2.33) - (2.35) і (4.9) вимагає наявності детермінованого визначення коефіцієнтів критерію (4.5) з врахуванням співвідношень (4.11), (4.13), (4.14).

В блоці 2 проводиться визначення параметрів функціоналу ціни згідно співвідношенням (4.11), (4.13), (4.14), (4.5) і наявних для обчислення критерію відбору синтезованих стратегій рішення конфлікту при переміщенні ОУ в цільову точку простору Q.

За наявністю даних щодо обчислення значень функціоналу ціни управління передається від блоку 3 до блоку 4. В іншому випадку управління передається блоку 28 для формування ознаки помилки та завершення роботи алгоритму.

Виконання розрахунків, які дозволяють визначити похідні дані, якими є термінальні множини початку і кінця траєкторії переміщення ОУ для імітаційної моделі формування простору рішень $G_{ріш}$, забезпечується блоком 4 за співвідношеннями (2.1) – (2.4). При цьому враховуються перетворення простору Q, які обумовлені часом затримки, пов'язаним з обчислювальними процедурами та прийняттям рішення згідно співвідношення (4.18) і (4.23).

При перетворенні простору Q здійснюється перевизначення границі простору $\Gamma_{гр}(Q)$, координат усіх ОС, координат цільової точки переміщення ОУ, а також похідних даних щодо траєкторії переміщення ОУ, яка відображає геометричне місце положень ОУ для стратегії управління і обирається згідно синтезованому ланцюжку управління.

Аналіз наявності ОС в просторі $G_{piш}$ забезпечується в блоці 5 з використанням співвідношень (4.6) - (4.8).

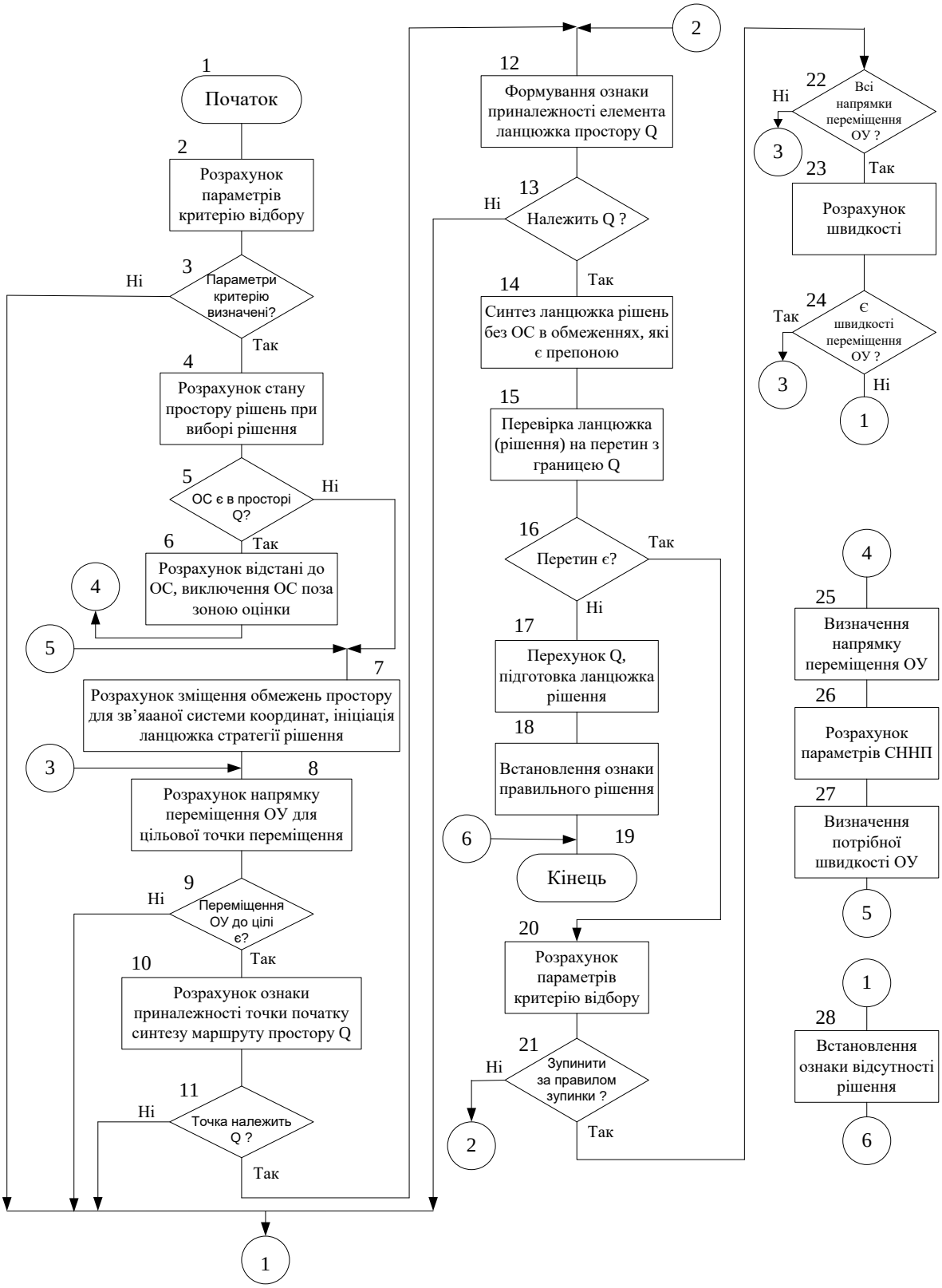


Рис. 4.5. Схема алгоритму методу інтегрального усікання варіантів та синтезу стратегій рішень

При наявності ОС в просторі $G_{ріш}$ управління передається блоку 6. В іншому випадку управління передається блоку 7.

Для визначення приналежності ОС до зон $R_{ос}$, $R_{рср}$ і $R_{нз}$, які визначають відстань зближення ОУ з ОС, в блоці 6 обчислюються відстані до ОС, що знаходяться в просторі $G_{ріш}$, та здійснюється ранжування ОС за відстанню до ОУ. Для оптимізації обчислювального процесу ОС, які не входять до зон ($R_{ос}$., $R_{рср}$, $R_{нз}$), не враховуються при рішенні конфлікту. Також здійснюється ініціація ланцюжка управління ОУ, як віртуальних даних в пам'яті обчислювального середовища СІУ.

Після виконання алгоритмічної частини блоку 6 управління передається засобам блоку 25, в якому визначається напрями переміщення ОУ в просторі Q.

В блоці 26 здійснюються процедури обчислення та оброблення даних щодо СННП для усіх ОС в просторі $G_{ріш}$.

Після визначення напрямків переміщення ОУ (блок 25) та простору рішення $G^{*}_{ріш}$ з врахуванням усіх обмежень СННП (блок 26) в блоці 27 визначаються швидкості переміщення ОУ при синтезі ланцюжків стратегій управління ОУ для рішення конфлікту взаємодії з ОС в просторі $G^{*}_{ріш}$ за умов обмежень та невизначеності. Значення швидкостей переміщення ОУ обираються з інформаційної множини ПКУО.

Після закінчення функціональних процедур блоку 27 управління передається блоку 7 для розрахунку параметрів перетворення простору $G_{ріш}$ та ініціації ланцюжків управління ОУ.

Перерахунок значення параметрів простору $G_{ріш}$ для зв'язаної з ОУ системи координат забезпечується в блоці 7. Перетворення простору спрощує подальші обчислення. За результатами перетворення простору здійснюється ініціація ланцюжка управління ОУ.

Засобами блоку 8 здійснюється розрахунок напрямку переміщення ОУ в цільову точку (кінцеву термінальну множину) простору $G^*_{ріш}$.

Аналіз наявності стану переміщення ОУ забезпечується шляхом визначення модуля вектору швидкості здійснюється в блоці 9. Якщо ОУ знаходиться в стані відсутності руху, управління передається блоку 28. В разі переміщення ОУ управління передається блоку 10.

Визначення ознаки приналежності точки початку переміщення синтезованому ланцюжку управління та траєкторії переміщення ОУ в просторі $G^*_{ріш}$ здійснюється в блоці 10.

Аналіз приналежності основних параметрів (координат) траєкторії переміщення ОУ простору $G^*_{ріш}$ здійснюється в блоці 11. При цьому аналізу підлягають координати точок початку, закінчення та зміни напрямків траєкторії переміщення ОУ (точки конкатенації елементів ланцюжків рішень) в просторі $G^*_{ріш}$.

В разі приналежності точок ламаної лінії траєкторії переміщення ОУ простору $G^*_{ріш}$ управління передається блоку 12. В іншому випадку управління передається блоку 28.

Визначення ознаки приналежності синтезованих ланцюжків траєкторії переміщення ОУ простору $G^*_{ріш}$ здійснюються в блоці 12.

Аналіз ознаки приналежності елемента ланцюжка синтезованого елемента ламаної лінії траєкторії переміщення ОУ простору $G^*_{ріш}$ виконується в блоці 13. В разі приналежності елемента траєкторії переміщення простору $G^*_{ріш}$ управління передається блоку 14. В іншому випадку - блоку 28.

Синтез траєкторії переміщення та управління для ОУ в разі відсутності обмежень на простір $G_{ріш}$ з боку ОС забезпечується блоком 14. Тобто має місце твердження $G^*_{ріш} \sim G_{ріш}$ і $\bigcup\limits_{G_{ріш}} W_i = \varnothing$.

Функціональні складові блоку 15 здійснюють перевірку ланцюжка траєкторії переміщення ОУ щодо наявності ознаки перетину з границею простору $\Gamma_{гр}(Q)$.

За результатами аналізу ознаки приналежності траєкторії переміщення простору Q та ланцюжка гарантованого управління ОУ множині ПКУО, що визначається блоком 15, управління передається блоку 17 в разі приналежності. В іншому разі управління передається блоку 20.

При переміщенні ОУ за обраною траєкторією визначення параметрів ланцюжка траєкторії переміщення ОУ простору Q і стратегії гарантованого управління множині ПЕУО здійснюється блоком 17.

Формування ознаки наявності правильного рішення, визначення значення критерію відбору та виведення результатів розрахунку параметрів траєкторії переміщення та ланцюжка рішень здійснюється блоком 18. В ТЕС здійснюється виведення параметрів траєкторії переміщення та ланцюжка управління ОУ для ОПР. При наявності рішення, яке відповідає вимогам критерію відбору (функції ціни) або правилу зупинки, процедура синтезу стратегій рішення завершується і управління передається блоку 19.

В блоці 19 виконуються процедури, що пов'язані з завершенням процесу синтезу стратегій рішення та функціонування алгоритму.

Обчислення значення параметрів критерію (4.5) за співвідношеннями (4.13) з врахуванням (4.4.14) та ознаки зупинки процесу синтезу стратегій за правилом зупинки (4.15) здійснюється в блоці 20.

Засобами блоку 21 здійснюється аналіз наявності ознаки зупинки обчислень за правилом зупинки при аналізі можливості переміщення ОУ за певним напрямком. За наявності ознаки зупинки управління передається блоку 22. В іншому разі для продовження процесу синтезу альтернативних стратегій рішення управління передається блоку 12.

Аналіз можливості подальшого синтезу стратегій рішення в просторі $G^{*}_{ріш}$ забезпечується блоком 22. За наявності можливості продовження

процесу пошуку стратегій рішення за можливим напрямком переміщення в просторі $G^{*}_{piш}$ управління передається блоку 8. В іншому випадку управління передається блоку 23.

Визначення можливості зміни швидкості переміщення ОУ за напрямком, який аналізується, здійснюється засобами блоку 23. В разі можливості зміни значення швидкості переміщення ОУ обчислюється значення швидкості та формується ознака наявності можливості зміни швидкості. В разі відсутності можливості зміни швидкості ОУ при переміщенні за напрямком, який аналізується, формується ознака неможливості зміни швидкості.

Аналіз ознаки можливості зміни швидкості переміщення ОУ, яку було визначено в блоці 23, здійснюється в блоці 24. За ознакою можливості зміни швидкості ОУ управління передається блоку 8, в іншому випадку - блоку 28.

За відсутністю синтезованих стратегій рішення в блоці 28 встановлюється ознака відсутності рішення і управління передається блоку 19.

Таким чином здійснюється синтез стратегій рішень за методом інтегрального усікання варіантів.

4.3. Рішення конфлікту в кібернетичному просторі

Відповідно до обраної політики безпеки та моделі доступу до об’єктів КС необхідно визначити технічні та технологічні рішення і засоби щодо вирішення задачі унеможливлення реалізації спроб несанкціонованого доступу (НСД) [164].

Вирішення такої задачі за своєю сутністю є вирішенням задачі конфлікту взаємодії об’єктів та суб’єктів КС з об’єктами КП, які можуть бути ініційовані будь-яким суб’єктом або об’єктом такого простору.

Для вирішення конфлікту взаємодії об’єктів в КП запропоновано метод інтегрального усікання варіантів [158-160, 162-164], який методологічно

дозволяє знаходити рішення задачі конфлікту в КП за умов наявності гарантованого управління ОУ, цілісності та цілеспрямованості його поведінки.

Виходячи з моделі конфлікту взаємодії об’єктів в кібернетичному просторі спостереження (КПСП), яка визначена співвідношенням (2.60) – (2/62), метод синтезу стратегій управління при вирішенні конфлікту вміщує два основних етапи: синтез моделі опису КП; синтез формального опису ПРКУ, в якому гарантовано існує комбіноване управління КС при вирішенні конфлікту.

В такому разі задача вирішення конфлікту представляється як варіаційна, а екстремальний елемент за критерієм вибору стратегій рішення є припустимим, незважаючи на можливу стрибкоподібну зміну вектора опису параметрів стану КС та КП.

Критерій оптимального управління враховує принципи гомеостатичності функціонування технічної ергатичної КС [160].

В такому разі підхід до розв’язання конфлікту за методом інтегрального усікання варіантів, що пов'язаний з побудовою мінімізуючої послідовності траєкторій (стратегій, ланцюжків) в просторі $G_{piш}$, відображає принцип оптимальності при синтезі та виборі стратегії рішення конфлікту.

При дослідженні систем захисту інформації (СЗІ) від НСД до суб’єктів та об’єктів, які складають множину елементів КС, оберемо дискреційну модель доступу (DAC).

Виходячи з того, що КП є метричним і, при необхідності, може бути декомпозований на класи еквівалентності [156], СЗІ можна представити у вигляді декартового добутку множин, складовими частинами яких є елементи системи захисту, якими є суб’єкти, об’єкти, рівні доступу, операції, тощо.

В такому разі матриця доступу Ω на множині суб’єктів O^S та об’єктів O^O КС згідно співвідношення (2.60) може бути визначена як декартовий добуток

$$\Omega = O^S \times O^O . \qquad (4.27)$$

Виходячи з співвідношення (4.27), для класів еквівалентності O^S і O^O визначимо права доступу для матриці Ω згідно множини прав доступу R та множини елементарних операцій

$$\alpha_z \in \alpha, \forall z \in Z, \tag{4.28}$$

які передбачені функціональністю КС та політикою безпеки СЗІ.

$$\Omega[O_i^S, O_j^O] = R_k, \forall O_i^S \in O^S, \forall O_j^O \in O^O \forall R_k \in R. \tag{4.29}$$

Значення Z в співвідношенні (4.28) визначається функціональністю КС.

Слід зазначити, що в співвідношенні (4.29) визначено

$$\begin{cases} i = \{1,\ldots,n\} \\ j = \{1,\ldots,m\}, \\ k = \{1,\ldots,l\} \end{cases} \tag{4.30}$$

де n – кількість суб'єктів O^S, m – кількість суб'єктів O^O, k – кількість станів R.

Результатом застосування політики безпеки, що задається моделлю безпеки для конкретного об'єкту захисту КС, є дозвіл або заборона виконання елементарних операцій α_z відповідно співвідношенню (4.28) над об'єктом захисту.

В результаті виконання елементарних операцій α відповідно матриці доступу Ω згідно визначеній множині прав доступу для класів еквівалентності O^S і O^O можна формально визначити перехід КС в результаті виконання елементарної операції з множини α із стану $Q = (O^S, O^O, \Omega)$ до стану $Q' = (O^S', O^O', \Omega')$. Тобто в результаті виконання примітивного оператора з множини α здійснюється перетворення стану КС

$$Q \underset{\alpha}{\mapsto} Q'. \tag{4.31}$$

Таким чином співвідношення (4.27) - (4.31) визначають переміщення КС в КП.

Визначимо рівні безпеки суб'єктів O^S і об'єктів O^O за допомогою функції рівня безпеки [164, 166]

$$\sigma : O^S \cup O^O \to L, \tag{4.32}$$

яка ставить у відповідність кожному об'єкту і суб'єкту рівень безпеки, з множини рівнів безпеки (секретності) КС

$$L = \bigcup_{d=1}^{D} L_d, \tag{4.33}$$

на якому визначена решітка

$$\Lambda = \left(L, \bigcup_{t=1}^{T} \Lambda_t \right). \tag{4.34}$$

В співвідношенні (4.33) L_d визначає рівні безпеки (секретності), наприклад, "нетаємноно", "таємно", тощо і може позначатись як вербально, так і кількісно. Загальна кількість рівнів безпеки КС для співвідношення (4.33) дорівнює D.

Для співвідношення (4.34) Λ_t визначає такі оператори, як нестроге відношення порядку для рівнів секретності, найменшої верхньої границі, найбільшої нижньої границі, тощо.

Слід зазначити, що в співвідношенні (4.32) функція рівня безпеки σ з решіткою рівнів визначають усі допустимі відносини доступу між суттєвостями КС.

В такому разі множина станів КС може бути визначена набором

$$\Re = (\sigma, \Omega). \tag{4.35}$$

З врахуванням (4.35) визначимо модель системи

$$\Sigma = (\nu_0, R, \Im), \tag{4.36}$$

яка описує процес переходу КС з початкового стану ν_0 в наступний стан ν' при запиті r за умови $r \in R$, де R – множина припустимих запитів до КС.

Тобто КС за функцією переходу

$$\Im : ((\Re \times R) \to \Re) \tag{4.37}$$

переходить в стан

$$\nu' = \Im(\nu, r). \tag{4.38}$$

Слід зауважити, що стан КС $\nu \in \Re$ може бути досягнуто при виконання послідовності n запитів та станів

$$\langle (\nu_0, r_0), ..., (\nu_n, r_n) \rangle \forall\, \nu_{i+1} = \Im(\nu_i, r_i), i = \overline{1, n}. \tag{4.39}$$

В такому разі з врахуванням основної теореми безпеки Белла-ЛаПадули та співвідношень (4.35) – (4.38) перехід КС із стану ν_0 в стан ν' є тривіально досяжним та безпечним. Тим самим теорема Мак-Ліна визначає умови безпечності функціонування КС за умов зміни її стану при виконанні дій.

Незважаючи на це твердження, слід мати на увазі, що КС може бути безпечною але не мати безпечної функції переходу.

При виконанні операції зчитування між віддаленими об’єктами виникає потік інформації від об’єкту, з якого інформація зчитується до об’єкту, який здійснив запит на зчитування інформації для відповідного суб’єкту. Потік, який з’являється, є безпечним у зв’язку з тим, що інформація є недоступною для неавторизованого суб’єкту. Разом з тим в розподіленій КС зчитування ініціюється запитом від одного об’єкту до іншого. Такий запит створює потік інформації, який йде в зворотньому (неправильному) напрямку (запис в об’єкт з нижчим рівнем секретності або в об’єкт, для якого не передбачена взаємодія з об’єктом, який створив запит).

Слід мати на увазі, що в такому разі необхідно забезпечити безпеку КС в початковому та кінцевому стані, включаючи процес переходу між станами. Також для КС необхідно реалізувати таку функцію переходу, яка відповідала би вказаним умовам безпеки, а саме

$$C : O^S \bigcup O^O \to \square\left(O^S\right), \tag{4.40}$$

де $\square\left(O^S\right)$ - множина усіх підмножин O^S і визначає підмножину суб’єктів, яким дозволено змінювати рівень безпеки, для заданого об’єкта або суб’єкта.

Тоді з врахуванням співвідношення (4.37 – (4.40)) модель конфліктуючої КС

$$\sum = \left(\nu_0, R, \Im^{\alpha}\right) \tag{4.41}$$

складається з початкового стану ν_0, множини запитів R і функції переходу $\Im^{\alpha}$, яка переводить КС зі стану в стан по мірі опрацювання запитів. В такому разі функції переходу (4.40), яка визначає наступний стан КС після виконання певним суб'єктом деякого запиту, з'явився ще один аргумент – суб'єкт, від якого надходить цей запит, оскільки результат переходу залежить від того, який суб'єкт його ініціював

$$\Im^{\alpha} : \left(O^S \times \Re \times R\right) \to \Re . \tag{4.42}$$

Виходячи з співвідношень (4.40) - (4.42), можна стверджувати, що КС, яка знаходиться у стані $\nu \in \Re$ переходить при отриманні запиту $r \in R$ із стану ν до стану $\nu' \in \Re$, при одержані запиту $r \in R$ від суб'єкта $o^s \in O^S$ переходить зі стану ν до стану

$$\nu' = \Im^{\alpha}\left(o^s, \nu, r\right). \tag{4.43}$$

Важливо зазначити, що функція переходу (4.43) є авторизованою за умови, якщо для кожного переходу згідно співвідношенню (4.35) $\nu = \left(\sigma, \Omega\right)$ та $\nu' = \left(\sigma', \Omega'\right)$ і

$$\forall x \subset O^O \bigcup O^S \Leftrightarrow \sigma' \neq \sigma \Rightarrow o_s \not\subset C(x), \tag{4.44}$$

що відповідає умовам безпеки згідно співвідношення (4.40).

З співвідношення (4.44) слідує те, що в ході авторизованого переходу рівень безпеки суб'єкта або об'єкта може змінитися тільки тоді, коли суб'єкт, що виконує перехід, належать множині суб'єктів, уповноважених змінювати рівень цього суб'єкта або об'єкта.

Відношення щодо ініціації об'єкта суб'єктом чи іншим об'єктом є ключовими для аналізу можливостей здійснення переходу системи в незахищений стан при зміні характеристик матриці доступу Ω, які передбачені функціональністю КС та політикою безпеки СЗІ згідно співвідношенню (4.29).

З метою моделювання можливості отримання Z - системи за результатами переведення КС в новий стан доцільним є використання підходів щодо вирішення задачі спостереження та пошуку, як в підпросторі КП, що притаманний конкретній КС, з використанням методів спостереження та пошуку, які використовуються в системах штучного інтелекту.

Виходячи з співвідношень (4.27) – (4.44) можна зробити висновок про те, що КС при переході з стану ν в стан ν' згідно моделі Σ з функцією переходу $\Im$ має набір правил та відносин в шарі гіперкубу існування КС з шарами у вигляді матриць доступу Ω з множиною можливих запитів R, множиною припустимих елементарних операцій α та множиною неприпустимих елементарних операцій α_z. Тобто маємо модель конфлікту взаємодії КС, як ОУ, що взаємодіє з варіативною множиною КС в КП за умов обмежень та невизначеностей, яка відповідає за своєю сутністю ТММ згідно співвідношенням (2.10) – (2.12).

Синтезуючи ланцюжки управління при переміщенні КС за синтезованими траєкторіями в КП можна здійснювати управління ОУ в СІУ СЗІ щодо запобігання НСД. Слід зазначити, що при використанні неприпустимих елементарних операцій α_z в СІУ, отримуємо систему несанкціонованого доступу до ресурсів КС в КП. КП за своєю сутністю є простором $G_{ріш}$.

Траєкторії переміщення та ланцюжки управління КС в просторі $G_{ріш}$ синтезуються за методом інтегрального усікання варіантів.

Стан КС, як траєкторія переміщення в просторі $G_{ріш}$ при гарантованому управлінні згідно ланцюжка дій, що визначені співвідношеннями (4.35) – (4.44), забезпечує виникнення нових відношень та правил, які фактично дозволяють побудувати Z - систему. В сенсі рішення конфлікту створення Z - системи визначає можливість переходу КС в стан, який визначається правилами і відношеннями, що виникають при взаємодії КС, як ОУ, з варіативною множиною КС в КП за умов обмежень та невизначеностей.

Висновки до четвертого розділу

1. На основі теоретико-множинного уявлення взаємодії об’єкту управління з варіативною множиною об’єктів спостереження за умов обмежень і невизначеностей запропоновано метод інтегрального усікання варіантів рішення конфлікту.

2. Задачу синтезу управління сформульовано як задачу розв'язання конфлікту взаємодії ОУ з ОС в просторі спостереження згідно принципу оптимальності та правилу зупинки.

3. Розглянуті підходи до вибору критерію оптимальності при синтезі стратегій та виборі управління ОУ в умовах конфлікту.

4. Отримані результати відрізняються від відомих здатністю:

1) процедурного рішення конфлікту, який:

- не залежить від сутності простору спостереження, а також природи об’єктів;

- не потребує необхідності адаптації до конкретного ОУ;

- ґрунтується на адитивному критерії оцінювання рішень за синтезованими траєкторіями переміщення ОУ за умови гарантованого безперервного управління;

2) забезпечує синтез стратегій управління ОУ при вирішенні конфлікту на основі інформаційних множин, що визначають простір рішення;

3) визначає ознаку завершення процесу рішення конфлікту, який ґрунтується на методі інтегрального усікання варіантів;

4) визначає ознаку розривності простору рішень при відсутності гарантованого управління ОУ.

5. Розроблений метод синтезу та вибору стратегій оптимального управління ОУ в ПР для варіативної множини ОС в умовах обмежень та невизначеностей, розроблений на основі теретико-множинної, семіотичної та семантичної моделей, а також методу інтегрального усікання варіантів за умов гарантованого управління ОУ дозволяє:

- обирати метод оптимального управління об’єктом в умовах конфлікту, обмежень і невизначеності при варіативній множині ОС в просторі спостереження;

- визначати чисельне, а не аналітичне подання моделі рішення конфлікту і обмежень конфліктуючої системи;

- визначати оптимальну траєкторію переміщення ОУ за умов оптимального управління;

- визначати формалізовану структуру алгоритму, що спрощує процес рішення конфлікту;

- забезпечувати *P*-час рішення конфлікту;

- забезпечувати стійкість функціонування СІУ ОУ.

6. Розглянуто модель взаємодії кібернетичних систем та їх елементів в кібернетичному просторі.

7. Розглянуто теоретико-множинну модель управління конфліктом взаємодії об’єкту управління в кібернетичному просторі з відкритою множиною об’єктів спостереження в умовах обмежень та невизначеності.

Основні результати розділу опубліковані у роботах [133, 134, 138, 139, 155, 156, 157, 161, 163, 164, 165, 166, 167, 168, 172, 175].

РОЗДІЛ 5. ПЕРСПЕКТИВИ ЗАСТОСУВАННЯ СИСТЕМИ ІНТЕЛЕКТУАЛЬНОГО УПРАВЛІННЯ ПРИ РІШЕННІ КОНФЛІКТУ ВЗАЄМОДІЇ ОБ'ЄКТУ УПРАВЛІННЯ З ВАРІАТИВНОЮ МНОЖИНОЮ ОБ'ЄКТІВ СПОСТЕРЕЖЕННЯ В УМОВАХ ОБМЕЖЕНЬ ТА НЕВИЗНАЧЕНОСТЕЙ

5.1. Удосконалення інформаційних технологій автоматизації процесів інтелектуального управління рішенням конфлікту

5.1.1. Розрахунок інформаційної множини простору комбінованого управління об'єктом

Похідними даними для розрахунку ПКУО є параметри, які визначені моделлю взаємодії об'єктів в ПР (2.10) - (2.11), граматикою $\Gamma_{пр}$ (2.48), алфавітом R^0 (2.46), простором $G^*_{ріш}$, мети переміщення ОУ, тощо.

Розрахунок інформаційної множини ПКУО здійснюється згідно співвідношень (3.9) – (3.32) за наступним алгоритмом:

Крок 1. Обчислення відстані до ОС.

В разі знаходження ОС в зоні $[R_{ос}, R_{нз}]$ (рис. 4.3) на відстані, яка не менша припустимої для зближення ОУ з будь-яким ОС, вирішується задача оцінки місця знаходження ОС та можливості зміни параметрів переміщення (напрямку та швидкості).

На основі даних щодо обмеження простору рішень $G_{ріш}$ або його перерізу площиною формується замкнена лінія границі $\Gamma_{гр}(Q)$ за правилом замикання обмежень простору Q з правого та лівого боку ("початок - початок", "кінець - кінець").

В зв'язаній з ОУ системі координат, визначається множина точок перетину границі ПР з траєкторіями переміщення ОС при лінійному прогнозі

їх переміщення, а також кількість перетинів з границею $\Gamma_{гр}(Q)$, координати точок перетину та відстань від точки перетину до вершини ламаної лінії границі $\Gamma_{гр}(Q)$ з лінією, що визначає траєкторію. За отриманими даними щодо перетину траєкторії переміщення ОС визначається ознака зміни напрямку переміщення ОС в межах $\Gamma_{гр}(Q)$, а також визначається додатній чи від'ємний знак зміни напрямку переміщення.

Крок 2. Визначаються ОС, з якими можливим є стан конфлікту взаємодії з ОУ при переміщенні в просторі Q.

Ознака можливості виникнення конфлікту визначається за умовою визначеного предикату $\eta_0^i = \left(d^0 \in [R_{ос}, R_{нз}]\right) \wedge \left(d^i \in [R_{ос}, R_{нз}]\right)$ для кожного i-го ОС. В разі $\eta_0^i = true$ для i-го ОС ситуація взаємного переміщення з ОУ визначається як конфліктна.

Крок 3. Аналіз приналежності ОС простору $G_{ріш}$ з границею $\Gamma_{гр}(Q)$.

Ознака приналежності визначається правилом (2.3) тільки для ОС, що визначені в кроці 2 як такі, з якими можливим є стан конфлікту взаємодії з ОУ при переміщенні в просторі Q.

Крок 4. Розраховуються значення параметрів, які необхідні для обчислення ПКУО.

В якості параметрів визначено діапазон можливих напрямків переміщення ОУ (4.2); крок зміни напрямків переміщення ОУ, що аналізуються; діапазон можливих значень швидкості переміщення ОУ (4.1); крок зміни значення швидкості переміщення ОУ.

Відповідно співвідношень (3.2) – (3.8) обчислюються значення коефіцієнтів для вирішуючого правила, яке визначено співвідношеннями (3.9) – (3.32).

Крок 5. Визначаються інтервали безпечної швидкості переміщення ОУ згідно вирішуючого правила (3.9) – (3.32) для кожного з ОС, щодо яких

можливим є стан конфлікту взаємодії з ОУ при переміщенні в просторі Q. Зазначені інтервали є ЛШ.

Формування ПКУО здійснюється згідно співвідношення (3.1).

В разі відсутності ОС в просторі Q ПКУО визначається без обмежень.

При наявності в просторі Q ОС визначається ЛШ згідно вирішую чого правила за співвідношеннями (3.9) – (3.32). За визначеними ЛШ формується інформаційний образ ПКУО (3.1) як об'єднання множин ЛШ для всіх ОС.

Розглянемо приклад розрахунку ПКУО в двомірному евклідовому просторі для трьох довільних ОС (рис. 5.1), дані про які наведено в табл. 5.1.

Таблиця 5.1.

Об'єкти спостереження

Номер об'єкта	Координати (одиниць відстані)	Вектор істинної швидкості	Припустиме зближення (одиниць відстані)
1.	(6.0, -4.5)	(-15.0, 0.0)	1.5
2.	(2.5, -5.0)	(-10.0, -15.0)	1.0
3.	(-5.0, 1.0)	(15.0, 0.0)	2.0

В прикладі розрахунку ПКУО для ПСП встановлена незамкнена границя простору (табл. 5.2). За правилами замикання отримуємо границю $Г_{гр}(Q)$ простору Q.

Таблиця 5.2.

Обмеження простору спостереження та пошуку

Вершини лінії лівої границі обмежень		Вершини правої границі обмежень	
№ з/п	Координати	№ з/п	Координати (одиниць відстані)
1.	(-8.0, 5.6)	1.	(-8.5, -1.0)
2.	(-0.5, 7.4)	2.	(-8.5, -7.0)
3.	(-3.0, 5.5)	3.	(-5.0, -2.0)
4.	(4.3, -3.3)	4.	(1.5, -7.0)
5.	(4.7, 1.8)	5.	(4.0, -7.0)
6.	(8.0, -3.0)	6.	(4.0, -5.0)
7.		7.	(10.0, -5.0)

Характеристики ОУ, а також дані, які необхідні для розрахунку ПКУО наведені в табл. 5.3 та відображені в імітаційній моделі, інтерфейс якої наведено на рис. 5.1.

Таблиця 5.3.

Об'єкт управління

№ з/п	Показник	Значення
1.	Найбільша швидкість ОУ	40 одиниць/одиниця часу
2.	Час запізнення на прийняття рішення	0,1 одиниць часу
3.	Радіус зони оцінки ситуації	16 одиниць
4.	Напрямок переміщення ОУ	315 градусів
5.	Швидкість ОУ	12 одиниць/одиниця часу
6.	Початок сектору перегляду ситуації (ПКУО)	225 градусів
7.	Кінець сектору перегляду ситуації (ПКУО)	45 градусів
8.	Крок перегляду ПКУО	5 градусів

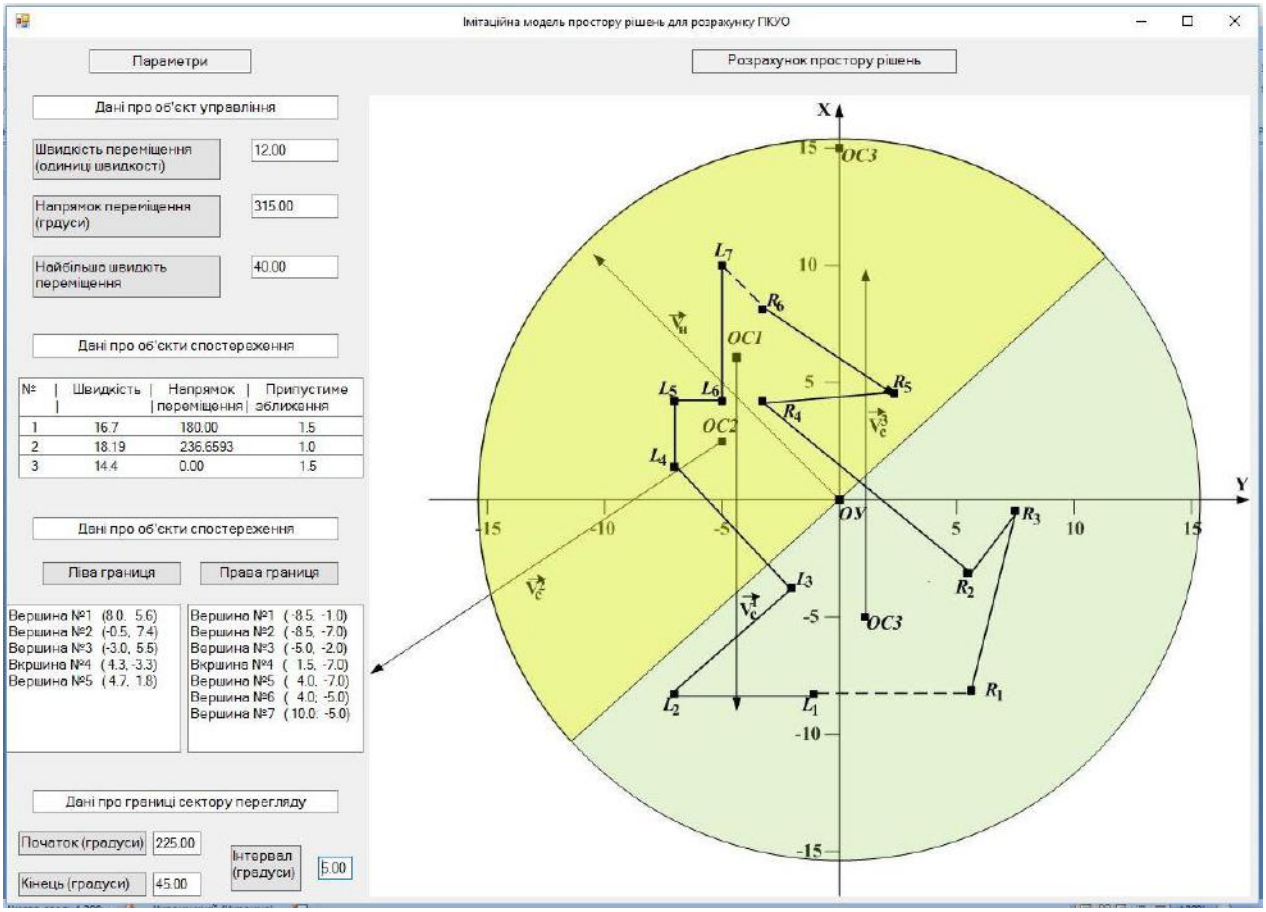


Рис. 5.1. Простір рішень для прикладу розрахунку ПКУО

Ламана лінія правої границі обмежень простору Q позначена, як лінія $(R_1R_2R_3R_4R_5R_6)$. Ламана лінія лівої границі обмежень простору Q позначена, як лінія $(L_1L_2L_3L_4L_5L_6L_7)$. За правилом замикання простір Q обмежений відрізками(R_6L_7) та (R_1L_1). ОС, які позначені $OC1$, $OC2$, $OC3$ мають відповідні вектори швидкості $\overrightarrow{V_c^1}$, $\overrightarrow{V_c^2}$, $\overrightarrow{V_c^3}$. Вектор швидкості ОУ позначено $\overrightarrow{V_н}$.

На рис. 5.2 наведено результат розрахунку інформаційної множини ПКУО за даними, які представлені в табл. 5.1 - 5.3.

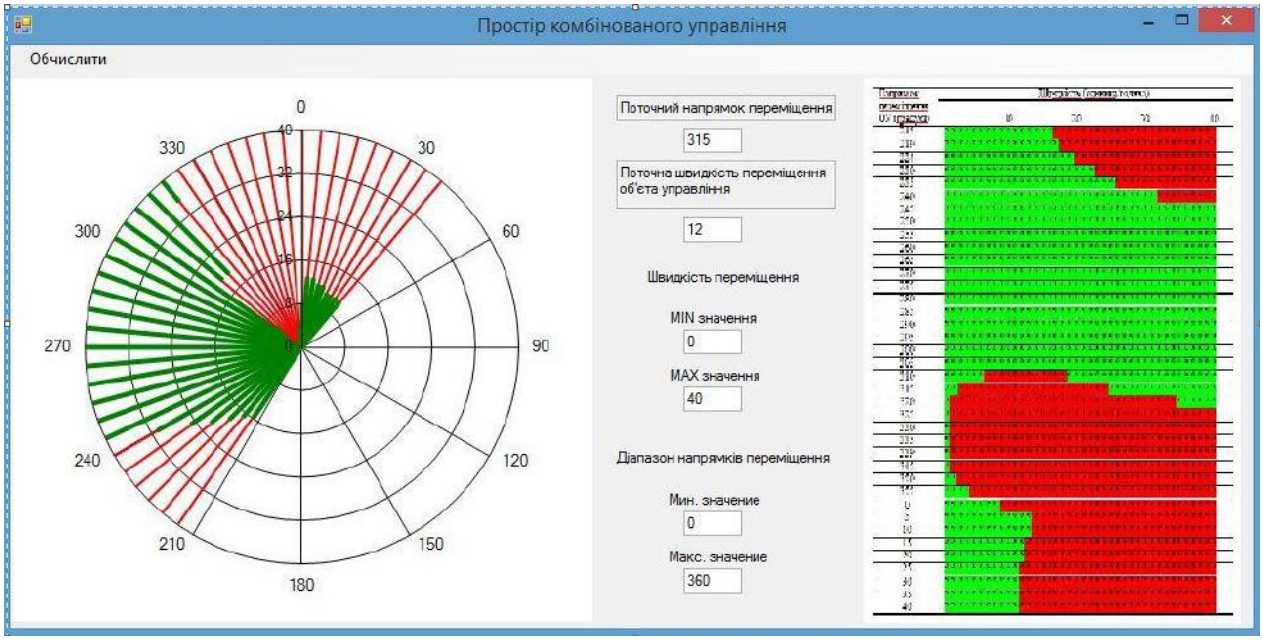


Рис. 5.2. Відображення результату розрахунку ПКУО

В лівій частині відображені напрямки можливого переміщення ОУ в просторі $G_{ріш}$. За кожним напрямком наведені відповідні множини ЛШ. Розрахунки здійснені з врахуванням значення часу запізнення на прийняття рішення, який заданий в таблиці 5.3. На рис. 5.2 зеленим кольором позначені безпечні швидкості переміщення ОУ, а червоним - небезпечні. Таким чином, для ОУ можна обрати варіант безпечного комбінованого управління при синтезі та виборі рішень щодо ланцюжка управління ОУ за синтезованою траєкторією згідно мови $L(\Gamma_{пр})$, яка забезпечує опис стратегії переміщення

ОУ в цільову точку, що належить множині термінальних кінцевих станів для мови $L(\Gamma_{пр})$ з використанням граматики $\Gamma_{пр}$ згідно співвідношення (2.48).

5.1.2. Синтез простору рішень

Для спрощення представлення простору $G_{ріш}$, як багатомірного і топологічного, та його перерізів площинами параметрів, визначаються формальні описи множин та ознак приналежності простору $G_{ріш}$: точки, елемента траєкторій переміщення ОУ, СННП, ІМНСПР, ланцюжка гарантованого управління ДО, ПКУО, збурень.

При формальному описі простору $G_{ріш}$ похідною є інформація щодо границі $\Gamma_{гр}(Q)$ простору Q, параметрів стану і переміщення ОУ та довільного ОС у відповідності до моделі (2.10) – (2.12).

Формальний опис простору $G_{ріш}$ визначається співвідношеннями (2.10), (2.11), (2.14) – (2.39), (3.1) – (3.44), (3.46) – (3.48). Переміщення ОУ і ОС є похідними для синтезу стратегій та вибору рішення щодо рішенні задачі конфлікту. Параметри переміщення ОС визначаються з геометричних співвідношень в топологічному просторі Q або його перерізі площиною параметрів.

Визначення ознаки приналежності точки простору Q здійснюється за співвідношенням (2.3).

Інформаційний образ границі простору $\Gamma_{гр}(Q)$ визначається за похідними даними відповідно правилу замикання (3.42).

Якщо термінальна множина кінцевих точок траєкторії переміщення ОУ в просторі $G_{ріш}$ не належить цьому простору ($T \not\subset G_{ріш}$) доповнення границі простору $\Gamma_{гр}(Q)$ здійснюється шляхом додавання додаткових двох точок до обмежень простору в напрямку точки, яка знаходиться поза межами простору.

Тобто, елемент лінії обмежень простору, який є найближчим до цільової точки, зміщується за напрямком, визначеним вектором, який побудовано з початку координат до цільової точки переміщення ОУ. Довжина вектору зміщення елемента (відрізку) лінії обмежень має бути більшою на довільне значення щодо відстані від точки до точки перетину побудованого вектору з відрізком лінії обмежень (рис. 5.3), що не порушує граматику загального виду $\Gamma_{пр}$ (2.48).

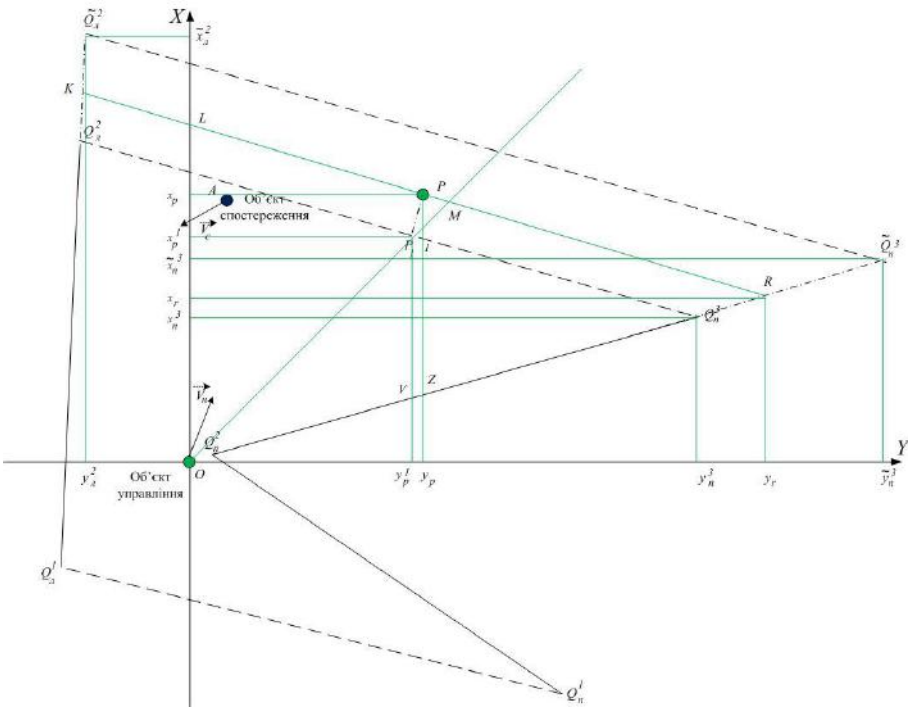


Рис. 5.3. Доповнення границі простору рішення

Границя простору $\Gamma_{гр}(Q)$ визначена лінією $\left(Q^1_л, Q^2_л, Q^3_п, Q^2_п, Q^1_п\right)$, яка враховує правило замикання (3.42). Кінцева точка $P\left(x_p, y_p\right) \notin G_{ріш}$, але кінцева точка переміщення належить множині термінальних позицій $P\left(x_p, y_p\right) \in T$, що відповідає граматиці загального виду $\Gamma_{пр}$ (2.48) і мові семіотичної системи $L\left(\Gamma_{пр}\right)$.

Виходячи з геометричного тлумачення такого випадку можна визначити правило доповнення простору Q або його проекції на площину параметрів, за яким точка переміщення ОУ $P\left(x_p, y_p\right)$ буде належати простору $G_{ріш}$.

Визначимо координати співвідношення для обчислення координат границі $\Gamma_{гр}(Q)$ простору $G_{ріш}$ або його проекції

$$\begin{cases} \tilde{x}_n^i = \dfrac{x_p}{x_p^1} x_n^i k_{под}, i = \{2, N_n\} \\ \tilde{y}_n^i = y_n^i - \dfrac{\tilde{x}_n^i}{x_n^i - x_n^{i-1}} \left(y_n^i - y_n^{i-1}\right) \\ \tilde{x}_л^j = \dfrac{y_p}{y_p^1} y_л^o k_{под}, j = \{2, N_л\} \\ \tilde{y}_л^j = y_л^j - \dfrac{\tilde{x}_л^j}{x_л^j - x_л^{j-1}} \left(y_л^j - y_л^{j-1}\right) \end{cases}, \qquad (5.1)$$

де $k_{под}$ - коефіцієнт подовження $(k_{под} \geq 1)$.

Ознака приналежності елемента ланцюжка траєкторії переміщення ОУ стратегій синтезованих рішень простору $G_{ріш}$ визначається правилом $П_\mu$ відповідно співвідношенням (3.44), (2.4) , (3.41) – (3.44).

Приклад відображення доповнення простору $G_{ріш}$ в імітаційній моделі наведено на рис. 5.4.

Суттєвими характеристиками стану простору $G_{ріш}$ при рішенні конфлікту взаємодії ОУ з відкритою множиною ОС в умовах невизначеності і обмежень є СННП і ІМНСПР.

Параметрами, які визначають СННП є координати, відстані та напрями переміщення на кінці СННП. В залежності від співвідношення швидкостей та напрямків взаємного переміщення ОУ і ОС в просторі $G_{ріш}$ для кожного i-го ОС кількість СННП визначається значенням ознаки $\varsigma_i \in [0,4]$. Ознака $\varsigma_i = 0$ визначає відсутність конфлікту при взаємному переміщенні ОУ та i-го ОС. Інші значення параметру ς_i відповідають кількості перетинів з колом (рис. 3.2) і при зустрічному та супутньому переміщенні ОУ та i-го ОС. Параметри, які характеризують взаємне переміщення ОУ та i-й ОС визначаються співвідношеннями (3.2) - (3.8).

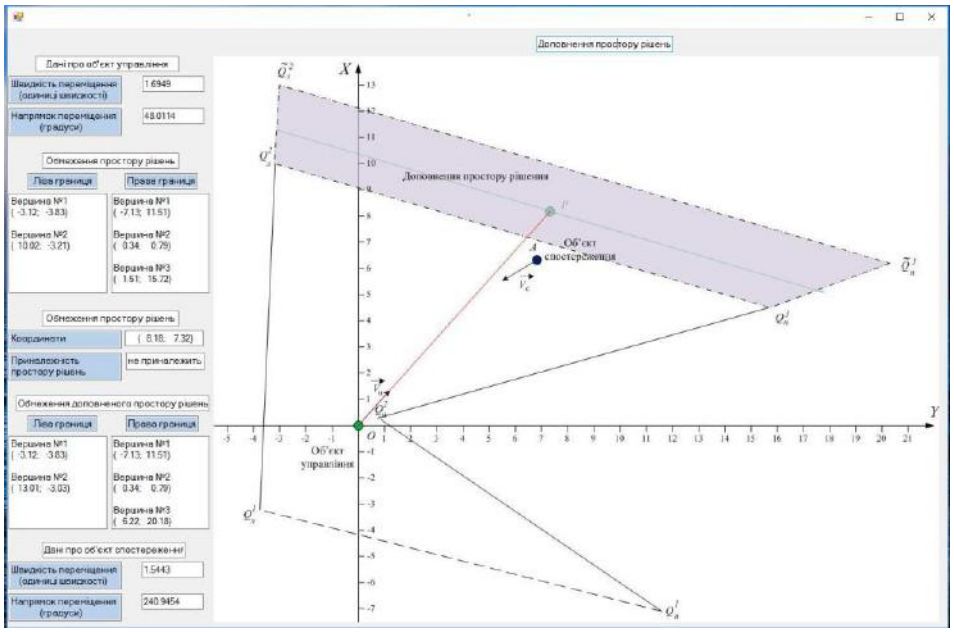


Рис. 5.4. Відображення простору рішень в імітаційній моделі

Алгоритм розрахунку параметрів СННП наведено на рис. 5.5. В наведеному алгоритмі значення $I2(1) \in [0,4]$ та $I2(2) \in [0,4]$ співпадають з значенням ознаки ς_i і визначають усі можливі випадки переміщення ОУ з i-м ОС при зустрічному переміщенні та переслідуванні.

Для кожного СННП в просторі $G_{ріш}$ згідно співвідношення (3.46) визначаються ІМНСПР при взаємодії ОУ з ОС. На підставі формального уявлення про ІМНСПР відповідно до співвідношення (3.47) визначається простір рішення $G^*_{ріш}$, в якому враховані геометричні місця точок можливого конфлікту взаємодії ОУ з усіма ОС в просторі Q, що визначено співвідношеннями (2.34) – (2.36).

Формальний опис процесу переміщенні ОУ в просторі Q за синтезованою траєкторією в множину термінальних кінцевих позицій T визначається згідно граматики $\Gamma_{пр}$ (2.48) відповідно граматиці загального виду $\Gamma^i_{пр}$ (2.37) за умови гарантованої реалізації ланцюжка управління U^0_j (2.41), формальної граматики $\Gamma_{пр}$(2.48) в моделі M, яка визначена співвідношеннями (2.33) – (2.36), (2.10) – (2.43), (2.37) – (2.40). При цьому

мова семіотичної системи $L(\Gamma_{пр})$ дозволяє побудувати дерево зв'язків та відношень лексичних структур опису простору $G^*_{ріш}$ та задовольняє вимогам перетворення та еквівалентності згідно співвідношенням (2.12), (2.13), (2.57).

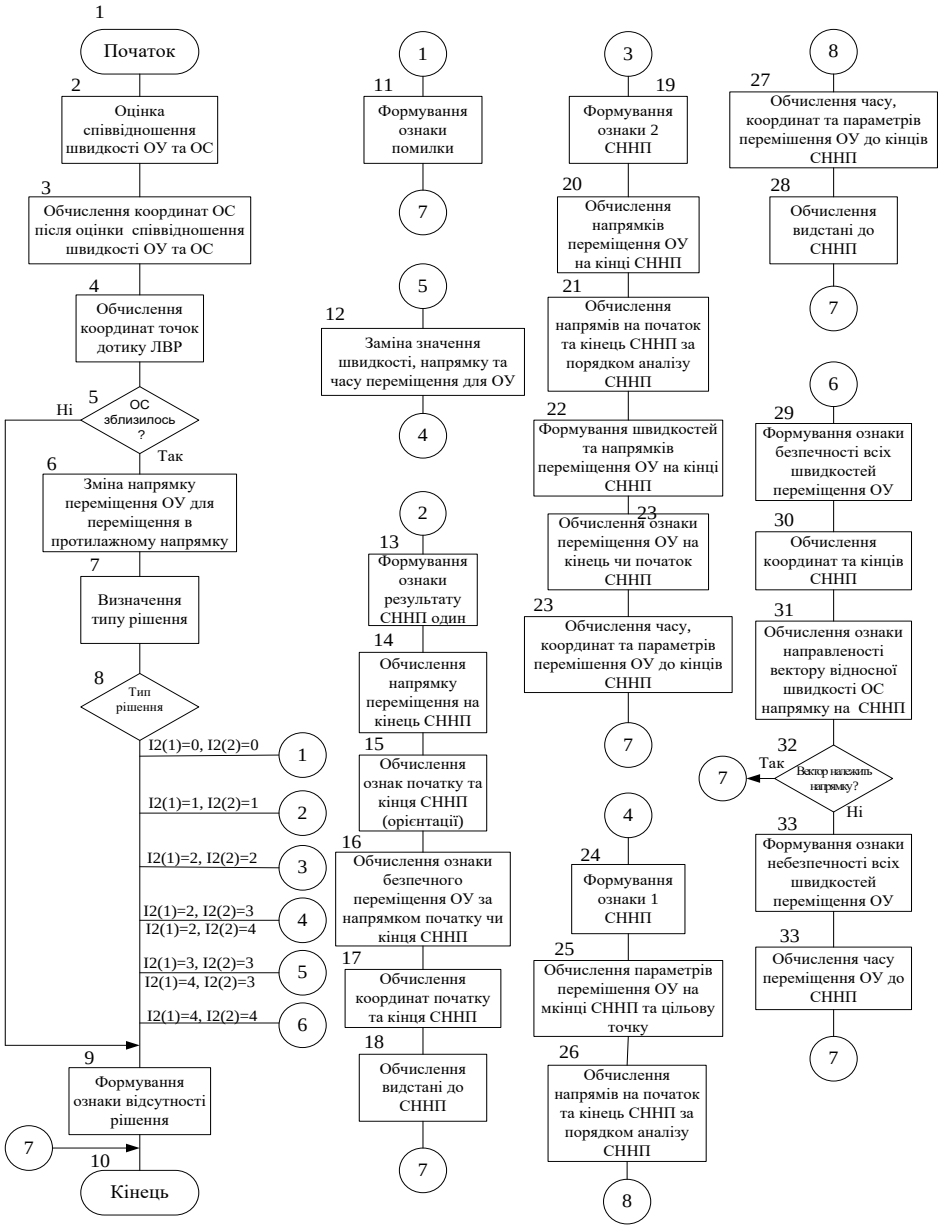


Рис. 5.5. Схема алгоритму визначення СННП при переміщенні ОУ

В такому разі автоматне відображенням S^0 (2.59) на множину простору $G_{ріш}$ дозволяє описати простір $G^*_{ріш}$ і задовольняє умовам повноти для ОУ.

Приклад розрахунку параметрів простору $G^*_{ріш}$ з використанням імітаційної моделі наведено на рис. 5.6.

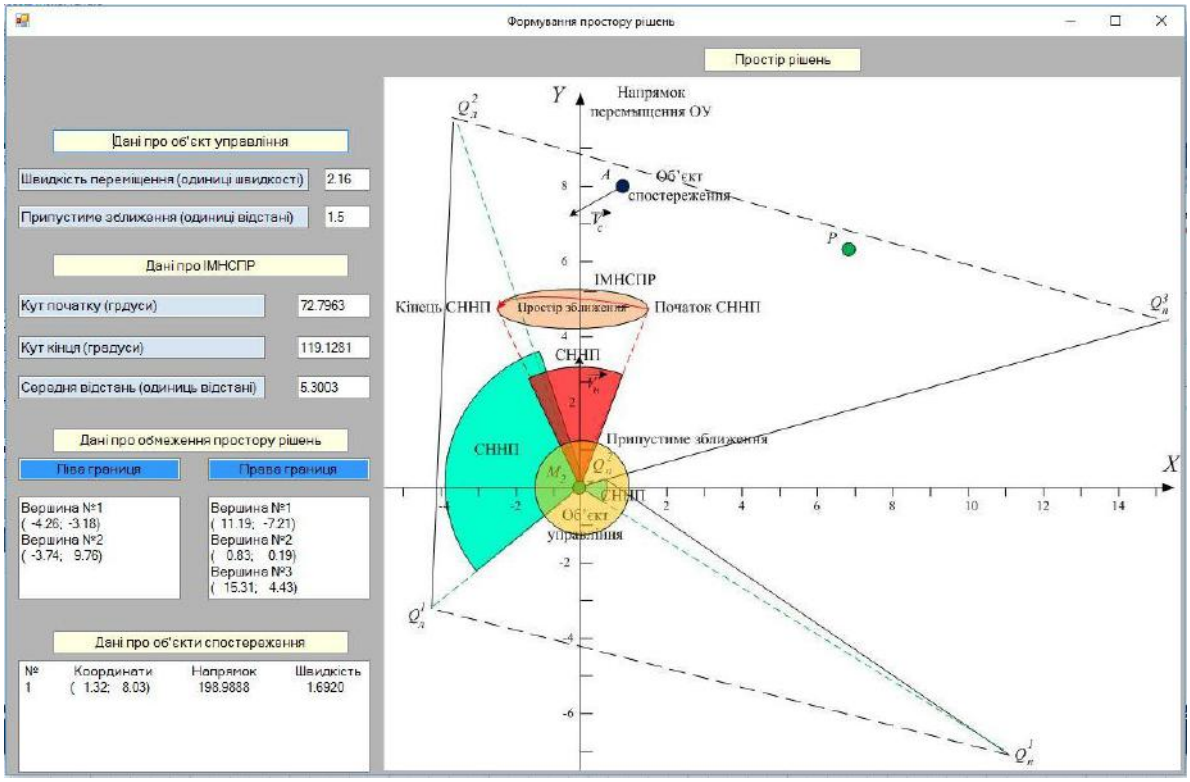


Рис. 5.6. Розрахунок параметрів простору рішення

Інтерфейс імітаційної моделі має чотири зони, в яких відображуються похідні дані та результати обчислень параметрів простору $G^*_{ріш}$.

СННП, які визначені в імітаційній моделі, мають відношення до ОУ та границі $\Gamma_{гр}(Q)$ простору $G_{ріш}$.

Червоним кольором позначено СННП, який розраховано виходячи з характеру взаємного переміщення ОУ та ОС.

ІМНСПР орієнтовано за напрямком переміщення в просторі $G^*_{ріш}$. В такому разі ПКУО визначає ланцюжок гарантованого управління ОУ, за яким має переміщатись ОУ в просторі $G^*_{ріш}$.

Похідні дані для розрахунку параметрів простору $G^*_{ріш}$ визначаються описом стану взаємодії ОУ з ОС з використанням мови семіотичної системи $L(\Gamma_{пр})$, яка визначена граматикою загального виду $\Gamma_{пр}$ відповідно співвідношенню (2.48).

5.1.3. Побудова простору гарантованого управління динамічного об’єкта

Проблема керованості ДС є самостійним напрямком теорії управління. Незважаючи на отримані в останній час результати для нелінійних ДС, проблема керованості залишається невирішеною. Досить важкою ця проблема залишається для нелінійних власно-нестійких ДС та ДО, область керованості яких за умов обмежень на управління та фазові змінні за математичною моделлю, не займають повністю простір станів ДС або ДО і має границю.

При дослідженні функціонування ДС та ДО з застосуванням математичної моделі визначається сукупність параметрів та їх кількісних характеристик, які при певних припущеннях визначають співвідношення між ними і описують функціонування реальної ДС та ДО в фізичному просторі переміщення [106, 166]. В такому разі встановлюється однозначне взаємне відображення параметрів, які характеризують переміщення ДС та ДО в фізичному просторі та параметрів, які визначають поведінку ДО в математичному просторі.

Метою рішення задачі конфлікту є забезпечення цілісності ДС та ДО, які знаходяться в стані конфлікту, при збереженні цілей їх переміщення в просторі Q [102, 137]. В такому разі ДО є ОУ в СУ, яка переміщується в просторі Q згідно синтезованих стратегій управління. СУ здійснює адаптивне управління ДО згідно співвідношень (3.49) – (3.54) в умовах конфлікту та невизначеності при забезпеченні основних принципів організмічної теорії систем [58, 102, 103].

В такому разі важливим є забезпечення функціонального гомеостазису СУ ДО, як адаптивної системи з структурою, що може перестроюватись, із збереженням необхідного гомеостазису та доцільності переміщення ОУ в просторі Q.

Синтез стратегії управління ДО визначається областями припустимих значень параметрів, які входять в математичну модель ДО і задовольняють співвідношенням (3.56) і (3.57).

З метою синтезу управлінь для СУ ДО необхідно визначити його властивості шляхом побудови динамічного образу простору керованих та напівкерованих станів ОУ шляхом дослідження властивостей математичної моделі (3.58) – (3.72). В разі моделювання переміщення ОУ в просторі Q відповідно співвідношенню (3.56) дослідження властивостей простору управління ОУ здійснюється шляхом аналізу властивостей системи диференційних рівнянь, які описують переміщення істотно-нелінійного ДО.

Простір рішень є безперервним, а рішення для СУ ДО обираються за критерієм, який враховує особливості переміщення ОУ в просторі Q.

Розглянемо алгоритм розрахунку та формального відображення динамічного образу простору керованих та напівкерованих станів ОУ на прикладі істотно-нелінійного ДО, яким є літальний апарат (ЛАП).

При перерізі динамічного образу простору керованих та напівкерованих станів ОУ площинами параметрів управління отримаємо відображення перерізу, який можна дослідити та обрати управління СУ ДО [159, 165, 166]. Перерізи простору відображають множину можливого розмаїття інваріантних функціональних поведінок ДС та ДО.

Відображення динамічного образу простору керованих та напівкерованих станів ОУ має особливе значення при побудові СІУ ДО для вирішення задачі конфлікту взаємодії з відкритою множиною об’єктів спостереження в умовах обмежень та невизначеності.

Динамічний образ простору керованих та напівкерованих станів нелінійного ОУ в перерізах простору площинами параметрів управління в загальному випадку представляє геометричне місце усіх оптимальних траєкторій, які побудовані у відповідному фазовому просторі. За умови безперервного управління динамічний образ простору керованих та

напівкерованих станів може розглядатись як простір гарантованого управління ДО [101, 102, 138].

Визначимо границі областей управління динамічного образу простору керованих та напівкерованих станів нелінійного ОУ на прикладі математичної моделі просторового переміщення ЛАП.

В загальному випадку динаміку просторового руху ДО та керуючі впливи можна описати системою рівнянь

$$\begin{cases} \dfrac{dv_1}{dt} = f_1(v_1,...,v_n,t,u_1,...,u_{n-r}) \\ \qquad\qquad ... \\ \dfrac{dv_r}{dt} = f_r(v_1,...,v_n,...,u_1,...,u_{n-r}), \\ v_{r+1} = u_{r+1}(v_1,...,v_n,t) \\ \qquad\qquad ... \\ v_n = u_n(v_1,...,v_n,t) \end{cases} \tag{5.2}$$

де $V(t)=\{v_1,...,v_r,t\}$ - множина параметрів математичної моделі; $U(t)=\{u_{r+1},...,u_{n-r},v_1,...,v_r,t\}$ - множина керуючих впливів; $n-$кількість рівнянь; $r-$кількість параметрів.

Визначимо множини параметрів, які впливають на переміщення ДО за напрямками в тримірній ортогональній системі координат, яка є зв'язаною з ДО

$$\begin{cases} X=\{x_1,...,x_q,u_{r+1},...,u_n,t\}, X\subset V, g\le n \\ Y=\{y_1,...,y_s,u_{r+1},...u_n,t\}, Y\subset V, s\le n \\ Z=\{z_1,...,z_w,u_{r+1},...,u_n,t\}, Z\subset V, w\le n \end{cases}. \tag{5.3}$$

З врахуванням (5.3) диференційні рівняння системи (5.2) можна представити у вигляді

$$\begin{cases} \overset{\bullet}{P}_X = F_X(X,Y,Z) \\ \overset{\bullet}{P}_Y == F_Y(X,Y,Z), \\ \overset{\bullet}{P}_Z = F_Z(X,Y,Z) \end{cases} \tag{5.4}$$

де $P_{\underset{X}{\bullet}} = \frac{dX}{dt}$, $P_{\underset{Y}{\bullet}} = \frac{dY}{dt}$, $P_{\underset{Z}{\bullet}} = \frac{dZ}{dt}$, ($P_X$, P_Y, P_Z) - множини значень параметрів, які визначають зміни напрямку переміщення ДО вздовж відповідних осей зв'язаної системи координат ОУ.

Визначимо систему рівнянь (5.4), яка описує поведінку ДО при наявності керуючого пристрою для СІУ або ЕСУ, у вигляді

$$\begin{cases} P_{\underset{X}{\bullet}} = K_X(P_X - X) \\ P_{\underset{Y}{\bullet}} == K_Y(P_Y - Y) \\ P_{\underset{Z}{\bullet}} = K_Z(P_Z - Z) \end{cases}. \qquad (5.5)$$

В співвідношенні (5.5) K_X, K_Y, K_X є коефіцієнтами передачі регулятора СІУ або ЕСУ, які враховують вимоги щодо якості процесу управління ДО.

Визначимо співвідношення для розрахунку простору значень припустимих кутових швидкостей при просторовому переміщенні ЛАП. За математичною моделлю просторового переміщення ЛА [58, 165, 166] маємо

$$\begin{cases} \overset{\bullet}{\omega}_x = \frac{1}{K_{\omega_x}}\left(\frac{Sl\rho V^2}{2J_x}\left(m_x^{cm} + \frac{l}{2V}m_x^{вр}\right) + \frac{J_y - J_z}{J_x}\omega_y\omega_z\right) \\ \overset{\bullet}{\omega}_y = \frac{1}{K_{\omega_y}}\left(\frac{Sl\rho V^2}{2J_y}\left(m_y^{cm} + \frac{l}{2V}m_y^{вр}\right) + \frac{J_z - J_x}{J_y}\omega_x\omega_z\right), \\ \overset{\bullet}{\omega}_z = \frac{1}{K_{\omega_z}}\left(\frac{Sl\rho V^2}{2J_z}\left(m_z^{cm} + \frac{b_A}{V}m_z^{вр}\right) + \frac{J_x - J_e}{J_z}\omega_x\omega_y\right) \end{cases} \qquad (5.6)$$

де J_x, J_y, J_z - проекції головного моменту інерції на вісі координат; S – площа крила; l – довжина крила; V – модуль вектора швидкості; ρ – щільність повітря; b_A – довжина середньої аеродинамічної хорди крила; m_x^{cm}, m_y^{cm}, m_z^{cm}, $m_x^{вр}$, $m_y^{вр}$, $m_z^{вр}$ – проекції моментів, які визначають обертання ЛАП навколо центра мас.

Згідно математичній моделі просторового переміщення ЛАП моменти в співвідношенні (5.6) залежать від певних параметрів, а саме:

α – кута атаки;

β – кута ковзання;

$\delta_н$ – кута відхилення керма напрямків;

δ_e – кута відхилення елеронів;

δ_i – кута відхилення інтерцепторів;

ω_x, ω_y, ω_z – проекцій вектора кутової швидкості;

$\varphi_{ст}$ – кута встановлення стабілізатора.

Згідно моделі просторового переміщення ЛАП маємо

$$\begin{cases} m_x^{ст} = m_x^{ст}(\alpha, \beta, \delta_з, \delta_н, \delta_e) \\ m_y^{ст} = m_y^{ст}(\alpha, \beta, \delta_з, \delta_н) \\ m_z^{ст} = m_z^{ст}(\alpha, \delta_з, \delta_в, \delta_i, \varphi_{ст}) \\ m_x^{вр} = m_x^{вр}(\alpha, V, \omega_x, \omega_y, \delta_з) \\ m_y^{вр} = m_y^{вр}(\alpha, \delta_з, \omega_x, \omega_y) \\ m_z^{вр} = m_z^{вр}(\omega_z) \end{cases}. \qquad (5.7)$$

З врахуванням співвідношень (5.5) і (5.6) визначимо значення припустимих кутових швидкостей ЛАП

$$\begin{cases} P_{\omega_x} = \dfrac{1}{K_{\omega_x}} \left(\dfrac{Sl\rho V^2}{2J_x} \left(m_x^{ст} + \dfrac{l}{2V} m_x^{вр} \right) + \dfrac{J_y - J_z}{J_x} \omega_y \omega_z \right) - \omega_x \\ P_{\omega y} = \dfrac{1}{K_{\omega_y}} \left(\dfrac{Sl\rho V^2}{2J_y} \left(m_y^{ст} + \dfrac{l}{2V} m_y^{вр} \right) + \dfrac{J_z - J_x}{J_y} \omega_x \omega_z \right) - \omega_y \\ P_{\omega_z} = \dfrac{1}{K_{\omega_z}} \left(\dfrac{Sl\rho V^2}{2J_z} \left(m_z^{ст} + \dfrac{b_A}{V} m_z^{вр} \right) + \dfrac{J_x - J_e}{J_z} \omega_x \omega_y \right) - \omega_z \end{cases}. \qquad (5.8)$$

Обчислення множин припустимих значень кутових швидкостей P_{ω_x}, P_{ω_y}, P_{ω_z} при переміщенні ЛА здійснюється послідовно для кожного моменту часу переміщення ОУ. Тобто з моменту початку процесу управління ДО t_0 до моменту закінчення процесу управління t_k маємо співвідношення (5.8), яке визначає динаміку зміни множини припустимих кутових швидкостей в часі за умови приналежності управління ДО за математичною моделлю згідно співвідношення (5.2)

$$\begin{cases} \overline{P_{\omega_x}} = \left\{P_{\omega_x}(t_0),...,P_{\omega_x}(t_k)\right\} \\ \overline{P_{\omega_y}} = \left\{P_{\omega_y}(t_0),...,P_{\omega_y}(t_k)\right\}. \\ \overline{P_{\omega_z}} = \left\{P_{\omega_z}(t_0),...,P_{\omega_z}(t_k)\right\} \end{cases} \quad (5.9)$$

Побудова множин припустимих значень кутових швидкостей при переміщенні ДО для моменту часу t_i може бути визначена як

$$\begin{cases} \overline{P^*_{\omega_x}(t_i)} = \bigcap_{j=0}^{i} P_{\omega_x}(t_j) \\ \overline{P^*_{\omega_y}(t_i)} = \bigcap_{j=0}^{i} P_{\omega_y}(t_j). \\ \overline{P^*_{\omega_z}(t_i)} = \bigcap_{j=0}^{i} P_{\omega_z}(t_j) \end{cases} \quad (5.10)$$

Співвідношення (5.9) і (5.10) визначають різноманітність стратегій управління ДО. За співвідношенням (5.10) фактично отримуємо багатомірний конус простору станів та гарантованого управління ДО.

Обчислення значень припустимих кутових швидкостей та прискорень здійснюється для множини припустимих швидкостей та висот для певних режимів польоту ЛАП та можливих значень параметрів управління ДО. Слід зазначити, що розмір та форма простору гарантованого управління ЛАП залежить від показників якості процесу управління K_{ω_x}, $K_{\omega_y}, K_{\omega_z}$ для кутових швидкостей, які входять в співвідношення (5.8).

Значення кутових швидкостей ω_x, ω_y, ω_z, які визначають з співвідношень (5.6) і (5.8), надалі використовуються для розрахунку значень інших істотних змінних, які визначають просторове переміщення ЛАП.

Гомеостатичність управління ДО визначаються співвідношенням (5.10) за умови

$$\begin{cases} \overline{P^*_{\omega_x}(t_i)} \neq \varnothing, \forall t_i \in [t_0,...,t_k] \\ \overline{P^*_{\omega_y}(t_i)} \neq \varnothing, \forall t_i \in [t_0,...,t_k]. \\ \overline{P^*_{\omega_z}(t_i)} \neq \varnothing, \forall t_i \in [t_0,...,t_k] \end{cases} \quad (5.11)$$

Співвідношення (5.11) описує умову безперервності простору гарантованого управління ДО в режимі стабілізації переміщення ОУ.

За імітаційним експериментом, який було проведено на моделі просторового переміщення сучасного ЛАП, було отримано результати, які наведені на рис. 5.7. В якості опорного переміщення ЛАП при моделюванні було обрано прямолінійне переміщення в горизонтальній площині (горизонтальний політ). В якості вихідних значень розрахованих параметрів просторового переміщення ЛАП було обрано кутову швидкість тангажу ω_x та кут атаки α, а параметром управління обрано відхилення $\delta_в$ керма висоти.

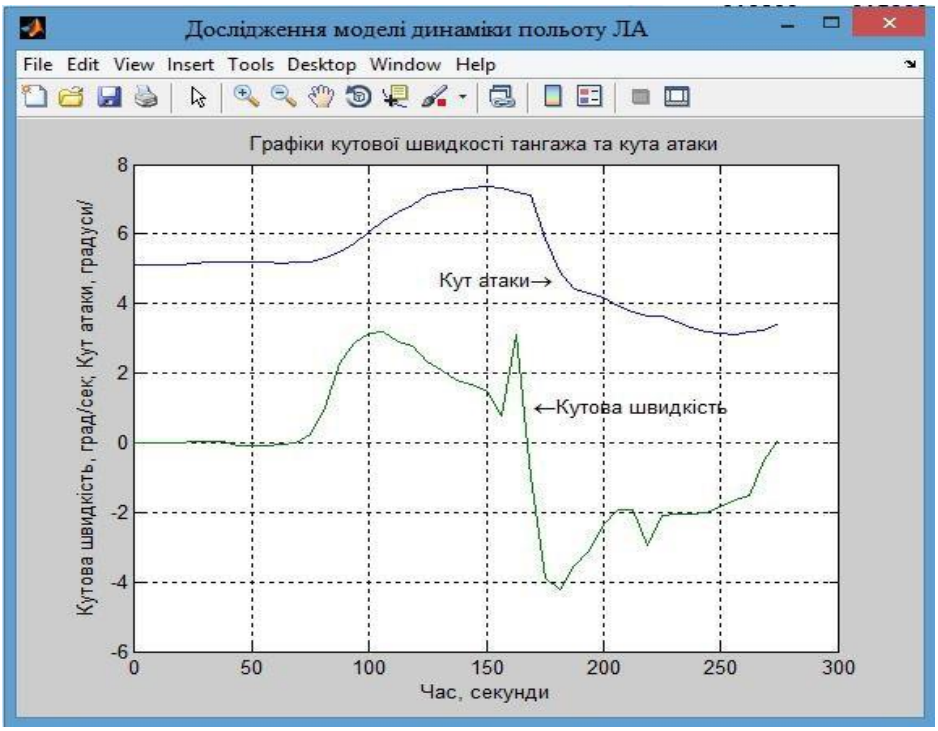


Рис. 5.7. Зміни в часі кутової швидкості тангажу ω_x та кута атаки α

За обчисленими даними, які представлено на рис. 5.7, було отримано фазову траєкторію в перерізі простору керованих станів для параметрів ω_x та α (рис. 5.8).

Отримані в результаті імітаційного експерименту дані визначають можливість здійснення управління в СУ ДО в просторі ПКУО.

Множина фазових траєкторій може бути отримана в процесі здійснення перебору в границях обмежень на значення параметрів системи, що визначено співвідношенням (5.2) та можливих керуючих впливів на ДО.

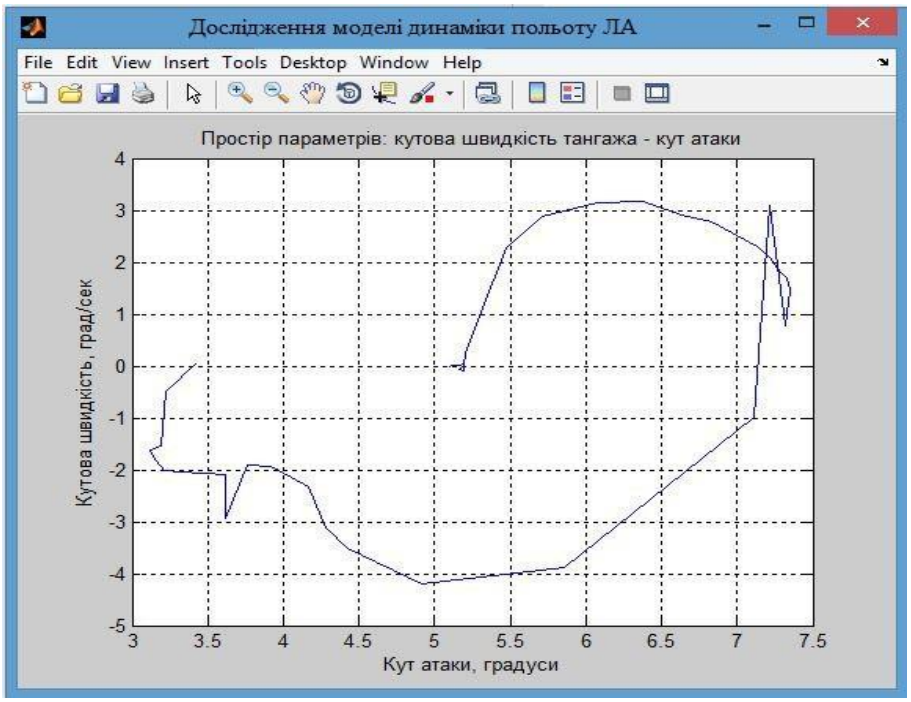


Рис. 5.8. Фазова траєкторія ω_x та α в перерізі простору керованих станів ЛАП

За результатами дослідження функціонування ДС та ДО з застосуванням математичної моделі визначено параметри, які при певних припущеннях визначають співвідношення між ними і описують простір керованих станів в фізичному просторі переміщення ОУ. При цьому визначено спосіб однозначного взаємного відображення параметрів, які характеризують переміщення ОУ в фізичному просторі та параметрів, які визначають поведінку ДО в математичному просторі.

Запропонований підхід дозволяє визначати припустимі граничні відхилення органів управління ДО на певних етапах переміщення ОУ в просторі спостереження.

Можливість визначення простору гарантованого управління ДО дозволяє вирішувати задачу конфлікту взаємодії ОУ з відкритою множиною об’єктів спостереження в умовах обмежень для моделі (2.10) – (2.12).

В якості прикладу для істотно-нелінійного ДО здійснено розрахунок та формальне відображення інформаційної множини динамічного образу керованих та напівкерованих станів ОУ на прикладі характеристик сучасного ЛАП.

5.1.4. Імітаційна модель збурень зовнішнього середовища динамічного об’єкту управління

Для опису та моделювання збурень зовнішнього середовища і генерації сигналів збурення розробляються додаткові моделі.

Математична модель відображає процес впливу об’єкту моделювання (зовнішнього середовища) на ОУ. Залежно від мети ідентифікації і початкової (апріорної) інформації про об'єкт моделювання та умови його функціонування застосовуються різні за формою і структурою стохастичні, статистичні і детерміновані математичні моделі, виходячи з яких створюється імітаційна модель. Адекватність моделі обов'язково оцінюється шляхом проведення імітаційного експерименту.

Моделювання турбулентності приземного шару атмосфери у вигляді тримірного анізотропного поля швидкостей є актуальним щодо опрацювання принципів управління ДО, яким є ЛАП, в ручному та директорному режимі управління при рішенні конфлікту взаємодії з іншими рухомими об’єктами в умовах обмежень та невизначеностей.

При обробці результатів метеорологічних спостережень і моделюванні руху ЛАП в приземному шарі атмосфери прийнято підрозділяти швидкість вітру на середній вітер, що має детермінований закон зміни в просторі, і атмосферну турбулентність, яка за сутністю є хаотичними вихровими рухами атмосфери.

Методи, які використовуються для моделювання атмосферної турбулентності, залежать від особливостей застосування результатів. Вони визначаються режимом польоту ЛАП, його конструкцією і геометричними

розмірами. Якщо геометричні розміри літального апарату малі в порівнянні з характерним розміром турбулентних вихорів, то можна вважати, що в усіх точках його поверхні швидкості поривів вітри однакові.

Якщо при цьому траєкторія польоту ЛАП близька до прямолінійної, задача моделювання атмосферної турбулентності може бути зведена до моделювання одновимірних випадкових процесів, для опису яких використовується їх спектральна щільність.

Якщо геометричні розміри ЛАП сумірні з масштабом турбулентності і/або траєкторія його польоту істотно відрізняється від прямолінійного польоту, з'являється необхідність моделювання атмосферної турбулентності у вигляді тривимірного випадкового поля. Це дозволяє врахувати додаткові прирости аеродинамічних сил і моментів, що виникають за рахунок нерівномірного розподілу поривів вітру по поверхні ЛАП, яким може бути безпілотний літальний апарат (БПС).

Зазвичай при аналізі результатів метеорологічних спостережень використовується земна система координат, яка орієнтована по напряму середнього вітру. Таким чином, вісь координат спрямована по напряму середнього вітру. Напрям середнього вітру в нижньому шарі висот вважається незмінним, що характерно для нейтрального стану атмосфери, який зустрічається в більшості метеорологічних спостережень.

Турбулентність за функцією щільності розподілу швидкості поривів відрізняється від нормального закону розподілу. В подальшому будемо виходити з припущення про горизонтальну однорідність і стаціонарність поля турбулентності [161]. Такі припущення виправдані для широкого класу задач. Відомо, що постійна часу зміни статистичних характеристик турбулентності у фіксованій точці простору складає декілька годин. Статистичні характеристики можуть змінюватися в результаті добового ходу температури, проходження атмосферних фронтів і т. п. Тому для малих інтервалів часу статистичні характеристики атмосферної турбулентності залишаються незмінними.

Атмосферна турбулентність має вплив на зміну траєкторії просторового руху ЛАП. З метою дослідження впливу атмосферної турбулентності на просторовий рух ДО створена імітаційна модель атмосферної турбулентності. В подальшому імітаційна модель може використовуватись в СІУ ЛАП або ергатичній системі управління (ЕСУ) ЛАП в автоматичному і автоматизованому режимах польоту та в тренажерах.

При створенні моделі атмосферної турбулентності в якості похідних даних були використані результати метеорологічних спостережень атмосферної турбулентності в районі здійснення польотів ЛАП.

Після опрацювання даних метеорологічних спостережень була отримана ідентифікаційна модель в класі поліномів шостого ступеня в канонічному вигляді [124] для обчислення значень математичного очікування

$$M(H) = (0{,}488 + 2{,}83 \times H - 0{,}8 \times H^2 + 0{,}989 \times H^3 + 0{,}00239 \times H^4 - 0{,}0000596 \times H^5 - 0{,}00000598 \times H^6) \times 0{,}00003 + k, \quad (5.12)$$

та дисперсії

$$D(H) = (17{,}9 + 6{,}65 \times H - 3{,}06 \times H^2 + 0{,}393 \times H^3 + 0{,}0196 \times H^4 - 0{,}000492 \times H^5 - 0{,}0000243 \times H^6) \times 0{,}000025 + m, \quad (5.13)$$

абсолютних значень поривів вітру.

Коефіцієнти k в співвідношенні (5.12) та m в співвідношенні (5.13) визначаються за отриманими даними метеорологічних спостережень атмосферної турбулентності в конкретному місті. Для співвідношень (5.12) та (5.13) визначені k=27, m=12.

Слід зазначити, що кількість членів поліномів (5.12) і (5.13) визначається вимогами точності до функції найкращого наближення за умовами Лагранжа і в даному випадку не перевищують 5%..

За ідентифікаційними моделями, які наведено в співвідношеннях (5.12) та (5.13) розроблено програмні засоби імітаційної моделі.

Результати імітаційного моделювання математичного очікування (рис. 5.9) та дисперсії (рис.5.10) за співвідношеннями (5.12) та (5.13) показали їх

відповідність ідентифікаційних моделей похідним даним метеорологічних спостережень.

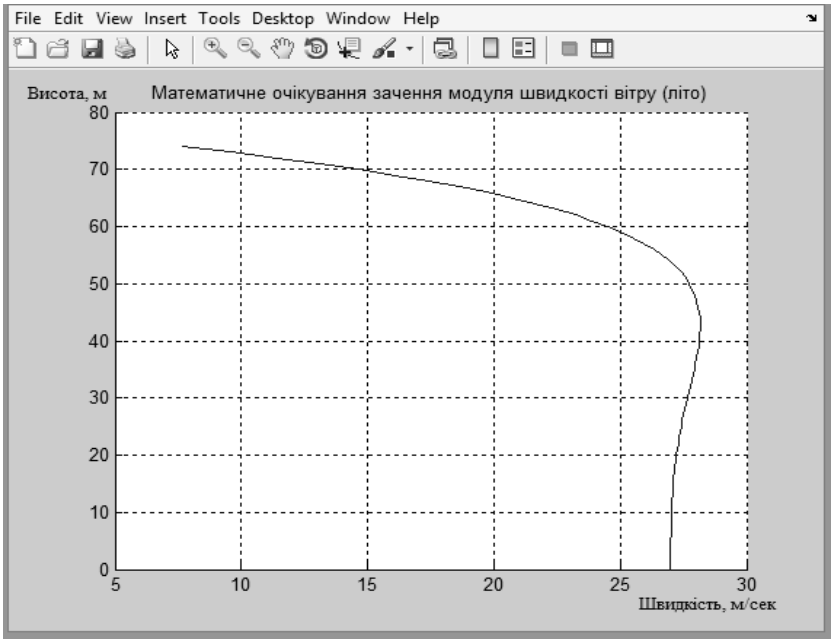


Рис. 5.9. Результати моделювання математичного спостереження

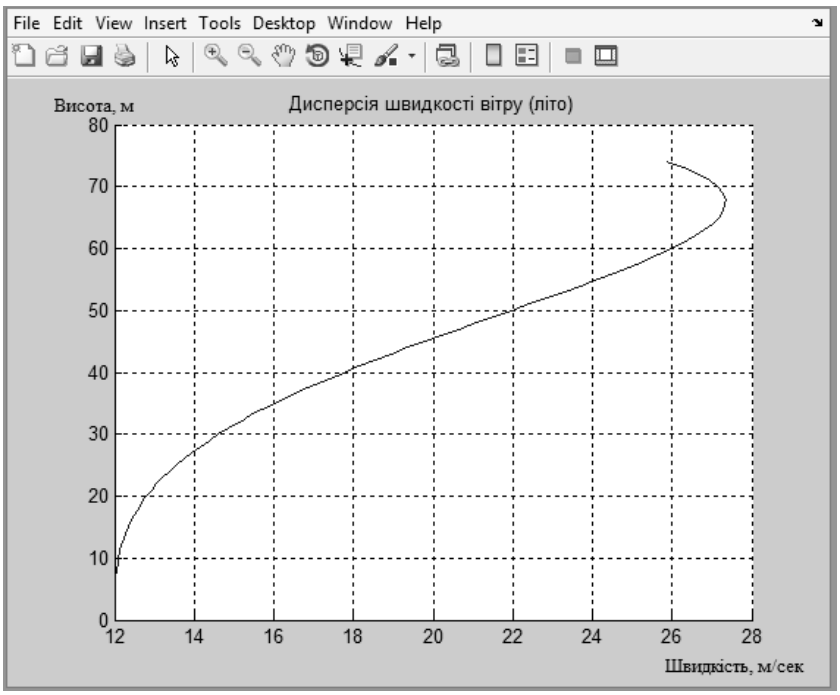


Рис. 5.10. Результати моделювання дисперсії

При визначенні проекцій зустрічної та бокової складової вектору швидкості вітру V_x були використані припущення про те, що значення

математичного очікування зустрічної складової швидкості вітру дорівнює значенню математичного очікування за результатами метеорологічних спостережень, а значення знаходяться в діапазоні

$$\left|\vec{V}_x\right| \in \left[M(H)-\Delta, M(H)+\Delta\right]. \qquad (5.14)$$

Величина Δ в співвідношенні (5.14) визначається на підставі теореми про те, що відхилення випадкової величини x, яка розподілена за нормальним законом, від математичного очікування $M(x)$ не буде більшим за значенням, ніж модуль величини $\Delta > 0$.

Визначимо функцію для інтегральної формули Муавра-Лапласа

$$\Phi\left(\frac{\Delta}{2}\right) = P\left(\left|x - M(x)\right|\right) \le \Delta, \qquad (5.15)$$

де $P\left(\left|x - M(x)\right|\right)$ - значення вірогідності, яке задається; $\Phi\left(\frac{\Delta}{2}\right)$- функція для інтегральної формули Муавра-Лапласа

$$\Phi\left(\frac{\Delta}{2}\right) = \frac{2}{\sqrt{2\pi}} \int_0^{\frac{\pi}{2}} e^{-\frac{t^2}{2}} dt. \qquad (5.16)$$

Для обчислення значення Δ з співвідношень (5.15) та (5.16) за результатами апроксимації знайдено функцію, яка є оберненою до функції $\Phi\left(\frac{\Delta}{2}\right)$,

$$\Delta = -0{,}21 + 9{,}78 \times y - 53 \times y^2 + 103 \times y^3 - 50{,}1 \times y^4 + 1 + 1{,}75 \times y^5 - 68{,}9 \times y^6 + 60{,}9 \times y^7, \qquad (5.17)$$

де y - випадкова величина середнього значення швидкості вітру.

Співвідношення (5.17) дозволяє обчислити діапазон значень $\left|\vec{V}_x\right|$ та $\left|\vec{V}_z\right|$ згідно співвідношення (5.14). Слід зазначити, що значення $\left|\vec{V}_x\right|$, яке визначене співвідношенням (5.14), відповідає рівномірному закону розподілу. Знак

проекції $\left|\vec{V}_x\right|$ визначається випадковим чином за рівномірним законом розподілу.

Бокова складова вектору швидкості вітру визначається як

$$\left|\vec{V}_z\right| = \left|\vec{V}_x\right| \times Sin\alpha \ , \tag{5.18}$$

де V_z - величина бокової складової швидкості вітру, V_x - величина зустрічної складової швидкості вітру,

$$\alpha = \overset{\wedge}{\vec{V}_x \vec{V}_z}. \tag{5.19}$$

Величина кута $\alpha \in [0,2\pi)$ і визначається для співвідношення (5.19) за нормальним законом розподілення.

Значення модуля проекції швидкості вітру V_z обчислюється згідно співвідношенню

$$\left|\vec{V}_y\right| = \sqrt{\left(M(H)\right)^2 - \left(\left|\vec{V}_x\right|\right)^2 - \left(\left|\vec{V}_z\right|\right)^2} \ , \tag{5.20}$$

а знак визначається за рівномірним законом розподілу.

Вертикальна проекція вектору швидкості вітру V_y має трапецієвидну форму, для якої визначається крутизна зростання фронту пориву витру, а знак визначається за рівномірним законом розподілу.

Виходячи з співвідношень (5.12) - (5.20), було розроблено імітаційну модель, результати роботи якої наведено на рис. 5.11.

Порівняння результатів імітаційного моделювання атмосферної турбулентності в діапазоні висот до 75 метрів з експериментальними даними показало відповідність результатів розрахунків даним метеорологічних спостережень.

Імітаційна модель дозволяє досліджувати властивості СУ та ЕСУ ЛАП при просторовому переміщенні ДО в умовах впливу атмосферних збурень.

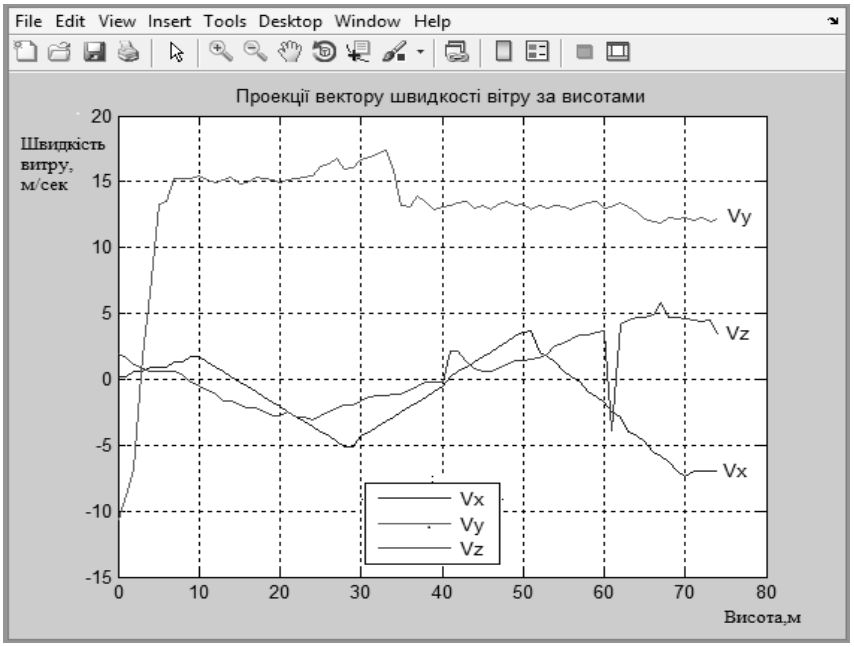


Рис. 5.11. Результати моделювання атмосферної турбулентності

Ідентифікаційна модель (5.12) і (5.13) визначається характеристиками спостережень за атмосферною турбулентністю в районі переміщення ДО. Сучасні геоінформаційні системи дозволяють побудувати динамічну модель простору рішень задачі конфлікту з врахуванням атмосферних збурень при переміщенні ЛАП у відповідності з географічними координатами ДО.

5.2. Імітаційні моделі вирішення задачі конфлікту взаємодії об'єктів

5.2.1. Опис та аналіз стану взаємодії об'єкту управління з середовищем простору спостереження

Виходячи з структурної схеми СІУ ОУ (рис. 1.7), синтез стратегій управління при вирішенні конфлікту взаємодії ОУ та ОС в просторі Q вміщує декілька етапів, а саме: синтез моделі опису простору $G^*_{ріш}$, опис стану взаємодії ОУ з ОС.

Для гіпотетичного прикладу розрахунку ланцюжків рішення задачі конфлікту взаємодії ОУ та ОС в неопуклій системі обмежень при переміщенні

ОУ в цільову точку простору Q, яка належить множині кінцевих термінальних позицій ОУ, що визначено граматикою загального виду $\Gamma_{пр}$ (2.48), визначимо показники спостереження за ОС та припустимі значення зближення ОУ з i-м ОС d^i згідно (2.15) – (2.17) та (2.27). Дані для розрахунку наведені в табл. 5.1.

В гіпотетичному прикладі визначаємо систему неопуклих обмежень простору Q (табл. 5.2). Замкнення простору $G_{ріш}$ здійснено за правилом замикання $\Gamma_{пр}$ (3.42) та мові семіотичної системи $L(\Gamma_{пр})$, що дозволяє визначити границю $\Gamma_{гр}(Q)$ простору $G_{обм}$. В загальному випадку отримаємо переріз простору $G_{обм}$.

Цільова точка P переміщення ОУ для гіпотетичного прикладу має координати $P(13.0, -6.0)$.

Таблиця 5.1

Об'єкти спостереження

Номер об'єкта	Координати	Вектор істинної швидкості	Вектор відносної швидкості	Припустиме зближення
1.	(9.7, -5.6)	(-10.3, 4.8)	(-18.8, 13.3)	2.0
2.	(11.7, -3.7)	(-15.0, 0.1)	(-23.5, 8.6)	2.0
3.	(5.9, -5.9)	(-9.9, 9.9)	(-18.4, 18.4)	2.0

Таблиця 5.2

Обмеження простору спостереження та пошуку

Вершини лінії лівої границі обмежень		Вершини правої границі обмежень	
№ з/п	Координати	№ з/п	Координати
1.	(-14.0, -4.0)	1.	(-10.0, 12.0)
2.	(14.0, 10.0)	2.	(6.0, -2.0)
		3.	(14.0, 6.0)

Дані щодо характеристик переміщення ОУ в просторі $G_{обм}$ наведено в табл. 5.3.

Таблиця 5.3

Об'єкт управління

№ з/п	Показник	Значення
1.	Найбільша швидкість ОУ	40 одиниць
2.	Час запізнення на прийняття рішення	6 хвилин
3.	Радіус зони оцінки ситуації	16 одиниць
4.	Напрямок переміщення ОУ	315 градусів
5.	Швидкість ОУ	12 одиниць/одиниця часу
6.	Координати вектору власної швидкості	(8.5, -8.5)
7.	Початок сектору перегляду ситуації (ПКУО)	225 градусів
8.	Кінець сектору перегляду ситуації (ПКУО)	45 градусів
9.	Крок перегляду ПКУО	5 градусів

Похідні дані для гіпотетичного прикладу (табл. 5.1 - табл. 3.3) використані в імітаційній моделі і відображені на рис. 5.12.

Виходячи з співвідношень (2.48) - (2.50) для КВ-граматики (2.52) з врахуванням співвідношення (2.48), що визначає граматику $\Gamma_{пр}$, для множини нетермінальних символів (2.54), що входять в граматику $\Gamma_{пр}$, визначимо семіотичну модель опису конфлікту взаємодії ОУ з ОС за умов неопуклого простору $G_{обм}$ та невизначеності переміщення ОС в просторі Q.

Для гіпотетичного прикладу конфлікту взаємодії ОУ та ОС визначимо опис поточної ситуації, який має вигляд:

"if ([A:(1);B(9.7,-5.6);H(2.0);D(-10.3,4.8)] and [A:(2);B(11.7,-3.7);H(2.0);D(-15.0,0.1)] and

[A:(3);B(5.9,-5.9);H(2.0);D(-9.9,9.9)]) else [A:(0);D(8.5,-8.5)] then [B:(13.0,-6.0)] &".

Наведений опис ситуації формується в автоматичному режимі за похідними даними, які надходять з модуля приймання та перетворення похідних даних згідно структурної схеми СІУ ОУ (рис. 1.7).

Синтезований опис поточної ситуації взаємодії ОУ з ОС в умовах обмежень та невизначеності згідно структурної схеми СІУ ОУ аналізується та

обробляється засобами модуля синтезу цілі та рішення, за допомогою яких виконується семіотичний аналіз синтезованого опису поточної ситуації згідно правилу формування лексем за формою Бекуса-Наура (2.50) та лексичних позначень даних про об'єкти згідно табл. 2.1.

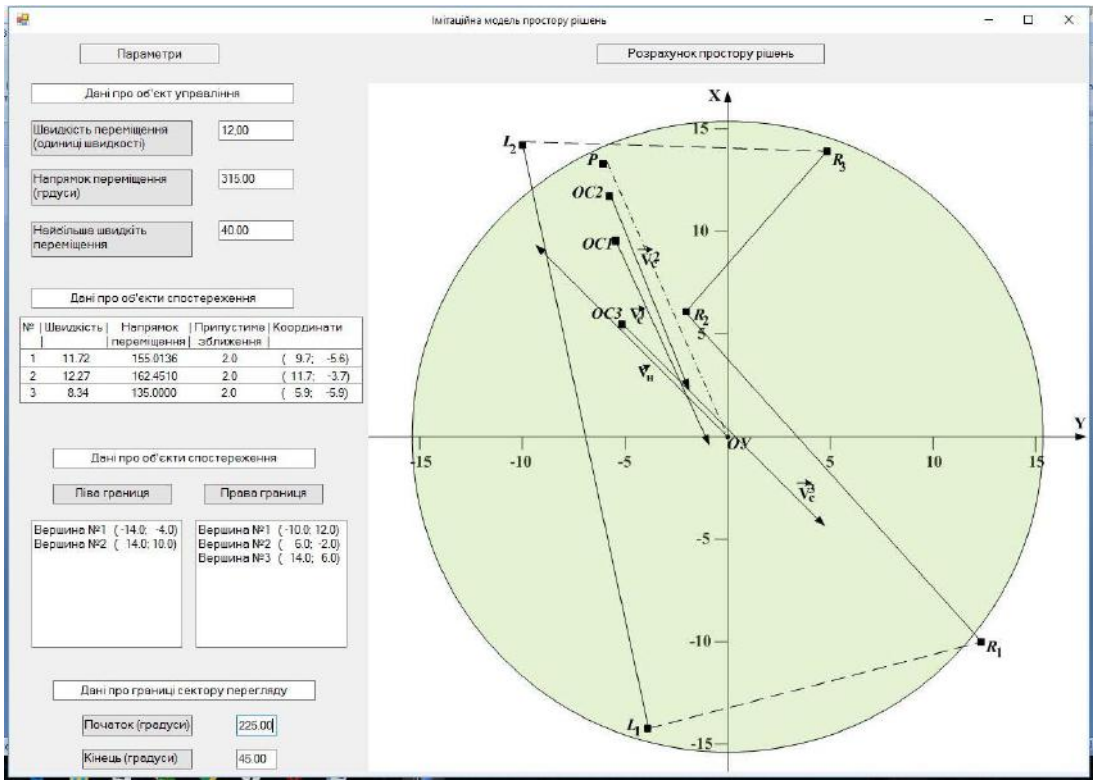


Рис. 5.12. Відображення простору рішень

Першим кроком при здійсненні процедури семіотичного аналізу є опрацювання опису ситуації за допомогою лексичного сканера (ЛС).

Вигляд екранної форми імітаційної моделі ЛС (рис. 5.13).

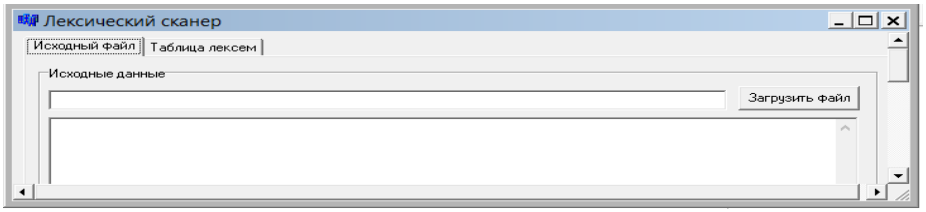


Рис. 5.13. Екранна форма імітаційної моделі лексичного сканера

Після введення похідних даних (табл. 5.1 – табл. 5.3) в СІУ генерується опис для опрацювання повідомлення щодо поточної ситуації (рис. 5.14).

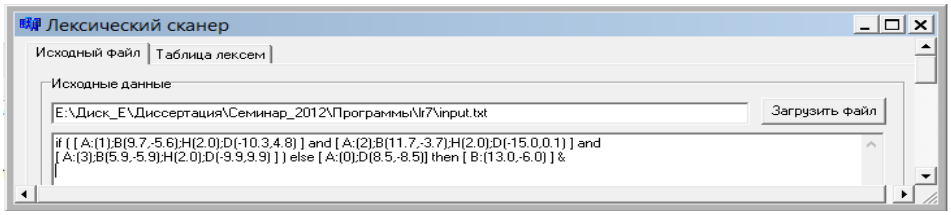


Рис. 5.14. Екранна форма лексичного сканера з похідними даними

За похідними даними опису повідомлення щодо поточної ситуації СІУ виконує ЛА повідомлення та створює відповідні таблиці лексем.

Результат ЛА похідних даних повідомлення щодо опису поточної ситуації наведено на рис. 5.15.

Після опрацювання похідних даних повідомлення щодо опису поточної ситуації ЛА формує значення лексем, змінних та виділяє ключові слова.

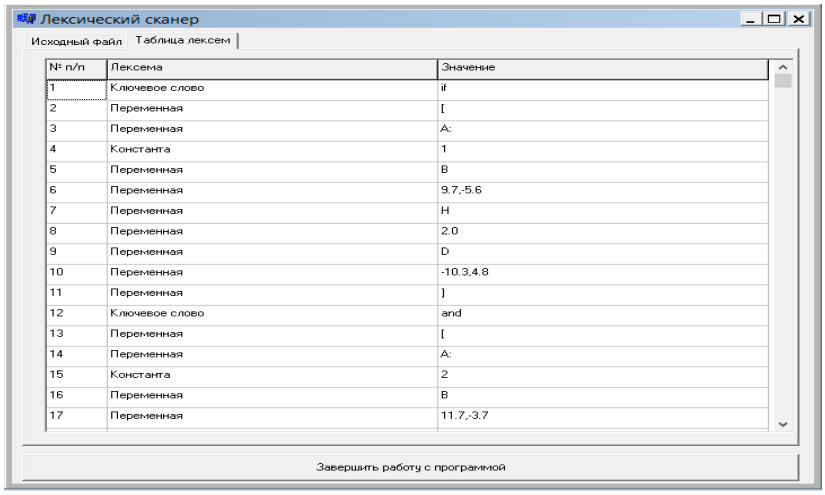


Рис. 5.15. Екранна форма опрацювання похідних даних лексичним сканером

Отримані за результатами функціонування ЛА дані вносяться до відповідних таблиць.

Екранна форма у вигляді консольного додатку імітаційної моделі формування таблиці ідентифікаторів з використанням попереднього хешування наведена на рис. 5.16.

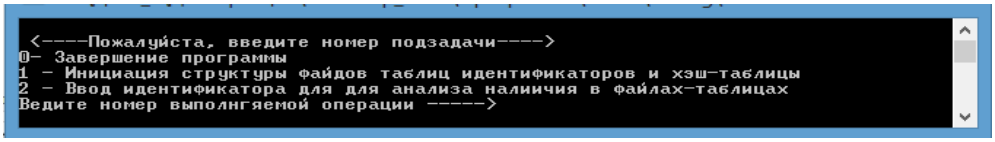


Рис. 5.16. Екранна форма формування таблиці ідентифікаторів

Після завантаження програмного додатку імітаційної моделі формування таблиці ідентифікаторів здійснюється первинна ініціація структури файлів хеш-таблиці (рис. 5.17).

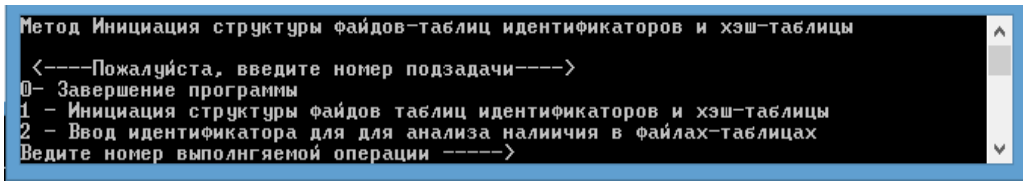


Рис. 5.17. Екранна форма результату ініціації таблиці ідентифікаторів

За результатами ініціації структури файлів хеш-таблиці здійснюється введення лексеми (рис.5.18).

Для введення ідентифікатора за алгоритмом обирається відповідна ознака (символ "2"). Для введення ідентифікатора з'являється екранна форма (рис. 5.18), яка передбачає введення ідентифікатора.

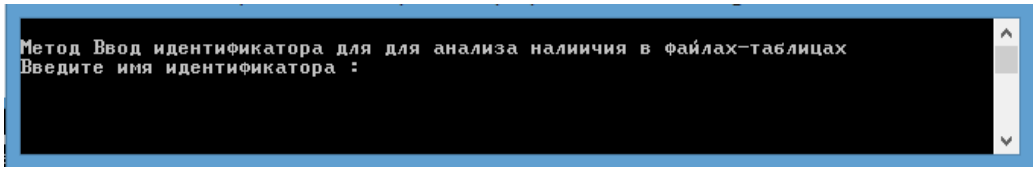


Рис. 5.18. Екранна форма запиту на введення лексеми

Після введення ідентифікатора, наприклад, "A:<0>" генерується екранна форма (рис. 5.19).

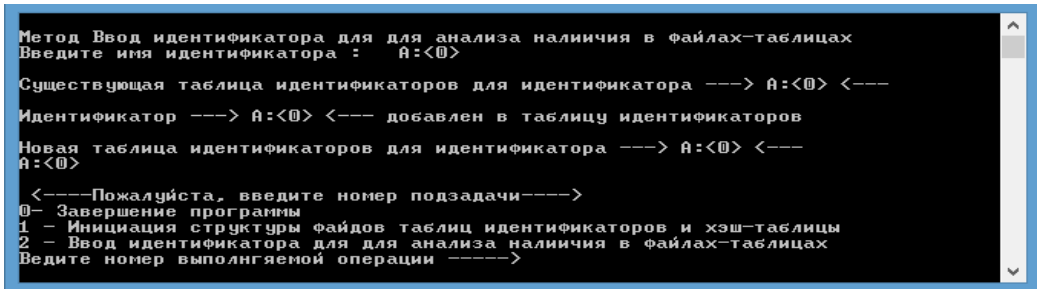


Рис. 5.19. Екранна форма результату на введення лексеми

В повідомленнях екранної форми наведено дані про існування відповідної до ідентифікатора хеш-таблиці, виведено повідомлення про виконання процедури додавання введеного ідентифікатора та виведено її вміст відповідної таблиці.

Процес додавання нових ідентифікаторів у відповідні таблиці наведено на рис. 5.20 та рис. 5.21. Введемо ідентифікатор "A:<1>".

Рис. 5.20. Екранна форма запиту на введення лексеми

Після введення відповідного ідентифікатора, наприклад, "A:<1>" для з'явиться екранна форма (рис. 5.21).

В повідомленнях екранної форми зазначено відомості про існування відповідної до ідентифікатора хеш-таблиці, зазначено про виконання процедури додавання введеного ідентифікатора та виведено її вміст.

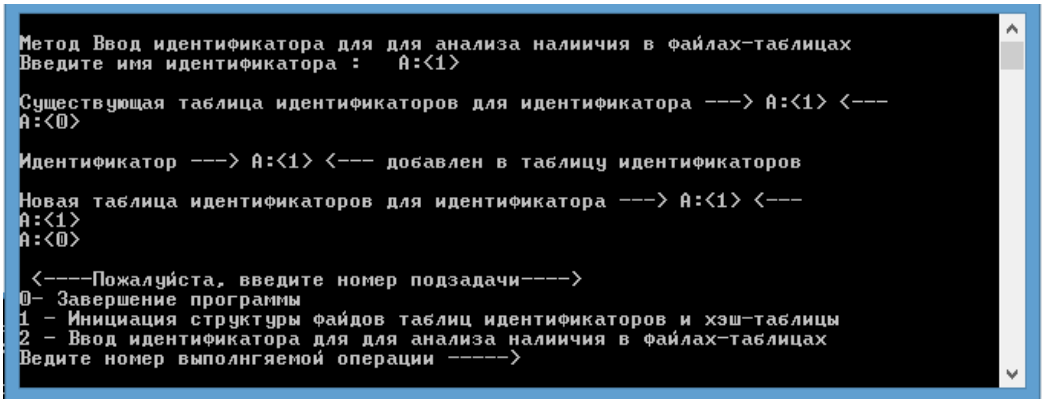


Рис. 5.21. Екранна форма результату на введення лексеми

За результатом введення ідентифікаторів буде створено хеш-таблицю у вигляді файлу, фрагмент якого наведено на рис. 5.22.

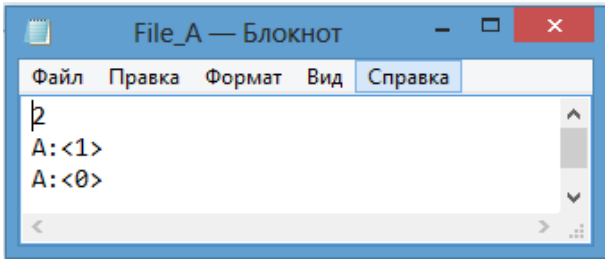


Рис. 5.22. Хеш-таблиця для ідентифікаторів, які починаються з літери "A"

Першим елементом хеш-таблиці є кількість ідентифікаторів у відповідній таблиці ідентифікаторів (рис 5.23).

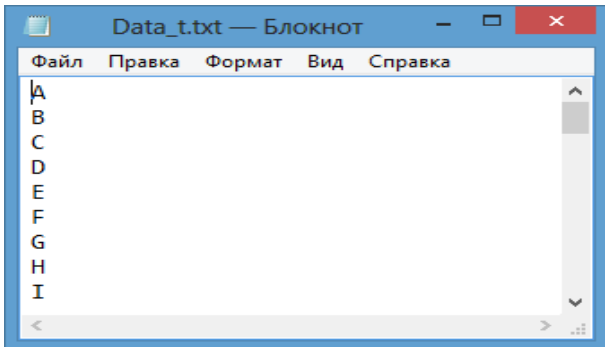


Рис. 5.23. Таблиця символів для формування хеш-таблиць ідентифікаторів

Значення перших символів ідентифікаторів створюваних хеш-таблиць визначається таблицею, яка орієнтована на множину символів похідної мови для формування хеш-таблиць ідентифікаторів. Перелік файлів хеш-таблиць ідентифікаторів наведено на рис. 5.24.

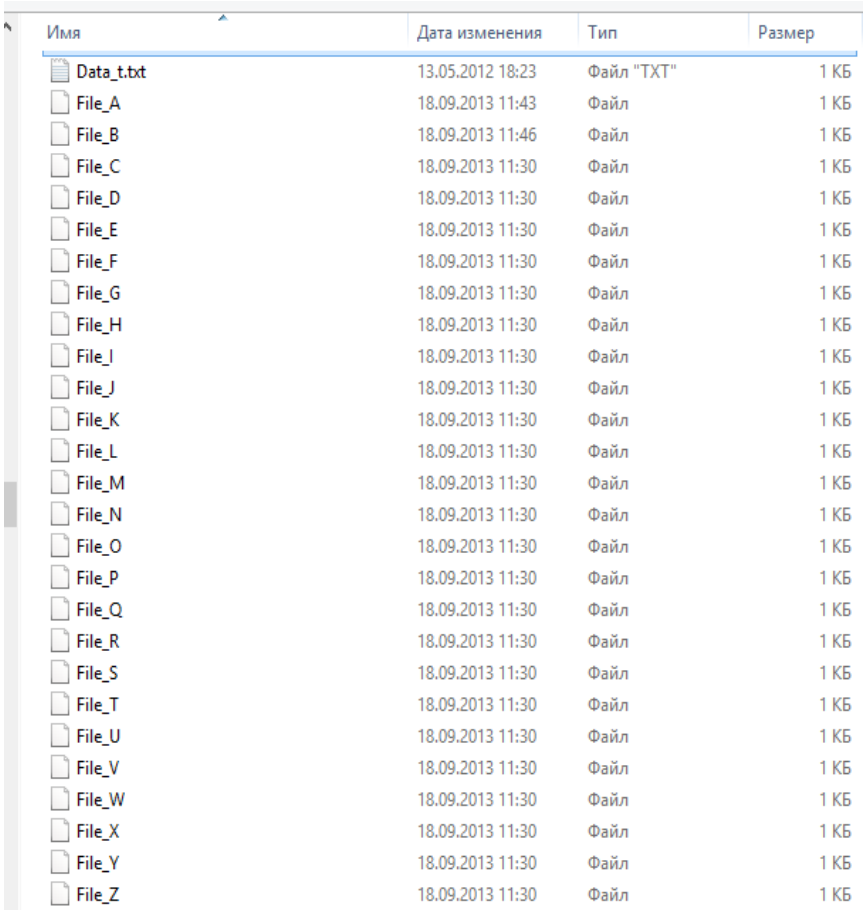


Рис. 5.24. Файли хеш-таблиць ідентифікаторів

Згідно даним хеш-таблиці ідентифікаторів (рис. 5.23) формуються відповідні хеш-таблиці ідентифікаторів з першою буквою, яка відповідає

значенню символу в певному рядку перших символів відповідних ідентифікаторів. Файл хеш-таблиці ідентифікатора має назву, за якою по ідентифікатору можна здійснити його бінарне значенням у представленні цілої константи без знаку.

Файли хеш-таблиці ідентифікаторів формуються згідно правилу формування лексем за формою Бекуса-Наура (2.50) та лексичних позначень даних про об'єкти згідно табл. 2.1 і розміщуються в пам'яті обчислювального середовища СІУ ОУ (рис. 1.7). Файли хеш-таблиць вміщують символьне значення ідентифікатора у вигляді строки запису. Таким чином значення ідентифікатора є бінарним кодом, який за методом хешування має сенс цілої константи без знаку.

Алгоритм функціонування ЛА наведено в Додатку Д.

Наступним етапом семіотичного аналізу похідного повідомлення про стан взаємодії ОУ та ОС в умовах обмежень та невизначеності є обробка похідного повідомлення синтаксичним сканером (рис. 5.25).

При здійсненні СА повідомлення, що сформовано згідно похідним даним, які наведені в табл. 5.1 – табл. 5.3, за результатом ЛА синтаксичний сканер генерує структуру повідомлення на внутрішній мові СІУ ОУ та відображення дерева розбору первинного повідомлення.

Алгоритм роботи СА наведено в Додатку Д.

За результатом лексичного аналізу синтаксичний сканер синтезує дерево опису ситуації взаємодії ОУ з ОС в умовах обмежень та невизначеності за формальною моделлю (2.10) – (2.12) на основі регулярної породжуючої граматики $\Gamma^{0}_{\text{пр}}$ у відповідності з співвідношенням (2.39). Тобто синтезоване дерево опису є похідним для розрахунку простору $G_{ріш}$ та $G^{*}_{ріш}$ з врахуванням співвідношень (2.39) – (2.41), (3.46) – (3.48) та (4.22).

Разом з тим дерево опису ситуації є похідним при породженні ланцюжків управління за співвідношенням (2.41).

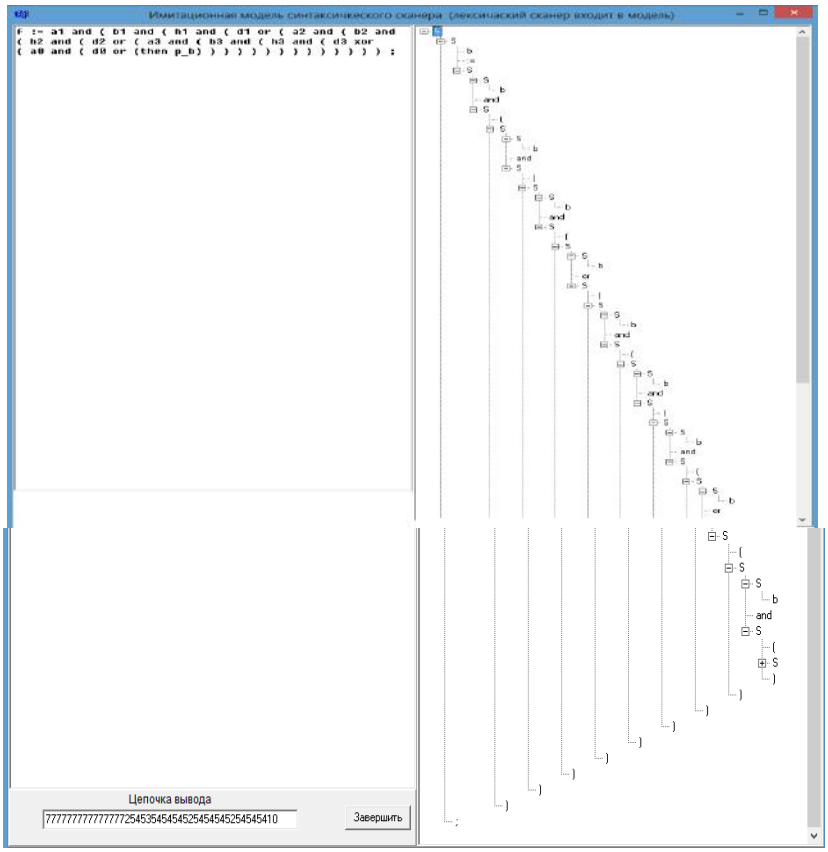


Рис. 5.25. Екранна форма результату синтаксичного аналізу

Таким чином, система (2.40) визначає приналежність характеристик ОУ множині припустимих при вирішенні для ОУ задачі конфлікту, спостереження та переслідування для всіх ОС в просторі $G^{*}_{piш}$.

5.2.2. Синтез варіантів припустимих рішень конфлікту за методом інтегрального усікання варіантів

При синтезі стратегій вирішення задачі конфлікту взаємодії ОУ з ОС в умовах обмежень та невизначеності згідно даним, які наведені в табл. 5.1 – табл. 5.3 за описом ОУ з ОС в умовах обмежень та невизначеностей за методом інтегрального усікання варіантів.

Виходячи з співвідношень (2.10) – (2.12), (2.30) - (2.32), (2.33) - (2.38), (4.22) та (4.24) - (4.53), за алгоритмом, який описано в розділі 4.2, було

досліджено результати функціонування імітаційної моделі рішення конфлікту взаємодії ОУ з ОС в умовах обмежень та невизначеностей.

При розрахунку стратегії управління ОУ при рішенні конфлікту для функціоналу ціни, що визначена співвідношеннями (4.5), (4.11), (4.13), (4.14) було обрано значення коефіцієнтів часткових критеріїв відбору

$$\begin{bmatrix} \overline{k_1} = 0.1 \\ \overline{k_2} = 0.2 \\ \overline{k_3} = 0.3 \\ \overline{k_4} = 0.4 \end{bmatrix}. \tag{5.21}$$

Значення зміни напрямку переміщення ОУ

$$\begin{bmatrix} |\Delta\psi_1| = 3^\circ \\ |\Delta\psi_2| = 3^\circ \end{bmatrix}. \tag{5.22}$$

При синтезі рішення визначимо значення часу затримки на прийняття рішення та виконання дій щодо управління ОУ $\tau = 0.1$ одиниці часу.

Визначимо максимальну кількість синтезованих варіантів рішень для кожного елементу траєкторії переміщення ОУ $(n = 15)$.

За похідними даними, які наведені в табл. 5.1 – табл. 5.3, та функцією ціни були обчислені значення параметрів управління ОУ щодо реалізації стратегій рішення задачі конфлікту взаємодії ОУ з ОС в умовах обмежень простору Q та невизначеності (рис. 5.26).

В результаті функціонування імітаційної моделі згідно функціоналу ціни синтезовано дві альтернативні траєкторії переміщення ОУ: OM_1M_3P при значенні функціоналу ціни $J = 0.244$, OM_1M_2P при значенні функціоналу ціни $J = 0.451$.

За похідними даними, які наведені в інтерфейсній формі імітаційної моделі, обчислено ПКУО для точки O (рис. 5.27) при початку переміщення за елементом траєкторії (OM_1). При переміщенні по елементу траєкторії (OM_1) обчислено ПКУО для точки M_1 в уявленні збереження напрямку переміщення на точку P (рис. 5.28).

При зміні напрямку переміщення в точці M_1 на точку M_3 обчислений ПКУО показує відсутність конфлікту взаємодії ОУ з ОС в умовах обмежень та невизначеності (рис. 5.29).

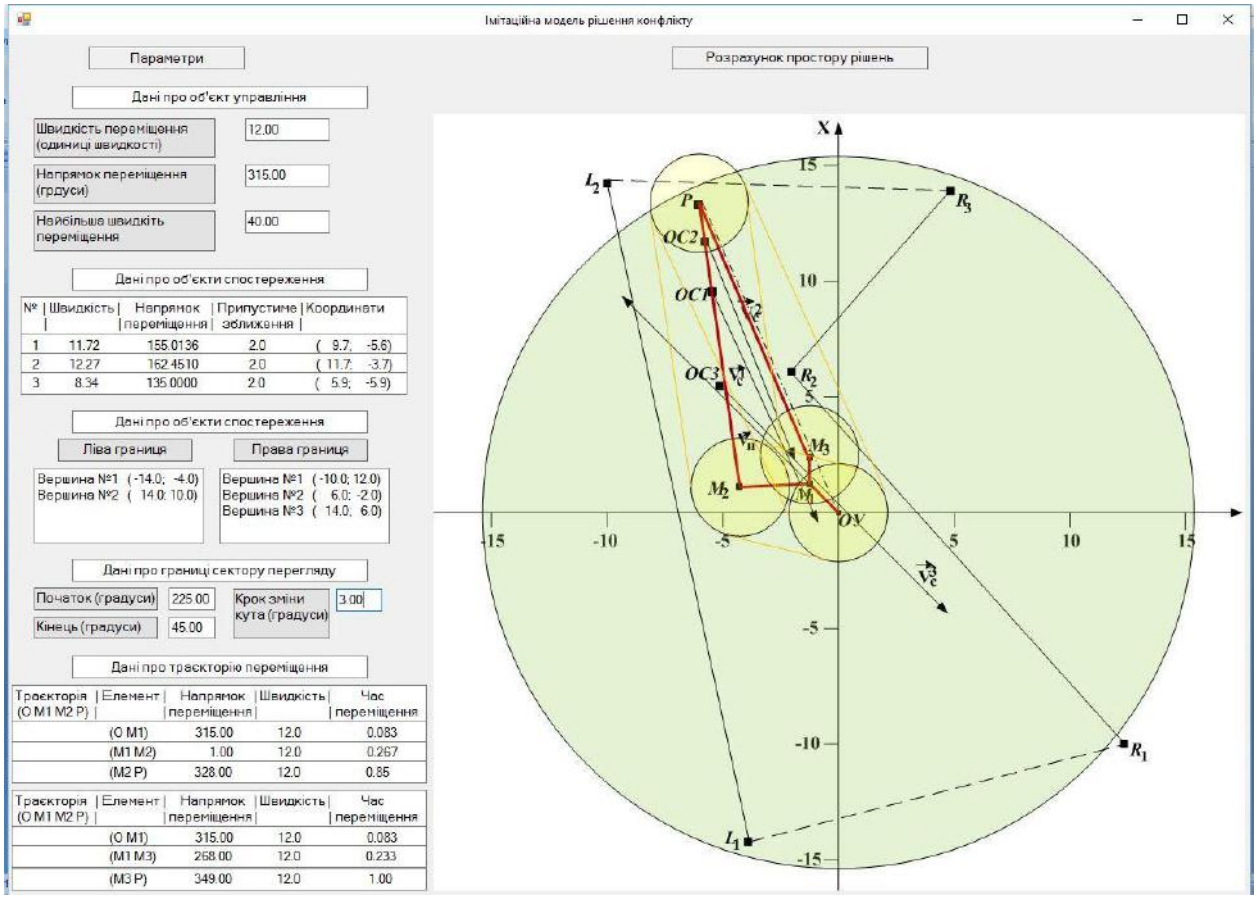


Рис. 5.26. Інтерфейс імітаційної моделі рішення конфлікту взаємодії ОУ з ОС

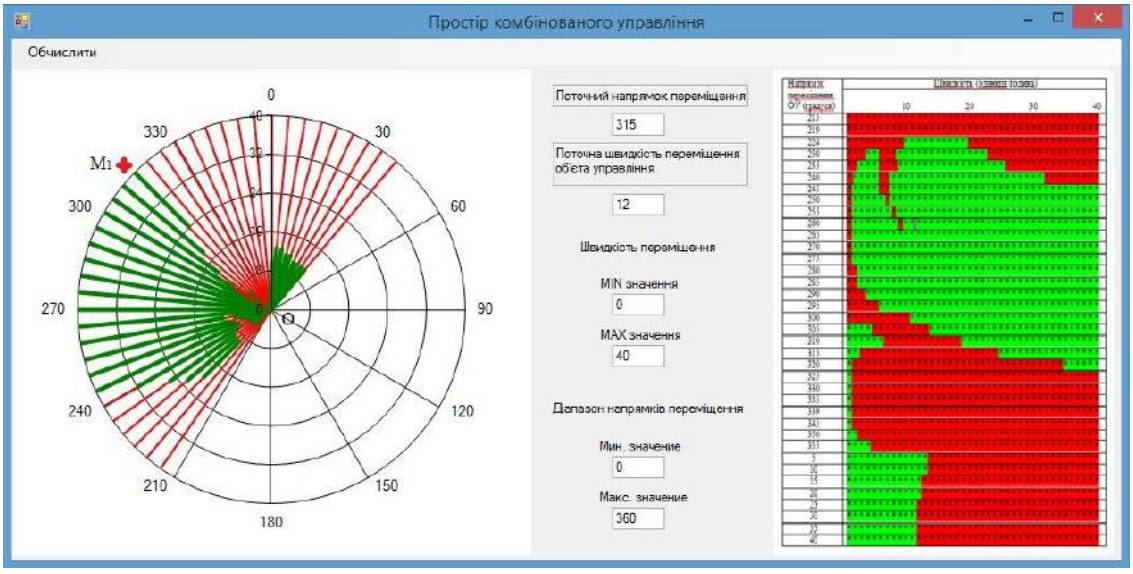


Рис. 5.27. Відображення результату розрахунку ПКУО в точці O

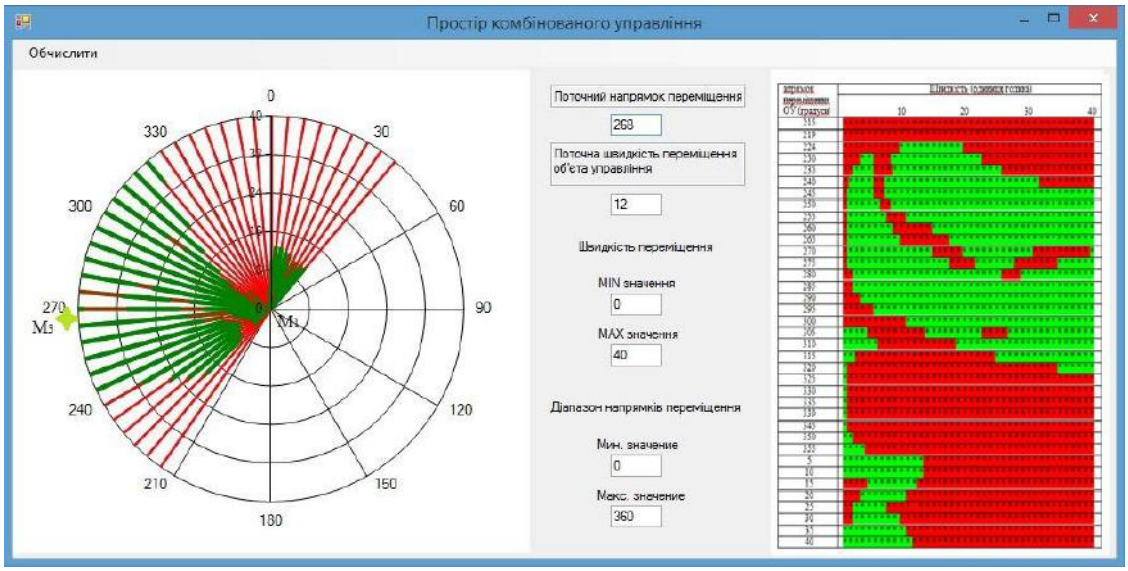


Рис. 5.28. Відображення результату розрахунку ПКУО в точці M_1

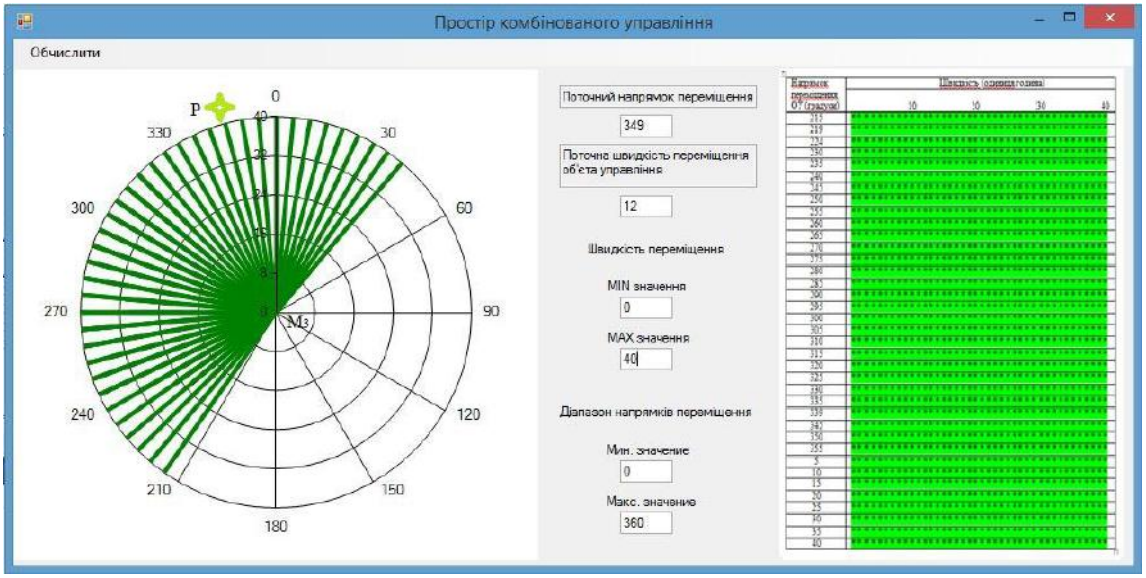


Рис. 5.29. Відображення результату розрахунку ПКУО в точці M_3

При переміщенні по елементу траєкторії (M_1M_3) обчислено ПКУО для точки M_3 в уявленні напрямку переміщення на точку P.

На рис. 5.27 – рис. 5.29 при відображенні ПКУО зеленим кольором позначені показники швидкості та напрямку переміщення ОУ, при яких немає зближення з i-м ОС на відстань меншу d^i. Значення швидкості та напрямку переміщення, при яких ОУ має зблизитись з i-м ОС на відстань меншу d^i, позначені червоним кольором.

Відображення ПКУО при синтезі ланцюжка оптимальної стратегії управління ОУ дозволяє зробити висновок про те, що стратегія забезпечує безконфліктну взаємодію ОУ з ОС в просторі Q за умов обмежень та невизначеностей.

Параметри переміщення і управління ОУ для траєкторій OM_1M_3P та OM_1M_2P відображені на інтерфейсі імітаційної моделі (рис. 5.26) .

Синтезована траєкторія переміщення ОУ OM_1M_3P не є оптимальною за значенням функціоналу ціни, але є непровокуючою ("обережною").

Синтезована траєкторія переміщення ОУ OM_1M_2P є оптимальною за значенням функціоналу ціни.

Коефіцієнти функціоналу ціни (5.21) визначають вимоги до синтезованого ланцюжка управління ОУ при вирішенні задачі конфлікту взаємодії з ОС в умовах обмежень та невизначеності, що дозволяє обирати варіант управління, який забезпечує сталість швидкості переміщення ОУ в просторі Q.

Обчислені значення функціоналу ціни визначено співвідношеннями (4.11), (4.13) та значенням коефіцієнтів часткових критеріїв відбору (5.21) і (5.22). При синтезі ланцюжка рішень щодо управління ОУ при рішенні конфлікту згідно похідним даним табл. 5.1 – табл. 5.3 функціонали ціни приймають значення, які наведені в табл. 5.4 для методу інтегрального усікання варіантів і табл. 5.5 для методу Брауна-Робінсона. Зазначені результати відображені на та гістограмі (рис. 5.30).

Таблиця 5.4

Значення функціоналу ціни для методу інтегрального усікання варіантів

Ітерація	4	21	23	34	37	52	60	63	80	110	114	162	180	200	220	240
Функція ціни	0,31	0,35	0,4	0,35	0,41	0,38	0,62	0,64	0,432	0,49	0,518	0,47	0,244	0,244	0,244	0,244

Таблиця 5.5

Значення функціоналу ціни для методу Брауна-Робінсона

Ітерація	4	21	23	34	37	52	60	63	80	110	114	162	180	200	220	240
Функція ціни	0,31	0,35	0,4	0,35	0,41	0,38	0,62	0,64	0,432	0,49	0,518	0,47	0,244	0,244	0,244	0,244

При рішенні конфлікту, синтез стратегій і ланцюжків управління за методом інтегрального усікання варіантів здійснювався при обмеженні глибини пошуку рішення $n=15$.

За похідними даними було передбачено здійснення 900 ітерацій пошуку при використанні повного перебору варіантів. За даними табл. 5.4 для синтезу оптимального ланцюжка рішення знадобилось 180 ітерацій, що дає значний виграш в часі синтезу та вибору рішення та вказує на коректність застосування правила зупинки $\Gamma_{зуп}$, яке визначено співвідношенням (4.15).

Для порівняння на рис. 5.30 наведено результати обчислення значення функціоналу ціни при рішенні задачі конфлікту за методом Брауна-Робертсона і за методом інтегрального усікання варіантів.

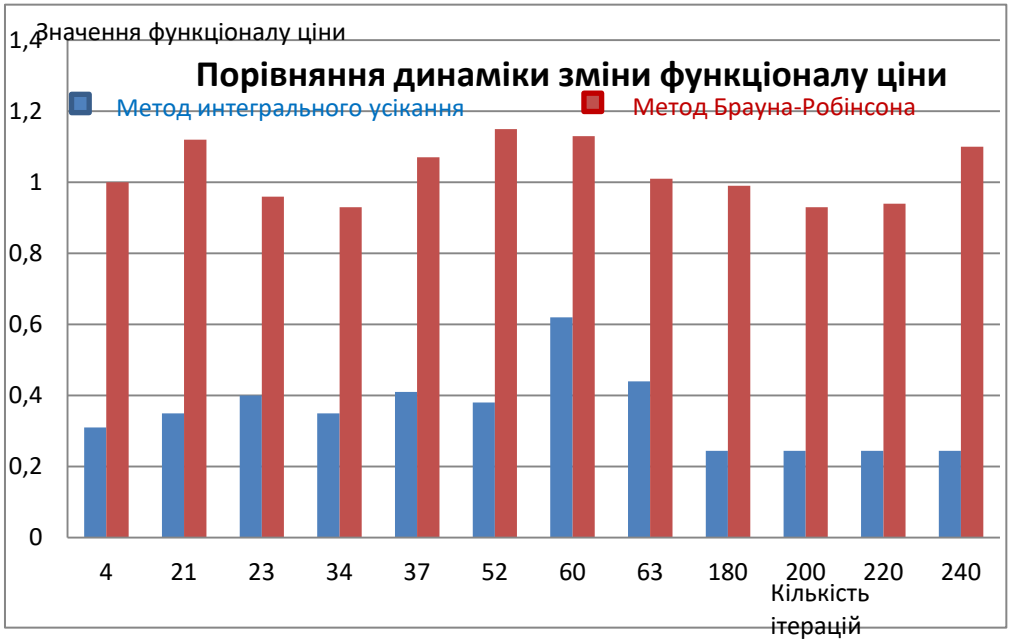


Рис. 5.30. Гістограма зміни значень функціоналу ціни

Для похідних даних, які визначені в табл. 5.1 – табл. 5.3, метод фіктивного розіграшу Брауна-Робертсона, що відноситься до ігрових методів дослідження операцій, в порівнянні з методом інтегрального усікання варіантів має повільну немонотонну збіжність при однаковій адитивній формі функції ціни (критерію вибору) відповідно співвідношенням (2.33) та (4.5).

За результатами імітаційного експерименту відповідно моделі взаємодії об'єктів в просторі Q (2.10) - (2.12) за граматикою загального виду (2.48) згідно співвідношення (4.5) синтезовано ланцюжок управління ОУ.

Після перетворення інформації згідно співвідношенням (2.43) – (2.45) синтезовані ланцюжки управління R_j^0 за співвідношенням (2.46) є результатом функціонування детермінованого кінцевого автомату, який визначений співвідношенням (2.41).

Тоді для алфавіту S^0 згідно співвідношення (2.59) синтезовано автоматне відображення $S^i\left(Б^0, Б^i, F^0, F^i, t\right)$, яке задовольняє умовам повноти для ОУ.

Тобто на рис. 5.28 наведені варіанти рішення, як підмножини множини ψ^0 просторового автоматного відображення S^i в множину Q згідно співвідношенню (2.59).

Таким чином, синтезовані ланцюжки рішень побудовані тільки в просторі гарантованого існування управлінь, які можуть бути реалізовані ОУ при вирішенні задачі ціле досяжності $C^0 \subset Q$ згідно співвідношення (2.58), та можуть бути здійснені в просторі S^0, що визначений співвідношенням (2.59), який фактично є простором $G_{ріш}$.

5.2.3. Евристичний пошук рішення конфлікту

Виходячи з співвідношень (4.9) - (4.11) та принципу оптимальності χ за співвідношеннями (2.34 – (2.36)), (4.16) правила вибору стратегії рішення для конфлікту (4.11), (4.13), (4.14), яке визначено співвідношенням (4.32), з

врахуванням формального опису простору $G^*_{ріш}$ згідно співвідношенням (3.46), (3.47), можна використати алгоритм евристичного пошуку для синтезу стратегій рішення (рис. 5.31).

При застосуванні методу евристичного пошуку при синтезі рішення в просторі $G^*_{ріш}$ на зазначений простір накладається координатна сітка, що дозволяє побудувати граф припустимих переходів (траєкторій) переміщення ОУ в просторі Q.

Метод евристичного пошуку є найбільш прийнятним при синтезі стратегій рішення та ланцюжка управління в зоні надзвичайного зближення ОУ з ОС за відсутності дії правил P^0 для граматики $\Gamma^0_{пр}$ відповідно співвідношенню (2.39) та співвідношень (4.41) – (4.44), які визначають модель рішення конфлікту взаємодії об’єктів КП .

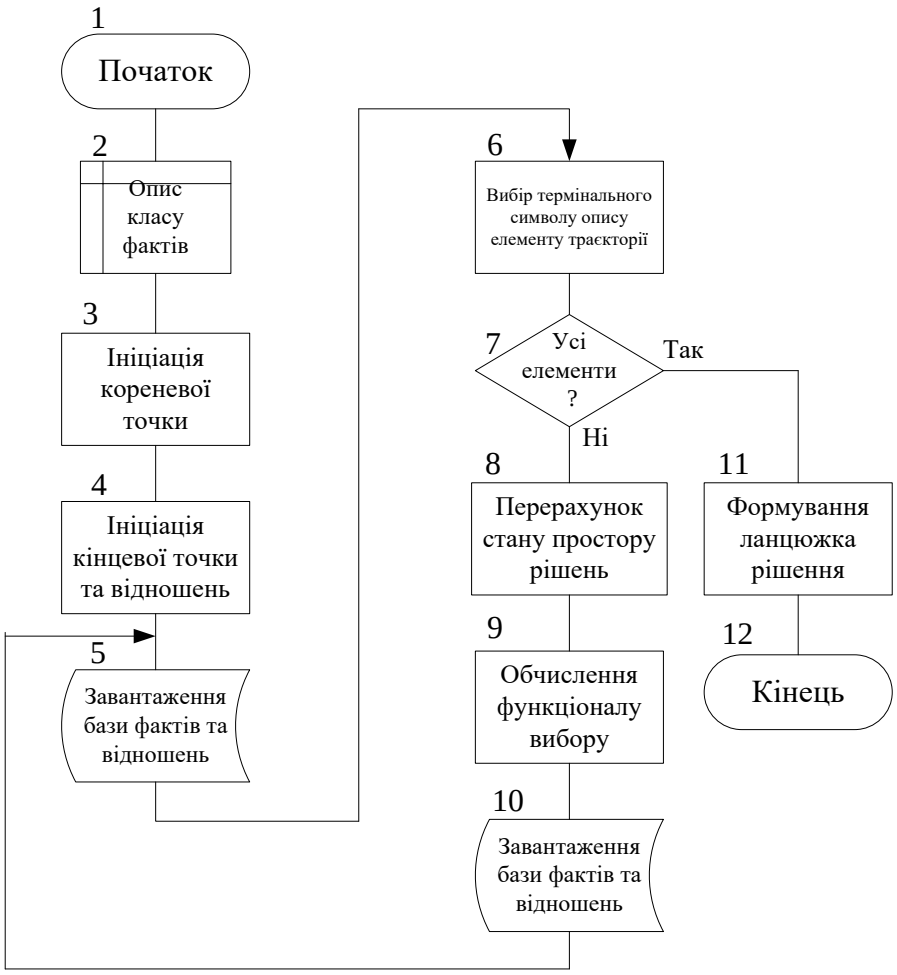


Рис. 5.31. Схема алгоритму евристичного пошуку

Для синтезу ланцюжків рішень накладемо координатну сітку на простір $G^*_{ріш}$ та визначимо позиції ОС, які позначені блакитними кругами, та термінальні початкові і кінцеві позиції, які позначені жовтими кругами (рис. 5.32). Координати вершин лінії границі простору $G^*_{ріш}$ та позицій ОС і термінальних кінцевих точок траєкторії переміщення ОУ наведені в табл. 5.6 і табл. 5.7 відповідно.

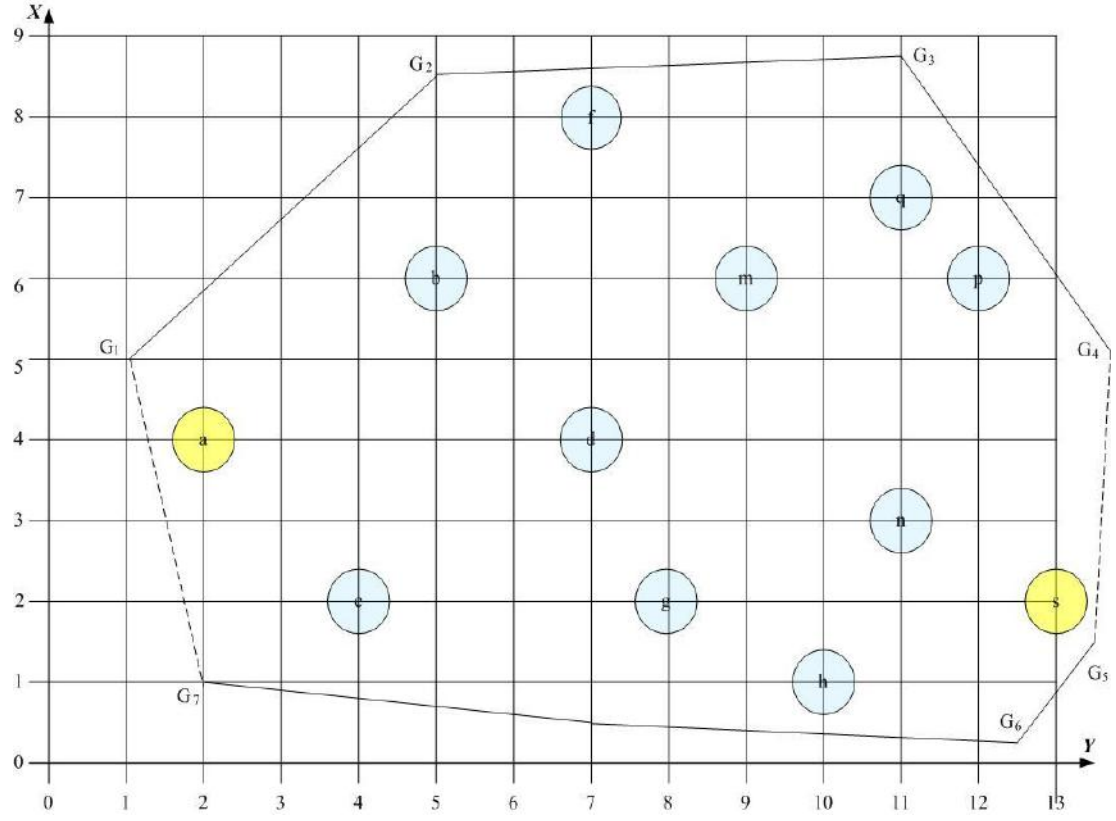


Рис. 5.32. Простір рішень

Для синтезу ланцюжків рішень визначимо ортограф припустимих переходів (траєкторій) переміщення ОУ в просторі $G^*_{ріш}$ (рис. 5.33), який відображає правила та відношення для граматики $\Gamma^0_{пр}$.

Простір $G^*_{ріш}$ враховує усі обмеження на переміщення ОУ. Синтезований граф може бути довільним і не має обмежень щодо можливості реалізації синтезованих ланцюжків управління.

Таблиця 5.6

Координати границі простору $G^*_{ріш}$

Лінія границі	G_1	G_2	G_3	G_4	G_5	G_6	G_7
Координати	(1.0; 5.0)	(5.0; 8,6)	(11,0;8,7)	(13.6;5.1)	(13.4;1.4)	(12.5;0.2)	(2.0;1.0)

Таблиця 5.7

Координати позицій ОС і термінальних кінцевих точок траєкторії

Позиція	a	b	c	d	e	g	q	h	p	m	n	s
Координати	(2.0;4.0)	(5.0;6.0)	(7.0;8.0)	(7.0;4.0)	(4.0;2.0)	(8.0;2.0)	(11.0;7.0)	(10.0;1.0)	(12.0;6.0)	(9.0;6.0)	(11.0;3.0)	(13.0;2.0)

Принцип оптимальності при виборі стратегії рішення визначається критерієм, який враховує можливості щодо управління ОУ і описується функціоналом ціни (2.33), (4.5), (4.11), (4.13), (4.14) та значеннями коефіцієнтів часткових критеріїв відбору.

Визначимо для гіпотетичного прикладу довільний простір (рис. 5.34), який обмежений багатокутником $(G_1G_2G_3G_4G_5G_6G_7)$. Сторони (G_1G_7) та (G_4G_5) багатокутника утворені за правилом замикання (3.42).

Для довільного простору $G^*_{ріш}$ визначимо правила та співвідношення (рис. 5.33).

Співвідношення та правила визначені у вигляді

op("вершина-початок","вершина-кінець"). (5.23)

Усі припустимі вершини визначені у вигляді співвідношень

xy ("назва вершини графа", абсциса вершини, ордината вершини). (5.24)

Виходячи з правил і відношень, які визначені співвідношеннями (5.23) і (5.24), граф переходів в просторі $G^*_{ріш}$ відображено на рис. 5.34.

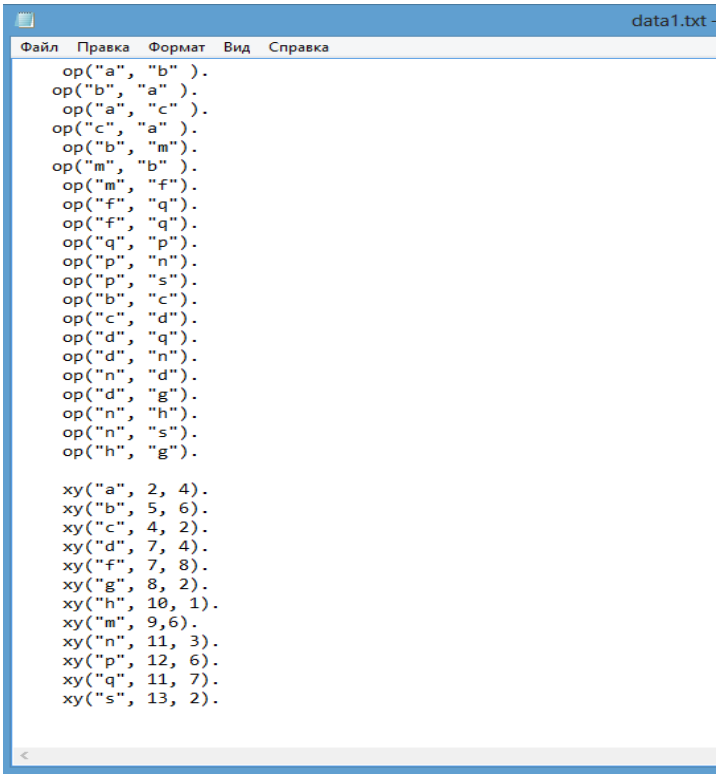


Рис. 5.33. Правила та відношення

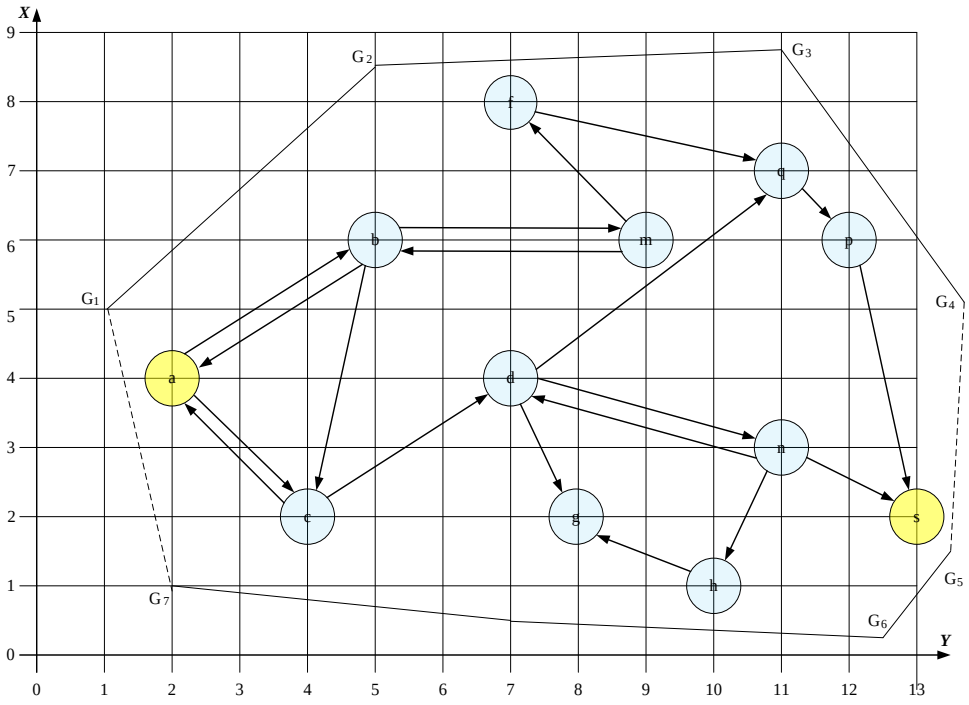


Рис. 5.34. Простір $G^*_{piш}$ та ортограф припустимих переходів

Імітаційна модель синтезу рішень за формальними правилами та відношеннями з врахуванням вимог функціоналу ціни наведена на рис.5.35.

За результатами розрахунку синтезовано траєкторію, яка позначена червоним кольором.

Слід зазначити, що алгоритм розрахунку траєкторії доцільно реалізувати в інструментальному середовищі функціонального та логічного програмування.

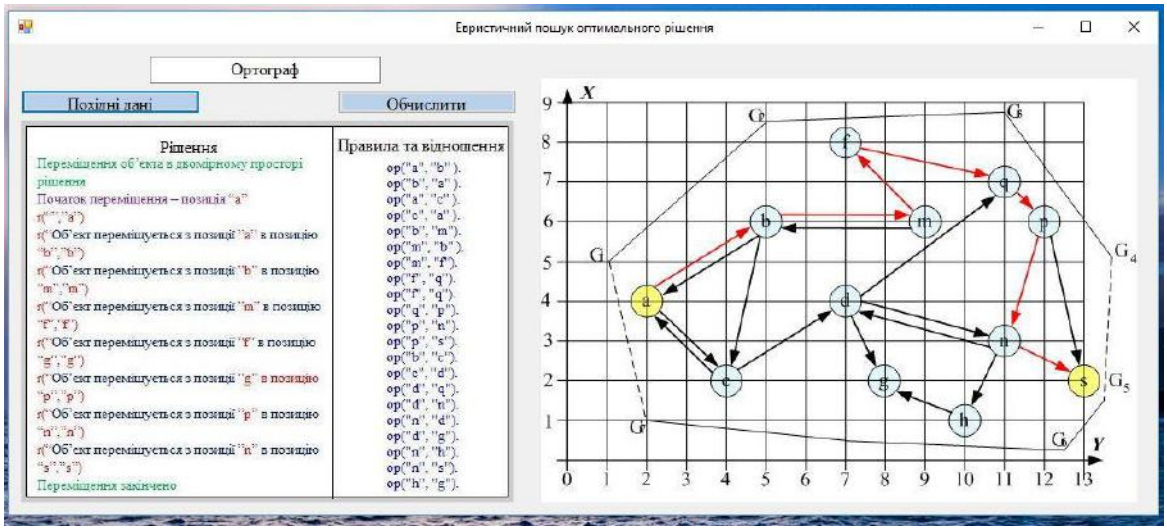


Рис. 5.35. Форма інтерфейсу імітаційної моделі синтезу рішень

Використання парадигми об'єктно-орієнтованого програмування дозволяє здійснювати перевантаження правил та відносин в процесі синтезу ланцюжків рішень в просторі $G^{*}_{piш}$, який змінюється при переміщенні ОУ (рис. 5.36).

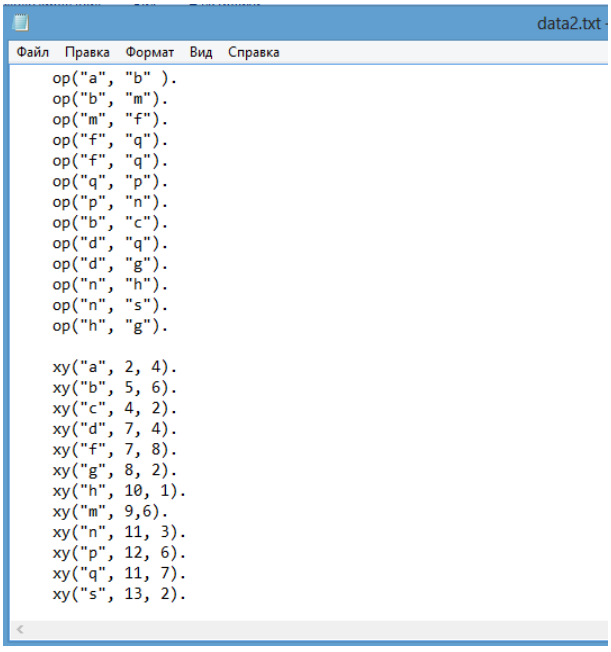


Рис. 5.36. Перевантажені правила і відношення

За перевантаженими правилами і відношеннями маємо орієнтований граф переходів в просторі $G^*_{ріш}$, який наведено на рис. 5.37. Відношення та правилами визначаються зміни в стані простору $G^*_{ріш}$.

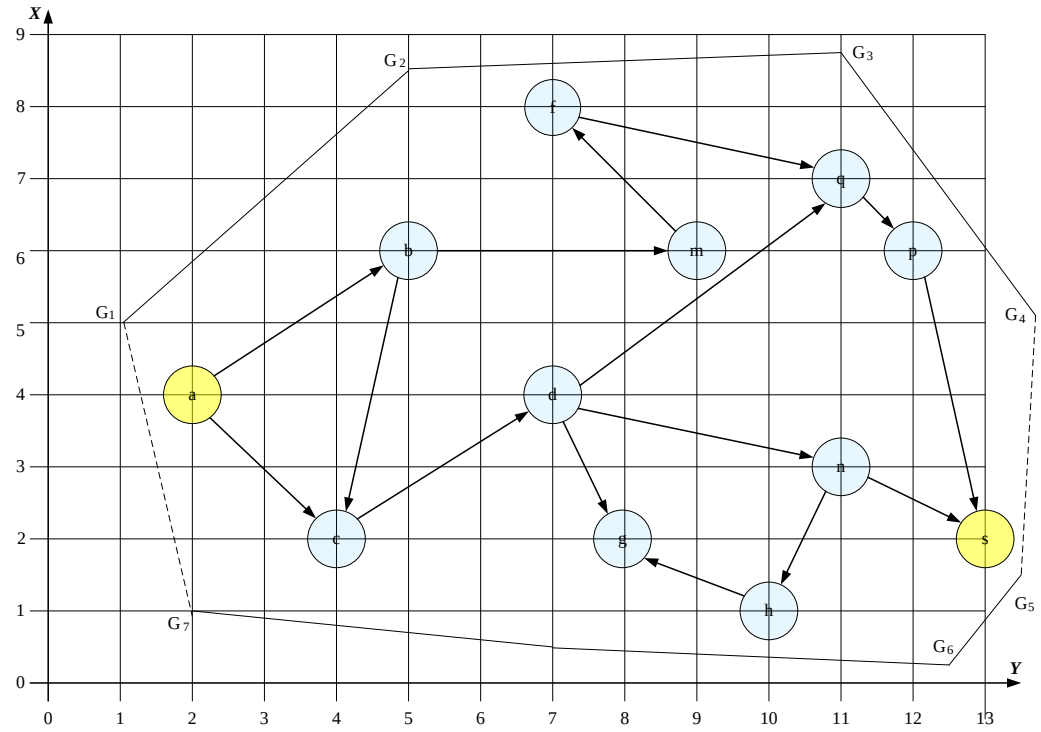


Рис. 5.37. Простір $G^*_{ріш}$ та орієнтований граф припустимих переходів

Після перевантаження правил і відношень імітаційна модель синтезує рішення, яке позначено у вигляді траєкторії, що виділена червоним кольором на рис. 5.38.

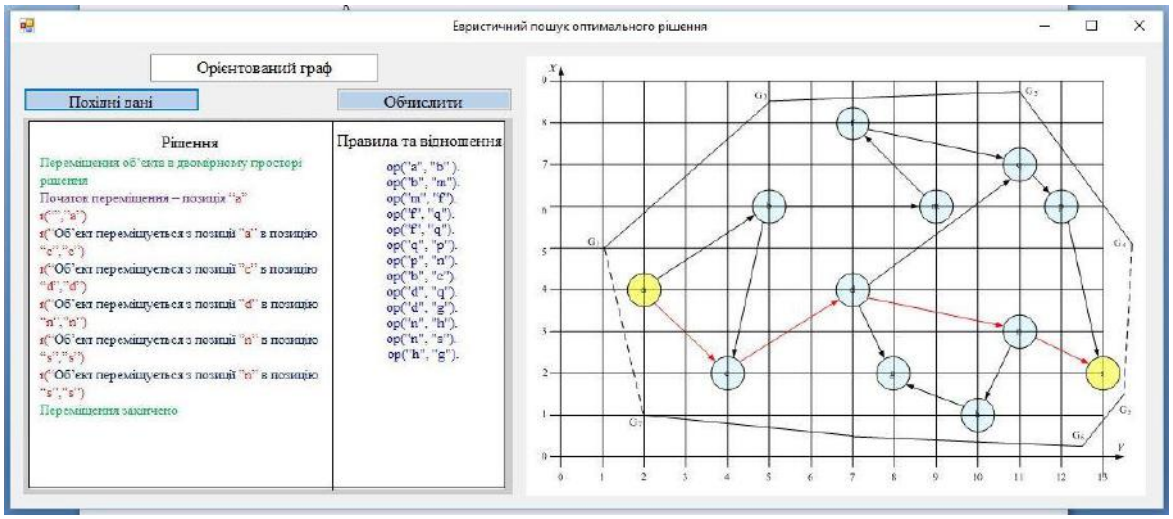


Рис. 5.38. Форма інтерфейсу імітаційної моделі синтезу рішень

За умов визначеного простору $G^{*}_{ріш}$ алгоритм евристичного пошуку дозволяє синтезувати стратегію управління ОУ, яка забезпечує вирішення задачі ухиляння від зближення ОУ з ОС, яка є характерною для методів пошуку та переслідування.

Застосування алгоритму евристичного пошуку дозволяє здійснювати модифікацію семіотичної системи шляхом зміни множини її аксіом. Сенс адаптації семіотичної системи полягає в зміні множини її правил та відносин за результатами дій за правилами та відносинами.

Застосування ідей семіотичних систем виявилося плідним для вирішення задач управління не тільки кінематичними та динамічними об'єктами, але і некласичними, наприклад, кібернетичними.

5.3. Дослідження ефективності алгоритму синтезу рішень за методом інтегрального усікання варіантів

Взаємодія різних за своєю сутністю об'єктів в просторі Q, як в просторі спостережень, пошуку, станів та поведінки щодо ОУ описана, як модель конфлікту.

Виходячи з особливостей функціонування ОУ в просторі Q, слід враховувати те, що при синтезі рішень може змінюватись структура управління ОУ та інформація щодо функціонування і переміщення ОС, а також властивості самого простору.

З метою дослідження ефективності алгоритму синтезу рішень за методом інтегрального усікання варіантів було досліджено часові характеристики рішення конфлікту за методами дослідження операцій. В якості методів рішення конфлікту були досліджені показники рішення задачі конфлікту методами теорії ігор та методом лінійного програмування. В табл. 5.8 наведені часові показники імітаційного експерименту в обчислювальному середовищі з продуктивністю 0,8 *gigaFLOPS* щодо рішення задачі конфлікту методами

теорії ігор та методом лінійного програмування з розмірністю задачі $m \times n$ (m- кількість стратегій першого учасника, n- кількість стратегій другого учасника). Значення похибки визначалось $\varepsilon \le 0{,}001$.

Таблиця 5.8

Результати імітаційного експерименту

Розмірність задачі		Час обчислення результату (секунди)		
n	m	Метод Брауна-Робінсона	Монотонний ітераційний алгоритм	Симплекс-метод
10	10	3.289	0.414	0.276
20	20	17.908	0.598	0.299
30	30	49.312	2.185	0.552
40	40	88.447	6.072	0.921
50	50	138.667	9.929	1.932

Для порівняння часових показників рішення задачі конфлікту за методом інтегрального усікання варіантів та методами дослідження операцій було проведено імітаційні експерименти за однаковими похідними даними.

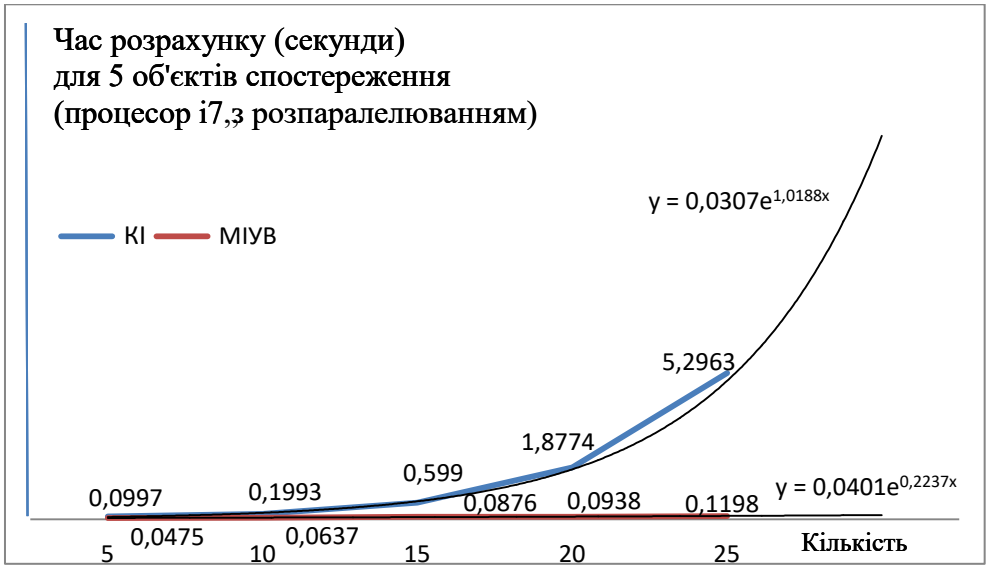
В результаті імітаційних експериментів було проведено порівняння показників швидкодії алгоритму методу інтегрального усікання варіантів з показниками швидкодії алгоритму вирішення біматричної коаліційної гри щодо 5 об’єктів спостереження для обчислювального середовища з продуктивністю 0,8 *gigaFLOPS*, що відповідає сучасному процесору *i*7 при застосуванні методів розпаралелювання обчислень (табл. 5.9).

Таблиця 5.9

Результати імітаційного експерименту

Кількість ітерацій	5	10	15	20	25
Час рішення біматричної коаліційної гри (сек)	0.0997	0.1993	0.599	1.8774	5.2963
Час рішення за методом інтегрального усікання варіантів (сек)	0.0475	0.0637	0.0876	0.0938	0.1198

На рис. 5.39 відображені дані табл. 5.9 з відповідними результатами апроксимації.



5.39. Залежність часу обчислення оптимального рішення від кількості ОС

Дослідження часових характеристик при синтезі траєкторій переміщення і ланцюжків управління ОУ при рішенні конфлікту в умовах обмежень, невизначеності та відкритої множини ОС показали незначну залежність часу обчислення оптимального рішення за методом інтегрального усікання варіантів від кількості ОС (рис. 5.40).

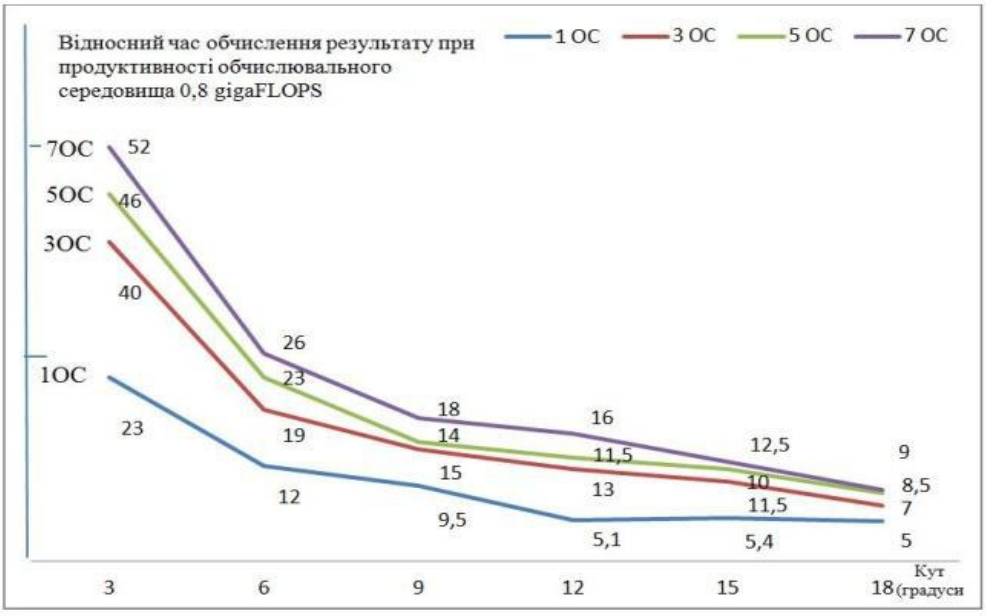


Рис. 5.40. Залежність часу обчислення оптимального рішення від кількості ОС за методом інтегрального усікання варіантів

Результати дослідження часових характеристик здійснення обчислювальних процедур при синтезі стратегій рішення конфлікту взаємодії ОУ з ОС в просторі $G_{piш}$ за методом інтегрального усікання варіантів в залежності від кількості ОС та інтервалу зміни напрямків переміщення ОУ на імітаційних моделях наведені на рис. 5.41.

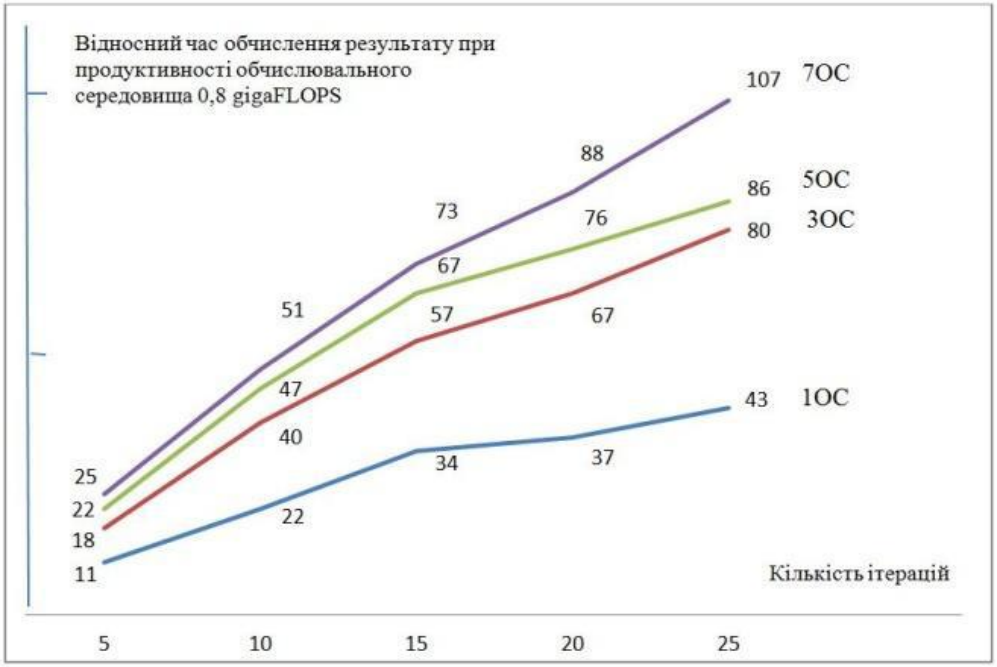


Рис. 5.41. Залежність часу обчислення від припустимої кількості ітерацій за методом інтегрального усікання варіантів

Дані, які наведені на рис. 5.40 і рис. 5.41, відображають лише тенденцію зміни часу обчислень за методом інтегрального усікання варіантів та вказують на незначну залежність часу здійснення обчислювальних процедур від кількості ОС та особливостей простору Q.

Подальші дослідження часових характеристик показали, що час рішення конфлікту за методом інтегрального усікання варіантів для 1 і для 9 ОС відрізняється в 7,143 рази і не перевищує 1 секунду при кількості кроків обчислення результату, яка визначається правилом зупинки. При застосуванні методу біматричних коаліційних ігор при 8 стратегіях для 1 і для 9 ОС час рішення конфлікту відрізняється в 7,768 рази і не перевищує 74 секунди для методу Брауна-

Робертсона, а для монотонного ітераційного методу відрізняється в 10,154 і не перевищує 28 секунд. В разі застосування методу інтегрального усікання варіантів для трьох ОС час рішення задачі конфлікту в 4,64 рази меншій, ніж при 8 ітераціях за методом біматричних коаліційних ігор, і в 8,243 рази менший, ніж при 8 ітераціях за монотонним ітераційним алгоритмом.

Тим самим, можна зробити висновок про те, що розглянуті ігрові методи не є ефективними щодо застосування в СІУ ОУ при рішенні конфлікту для активних ТС і ТЕС.

Імітаційні експерименти щодо дослідження алгоритмів методу інтегрального усікання варіантів дозволили отримати кількісні характеристики, за якими отримано співвідношення для оцінювання часу синтезу ланцюжків оптимальних рішень щодо управління ОУ при взаємодії з ОС в просторі Q

$$t = \frac{N_в}{N_o}(5{,}1851 + 39{,}392\frac{n}{(\Delta\psi)^2} - 1{,}7546\frac{n^2}{(\Delta\psi)^2} - 188{,}83\times k\times(\Delta\psi)^2 + 0{,}42559\frac{n^2}{\Delta\psi} + 0{,}2649\times k\times n - 0{,}091951\times k\times\Delta\psi - 0{,}5177\frac{n}{k} - 0{,}057979\times k\times n^2 + 0{,}2754\frac{\Delta\psi}{k} - 0{,}00999\times\Delta\psi\times n),(сек), \quad (5.25)$$

де $N_в$ - продуктивність обчислювального середовища, яке здійснює розрахунок стратегій рішення (*gigaFLOPS*);

N_o - базова продуктивність обчислювального середовища обчислювального середовища PDP11 (*gigaFLOPS*);

k – глибина пошуку (кількість ітерацій напрямком переміщення ОУ, який підлягає аналізу);

n – кількість ОС в просторі Q;

$\Delta\psi$ - інтервал зміни напрямків переміщення ОУ в просторі $G^*_{ріш}$.

Співвідношення (5.25) є функцією найкращого наближення в класі поліномів Чебишева [124] для оцінювання часових показників рішення конфлікту.

На основі дослідження даних, що отримані при імітаційному моделюванні процесів рішення задачі конфлікту в умовах обмежень та невизначеностей за методом інтегрального усікання варіантів, визначено доволі слабку залежність часу здійснення обчислювальних процесів від кількості ОС та ітерацій обчислення напрямків переміщення ОУ в просторі $G_{piш}$ (рис. 5.42). В наведеному прикладі дані отримані при моделюванні ситуації, що визначена табл. 5.1 - табл. 5.3, в обчислювальному середовищі з продуктивністю 106 *gigaFLOPS*.

Отримані результати дозволяють зробити висновок про те, що метод інтегрального усікання варіантів в сполученні з сучасними обчислювальними середовищами дозволяє в певній мірі вирішити проблеми, які визначені для методів дослідження операцій (рис. 1.1) при рішенні задачі конфлікту засобами ІП СІУ ОУ.

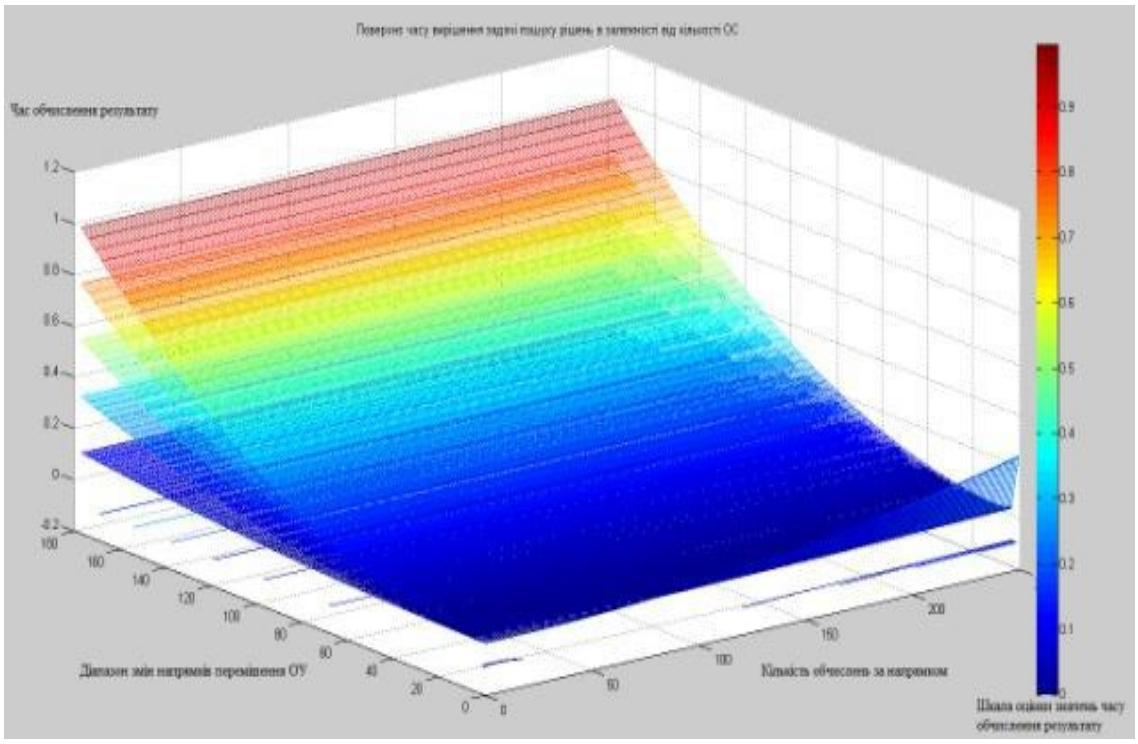


Рис. 5.42. Залежність часу рішень задачі конфлікту за методом інтегрального усікання варіантів

Виходячи з співвідношення (5.25) можна зробити висновок про те, що час рішення задачі конфлікту взаємодії ОУ з варіативною множиною ОС в умовах обмежень та невизначеностей у загальному випадку залежить тільки від кількості ОС в просторі Q. Тобто, в залежності від структур похідних даних часові значення здійснення обчислень можна характеризувати алгоритм, як такий, якому притаманна середня та максимальна просторова складність.

Відносно параметра *n*, який в співвідношенні (5.25) визначає кількість ОС, алгоритм є поліноміальним і може бути визначеним як "добрий" і віднесений до алгоритмів класу P. Таким чином алгоритм методу інтегрального усікання варіантів є алгоритмом з поліноміальним часом.

Співвідношення (5.25) показує, що функція часу обчислення рішення не має сідлової точки. Тобто запропонований метод синтезу рішень не має істотних обмежень щодо властивостей простору $G_{ріш}$ та кількості ОС.

Розроблений поліноміальний алгоритм синтезу траєкторій переміщення і ланцюжків гарантованих управлінь ОУ при взаємодії з варіативної множиною ОС в умовах конфлікту, обмежень та невизначеностей дозволяє вважати задачу конфлікту "добре розв'язуваною", а алгоритм визначити як P-складний.

В такому разі час вирішення задачі конфлікту взаємодії ОУ з варіативною множиною ОС є гарантовано оціненим.

Розроблені алгоритми обчислення ПКУО, СННП, простору $G^{*}_{ріш}$, параметрів пошуку рішення за методом інтегрального усікання варіантів, евристичного пошуку та інші, які використовуються для рішення конфлікту ІП СІУ ОУ, забезпечують формування інформаційних повідомлень, їх обробку, аналіз та синтез рішень і базуються на алгоритмах класу P, що використовує вузький клас функцій – клас предикатів $f:\Sigma^{*}\to\{0,1\}$. Використання класу предикатів стало можливим за рахунок декомпозиції теоретико-множинної моделі конфлікту взаємодії ОУ з варіативною множиною ОС в умовах обмежень та невизначеностей, яка визначена співвідношеннями (2.10) – (2.12).

Алгоритми класу P, що є одним з найвужчих класів складності, належать також класу *NP* і допускає поліноміальну реалізацію з нульовою помилкою.

Специфічною особливістю методів вирішення задач конфлікту, як оптимальних задач, є те, що до деякого етапу таку задачу вирішують аналітично, а вже потім визначають оптимальне рішення, що отримується при обчисленні критерію оптимальності, зміна якого служить оцінкою ефективності тієї чи іншої дії.

Відповідно результатів досліджень імітаційних моделей оптимального рішення конфлікту за методами дослідження операцій було оцінено тип обмежень на параметри процесу рішення, кількість змінніх і параметрів процесу, показники швидкодії, збіжності і стійкості отриманих рішень (5.43).

Тип рівняння опису		Кінцеві рівняння						Диференційні рівняння					
Обмеження типу змінних		Ні		Рівності		Нерівності		Ні		Рівності		Нерівності	
Кількість змінних		?(5)	>5	?(5)	>5	?(5)	>5	?(5)	>5	?(5)	>5	?(5)	>5
Метод оптимізації	Класичний аналіз	А	Б	Г	Г	Г	Г	В	Г	Г	Г	Г	Г
	Динамічне програмування	А, Д	В, Д	А, Д, Є	В, Д, Є	А, Д	В, Д	Б	В	В	В	В	В
	Лінійне програмування				Б, Е	Б, Е	А, Е						
	Нелінійне програмування	Б	А	Б	А	Б	А	Г	Г	Г	Г	Г	Г
А – ефективне використання; Б – використовується; В – використання припустиме; Г –використання як допоміжного; Д – багатокрокові (визначається окремо для кожного кроку); Е – використання лінійних критеріїв і обмежень; Є – використання множників Лагранжа													

Рис. 5.43. Результати аналізу можливостей застосування методів дослідження операцій

Наведені результати не протирічать рельтатам [188], які були отримані іншими дослідниками. Отримані оцінки у визначній мірі залежать від якості алгоритмічної і програмної реалізації та продуктивності обчислювального середовища.

5.4.Аналіз характеристик функціонування системи інтелектуального управління об’єктом при рішенні конфлікту

Стабілізуючий регулятор, яким є обчислювальне середовище ІП СІУ, в якому функціонує алгоритм методу синтезу рішень конфлікту щодо управління ОУ, при управлінні ДО на прикладі ЛАП наведено на рис. 5.44.

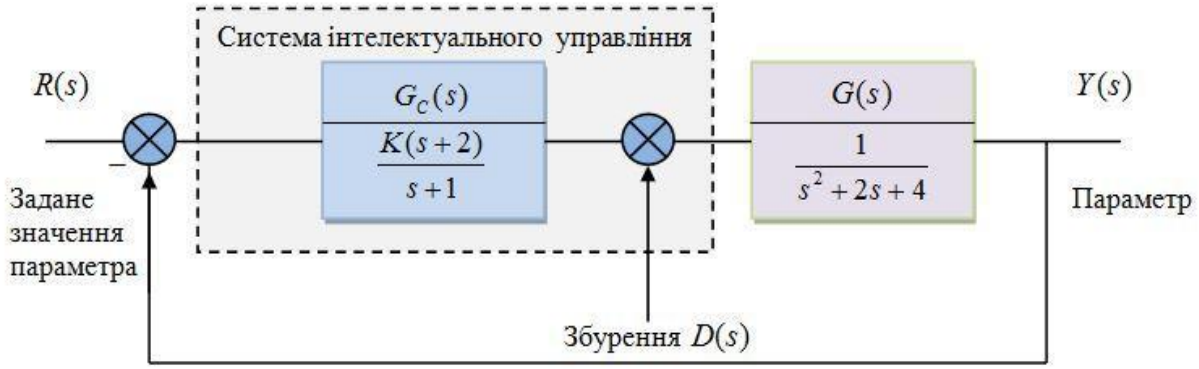


Рис. 5.44. Система інтелектуального управління з квазілінійною моделлю

Для ТС сигнал управління СІУ передається до ОУ. Збурення $D(s)$, які виникають в просторі Q та середовищі функціонування ТС, впливають на процес управління ОУ.

Передаткова функція СІУ [104, 114, 115, 116, 223] має вигляд

$$G_c(s) = \frac{K(s+2)}{s+1}, \tag{5.26}$$

де K - коефіцієнт підсилювання.

Передаткова функція ОУ [104, 223] має вигляд

$$G(s) = \frac{1}{s^2 + 2s + 4}, \tag{5.27}$$

Для дискретного ступінчастого вхідного сигналу $R(s)$ в ТС визначимо похибку

$$e_{ss} = \lim_{s\to 0} sF(s) = \lim_{s\to 0}\left[\frac{R(s)}{1-G_c(s)G(s)}\right] = \frac{1}{1-G_c(s)G(0)} = \frac{1}{1+\frac{K}{2}} = \frac{2}{2+K}, \qquad (5.28)$$

Дослідження стійкості функціонування СІУ доцільно здійснити за критерієм Найквіста.

Для визначення стійкості СІУ дослідимо стійкість системи в розімкненому стані шляхом аналізу АФЧХ за домогою діаграм Ніколса і Боде. Оцінку характеристик перехідного процесу при впливі визначимо за результатами аналізу перехідних процесів в СІУ ОУ при різних значеннях параметрів системи управління.

Для ТС з квазілінійною СІУ, яка наведена на рис. 5.45, побудуємо діаграму Ніколса для значень коефіцієнта підсилення $K = 5$, $K = 10$, $K = 20$ (рис. 5.45).

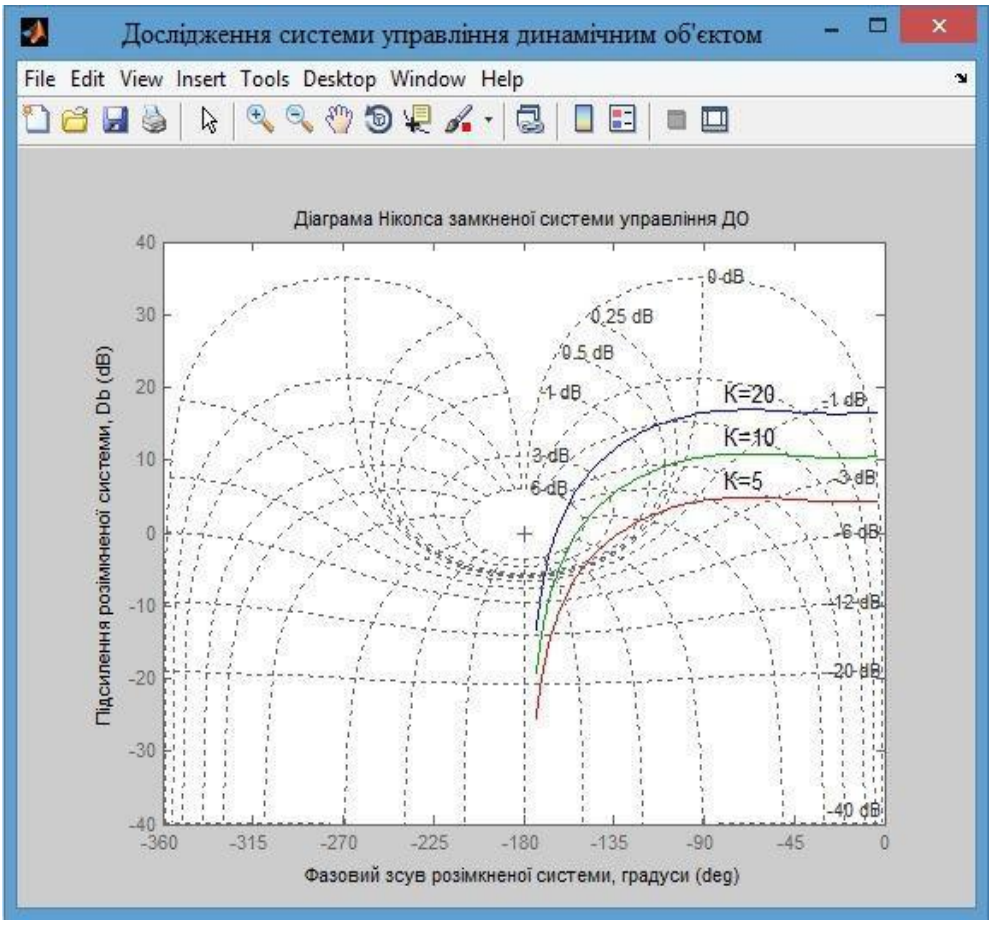


Рис. 5.45. Діаграма Ніколса

За діаграмою Ніколса визначимо, що для $K = 20$ значення сталої похибки згідно співвідношення (5.28) дорівнює $e_{ss} \approx 0{,}09$. В такому разі для побудови діаграми Боде визначимо

$$G_c(s)G(s) = \frac{10(1+\frac{s}{2})}{(1+s)(1+\frac{s}{2}+\frac{s^2}{4})} = \frac{20(2+s)}{(1+s)(4+2s+s^2)}. \qquad (5.29)$$

Для розрахованої діаграми Ніколса з врахуванням співвідношення (5.26) визначимо частотні характеристики (табл. 5.10).

Таблиця 5.10

Частотні характеристики

ω	0	1,2	1,6	2,0	2,8	4	6
дБ	20	18,4	17,8	16,0	10,5	2,7	-5,2
градуси	0	-65	-86	-108	-142	-161	-170

Фазові характеристики ТС, яка розглядається, розраховані та відображені на діаграмі Боде (рис. 5.46).

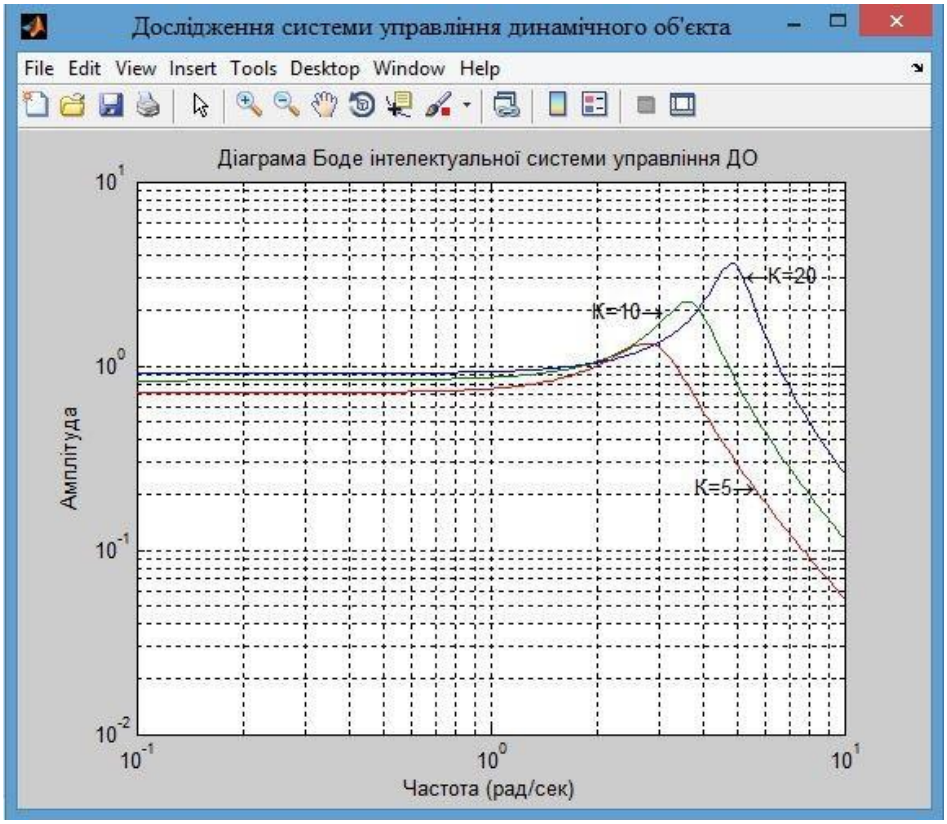


Рис. 5.46. Діаграма Боде

З табл. 5.10 та діаграми Ніколса (рис. 5.46) визначимо, що для значення підсилення розімкненого контуру управління $M_{p_\omega} = 12 \partial Б$ запас по фазі дорівнює 15°. При цьому система є недемпфованою, а перерегулювання становить приблизно 60%.

Діаграма Боде дозволяє встановити зв'язок між часовими та частотними характеристиками ТС та визначити діапазони резонансних частот при управлінні ОУ з застосуванням СІУ.

Перехідні характеристики запропонованої ТС відображені на рис. 5.47.

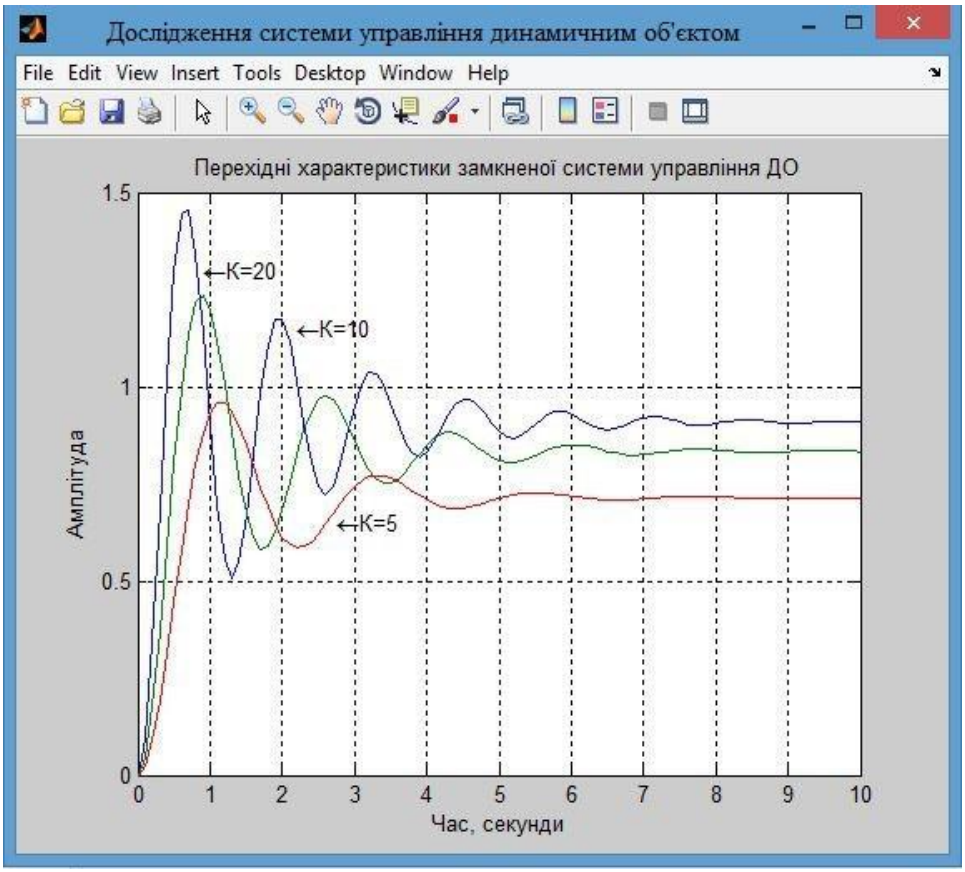


Рис. 5.47. Перехідні характеристики ТС

Як показує аналіз перехідних характеристик ТС, збільшення коефіцієнту K може призвести до виникнення явища автоколивань. Для аналізуємої ТС найбільш прийнятним є значення $K = 10$ за умов визначення пере регулювання та коефіцієнту затухання.

З метою більш детального розуміння сутності процесів в ТС досліджені частотні характеристики ІП СІУ та ОУ згідно співвідношенням (5.26) та (5.27)

Діаграма Ніколса СІУ (рис. 5.48) показує, що функція $G_c(s)$ не має полюсів в правій півплощині, а контур не охоплює точку “-1” при будь-яких значеннях K. Тобто СІУ завжди буде стійкою при усіх додатних значеннях коефіцієнта K.

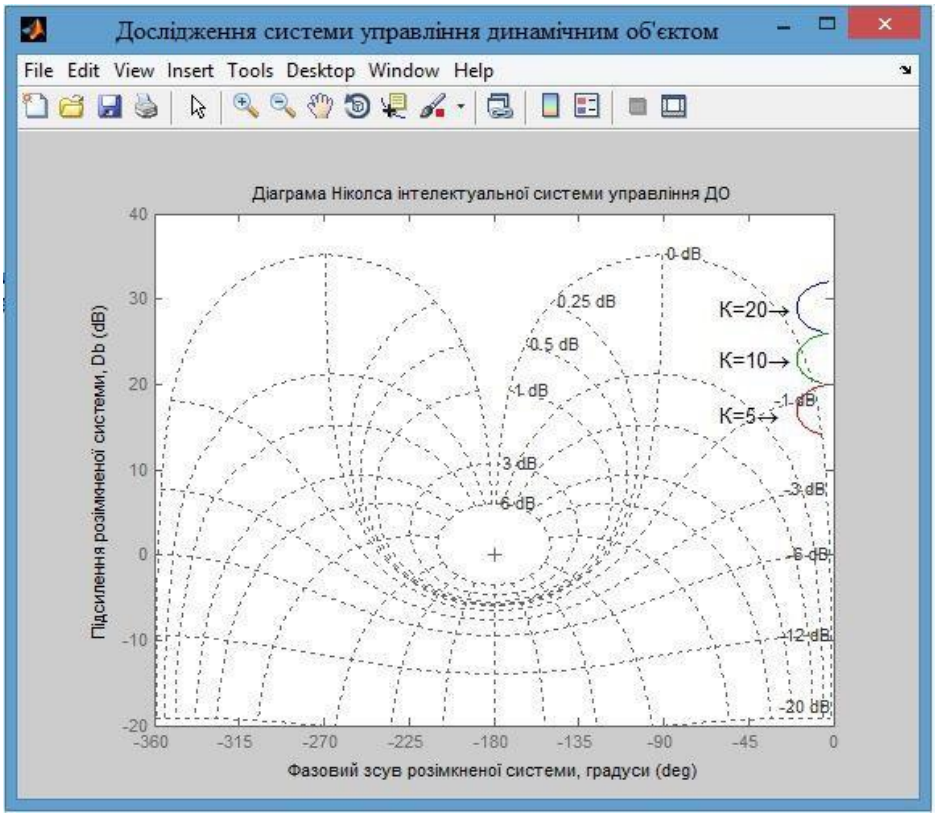


Рис. 5.48. Діаграма Ніколса СІУ

Діаграма Боде ІП СІУ (рис. 5.49) показує, що фактично відсутній зв’язок між часовими та частотними характеристиками і ІП СІУ є стійкім.

Тобто СІУ не має виразних властивостей до виникнення резонансних явищ щодо управління ТС.

Перехідна характеристика СІУ (рис. 5.50) підтверджує сталість та відсутність впливу на характеристики процесу управління ТС, крім наявної затримки, що обумовлено обчислювальними процесами при синтезі управління ОУ.

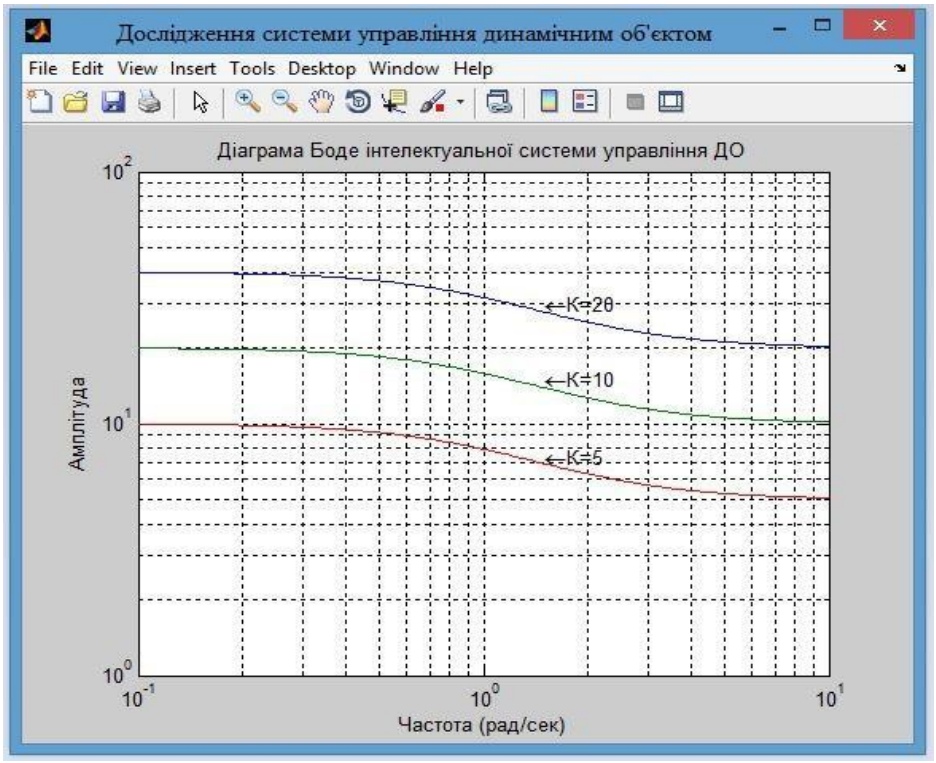


Рис. 5.49. Діаграма Боде СІУ

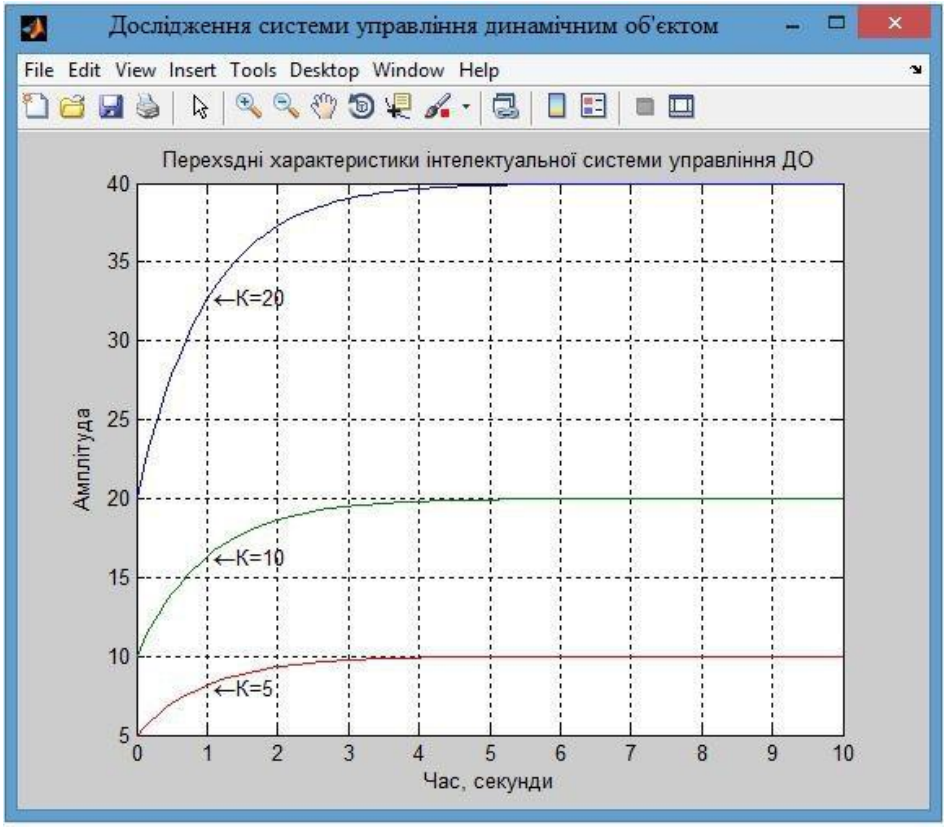


Рис. 5.50. Перехідні характеристики СІУ

Згідно співвідношення (5.27) проаналізуємо частотні характеристики ОУ.

Діаграма Ніколса для ОУ (рис. 5.51) показує, що функція $G(s)$ не має полюсів в правій півплощині і а контур не охоплює точку “-1”. Тим самим ОУ є стійким елементом ТС.

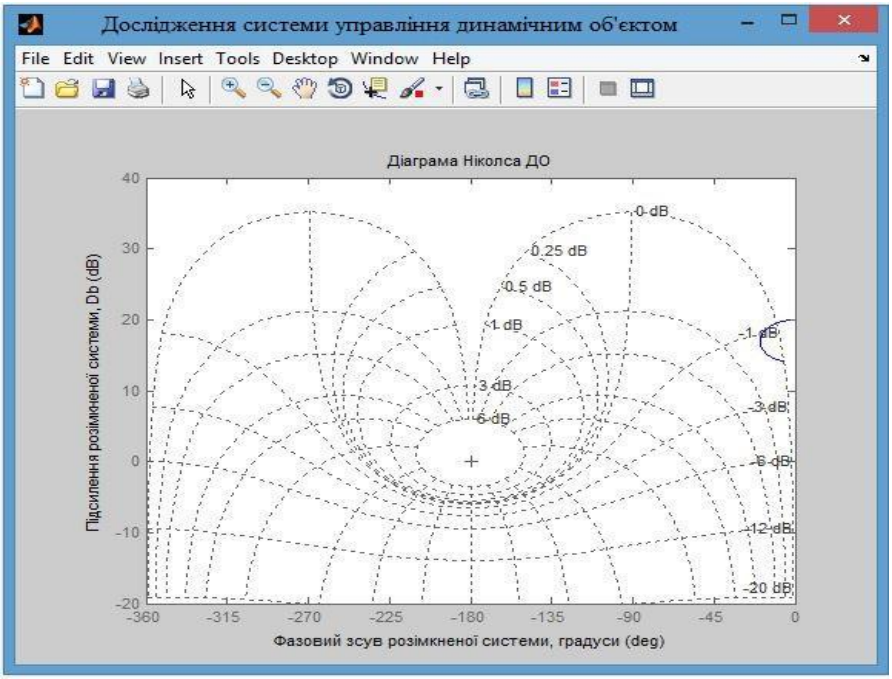


Рис. 5.51. Діаграма Ніколса для ОУ

Діаграма Боде ОУ (рис. 5.52) показує, що фактично відсутній зв’язок між часовими та частотними характеристиками. Тим самим ОУ є стійким.

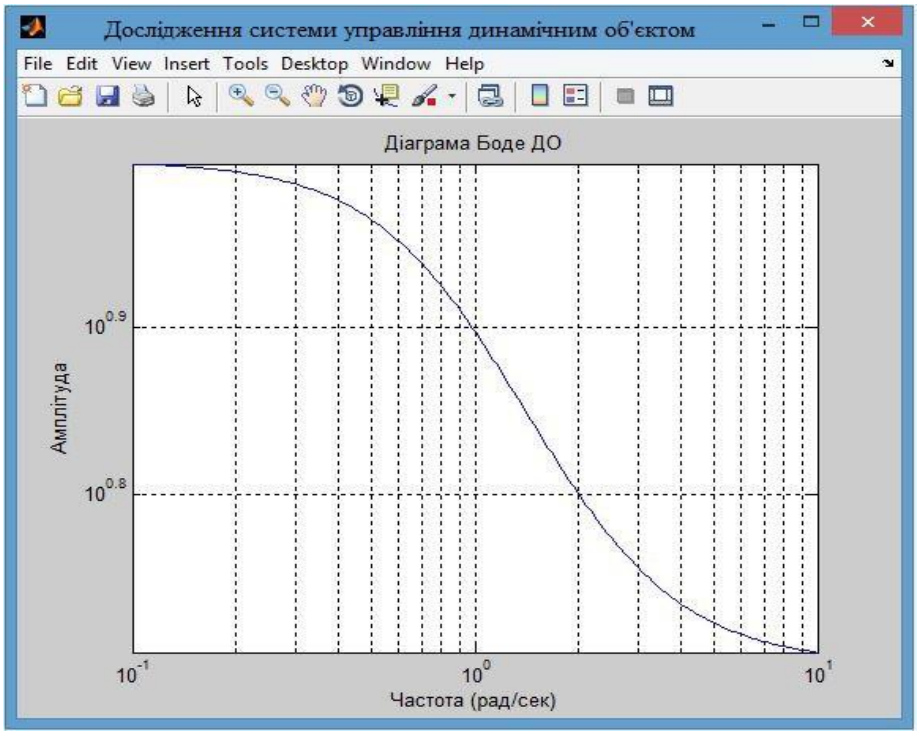


Рис. 5.52. Діаграма Боде ОУ

Перехідні характеристики ОУ (рис. 5.53) показують сталість стану в просторі функціонування.

Аналіз частотних характеристик СІУ та ОУ проведено для розімкнених систем управління.

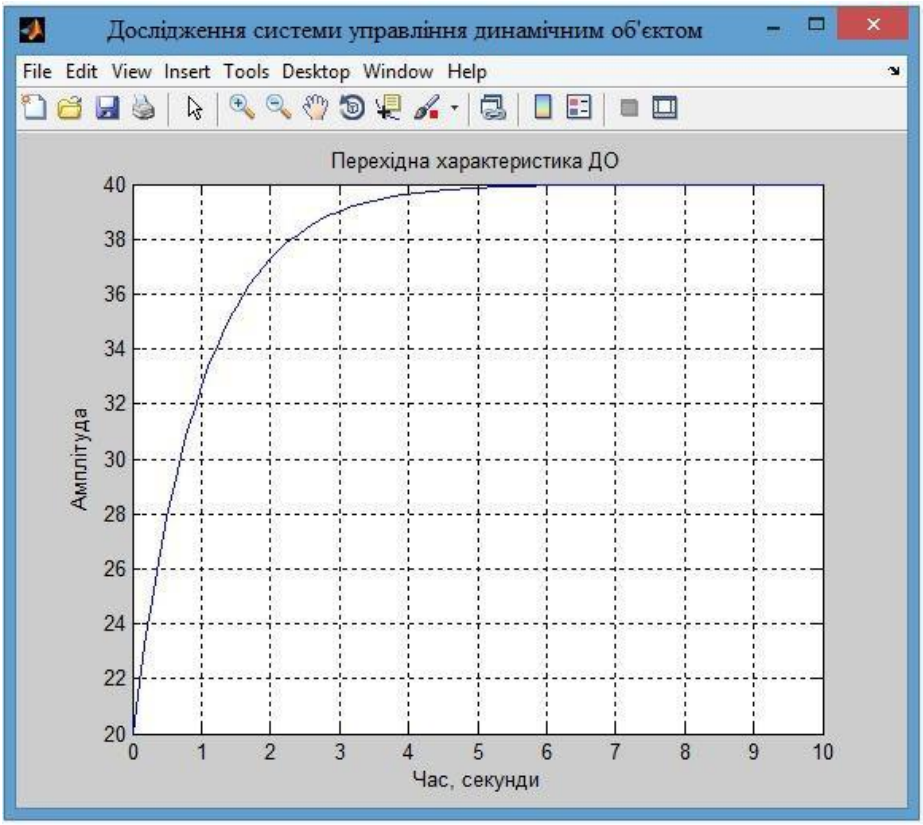


Рис. 5.53. Перехідні характеристики ОУ

Для дослідження області стійкості ТС запишемо передаточну функцію СІУ у вигляді

$$G_c(s) = \frac{K(s+z)}{s+p}, \tag{5.30}$$

де $z > 0$ і $p > 0$.

В такому разі з врахуванням співвідношення (5.30) маємо передаточну функцію ТС у вигляді

$$G_c(s)G(s) = \frac{K(s+z)}{(s+p)(s^2+2s+4)}. \tag{5.31}$$

Отримана передаточна функція ТС є стійкою в розімкненому стані і за виглядом характеристичного рівняння

$$(s + p)(s^2 + 2s + 4) = 0. \qquad (5.32)$$

Співвідношення (5.31) – (5.32) підтверджують стійкість системи, яка зображена на рис. 5.45.

Таким чином, визначено, що СІУ ОУ забезпечує стій

Висновки до п’ятого розділу

Досліджено стійкість та сталість стану системи інтелектуального управління об’єкту управління при рішенні конфлікту з використанням розробленої інформаційної технології, що забезпечується відсутністю зв’язку між часовими та частотними характеристиками системи.

До нових наукових результатів відносяться:

1. Удосконалена інформаційна технологія автоматизації процесів інтелектуального управління рішенням конфлікту в умовах обмежень та невизначеностей за рахунок розроблення і дослідження властивостей алгоритмів і програмних засобів формування:

- інформаційної множини простору комбінованого управління об'єктом;
- синтезу простору рішень;
- простору гарантованого управління динамічним об’єктом;
- моделі збурень зовнішнього середовища динамічного об’єкту управління.

2. Удосконалена інформаційна технологія вирішення задачі конфлікту взаємодії об’єктів за рахунок розроблення і дослідження властивостей алгоритмів і програмних засобів:

- опису та аналіз стану взаємодії об‘єкту управління з середовищем простору спостереження;

- синтезу варіантів припустимих рішень конфлікту за методом інтегрального усікання варіантів;

- евристичного пошуку рішення конфлікту;

3. Шляхом імітаційного моделювання досліджено ефективність алгоритму синтезу рішень конфлікту за методом інтегрального усікання варіантів.

4. Аналіз часових характеристик функціонування СІУ ОУ при рішенні конфлікту взаємодії з варіативною множиною ОС, яка забезпечує синтез та вибір стратегій рішення за методом інтегрального усікання варіантів згідно методології оптимального управління об’єктом в умовах конфлікту обмежень і невизначеності, дозволяє вирішити:

- методологічні проблеми рішення конфліктів взаємодії об'єктів, які властиві методам дослідження операцій (ефект “доміно”, ефект “прокляття розмірності”, гіпотезу про неконтрольовану природу неконтрольованих збурень, топологія силових полів);

- проблему забезпечення рішення конфлікту взаємодії ОУ з варіативною множиною ОС в ПС за поліноміальний час;

- проблему забезпечення вимог гомеостатичності функціонування ТС і ТЕС (функціональну і технологічну активність та грубість; функціональний і технологічний гомеостазис; стаціонарність і технологічність взаємодії; інваріантність та автономність; умовну та безумовну рефлекторну дію; багаторівневий конфліктний процес балансу можливостей, мети і ресурсів; гарантоване забезпечення цілісності рішень).

5. Отримані результати відрізняються від відомих в тому, що:

- на відміну від традиційних підходів щодо оптимального управління конфліктом вперше запропоновано метод процедурного рішення, який забезпечує стійкість СІУ незалежно від сутності простору спостереження і пошуку та природи об’єктів;

- вперше запропоновано універсальний підхід, який дозволяє створити алгоритми з поліноміальним часом вирішення конфлікту.

6. Розроблений метод динамічної дискретної оптимізації рішення конфлікту взаємодії ОУ з варіативною множиною ОС в умовах обмежень та невизначеності на основі теретико-множинної, семіотичної, семантичної моделей та методу інтегрального усікання варіантів за умов гарантовано оціненого управління ОУ дозволяє:

- визначити технологію оптимального управління об’єктом в умовах конфлікту, обмежень і невизначеності при варіативній множині ОС;
- забезпечити стійкість функціонування СІУ ОУ при застосуванні теоретико-множинної і семіотичної моделей опису конфлікті, методів і алгоритмів їх реалізації та використанні методу інтегрального усікання варіантів;
- забезпечити поліноміальний час рішення конфлікту за алгоритмом інтегрального усікання варіантів, що спрощує процес створення СІУ ОУ.

7. Стійкість системи інтелектуального управління забезпечується:

- відсутністю зв’язку між часовими та частотними характеристиками системи, що визначено діаграмами Боде;
- додатнім значенням коефіцієнту пропорційної ланки передаткової функції системи;
- розраховані і побудовані діаграми Ніколса не мають полюсів передаткової функції в правій півплощині і а контур не охоплює точку “-1”;

8. СІУ не має виразних властивостей до виникнення резонансних явищ щодо управління ОУ.

9. Перехідна характеристика СІУ підтверджує сталість та відсутність впливу на характеристики процесу управління ОУ, крім наявної затримки, що обумовлено обчислювальними процесами при синтезі управління ОУ.

10. Проведено порівняльний аналіз показників швидкодії розробленого методу рішення задічі конфлікту з методами дослідження операцій.

11. За класифікацією Міжнародної федерації з автоматичного управління IFAC (International Federation of Automatic Control) СІУ ОУ відноситься до нового класу систем (Integrated Intellectual Communication Computer Command

Control System), які функціонують за замкненим контуром управління з процесами жорсткого реального часу.

Основні результати розділу опубліковані у роботах [138, 142, 143, 148, 149, 150, 159 - 164, 166, 170 - 175].

ВИСНОВКИ

В дисертаційній роботі отримано принципово нове вирішення актуальної наукової проблеми, що полягає у забезпеченні поліноміального часу рішення задач конфлікту та стійкості функціонування СІУ об’єктом в цілому при взаємодії з варіативною множиною об’єктів спостереження в умовах існуючих обмежень та невизначеностей. За результатами аналізу та узагальнення наукових і методологічних засад щодо існуючих підходів і методів вирішення конфліктів в ТС та ТЕС в роботі, по-перше, запропоновано структуру математичної моделі інтелектуального перетворювача, по-друге, запропоновано структуру математичних, семіотичної і семантичної моделей, які формально описують конфлікт в ТС та/або ТЕС та сприяють вирішенню задачі синтезу стратегій та вибору оптимального рішення в СІУ ОУ, по-третє, сформульовано низку аксіом, які застосовуються при формуванні математичної моделі сукупності елементів ДО або ДС, по-четверте, запропоновано структурну схему системи інтелектуального управління об'єктом , по-п’яте, сформульовано наукову проблему, що підлягає вирішенню та запропоновано загальну концепцію дисертаційного дослідження.

До основних результатів дисертаційного дослідження слід віднести:

1) теоретико-множинну модель взаємодії об’єкту управління з варіативною множиною об’єктів спостереження за умов конфлікту, обмежень та невизначеностей, впровадження якої дозволило, по-перше, визначати основні взаємозв'язки конфлікту з середовищем, в якому конфлікт розвивається, по-друге, визначати характер і способи біфуркації елементів і підструктур конфлікту, по-третє, досліджувати властивості топологічності математичного простору представлення для ТММ конфліктної системи, по-четверте, формувати формальні моделі конфлікту взаємодії об'єктів конфліктуючої системи в просторі спостереження й, по-п’яте, визначати шляхи запобігання та рішення конфлікту в класі поліноміальних алгоритмів;

2) семіотичну модель взаємодії об’єкту управління з варіативною множиною об’єктів спостереження в просторі спостереження, впровадження якої дозволило, по-перше, використовувати правила змін семіотичної моделі інтелектуального перетворювача відповідно до ТММ, по-друге, використовувати мову і граматику загального виду для опису процесів взаємодії об’єктів в ПС, по-третє, використовувати формальну мову для опису множини ланцюжків управління об’єкту при переміщенні з початкової в кінцеву термінальну позицію в ПР;

3) семантичну модель взаємодії об’єкта управління з варіативною множиною об’єктів спостереження в просторі спостереження, впровадження якої дозволило, по-перше, використовувати мову і КВ-граматику опису подій в СІУ для МП-автомату аналізу повідомлень, по-друге, здійснювати синтез стратегій та ланцюжків гарантовано оцінених керуючих впливів в ІП СІУ при рішенні конфлікту взаємодії об’єкта управління з варіативною множиною об’єктів спостереження в просторі рішення;

4) метод рішення конфлікту взаємодії об’єкта управління з варіативною множиною об’єктів спостереження в умовах обмежень та невизначеностей, як NP-повної перебірної задачі динамічної дискретної оптимізації, впровадження якого дозволило, по-перше, визначити правила продукції для формування гарантовано оцінених ланцюжків управління при переміщенні ОУ з початкової в кінцеву термінальну позицію ПР, по-друге, визначити, що критерій відбору не входить до правил мови класу предикатів і входить в граматику загального виду ТММ ОУ, по-третє, забезпечити вирішення конфліктів великої розмірності з високим рівнем абстракції в гарантовано оціненому ПР, по-четверте, визначати властивості конфліктів при множинному уявленні ПР;

5) вирішуючі правила і формальні методи визначення інформаційних множин, доповнення простору рішень, гарантованого управління об’єктом при взаємодії з варіативною множиною об’єктів спостереження, впровадження яких дозволило, по-перше, формувати множини простору комбінованого управління об’єктом, по-друге, доповнювати простір рішень, по-третє, визначати сектори

небезпечних напрямків переміщення об’єкта управління, по-четверте, формально описувати простір рішень;

6) метод формування функціонального віртуального простору рішень для варіативної множини об’єктів спостереження, впровадження яких дозволило, по-перше, здійснювати інтегральне врахування інформаційних множин небезпечних станів простору рішення, по-друге, визначати множини гарантовано оціненого управління ОУ при переміщенні в ПР, по-третє, визначати напрямки переміщення інформаційних множин небезпечних станів в просторі рішення конфлікту;

7) метод синтезу та вибору стратегій (траєкторій) переміщення та гарантованого управління об’єктом в просторі рішень при взаємодії з варіативною множиною ОС в умовах конфлікту, обмежень і невизначеностей (метод інтегрального усікання варіантів), впровадження яких дозволило, по-перше, визначати чисельне, а не аналітичне подання моделі рішення конфлікту і обмежень конфліктуючої системи; по-друге, визначати оптимальну траєкторію переміщення об’єкта управління за умов його оптимального гарантовано оціненого управління; по-третє, визначити критерій відбору (функцію ціни) при рішенні конфлікту; по-четверте, визначити ознаку розривності ПР, по-п’яте, визначити правило зупинки при рішенні конфлікту, по-шосте, визначити алгоритм рішення конфлікту, по-шосте, забезпечити Р-час рішення конфлікту.

Результати імітаційного моделювання процесів щодо удосконалення інформаційних технологій автоматизації процесів оптимального управління об’єктом в умовах конфлікту, обмежень та невизначеності, а також дослідження частотних характеристик та характеристик стійкості СІУ ОУ для обчислювального середовища ІП з продуктивністю 0,8 gigaFLOPS свідчать, про те, що:

- стійкість функціонування СІУ забезпечується при додатньому значенні коефіцієнті підсилення пропорційного елемента (К>0) в передатковій функції;

- значення сталої похибки СІУ з регулятором, яким є підсистема синтезу цілі та рішень, не перевищує $e_{ss} \leq 0{,}09$ при запасі по фазі не більш 15° і перерегулюванні 60% та граничному щодо виникнення автоколивань значенні коефіцієнта підсилення пропорційного елемента $K \leq 20$;

- можливість застосування СІУ об'єктом для кібернетичних систем при значенні коефіцієнта підсилення пропорційного елемента K<3 та для динамічних об'єктів, включаючи нелінійні, при значенні K<10;

- застосування розробленої методології рішення конфлікту в інтелектуальному перетворювачі дає виграш у показнику швидкодії від 1,5 до 8,243 рази в порівнянні з методами рішення біматричних коаліційних ігор при взаємодії ОС в кількості до 4;

- поліноміальну залежність значення часу рішення конфлікту за алгоритмами, які реалізовують запропоновану методологію оптимального управління в СІУ об'єктом;

- ефективність використання розробленої методології при варіативній кількості об'єктів спостереження.

Достовірність результатів дисертаційного дослідження підтверджується результатами імітаційного моделювання рішення конфлікту за розробленою методологією та за методом рішення біматричної коаліційної гри.

У своїй сукупності отримані в роботі наукові результати утворюють нову інформаційну технологію рішення конфлікту для СІУ об'єктами різної фізичної сутності, включаючи динамічні та кібернетичні. Конкретні моделі й процедури синтезу і вибору альтернативних рішень при рішенні конфлікту взаємодії ОУ з варіативною множиною ОС складають при цьому методологічну основу оптимального управління об'єктом в умовах конфлікту, обмежень та невизначеностей.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Айзекс Р. Дифференциальные игры / Р.Айзекс. - М.: Мир, 1967. - 480с.

2. Абрамова Н.Т. Целостность и управление / Н.Т.Абрамова. - М.: Наука,1974. - 248с.

3. Адельсон-Вельский Г.М., Арлазаров В.А., Донской М.В. Программирование игр / Г.М.Адельсон-Вельский, В.А.Арлазаров, М.В.Донской. - М.: Наука, 1978. - 256с.

4. Адельсон-Вельский Г.М., Арлазаров В.Л., Битман А.Р., Донской М.В. Машина играет в шахматы / Г.М.Адельсон-Вельский, В.А.Арлазаров, М.В.Донской. - М.: Наука, 1983. - 208с.

5. Адельсон-Вельский Г.М., Динц Е.А., Карзанов А.В. Потоковые алгоритмы / Г.М.Адельсон-Вельский, Е.А.Динц, А.В..Карзанов. - М.: Наука, 1975. - 118с.

6. Адрианова В.Е. Деятельность человека в системах управления / В.Е.Адрианова. - Л.: ЛГУ, 1974. - 135с.

7. Айзерман М.А. Выбор вариантов: основы теории. / М.А.Айзерман, Ф.Т.Алескеров. - М.: Наука, 1990. - 240с.

8. Алексеев В.М., Тихомиров В.И., Фомин С.В. Оптимальное управление / В.М.Алексеев, В.И.Тихомиров, С.В.Фомин. - М.: Наука, 1979. - 223с.

9. Александров А.Д. Геометрия и приложения / А.Д.Александров - Новосибирск: Наука, 2006. - 748с.

10. Белман Р. Динамическое программирование / Р.Белман. - М.: ИЛ, 1960. - 400с.

11. Белман Р. Процессы регулирования с адаптацией / Р.Белман. - М.: Наука, 1964. - 359с.

12. Берж К. Общая теория игр нескольких лиц / К.Берж. - М.: Физматгиз, 1961. - 126с.

13. Бертсекас Д., Шрив С. Стохастическое оптимальное управление: случай дискретного времени / Д.Бертсекас, С.Шрив . - М.: Наука, 1985. - 280с.

14. Бортовая система предупреждения столкновений. // Flug revue, N9, p.p. 81 - 82.

15. Брайсон А., Хо Ю-Ши. Прикладная теория оптимального управления. Оптимизация, оценка и управление / А.Брайсон, Хо Ю-Ши. - М.: Мир, 1972. - 515с.

16. Бухановский И.Л. Радиолокационные методы судовождения / И.Л.Бухановский. - М.: Транспорт, 1970. - 248с.

17. Бурбаки Н. Общая топология. Основные структуры / Н.Бурбаки. - М.: Наука, 1968. - 278с.

18. Бурбаки Н. Элементы математики. Общая топология. Структуры / Н.Бурбаки. - М.: Наука, 1958. - 305с.

19. Вагущенко Л.Л., Вагущенко А.Л. Поддержка решений по расхождению с судами / Л.Л.Вагущенко, А.Л.Вагущенко. - Одесса: Фенікс, 2010. - 229с.

20. Васин В.Н. Дедукция и обобщение в системах принятия решений / В.Н.Васин. - М.: Наука, 1988. - 384с.

22. Васильев В.И. Распознающие системы: справочник / В.И.Васильев. - К.: Наукова думка, 1983. - 422с.

23. Вайман М.Я. Исследование систем, устойчивых в "большом" / М.Я.Вайман. - М.: Наука, 1981. - 255с.

24. Вентцель А.Д. Курс теории случайных процессов / А.Д.Вентцель. - М.: Наука, 1975. - 319с.

25. Вирт Н. Алгоритмы и структуры данных / Н.Вирт. - М.: Мир, 1989. - 360с.

26. Вилкас Э.И. Оптимальность в играх и решениях / Э.И.Вилкас. - М.: Наука, 1990. - 256с.

27. Волков Е.А. Численные методы / Е.А.Волков. - М.: Наука, 1982. - 254с.

28. Воробъев Н.Н. Основы теории игр. Бескоалиционные игры / Н.Н.Воробъев. - М.: Наука, 1984. - 496с.

29. Гладкий А. В. Формальные грамматики и языки / А. В. Гладкий. - М. : Наука. 1973. - 368 с.

30. Гаек П., Гавранек Е. Автоматическое образование гипотез: математические основы общей теории / П.Гаек, Е.Гавранек. - М.: Наука, 1984, 1984. - 280с.

31. Гермейер Ю.Б. Введение в теорию исследования операций / Ю.Б.Гермейер. - М.: Наука, 1971. - 383с.

32. Гладун П.В. Эвристический поиск в сложных средах / П.В.Гладун. - Киев: Наукова думка, 1977. - 166с.

33. Глушков В.М. Введение в кибернетику / И.М.Глушков. . - К: АН УССР, 1964. - 324с.

34. Гмурман В.Е. Теория вероятностей и математическая статистика / В.Е.Гмурман. - М.: Высшая школа, 1977. - 479с.

35. Гогин Ю.А. Логико-математическое моделирование больших или сложных систем (основания и математический аппарат) / Ю.А.Гогин. - Л.: ЛВИКА им. Мщжайского, 1972. - 261с.

36. Гогин Ю.А. Системы линейных логических уравнений и способы их решения / Ю.А.Гогин. - Л.: ЛВИКА им. Можайского, 1966. - 10с.

37. Грей П. Логика, алгебра и базы данных / П.Грей. - М.: Машиностроение, 1989. - 359с.

38. Григоренко Н.Л. Задача преследования несколькими объектами / Н.Л.Григоренко. // - М.: Труды математического ин-та АН СССР - 1984.- Т. 166 - С. 61-75.

39. Гроп Д. Методы идентификации систем / Д.Гроп. - М.: Мир, 1979. - 302с.

40. Грис Д. Конструирование компиляторов для цифровых вычислительных машин / Д.Грис. - М.: Мир, 1975. - 544с.

41. Дракин В.И. Общение конечных пользователей с системами обработки данных / В.И.Дракин, Э.В.Попов, А.Б.Преображенский. - М.: Радио и связь, 1988. - 288с.

42. Дружинин В.В. Введение в теорию конфликта / В.В.Дружинин, Д.С.Конторов, М.Д.Конторов. - М.: Радио и связь, 1989. - 288с.

43. Дружинин В.В. Системотехника / В.В.Дружинин, Д.С.Конторов. - М.: Радио и связь, 1985. - 200с.

44. Дынкин Е.Б. Управляемые марковские прооцессы / Е.Б.Дынкин, А.А.Юшкевич. - М.: Наука, 1975. - 341с.

45. Дюбин Г.Н. Введение в прикладную теорию игр / Г.Н.Дюбин, В.Г.Суздаль. - М.: Наука, 1969. - 336с.

46. Ермаков С.М. Математическая теория оптимального эксперимента / С.М.Ермаков, А.А.Жиглявский. - М.: Наука, 1987. - 320с.

47. Ермольев Ю.М. Методы стохастического программирования / Ю.М.Ермольев. - М.: Наука, 1976. - 240с.

48. Завьялов Ю.С.Сплайны в инженерной геометрии / Ю.С.Завьялов, В.А.Леус, В.А.Скороспелов. - М.: Машиностроение, 1985. - 224с.

49. Зараковский Г.М. Введение в эргономику / Г.М.Зараковский, Б.А.Королев, А.И.Медведев, П.Я.Шлаен. - М.: Сов. радио, 1974. - 352с.

50. Зараковский Г.М. Закономерности Функционирования эргатических систем / Г.М.Зараковский, В.В.Павлов. - М.: Радио и связь, 1987. - 232с.

51. Иллингоурт В. Толковый словарь по вычислительным системам / В.Иллингоурт . - М.: Машиностроение, 1989. - 568с.

52. Иоффе А.Д. Теория экстремальных задач / А.Д.Иоффе, В.М.Тихомиров. - М.: Наука, 1974. - 480с.

53. Исследование операций. Методологические основы и математические методы / под ред. Дж. Моудера, С.Элмаграби. - М.: Мир, 1981. - 712с.

54. Карманов В.Г. Математическое программирование: Учебное пособие / В.Г.Карманов. - М.: ФИЗМАТЛИТ, 2004. — 264 с.

56. Карпов В.Э. Классическая теория компиляторов /В.Э.Карпов. - М.: Московский государственный институт электроники и математики, 2011. - 91с.

57. Касти Дж., Большие системы. Связность, сложность и катастрофы / Дж.Касти. - М.: Мир, 1982. - 216с.

58. Касьянов В.А. Субъективный анализ: Монография / В.А.Касьянов. - К.: НАУ, 2007. - 512с.

59. Катковник В.Я. Многомерные дискретные системы управления / В.Я.Катковник, Р.А.Полуэктов. - М.: Мир, 1976. - 411с.

60. Ким Д.П. Методы поиска и преследования подвижных объектов / Д.П.Ким. - М.: Наука, 1989. - 336с.

61. Кнут Д. Искусство программирования на ЭВМ. Том 1. Основные алгоритмы / Д.Кнут. - М.: Мир, 1976. - 735с.

62. Кнут Д. Искусство программирования на ЭВМ. Том 2. Получисленные алгоритмы / Д.Кнут. - М.: Мир, 1977. - 724с.

63. Кнут Д. Искусство программирования на ЭВМ. Том 3. Сортировка и поиск / Д.Кнут.- М.: Мир, 1978. - 844с.

64. Колмогоров, С. В. Фомин. Элементы теории функций и функционального анализа / А. Н. Колмогоров, С. В. Фомин. - М.: ФИЗМАТЛИТ, 2006 г. - 572с.

65. Комплекс проблемно-ориентированной разработки и отладки автоматизированных систем предотвращения столкновений транспортных средств: (заключит. отчет) / ВНТИЦентр; руководитель В.В.Павлов - О.Ц.027.02.38; № ГР01824012234; инв. №02860024050. -М.: 1986. - Ч.3. - С.90-94.

66. Коньков В. Г. Интеллектуальные системы / В.Г.Коньков, К.А. Пупков. – М.: Издательство МГТУ им. Н. Э. Баумана, 2003. - 348с.

67. Коршунов Ю.М. Математические основы кибернетики / Ю.М.Коршунов. - М.:Энергоатомиздат, 1987. - 494с.

68. Кофман А. Методы и модели исследования операций / А.Кофман, А.Анри-Лабордер. - М.: Мир, 1977. - 432с.

69. Кофман А. Сетевые методы планирования и их применение / А.Кофман, Г.Дебазей. - М.: Прогресс, 1968. - 181с.

70. Красовский Н.Н. Позиционные дифференциальные игры / Н.Н.Красовский, А.И.Субботин. - М.: Наука, 1974. - 456с.

71. Красовский Н.Н. Управление динамической системой: задаче о минимуме гарантированного результата / Н.Н.Красовский. - М.: Наука. 1985. - 518с.

72. Кривонос Ю.Г. Динамические игры с разрывными траекториями. / Ю.Г.Кривонос, И.И.Матичин, А.А.Чикрий - К.: "Наукова думка", 2005. - 220с.

73. Кристофедис Н. Теория графов. Алгоритмический подход / Н.Кристофедис. - М.: Мир, 1978. - 432с.

74. Кузьмин В.Б. Построение групповых решений в пространствах четких и нечетких бинарных отношений / В.Б.Кузьмин. - М.: Наука, 1982. - 268с.

75. Кунцевич В.М. Управление в условиях неопределенности: гарантированные результаты в задачах управления и идентификации / В.М.Кунцевич. - К: Наук. думка, 2006. - 264с.

76. Лисков Б., Гатег Дж. Использование абстракций и спецификаций при разработке программ /Б. Лисков, Дж.Гатег. - М.: Мир, 1989. - 424с.

77. Ломов Б.Ф. Математика и психология в изучении процессов принятия решений / Б.Ф.Ломов. - М.: Наука, 1981.

78. Липский В. Комбинаторика для программистов / В.Липский. - М.: Мир, 1988. - 213с.

79. Льюс Р.Д., Райфа Х. Игры и решения / Р.Д.Льюс, Х.Райфа. - М.: ИЛ, 1961.

80. Мамиконов А.Г. Принятие решений и инрформация / А.Г.Мамиконов. - М.: Наука, 1983. - 184с.

81. Математическое моделирование / под ред. Дж.Эндрюс, Р.Мак-Лоун. - М.: Мир, 1979. - 248с.

82. Медич Дж. Статистические оптимальные оценки и управление / Дж.Медич.- М.: Энергия, 1973. - 440с.

83. Мессарович М., Макон Д., Такахара И. Теория иерархических многоуровневых систем / М.Мессарович, Д.Макон, И.Такахара. - М.: Мир, 1973. - 344с.

84. Мессарович М., Такахара Я. Общая теория систем: математические основы / М.Мессарович, И.Такахара. - М.: Мир, 1978. - 311с.

85. Мейстер Д. Эргономические основы разработки сложных систем / Д.Мейстер. - М.: Мир, 1979. - 456с.

86. Минский М. Фреймы для представления знаний / М.Минский. . - М.:Энергия, 1979. - 151с.

87.Михалевич В.С. Оптимизационные задачи производственно-транспортного планирования: модели, методы, алгоритмы / В.С.Михалевич, В.А.Трубин, Н.З.Шор. - М.: Наука, 1986. - 264с.

88. Михалевич В.С., Шор Н.З., Галустова Л.А., Журбенко Н.Б., Момот А.И., Трубин В.А., Юн Г.Н. Вычислительные методы выбора оптимальных проектных решений / В.С.Михалевич, Н.З.Шор, Л.А.Галустова, Н.Б.Журбенко, А.И.Момот, В.А.Трубин, Г.Н.Юн. - Киев: Наук. думка, 1977. - 178с.

90. Модель Б.И. Элементы теории многошаговых процессов последовательного выбора решений / Б.И.Модель. - М.: Наука, 1985. - 92с.

91. Молодцов Д.А. Устойчивость принципов оптимальности / Д.А.Молодцов. - М.:Наука, 1987. - 280с.

92. Моисеев Н.Н. Методы оптимизации / Н.Н.Моисеев, Ю.П.Иванилов, Е.М.Столярова. - М.: Наука, 1978. - 351с.

93. Молчанов А.Ю. Системное программное обеспечение: Учебник для вузов / А.Ю.Молчанов. - М.: Наука, 1978. - 396с.

94. Мулен Э. Теория игр с примерами из математической экономики / Э.Мулен. - М.: Мир, 1985. - 200с.

95. Назин А.В. Адаптивный выбор вариантов. Рекурентные алгоритмы / А.В.Назин, А.С.Поздняк. - СПб.: Питер, 2003. - 288с.

96. Нильсон Н. Принципы искусственного интеллекта. - М.: Радио и связь, 1985. - 375с.

97. Новосельцев В.Н. Теория управления и биосистемы / В.Н.Новосельцев. - М.: Наука, 1978. - 320с.

98. Орлов Ю.В. Теория оптимальных систем с обобщенными управлениями / Ю.В.Орлов. - М.: Наука, 1988. - 192с.

99. Осуга С. Обработка знаний / С.Осуга. - М.: Мир, 1989. - 293с.

100. Павлов В.В. Проблемы разработки систем разрешения конфликтов в технических системах / В.В.Павлов // Автоматика. - Киев: - 1982. - №5. - С.27-36.

101. Павлов В.В. Инвариантность и автономность нелинейных систем управления / В.В.Павлов. - К: Наук. думка, 1975. - 272с.

102. Павлов В.В. Конфликты в технических системах / В.В.Павлов. - К: Вища школа, 1982. - 184с.

103. Павлов В.В. Системы человек-машина: проблемы и синтез / В.В.Павлов. - К.: Вища школа, 1987. - 55с.

104. Павлов В.В. Начала теории эргатических систем / В.В.Павлов - К.: Наук. думка, 1975. - 240с.

105. Павлова С.В. Свойство слоения прообразов векторных полей динамически управляемых глобальных систем / С.В. Павлова // Кибернетика и вычисл. техника. – 2004. – Вып.144. – С.72–82.

106. Павлова С.В. Метод гарантованого оцінювання області повністю керованого стану складної нелінійної динамічної системи / С.В.Павлова, В.В.Павлов, В.І.Чепіженко // Вісник НАУ. –2011. – №2. – С. 27–32.

107. Патрик Э. Основы теории распознавания образов / Э.Патрик. - М.: Сов. радио, 1980. - 408с.

108. Палис Ж. Геометрическая теория динамических систем. Введение / Ж.Палис, В.Димелу. - М.: Мир, 1986. - 301с.

109. Пападимитру Х., Стайглиц К. Комбинаторная оптимизация. Алгоритмы и сложность / Х.Пападимитру, К.Стайглиц. - М.: Мир, 1985. - 510с.

110. Подиновский В.В. Математическая теория выработки решений в сложных ситуациях. Учебник / В.В.Подиновский. - М.: МО СССР, 1981. - 211с.

111. Понтрягин Л.С. Основы комбинаторной топологии / Л.С.Понтрягин. - М.: Наука, 1986. - 120с.

112. Понтрягин Л.С., Болтянский В.В., Гамкрелидзе Р.В., Мищенко Е.Ф. Математическая теория оптимальных процессов / Л.С.Понтрягин, В.В.Болтянский, Р.В.Гамкрелидзе, Е.Ф.Мищенко. - М.: Наука, 1983. - 393с.

113. Попов Э.В. Алгоритмические основы интеллектуальных роботов и искусственного интеллекта / Э.В.Попов, Г.Р.Фридман. - М.: Наука,1976. - 456с.

114. Поспелов Д.А. Ситуационное управление: теория и практика / Д.И.Поспелов . - М.: Наука, 1986. - 288с.

115. Поспелов Г.С. Искусственный интеллект – основа новой информационной технологии / Г.С.Поспелов. – М.: Наука, 1988. - 280с.

116. Поспелов Д.А. Мышление и автоматы / Д.А.Поспелов, В.Н.Пушкин. - М.: Сов. радио, 1972. - 226с.

117. Пшеничный Б.Н. Дифференциальные игры / Б.Н.Пшеничный, В.В.Остапенко. - К.: Наук. думка, 1991. - 264с.

118. Растригин Л.А. Современные принципы управления сложными системами / Л.А.Растригин. - М.: Сов. радио, 1980. - 232с.

119. Рейнгольд Э. Комбинаторные алгоритмы. Теория и практика / Э.Рейнгольд, Ю.Нивергельт, Н.Део. - М.: Мир, 1980. - 476с.

120. Рихтер К. Динамические задачи дискретной оптимизации / К.Рихтер. - М.: Радио и связь, 1985. - 136с.

121. Рокафеллар Р. Выпуклый анализ / Рокафеллар Р. - М.: Мир, 1973. - 470с.

122. Рубашкин В.Ш. Представление и анализ смысла в интеллектуальных информационных системах / В.Ш.Рубашкин. - М.: Наука, 1989. - 192с.

123. Семко В.В. Моделирование деятельности пилота по данным бортовых самописцев / В.Н.Голего, В.А.Коротеев, В.В.Семко //Кибернетика и вычисл. техника. - 1984. - вып.61 - С.44-48.

124. Семко В.В. Диалоговая система восстановления многомерных эмпирических зависимостей / В.Н.Голего, В.А.Коротеев, В.В.Семко //Кибернетика и вычисл. техника. - 1985. - вып.68. - С.72-78.

125. Семко В.В. Розробка інформаційно-аналітичної системи з безпеки цивільної авіації України / В.Н.Голего, В.В.Семко, О.І.Запорожець, С.О.Тарасенко, О.Й.Левіт // Обеспечение безопасности полетов в новых экономических условиях. Материалы международной научно-практической конференции- Киев: КМУГА, 1997. - С.50-51.

126. Семко В.В. Розробка автоматизованої системи супроводження нормативно-технічної і керівної документації цивільної авіації України / В.Н.Голего, В.В.Семко, С.О.Тарасенко, О.Й.Левіт // Обеспечение безопасности полетов в новых экономических условиях. Материалы международной научно-практической конференции - Киев: КМУГА, 1997. - С.51-53.

127. Семко В.В. Управление учебной деятельностью с помощью специализированной АОС для теоретической подготовки летного состава / Л.А.Журавлева, В.В.Семко // Методология создания и опыт эксплуатации АСУ в гражданской авиации. Тезисы докладов всесоюзной научно-технической конференции. - Рига: ЦНИИАСУГА, 1989. - С.152-153.

128. Семко В.В. Автоматизация процесса формирования знаний инженера-пилота с целью исключения ошибок, связанных с недостатками теоретической подготовки / Л.А.Журавлева, В.В.Семко // Тез. докл. всесоюзн. конф. "Безопасность полетов и профилактика авиационных происшествий". - Л.: ОЛАГА, 1988. - С.60.

129. Семко В.В. Структура системы автоматизации этапа теоретической подготовки бортинженеров (инженеров-пилотов) / Л.А.Журавлева, В.В.Семко // Проблемы профессиональной подготовки специалистов на этапе научно-технического прогресса. Часть 2. Тезисы докладов научно-практической конференции. - Кировоград: Кировоградская правда, 1989. - С.150-151.

130. Семко В.В. Построение банка ситуационных моделей для системы автоматизации этапа подготовки бортинженеров (инженеров-пилотов) / Л.А.Журавлева, В.В.Семко / Киев. ин-т инж. гражд. авиации. - Киев: 1989. - 10с. - Деп. в ЦНТИГА 16.11.89., №765-ГА89.

131. Семко В.В. Ситуационное управление процессом обучения при освоении деятельности бортинженера (инженера-пилота) в особых случаях полета / Л.А.Журавлева, В.В.Семко. / Киев. ин-т инж. гражд. авиации. - Киев: 1989. - 10с. - Деп.в ЦНТИГА 16.11.89., №766-ГА89.

132. Семко В.В. Применение метода интегрального усечения вариантов при синтезе стратегий управления подвижным объектом / В.В.Семко, В.В.Павлов //Кибернетика и вычислительная техника. - 1989. - Вып. 84. - С.1-6.

133. Семко В.В. Автоматизация управления подвижным объектом в условиях конфликта / В.В.Семко // Моделирование в обеспечении безопасности полетов. - Киев: КИИГА, 1987. - С.67-73.

134. Семко В.В. Автоматизированный метод построения областей активных управляемых состояний динамических объектов / В.В.Семко // Тр. молодых ученых /Киев. ин-т инж. гражд. авиации. - Киев; 1981. С.156-165. - Деп. в УкрНИИНТИ 07.09.81, №2981.

135. Семко В.В. Концепция решения задачи предотвращения столкновения самолетов в воздухе / В.В.Семко // Тез. докл. всесоюзн. конф. "Безопасность полетов и профилактика авиационных происшествий". - Л.: ОЛАГА, 1988. - С.22-23.

136. Семко В.В. Разработка и исследование метода и алгоритмов синтеза бесконфликтных траекторий движения судов в условиях ограничений : автореф. дис. на соискание учен. степени канд. техн. наук. : спец. 01.01.11 "Системный анализ и автоматическое управление" / В.В.Семко. - К:, 1987. - 15с.

137. Семко В.В. Применение теории конфликта в задаче предотвращения столкновений воздушных судов / В.В.Семко, В.В.Павлова // Методы и средства оценки уровня безопасности полетов гражданских воздушных судов. - Киев: КИИГА, 1985. - С.97-102.

138. Семко В.В. Построение области безопасного маневра при расхождении судов / В.В.Семко, В.В.Павлова / Киев. ин-т инж. гражд. авиации. - Киев: 1989. - 13с. - Деп. в ЦНТИГА 14.02.90., №789-ГА90.

139. Семко В.В. Система автоматизации этапа теоретической подготовки бортинженеров / В.В.Семко, Л.А.Журавлева // Предотвращение авиационных происшествий в гражданской авиации: Сб. научн. тр. - Киев: КИИГА, 1988. - С.93-102.

140. Семко В.В. Игровой подход к синтезу стратегий поведения сторон при предотвращении навигационного конфликта / В.В.Семко // Методология создания и опыт эксплуатации АСУ в гражданской авиации. Тезисы докладов всесоюзной научно-технической конференции. - Рига: ЦНИИАСУГА, 1989. С.71-72.

141. Семко В.В. Системный анализ функций и структур эргатической системы формирования знаний предметной области в специализированной АОС для летного состава / В.В.Семко, Л.А.Журавлева // Проблемы применения технических средств в формировании профессиональных навыков при первоначальной подготовке летного состава. Тезисы докладов второй всесоюзной научно-практической конференции. - Актюбинск: АВЛУГА, 1989. - С.93-94.

142. Семко В.В. Решение задачи распознавания образов при синтезе бесконфликтных траекторий движения динамического объекта. / В.В.Семко /

Киев. ин-т инж. гражд. авиации. - Киев: 1989. - 15с. - Деп. в ЦНТИГА 14.02.90, №790 - ГА90.

143. Семко В.В. Эвристический метод построения рекурсивного алгоритма принятия решений в условиях конфликта / В.В.Семко / Киев. ин-т инж. гражд. авиации. - Киев: 1989. - 14с. - Деп. в ЦНТИГА 14.02. 90, №788 - ГА90.

144. Семко В.В. Повышение безопасности компьютерных систем на основе учета психофизиологического состояния операторов / В.В.Семко, В.А.Темников, А.И.Арифов.// Збірник тез. II міжнародної науково-технічної конференції "Комп'ютерні системи та мережні технології". - Київ.: "НАУ-друк", 2009. - С.81.

145. Семко В.В. Правові питання інформатизації державного регулювання підприємництва в Україні /А.О.Селівванов, В.В.Семко // Право України. - Киев: ВАТ "Книжкова друкарня наукової кгниги", 1998. - С.110-115.

146. Семко В.В. Модель конфликта взаимодействия объектов в кибернетическом пространстве / В.В.Семко .// Комп'ютерні системи та мережні технології. Тези доповідей V міжнародної науково-технічної конференції. - Київ: НАУ, 2012.- С.116.

147. Семко В.В Концептуальные основы создания и развития дистанционной системы обучения / В.В.Семко, В.А.Темников, О.О.Михайловский // Український журнал телемедицини та медичної телематики. - 2007. – том 5, №3 - С.320-325.

148. Свідоцтво про реєстрацію авторського права на твір №25553 від 24.09.2008. Комп'ютерна программа "Програма для ПЕОМ щодо ведення бази даних дозволів на викиди забруднюючих речовин у атмосферне повітря та надання інформації громадськості" ("Дозвіл") / В.В.Семко, О.О.Михайловський; заявник та власник авторських прав Товариство з обмеженою відповідальністю "Елан"; заявл. №26809 від 22.08.2008; видане

Державним департаментом інтелектуальної власності Міністерства освіти і науки України; опубл. Кат. №12, Бюл. №17.

149. Свідоцтво про реєстрацію авторського права на твір №26411 від 11.11.2008. Комп'ютерна программа ”Психодіагностичні методики” до тематичного блоку ”Пізнай себе” Програмно-апаратного комплексу ”Профорієнтаційний термінал” / В.В.Семко, О.О.Михайловський; заявник та власник авторських прав Товариство з обмеженою відповідальністю "Елан"; заявл. №27406 від 27.10.2008; видане Державним департаментом інтелектуальної власності Міністерства освіти і науки України; опубл. Кат. №12, Бюл. №17.

150. Свідоцтво про реєстрацію авторського права на твір №39113 від 11.07.2011. Комп'ютерна программа ”Крос-платформена клієнт-серверна технологія захищеного обміну електронними даними з використанням терміналів мобільного телефонного зв’язку” (”ТЗОД”) / Семко В.В., Михайловський О.О., Гіденко О.В., Зінов'єв О.В., Мельник В.Т., Барткова В.А; заявник та власник авторських прав Товариство з обмеженою відповідальністю "Елан"; заявл. №39558 від 20.05.2011; видане Державним департаментом інтелектуальної власності Міністерства освіти і науки України; опубл. Кат. №15, Бюл. №25.

151. Семко В.В., Модель конфлікту взаємодії об’єктів кібернетичного простору / В.В.Семко //Проблеми інформатизації та управління. - 2012. - Вип. 2(38). - С.88-92.

152. Семко В.В. Система обеспечения безопасности на социально-экономических и информационных объектах с учетом психофизического состояния людей/ В.А.Темников, В.В.Семко, Е.Л. Темникова // Вісник Східноукраїнського національного університету імені Володимира Даля – 2007. - №5(111) - С.121-124.

153. Семко В.В. Построение системы определения психофизиологического состояния личности для профессиональной диагностики. / В.О.Темніков, В.В.Семко // Вісник Східноукраїнського

національного університету імені Володимира Даля – 2008. - №8(126). - С.195-199.

154. Семко В.В. Захищена операційна система. Досвід створення / В.В.Семко, В.О.Темніков // Комп'ютерні системи та мережні технології. Тези доповідей VI міжнародної науково-технічної конференції. - Київ: НАУ, 2013.- С.109.

155. Семко В.В., Формальний опис простору пошуку при синтезі рішень / В.В.Семко //Проблеми інформатизації та управління. - 2013. - Вип. 2(42). - С.104-111.

156. Семко В.В., Модель взаємодії кібернетичних організмів та синтез стратегій оптимального керування в кібернетичному просторі / В.В.Семко //Проблеми інформатизації та управління. - 2013. - Вип. 3(43). - С.75-82.

157. Семко В.В., Екологія кібернетичного простору / В.В.Семко, Г.М.Гулак // Інформаційна безпека людини, суспільства, держави. - 2014. - №1(14). - С.95-103.

158. Семко В.В., Логіко-математична модель опису простору рішень /В.В.Семко //Вісник Чернігівського державного технологічного університету. - 2013. - Вип. 2(65). - С.147-156.

159. Семко В.В., Дослідження властивостей рішення задачі конфлікту за методом інтегрального усікання варіантів / В.В.Семко, О.В.Семко //Проблеми інформатизації та управління. - 2014. - Вип. 2(46). - С.60-71.

160. Семко В.В. Використання методу інтегрального усікання варіантів при вирішенні задач конфлікту взаємодії об'єктів в просторі спостереження / В.В.Семко // Телекомунікаційні та інформаційні технології. - 2015. - Вип. № 1. - С.59-66.

161. Семко В.В. Моделювання турбулентності приземного шару атмосфери /В.В.Семко //Вісник Чернігівського державного технологічного університету. - 2015. - № 2(78). - С.230-234.

162. Семко В.В. Вирішення задачі конфлікту за методом інтегрального усікання варіантів / В.В.Семко // Телекомунікаційні та інформаційні технології. - 2015. - Вип. № 2. - С.40-50.

163. Семко В.В. Модель управління захистом інформації в інформаційно-телекомунікаційній системі / В.В.Семко, В.Л.Бурячок, С.В.Толюпа, П.М.Складанний // Вісник Національного університету «Львівська політехніка»: Радіоелектроніка та телекомунікації.- 2015. - № 818. - С.151-155.

164. Семко В.В. Ситуаційне управління доступом в інформаційно-телекомунікаційній системі [Електронний ресурс] / В.В.Семко, В.Л.Бурячок, С.В.Толюпа, П.М.Складаний // Проблеми телекомунікацій // ХНУРЕ. - 2015. - № 2(17). - С.54-61. Режим доступу до журн.: http://pt.journal.kh.ua/2015/2/1/152_semko_access.pdf.

165. Семко В.В. Метод побудови областей керованих станів динамічного об’єкту / В.В.Семко // Математичні машини і системи. - 2015. - Вип. № 2. - С.44-52.

166. Семко В.В. Побудова областей керованих станів динамічного об’єкта / В.В.Семко, В.І.Чепіженко // Телекомунікаційні та інформаційні технології. - 2015. - Вип. № 3. - С.25-30.

167. Семко В.В. Модель системи виявлення вторгнень з використанням двоступеневого критерію виявлення мережних аномалій / В.В.Семко, Г.М.Гулак, П.М.Складаний // Сучасний захист інформації. - 2015. - Вип. № 4. - С.81-85.

168. Семко В.В. Концепція топологічного ситуаційного аналізу та синтезу в “малому” управління об'єктом в умовах конфлікту, невизначеності поведінки та варіативної множини об’єктів спостереження / В.В.Семко // Актуальні проблеми забезпечення інформаційної безпеки держави. Матеріали науково-технічної конференції студентів, аспірантів, викладачів та науковців: 18 грудня 2014 року. - Київ: Державний університет телекомунікацій, - 2014. - С.90-91.

169. Семко В.В. Модель конфликта взаимодействия объектов в кибернетическом пространстве / В.В.Семко // Комп'ютерні системи та мережні технології. Тези доповідей V міжнародної науково-технічної конференції. - Київ: НАУ, 2012.- С.116.

170. Семко В.В. Інформаційний та кіберпростори: проблеми безпеки, методи та засоби боротьби / В.Л.Бурячок, В.В.Семко, П.М.Складанний, Н.В. Лукова-Чуйко – К.: ДУТ - КНУ, 2016. – 178с. (з них власних 32 с.) Гриф надано Міністерством освіти і науки України згідно з листом №1/11-8662 від 22.06.2015р.

171. Семко В.В. Топологічний ситуаційний аналіз та синтез стратегій управління об'єктом в умовах конфлікту, невизначеності поведінки та варіативної множини об'єктів спостереження / В.В.Семко // ScienceRise. - 2016. - № 9/2(26). - С.62-69.

172. Семко В.В. Квазілінійна система інтелектуального управління конфліктом / В.В.Семко // Зв'язок. - 2016. - № 5. - С.68-71.

173. Семко В.В. Алгебраїчний підхід до рішення конфлікту за методом інтегрального усікання варіантів / В.В.Семко // Сучасний захист інформації. - 2016. - № 3. - С.4-10.

174. Семко В.В. Модель функціонування системи інтелектуального управління об'єктом / В.В.Семко, В.Л.Бурячок // Наукові записки Українського науково-дослідного інституту зв'язку. - 2016. - № 3(43). - С.21-29.

175. Семко В.В. Технологія проведення порівняльного аналізу та оцінювання стану захищеності автоматизованих інформаційних систем / Л.В.Бурячок, В.Л.Бурячок, В.В.Семко // Сучасний захист інформації. - 2016. - № 4. - С.16-24.

176. Словарь по кибернетике / под ред. В.М.Глушкова. - К:Главная редакция Украинской Советской энциклопедии, 1979. - 423с.

177. Соболь И.М., Статников Р.Б. Выбор оптимальных параметров в задачах со многими критериями / И.М.Соболь, Р.Б.Статников. - М.: Наука, 1981. - 110с.

178. Субботин А.И., Ченцов А.Г. Оптимизация гарантии в задачах управления / А.И.Субботин, А.Г.Ченцов. М.: Наука. 1981.

179. Субботин А.И. Минимаксные неравенства и уравнения Гамильтона-Якоби. М. Наука. 1991.

180. Суздаль В.Г. Теория игр для флота / В.Г.Суздаль. - М.: Воениздат, 1976. - 317с.

181. Теория эргатических систем / под ред. В.В.Палова. - К.: Вища школа, 1977. - 344с.

182. Теория иерархических многоуровневых систем / под ред. М.Мессаровича. - М.: Мир, 1973. - 344с.

183. Теория прогнозирования и принятия решений / под ред. С.А.Саркисяна. - М.: Высшая школа, 1977. - 351с.

184. Теория систем. Математические методы и моделирование. - М.: Мир, 1989. - 384с.

185. Тей А. Логический подход к искусственному интеллекту: от классической логики к логическому программированию / А.Тей, П.Грибомон, Ж.Луи, Д.Спийерс, П.Водон, П.Гоше, Э.Грегуар, Э.Санлес, Ф.Дельсарт. - М.: Мир, 1990. - 432с.

186. Технические эргатические системы / под ред. В.В.Павлова. - К.: Вища школа, 1977. - 344с.

187. Трауб Дж. Информация, неопределенность, сложность / Дж.Трауб, Г.Васильковский, Х.Вожьняковский - М.: Мир, 1988. - 184с.

188. Трифонов А.Г. Постановка задачи оптимизации и численные методы ее решения / А.Г.Трифонов. - М.: Дело, 2003. - 198с.

189. Ту Дж. Принципы распознавания образов / Дж.Ту, Р.Гонсалес. - М.: Мир, 1978. - 411с.

190. Уинстон П. Искусственный интеллект / П.Уинстон. - М.: Мир, 1980. - 519с.

191. Уотермен Д. Руководство по экспертным системам / Д.Уотермен. - М.: Мир, 1989. - 388с.

192. Уэно Х. Представление и использование знаний / Х.Уэно, М.Исидзука. - М.: Мир. - 220с.

193. Финкельштейн Ю.Ю. Приближенные методы и прикладные задачи дискретного программирования / Ю.Ю.Финкельштейн. - М.: Наука, 1976. - 264с.

194. Хедли Д. Нелинейное и динамическое программирование / Д.Хедли. - М.: Мир, 1967. - 506с.

195. Хеллман О. Введение в теорию оптимального поиска / О.Хеллман. - М.: Наука, 1985. - 248с.

196. Ховард Р. Динамическое программирование и марковские процессы / Р.Ховард. - М.: Сов. радио, 1964. - 188с.

197. Хоменюк В.В. Элементы теории многоцелевой оптимизации / В.В.Хоменюк. - М.: Наука, 1983. - 128с.

198. Цаленко М.Ш. Моделирование семантики в базах данных / М.Ш.Цаленко. - М.: Наука, 1989. - 288с.

199. Цыпкин Я.З. Основы теории автоматических систем / Я.З.Цыпкин. - М.: Наука, 1977. - 560с.

200. Чень Ч. Математическая логика и автоматическое доказательство теорем / Ч.Чень, Р.Ли. - М.: Наука, 1983. - 360с.

201. Черноусько Ф.Л. Оценивание фазового состояния динамических систем. Метод эллипсоидов / Ф.Л.Черноусько. - М.: Наука, 1988. - 320с.

202. Черноусько Ф.Л. Вариационные задачи механики и управления (Численные методы) / Ф.Л.Черноусько, Н.В.Баничук. - М.: Наука, 1973. - 240с.

203. Черноусько Ф.Л. Игровые задачи управления и поска / Ф.Л.Черноусько, А.А.Меликян. - М.: Наука, 1978. - 270с.

204. Чикрий А.А. Конфликтно управляемые процессы / А.А.Чикрий. - К.: Наук. думка, 1992. - 382с.

205. Шеннон К. Работы по теории информации и кибернетике / К.Шеннон. - М.: ИЛ, 1963. - 829с.

206. Шеридан Т.Б. Системы человек-машина / Т.Б.Шеридан, У.Р.Феррелл. - М.:Машиностроение, 1980. - 400с.

207. Шоломов Л.А. Логические методы исследования дискретных моделей выбора / Л.А.Шоломов. - М.: Наука, 1989. - 288с.

208. Шнейдерман Б. Психология программирования: Человеческие факторы в вычислительных и информационных системах / Б.Шнейдерман. - М.: Радио и связь, 1984. - 304с.

209. Шор Н.З. Методы минимизации недифференцируемых функций и их приложения / Н.З.Шор. - Киев: Наукова думка, 1979. - 199с.

210. Шредингер Э. К принципу неопределенностей Гейзенберга // Избранные труды по квантовой механике - М.: Наука, 1976. - С.210-217.

211. Штейнгауз Г.Д. Математический калейдоскоп / Г. Д. Штейнгауз. - М.: Государственное издательство технико-теоретической литературы, 1949. – 140с.

212. Энгелькинг Р. Общая топология / Р.Энгелькинг. - М.: Мир, 1986. - 752с.

213. Эндрю А. Искусственный интеллект / А.Эндрю. - М.: Мир, 1985. - 264с.

214. Эшби У.Р. Введение в кибернетику / У.Р.Эшби. - М.: Изд-во иностр. лит., 1959. - 432с.

215. Эшби У.Р. Конструкция мозга / У.Р.Эшби. - М.: Мир, 1964. - 412с.

216. Юдин Д.Б. Математические методы управления в условиях неполной информации / Д.Б.Юдин. - М.: Сов. радио, 1974. - 400с.

217. Akgul M. Topics in relaxation and elepsoidal methods. - Pitman publishing inc., 1984. - 322p.

218. Edmonds J., Fulkerson D.R. Bottleneck extrema. - J.Combinatorial theory, 1970, 8, N8,p.299-306.

219. Fulkerson D.R. Blocking and antiblocking pairs of polyhedra. - Math. programming, 1971, 1, N2, p.168-194.

220. Fulkerson D.R., Weinberger D.B. Blocking pairs of polyhedra arising from networks flows. - J. Combinatorial theory, 1975, 18, N2, p.265-283.

221. Held M., Wolfe P., Crowder H. Validation of subgradient optimization. - Math. Programming, 1974, 6, p.62-88.

222. Lenmarehal C. Numerical experiments in nonsmooth optimization& In; progress in nondifferentiable optimization./ Ed. E.A.Nurminsky. p.58-82, International Institute for
applied system analysts: Zaxenburg, Austria, 1982, p.61-84.

223. Miller C.A. Aircraft collision avoidance: requirements and consepts. Anachaim, Conference records, 1980,p.p.1-7.

224. Miloh T., Sharma S.D. Maritime Collision Avoidance as a Differential Game. - Proccedings 4-th Ship Control Systems Symposium& Washington, 1975, October 27-31, v.2, p.109-128.

225. Orlando V.A., Dronilhet P.R. Dicret address beacon system functional description. Report FAA-RD-80-41. Federal Aviation Administration, Washington, D.C. April 1980.

226. Scolnich M.I. Radar handbook. Mc. Graw-Hill book company, New York, 1970.

227. Smale S. On the average speed of the simplex methode of liner programming. - Math. Programming, 1983, v.27, N3, p.241- 262.

228. Tind J. Blocking and antiblocking polyhedra. - Annals of discrete mathematics, 1979, 4, p.159-174.

229. Zadeh N. A bad network problem for the simplex method and other minimum cost flow algoritms. - Math. Programming, 1973, 5, N3, p.255-266.

230. Modern Control Systems. Twelfth edition / Richard C. Dorf, Robert H. Bishop – Prentice Hall, 2008. - 1140 pp.

231. Schriften zur Informations-Energetik 1 / Reinhard R. Kocher – VDM 2009, 1.Auflage, S. 23-28.

232. The Computer Contradictionary. Twelfth edition / Stan Kelly-Bootle — MIT Press, 1995. — 260 p.

Додаток А

Акти впровадження результатів дисертаційної роботи

ЗАТВЕРДЖУЮ

Перший заступник Голови Державного агентства з питань електронного урядування України

_____ Т.І. Олійник

"05" березня 2015 року

АКТ

про впровадження результатів дисертаційного дослідження

Семка Віктора Володимировича

на тему "Синтез стратегій оптимального управління об'єктом в умовах конфлікту, обмежень та невизначеності"

на здобуття наукового ступеня доктор технічних наук

за спеціальністю 05.13.06 - інформаційні технології

При впровадженні системи електронної взаємодії органів виконавчої влади України було використано та впроваджено наукові результати та інженерні рішення дисертаційного дослідження Семка Віктора Володимировича, а саме:

1. Методи і моделі опису взаємодії об'єктів в інформаційному просторі системи електронної взаємодії.

2. Метод синтезу оптимальних рішень при управлінні інформаційними потоками.

3. Методи і моделі синтезу стратегій управління при виникненні стану інформаційного конфлікту.

4. Методи і моделі оцінювання ризиків взаємодії об`єктів в інформаційному просторі.

5. Рекомендації щодо побудови системи безпечного функціонування інформаційно-телекомунікаційної системи електронної взаємодії.

Отримані автором результати дозволяють підвищити ефективність інформаційно-аналітичного процесу на основі використання методу синтезу рішень в інформаційному просторі за умов невизначеності та відкритої множини об'єктів інформаційної взаємодії.

Генеральний директор
ДП "Держінформресурс" Тертичний О.С.

ЗАТВЕРДЖУЮ
ТВП начальника Української
військово-медичної академії
доктор медичних наук, професор

Рум'янцев Ю.В
09 2011р.

АКТ

про використання результатів дисертаційної роботи докторанта Інституту телекомунікацій і глобального інформаційного простору НАН України Семка Віктора Володимировича

В процесі досліджень показників психофізичного стану контингенту слухачів Військово-медичної академії Збройних сил України з використанням апаратно-програмних комплексів вимірювання та оцінювання електричних сигналів людського організму були використані результати дисертаційної роботи Семка Віктора Володимировича, а саме:

1. Алгоритми та програмне забезпечення введення та оцінювання параметрів, за якими визначаються психофізіологічні резерви людини-оператора ергатичної системи, функціонуючої в кібернетичному просторі.
2. Методи та алгоритми розрахунку та формування простору станів ергатичного організму в кібернетичному та фізичному просторах.
3. Спосіб та алгоритми розрахунку і формування простору гарантованих рішень при запобіганні конфлікту взаємодії ергатичних організмів.
4. Спосіб, алгоритми та програмне забезпечення розпізнавання образів та виділення простих аналогій при оцінюванні динаміки перетворення простору станів ергатичного організму.

Створені автором способи і алгоритми можуть ефективно використовуватись при синтезі стратегій вирішення та запобігання конфлікту взаємодії ергатичних організмів в фізичному і віртуальному просторі.

Відповідальний за впровадження:
Начальник кафедри авіаційної, морської медицини і психофізіології
Української військово-медичної академії
доктор медичних наук, професор Єна А.І.

Національне космічне агентство України
Національний центр управління та випробувань космічних засобів
ГОЛОВНИЙ ЦЕНТР СПЕЦІАЛЬНОГО КОНТРОЛЮ

ЗАТВЕРДЖУЮ
Начальник Головного центру
спеціального контролю
І.В. Толчонов
15 лютого 2011 року

АКТ

про використання результатів дисертації докторанта Інституту телекомунікацій і глобального інформаційного простору НАН України Семка Віктора Володимировича

При виконанні дослідно-конструкторських робіт "Розробка технічного завдання на створення системи технічного захисту інформації в автоматизованій системі обробки інформації Головного центру спеціального контролю (ГЦСК) НКАУ" та "Розробка заходів технічного захисту інформації автоматизованої системи збору та обробки геофізичної інформації станції PS-45" в Головному центрі спеціального контролю Національного космічного агентства України було використано та впроваджено наукові результати та інженерні рішення дисертаційного дослідження Семка Віктора Володимировича, а саме:

1. Методи і моделі опису конфлікту взаємодії ергатичних організмів в інформаційному просторі.
2. Методи формального опису простору припустимих станів при оптимальному управлінні об'єктами в інформаційному просторі.
3. Методи і моделі синтезу стратегій запобігання конфліктів при виникненні стану інформаційного протиборства.
4. Методи і моделі оцінювання ризиків взаємодії об'єктів в інформаційному просторі.
5. Рекомендації щодо побудови системи безпечного функціонування інформаційно-телекомунікаційної системи Головного центру спеціального контролю Національного космічного агентства України.

Отримані автором результати мають практичний інтерес при створенні розподілених інформаційно-телекомунікаційних систем, функціонуючих в недетермінованих умовах інформаційного простору.

ВО начальника відділу інформаційно-обчислювального
О.Є. Іващенко

ЗАТВЕРДЖУЮ
Директор Державного підприємства "Український медичний центр безпеки дорожнього руху та інформаційних технологій" Міністерства охорони здоров'я України

Ю.О.Чорний
" 19 " грудня 2014 року

АКТ
впровадження (використання) результатів дисертаційного дослідження
Семка Віктора Володимировича
на тему "Синтез стратегій оптимального управління об'єктом в умовах конфлікту, обмежень та невизначеності"
на здобуття наукового ступеня доктор технічних наук
за спеціальністю 05.13.06 - інформаційні технології

При виконанні дослідно-конструкторських робіт "Ведення бази даних медичних комісій з огляду кандидатів у водії та водіїв транспортних засобів та виданих ними медичних довідок щодо придатності до керування транспортним засобом" (Шифр "Медіана")" та "Ведення бази даних медичних працівників та бази даних щодо проходження водіями транспортних засобів передрейсових та післярейсових медичних оглядів" (шифр "Огляд") було використано та впроваджено наукові результати та інженерні рішення дисертаційного дослідження Семка Віктора Володимировича, а саме:

1. Методи і моделі опису взаємодії об'єктів та суб'єктів інформаційного простору ведення баз даних медичного призначення.
2. Методи і моделі синтезу стратегій запобігання конфліктів взаємодії об'єктів та суб'єктів інформаційного простору ведення баз даних медичного призначення при виникненні стану інформаційного протиборства.
3. Методи і моделі оцінювання ризиків взаємодії об'єктів та суб'єктів в інформаційному просторі медичного призначення.
4. Топологічні рішення щодо створення єдиного медичного простору ДП "Український медичний центр безпеки дорожнього руху та інформаційних технологій"Міністерства охорони здоров'я України.

Отримані автором результати підвищують наукову та практичну цінність проектування, створення, впровадження та експлуатації для різного рівня складності сучасних комп'ютерних мереж та інформаційно-телекомунікаційних систем медичного призначення.

Заступник директора
ДП "УМЦБДР та ІТ" МОЗ України Г.І. Шлома

ЗАТВЕРДЖУЮ
Комерційний директор
ТОВ «БМС Консалтінг»
_____ М.О. Тищенко
"11" лютого 2016 року

АКТ
про впровадження результатів дисертаційного дослідження
Семка Віктора Володимировича
на здобуття наукового ступеня доктор технічних наук
за спеціальністю 05.13.06 – інформаційні технології

Результати досліджень Семка Віктора Володимировича, виконаних ним у рамках дисертаційної роботи на здобуття наукового ступеня доктора технічних наук, зокрема такі наукові положення та інженерні рішення, як:

1) концепція та методи інтегрально-топологічного аналізу та синтезу простору гарантованого рішення задачі конфлікту в кібернетичному просторі:

2) теоретико-множинна модель опису простору гарантованого рішення задачі конфлікту взаємодії недетермінованих об'єктів кібернетичного простору в умовах невизначеності їх поведінки;

3) метод розрахунку поточних характеристик простору безконфліктного переміщення об'єкту управління в довільну точку простору спостереження при варіативній множини об'єктів спостереження;

4) метод інтегрального усікання варіантів і алгоритм його реалізації, як інструмент вирішення задачі синтезу стратегій ситуаційного управління об'єктом в ресурсному неопуклому небезперевному просторі рішень;

5) досліджено вплив обмежень на структуризацію простору рішень об'єкту управління та введено інтегроване уявлення його відображення на простір гарантованого повністю керованого стану.

6) метод виміру області гарантованого управління щодо реалізації стратегій управління об'єктом при його переміщенні в обрану точку простору спостережності.

7) структура системи ситуаційного інтелектуального управління, яка використовує автоматні граматики при синтезі управління;

8) імітаційна модель алгоритму функціонування системи синтезу стратегій ситуаційного управління об'єктом при вирішенні задачі його переміщенні в обрану точку кібернетичного простору спостережень за наявності довільної варіативної множини нетермінованих об'єктів спостереження, -

використані у роботі фахівцями компанії «БМС Консалтінг» при розробці і реалізації проекту з доопрацювання складових компонентів інформаційно-телекомунікаційної системи електронної взаємодії органів виконавчої влади (шифр НДР «ДЦІР5-1», державний номер реєстрації 0115U006734), що забезпечуватиме автоматизацію процесів створення, відправлення, передавання тощо електронних документів та копій паперових документів в електронному вигляді.

Отримані автором результати дозволяють підвищити ефективність інформаційно-аналітичного процесу системи електронного врядування держави.

Відповідальний виконавець Стасюк О.В.

Association YUG, Ltd.
04073, 21, av.Moskowskii, Kyiv, Ukraine
phone/fax +380-44-5024472 / 73
E-mail: info@yug.com.ua
http://www.yug.com.ua

ОБ'ЄДНАННЯ ЮГ
Інформаційна безпека

ТОВ «Об'єднання ЮГ»
Україна, 04073, м.Київ, Московський пр-т, 21
тел/факс +380-44-5024472 / 73
E-mail: info@yug.com.ua
http://www.yug.com.ua

ЗАТВЕРДЖУЮ
Генеральний директор
ТОВ "Об'єднання ЮГ"
_____ В.О.Поляков
"05" липня 2016 року

АКТ
про впровадження результатів дисертаційного дослідження Семка Віктора Володимировича

на здобуття наукового ступеня доктора технічних наук
за спеціальністю 05.13.06 – інформаційні технології

Результати досліджень Семка Віктора Володимировича, що виконані ним у рамках дисертаційної роботи на здобуття наукового ступеня доктора технічних наук, зокрема, такі наукові положення та інженерні рішення, як:

1) концепція та методи інтегрально-топологічного аналізу стану взаємодії об'єктів в кібернетичному просторі в умовах конфлікту, довільних обмежень та невизначеності;
2) теоретико-множинна формальна модель взаємодії варіативної множини об'єктів та суб'єктів автоматизованої системи класу 3 щодо виявлення Z-систем несанкціонованого доступу до інформаційних ресурсів в межах, які визначені матрицею доступу, правилами та відношеннями;
3) інформаційна технологія попередження та рішення конфлікту при виявленні несанкціонованої активності в інформаційно-телекомунікаційній системі, яка взаємодіє з мережею *Internet*;
4) методологія та інформаційна технологія аналізу стійкості та передатних характеристик системи інтелектуального управління комплексу засобів захисту системи захисту інформації інформаційно-телекомунікаційної системи при несанкціонованій мережевій активності, -

використані у роботі фахівцями Товариства з обмеженою відповідальністю "Об'єднання ЮГ" при розробці і реалізації перспективних систем захисту інформації від несанкціонованого доступу в автоматизованих системах класу 3.

Отримані автором результати дозволяють підвищити ефективність застосування систем захисту інформації від несанкціонованого доступу, надійність та якість інформаційно-аналітичного процесу в інформаційно-телекомунікаційних системах.

Начальник відділу захисту інформації ***Снопченко Д.В.***

Додаток Б

Копі

Копії свідоцтв про авторське право

УКРАЇНА

Міністерство освіти і науки України

Державний департамент інтелектуальної власності

СВІДОЦТВО

про реєстрацію авторського права на твір

№ 26411

Комп'ютерна програма "Психодіагностичні методики" до тематичного блоку "Пізнай себе" Програмно-апаратного комплексу "Профорієнтаційний термінал"

(вид, назва службового твору)

Автор(и) **Семко Віктор Володимирович, Михайловський Олег Олегович**

(повне ім'я, псевдонім (за наявності))

Авторські майнові права належать **Товариство з обмеженою відповідальністю "Елан", вул. Заньковецької, 6, кв. 43, 01001**

(повне ім'я фізичної та/або повне офіційне найменування юридичної особи, адреса)

Дата реєстрації 11.11.2008

Голова Державного департаменту інтелектуальної власності М.В.Паладій

М.П.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ ДЕПАРТАМЕНТ ІНТЕЛЕКТУАЛЬНОЇ ВЛАСНОСТІ

Україна, МСП 03680, Київ-35, вул. Урицького, 45
Тел.: (044) 494-06-06 Факс: (044) 494-06-67

РІШЕННЯ

ПРО РЕЄСТРАЦІЮ АВТОРСЬКОГО ПРАВА НА ТВІР

Державний департамент інтелектуальної власності розглянув заяву
Товариство з обмеженою відповідальністю "Елан", вул. Заньковецької, 6, кв. 43, 01001
(повне ім'я фізичної або повне офіційне найменування юридичної особи, адреса)

заявка від **27.10.2008** **№ 27406**

про реєстрацію авторського права на твір і прийняв рішення зареєструвати авторське право на службовий твір **Комп'ютерна програма "Психодіагностичні методики" до тематичного блоку "Пізнай себе" Програмно-апаратного комплексу "Профорієнтаційний термінал"; Семко Віктор Володимирович, Михайловський Олег Олегович; Товариство з обмеженою відповідальністю "Елан"**
(вид, повна, скорочена (за наявності) назва твору, повне ім'я, псевдонім (за наявності) автора (ів), повна офіційна назва роботодавця)

Внесення відомостей до Державного реєстру свідоцтв про реєстрацію авторського права на твір та видача свідоцтва будуть здійснені за умови сплати збору за оформлення і видачу свідоцтва про реєстрацію авторського права на твір відповідно до п.3 постанови Кабінету Міністрів України від 27 грудня 2001 року № 1756 "Про державну реєстрацію авторського права і договорів, які стосуються права на твір".

Якщо протягом трьох місяців від дати одержання заявником рішення про реєстрацію авторського права на твір Державний департамент не одержав документ про сплату збору за оформлення і видачу свідоцтва у розмірі та порядку, визначених законодавством, або копію документа, що підтверджує право на звільнення від сплати зазначеного збору, заявка вважається відхиленою і реєстрація авторського права та публікація відомостей про реєстрацію Державним департаментом не проводиться.

Голова Державного департаменту інтелектуальної власності **М.В. Паладій**

Міністерство освіти і науки України * Державний департамент інтелектуальної власності * Ідентифікаційний код 21716978

УКРАЇНА
Міністерство освіти і науки України
Державний департамент інтелектуальної власності

СВІДОЦТВО

про реєстрацію авторського права на твір

№ 39113

Комп'ютерна програма "Крос-платформена клієнт-серверна технологія захищеного обміну електронними даними з використанням терміналів мобільного телефонного зв'язку" ("ТЗОД")

(вид, назва службового твору)

Автор(и) **Семко Віктор Володимирович, Михайловський Олег Олегович, Гіденко Олексій Володимирович, Зінов'єв Олексій Вячеславович, Мельник Віра Трохимівна, Барткова Валентина Андріївна**

(повне ім'я, псевдонім (за наявності))

Авторські майнові права належать **Товариство з обмеженою відповідальністю "Елан", вул. Заньковецької, 6, кв. 43, 01001**

(повне ім'я фізичної та/або повне офіційне найменування юридичної особи, адреса)

Дата реєстрації 11.07.2011

Голова Державного департаменту інтелектуальної власності М.В.Паладій

Ідентифікаційний код 21716978

М.П.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ ДЕПАРТАМЕНТ ІНТЕЛЕКТУАЛЬНОЇ ВЛАСНОСТІ

Україна, МСП 03680, Київ-35, вул. Урицького, 45
Тел.: (044) 494-06-06 Факс: (044) 494-06-67

РІШЕННЯ

ПРО РЕЄСТРАЦІЮ АВТОРСЬКОГО ПРАВА НА ТВІР

Державний департамент інтелектуальної власності розглянув заяву
Товариство з обмеженою відповідальністю "Елан", вул. Заньковецької, 6, кв. 43, 01001
(повне ім'я фізичної або повне офіційне найменування юридичної особи, адреса)

заявка від **20.05.2011** **№ 39558**

про реєстрацію авторського права на твір і прийняв рішення зареєструвати авторське право на службовий твір **Комп'ютерна програма "Крос-платформена клієнт-серверна технологія захищеного обміну електронними даними з використанням терміналів мобільного телефонного зв'язку" ("ТЗОД"); Семко Віктор Володимирович, Михайловський Олег Олегович, Гіденко Олексій Володимирович, Зінов'єв Олексій Вячеславович, Мельник Віра Трохимівна, Барткова Валентина Андріївна; Товариство з обмеженою відповідальністю "Елан"**
(вид, повна, скорочена (за наявності) назва твору, повне ім'я, псевдонім (за наявності) автора (ів), повна офіційна назва роботодавця)

Внесення відомостей до Державного реєстру свідоцтв про реєстрацію авторського права на твір та видача свідоцтва будуть здійснені за умови сплати збору за оформлення і видачу свідоцтва про реєстрацію авторського права на твір відповідно до п.3 постанови Кабінету Міністрів України від 27 грудня 2001 року № 1756 "Про державну реєстрацію авторського права і договорів, які стосуються права на твір".

Якщо протягом трьох місяців від дати одержання заявником рішення про реєстрацію авторського права на твір Державний департамент не одержав документ про сплату збору за оформлення і видачу свідоцтва у розмірі та порядку, визначених законодавством, або копію документа, що підтверджує право на звільнення від сплати зазначеного збору, заявка вважається відхиленою і реєстрація авторського права та публікація відомостей про реєстрацію Державним департаментом не проводиться.

Голова Державного департаменту інтелектуальної власності **М.В. Паладій**

УКРАЇНА

Міністерство освіти і науки України

Державний департамент інтелектуальної власності

СВІДОЦТВО

про реєстрацію авторського права на твір

№ 25553

Комп'ютерна програма "Програма для ПЕОМ щодо ведення бази даних дозволів на викиди забруднюючих речовин у атмосферне повітря та надання інформації громадськості" ("Дозвіл")

(вид, назва службового твору)

Автор(и) **Семко Віктор Володимирович, Михайловський Олег Олегович**

(повне ім'я, псевдонім (за наявності))

Авторські майнові права належать **Товариство з обмеженою відповідальністю "Елан", вул. Заньковецької, 6, кв. 43, 01001**

(повне ім'я фізичної та/або повне офіційне найменування юридичної особи, адреса)

04.09.2008

Дата реєстрації

Голова Державного департаменту інтелектуальної власності М.В.Паладій

Ідентифікаційний код 21616178

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ ДЕПАРТАМЕНТ ІНТЕЛЕКТУАЛЬНОЇ ВЛАСНОСТІ

Україна, МСП 03680, Київ-35, вул. Урицького, 45
Тел.: (044) 494-06-06 Факс: (044) 494-06-67

РІШЕННЯ

ПРО РЕЄСТРАЦІЮ АВТОРСЬКОГО ПРАВА НА ТВІР

Державний департамент інтелектуальної власності розглянув заяву
Товариство з обмеженою відповідальністю "Елан", вул. Заньковецької, 6, кв. 43, 01001
(повне ім'я фізичної або повне офіційне найменування юридичної особи, адреса)

заявка від **22.08.2008** **№ 26809**

про реєстрацію авторського права на твір і прийняв рішення зареєструвати авторське право на службовий твір **Комп'ютерна програма "Програма для ПЕОМ щодо ведення бази даних дозволів на викиди забруднюючих речовин у атмосферне повітря та надання інформації громадськості" ("Дозвіл"); Семко Віктор Володимирович, Михайловський Олег Олегович; Товариство з обмеженою відповідальністю "Елан"**
(вид, повна, скорочена (за наявності) назва твору, повне ім'я, псевдонім (за наявності) автора (ів), повна офіційна назва роботодавця)

Внесення відомостей до Державного реєстру свідоцтв про реєстрацію авторського права на твір та видача свідоцтва будуть здійснені за умови сплати збору за оформлення і видачу свідоцтва про реєстрацію авторського права на твір відповідно до п.3 постанови Кабінету Міністрів України від 27 грудня 2001 року № 1756 "Про державну реєстрацію авторського права і договорів, які стосуються права на твір".

Якщо протягом трьох місяців від дати одержання заявником рішення про реєстрацію авторського права на твір Державний департамент не одержав документ про сплату збору за оформлення і видачу свідоцтва у розмірі та порядку, визначених законодавством, або копію документа, що підтверджує право на звільнення від сплати зазначеного збору, заявка вважається відхиленою і реєстрація авторського права та публікація відомостей про реєстрацію Державним департаментом не проводиться.

Голова Державного департаменту інтелектуальної власності **М.В. Паладій**

Міністерство освіти і науки України Державний департамент інтелектуальної власності
Ідентифікаційний код 21716978

Додаток В. Алгоритми і програмні компоненти розрахунку інформаційних множин опису стану об'єкту управління

В.1 - алгоритми програмних компонент розрахунку інформаційних множин опису стану об'єкту управління

Алгоритми створені та реалізовані в програмному засобі імітаційної моделі згідно співвідношеннями, які визначають модель взаємодії ОУ з варіативною множиною ОС в умовах конфлікту, обмежень і невизначеностей.

Алгоритм розрахунку та побудови ПКУО наведено на рис. 1.

Рис. 1. Схема алгоритму розрахунку ПКУО

Виходячи з сучасних технологій програмування, алгоритм передбачає наявність інформації щодо ОС, ОУ, простору Q, мети переміщення ОУ, границі простору $\Gamma_{гр}(Q)$, значень припустимого зближення ОУ з ОС, тощо. Таким чином, при роботі алгоритму не передбачається введення похідних даних для розрахунків.

Слід зазначити, що вибір мови програмування для реалізації алгоритмів синтезу та вибору рішень конфлікту пов'язане з можливістю використання технологій паралельних обчислень, що дозволяє оптимально використовувати можливості обчислювального середовища ІП. Для використання засобів розпаралелювання обчислень та балансування навантаження обчислювального середовища ІП СІУ використовується програмний акселератор середовища розробки MS Visual Studio 2010/2012, який орієнтований на використання мови програмування середовища розробки PGI Fortran та MS Visual Studio C/C++.

Робота алгоритму розрахунку ПКУО (рис. 1) починається з обчислення відстані до ОС (блок 2).

В разі знаходження ОС в зоні надзвичайного зближення на відстані, яка не менша припустимої для зближення ОУ та ОС, вирішується задача оцінки місця знаходження ОС та можливості зміни параметрів переміщення ОУ (нарпрямку та швидкості) за алгоритмом, який наведено на рис. 2.

В блоці 3 здійснюється аналіз та виділення ОС, з якими є можливим стан конфлікту взаємодії при переміщенні в ПСП.

Аналіз приналежності ОС простору Q здійснюється в блоці 4.

В блоці 5 забезпечується розрахунок параметрів, що необхідні для обчислення ПКУО, а саме - діапазону можливих напрямків переміщення ОУ, кроку зміни напрямків переміщення, що аналізуються, діапазону можливих значень швидкості переміщення ОУ, кроку зміни значення швидкості переміщення ОУ.

В разі знаходження ОС в зоні надзвичайного зближення на відстані, яка не менша припустимої для зближення ОУ та ОС, вирішується задача оцінки місця знаходження ОС та можливості зміни його парараметрів переміщення (нарпрямку та швидкості) за алгоритмом, наведеним на рис. 2.

В блоці 3 здійснюється аналіз та виділення ОС, з якими є можливим стан конфлікту взаємодії при переміщенні в ПСП.

Аналіз приналежності ОС простору Q бзійснюється в блоці 4.

В блоці 6 здійснюються необхідні розрахунки з використанням алгоритму побудови ЛС для кожного ОС та розрахунку вихідної множини, яка

утворюється при перетині ПКУО за окремими ОС за алгоритмом, наведеним на рис. 3.

Відображення розрахованої множини ПКУО здійснюється в блоці 7.

Алгоритм розрахунку ПКУО реалізовано засобами алгоритмічної мови стандарту Fortran 2003 середовища розробки MS Visual Studio 2010 з інтегрованими засобами PGI Fortran 12.10 та акселератором.

Програмний модуль розрахунку ПКУО оформлено у вигляді підпрограми (subroutine komby ()), текст якої наведено в Додатку В.2.

Розглянемо алгоритм оцінки місця знаходження ОС та можливості зміни параметрів його переміщення (рис. 2) згідно блоку 2 алгоритму, який наведено на рис. 1.

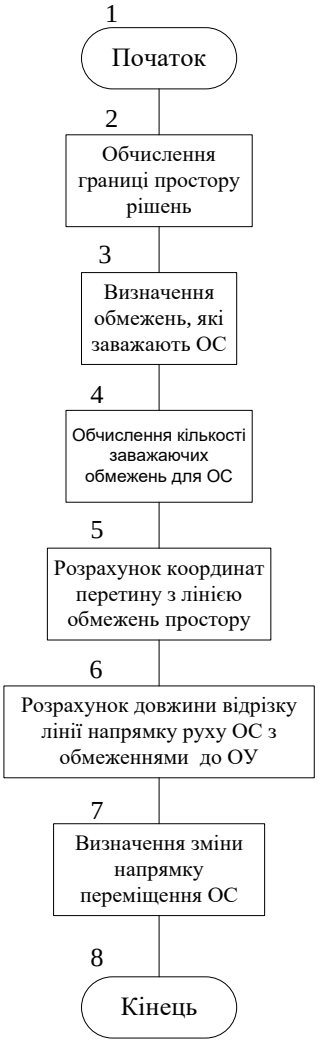


Рис. 2. Схема алгоритму розрахунку ознаки зміни напрямку переміщення ОС

В блоці 2, використовуючі наявні дані про обмеження простору рішень (перерізу), формується замкнена лінія границі за правилом замикання "початок - початок", "кінець - кінець".

Переходячи в зв'язану з ОУ системи координат, в блоці 3 визначається множина точок перетину лінії руху ОС при збереженні напрямку переміщення ОУ та ОС з лінією границі простору рішень.

За результатами обчислень блоку 3 в блоці 4 визначається кількість перетинів з границею простору, координати точок перетину (блок 5) та відстань від точки перетину до вершини ламаної лінії границі обмежень (блок 6).

В блоці 7 визначається ознака зміни напрямку руху ОС для переміщення в межах границі простору рішень, визначаючи додатній чи від'ємний знак зміни напрямку переміщення.

В Додатку В.2 наведено текст програмного коду, який реалізує алгоритм, який наведено на рис. 2 (subroutine pnt()).

Розглянемо алгоритм побудови ЛШ при взаємодії ОУ з ОС (рис. 3) згідно блоку 6 алгоритму, який наведено на рис. 1.

В блоці 2 здійснюється підрахунок кількості ОС в просторі рішень.

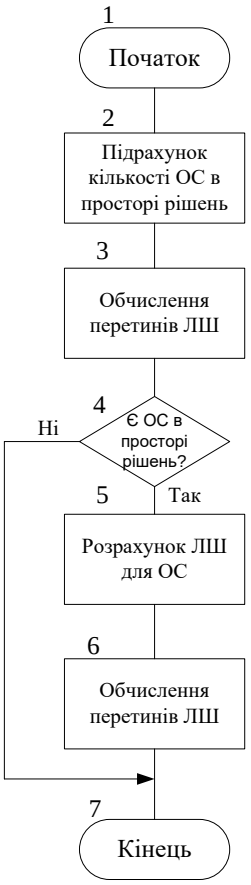


Рис. 3. Схема алгоритму побудови ЛШ при взаємодії ОУ з ОС

Формування вільного від конфліктів ввзаємодії ОУ з ОС ПКУО здійснюється в блоці 3.

В разі відсутності ОС в просторі Q (блок 4) ПКУО визначається без обмежень і управління передається блоку 7, тобто функціонування програмного модуля завершується.

При наявності в просторі Q ОС (блок 3) управління передається блоку 5, в якому викликається процедура (підпрограма subroutine stick()), яка здійснює розрахунок ЛШ для кожного ОС в просторі Q. Схема алгоритму розрахунку ЛШ наведена на рис. 4.

Після розрахунку ЛШ формується інформаційний образ ПКУО (блок 6), в якому здійснюється формування перетинів ЛШ для ОС (subroutine aria()), схема алгоритму якої наведена на рис. 5.

В Додатку В.2 наведено текст програмного коду, який реалізує алгоритм, який наведено на рис. 5 (subroutine cross1()).

Розглянемо алгоритм побудови ЛШ при взаємодії ОУ з ОС (рис. 4) згідно блоку 5 алгоритму, який наведено на рис. 3.

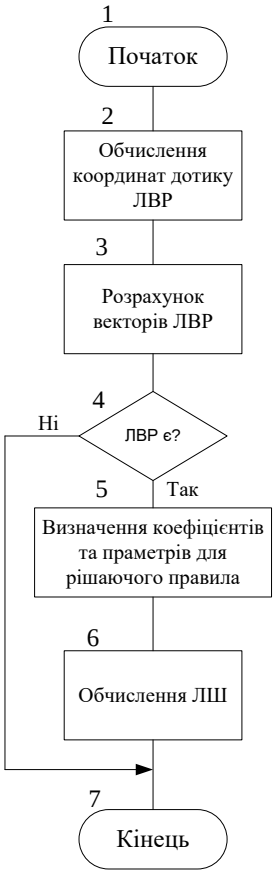


Рис. 4. Схема алгоритму розрахунку ЛШ

В блоці 2 (рис. 4) здійснюється розрахунок координат точок дотику ЛВР засобами програмного модуля (subroutine line()), алгоритм функціонування якго наведено на рис. 6.

Розрахунок координат векторів, які співпадають з дотичними, здійснюється в блоці 3.

В разі неможливості отримання результату (блоки 2, 3) управління передається блоком 3 блоку 7, який завершує функціонування програмного модуля.

При наявності результату обчислень блок 3 передає управління блоку 5, в якому здійснюються розрахунки коефіцієнтів та параметрів для вирішуючого правила за допомогою програмного модуля розрахунку потрібної швидкості (subroutine vv()), схема алгоритму якого наведена на рис. 7.

В блоці 6 здійснюються розрахунки інтервалів безпечної швидкості ОУ згідно вирішуючого правила.

Завершення функціонування алгоритму забезпечується в блоці 7.

В Додатку Д наведено текст програмного коду, який реалізує алгоритм, що наведений на рис. 4 (subroutine stick()).

Розглянемо алгоритм перетинів ЛШ для ОС (рис. 5) згідно блоку 6 алгоритму, який наведено на рис. 3.

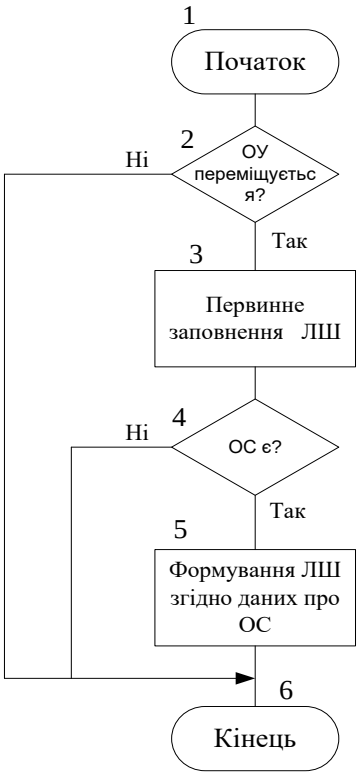


Рис. 5. Схема алгоритму розрахунку перетинів ЛШ

В блоці 2 (рис. 5) здійснюється аналіз ознаки переміщення ОУ за наявності ОС. За відсутністю ОС ЛШ формується з умови "всі припустимі швидкості ОУ дозволені" і управління передається блоку 6 для завершення

функціонування алгоритму. В іншому випадку управління передається блоку 3.

Блок 3 забезпечує первинне заповнення ознакою "1" - припустимого значення швидкості для формування ЛШ.

В блоці 4 здійснюється аналіз наявності ОС. За відсутністю ОС управління передається блоку 6 для завершення функціонування алгоритму. В іншому випадку управління передається блоку 5.

Для кожного ОС засобами блоку 5 забезпечується уточнення сформованої ЛШ ознаками неприпустимих значень швидкості ОУ. Таким чином забезпечується формування перетину всіх ЛШ при взаємодії ОУ з усіма ОС.

Після формування ЛШ для напрямку переміщення ОУ управління передається блоку 6, який забезпечує процес завершення функціонування алгоритму.

В Додатку В.2 наведено текст програмного коду, який реалізує алгоритм, який наведено на рис. 5 (subroutine aria()).

Розглянемо алгоритм розрахунку координат точок дотику (рис. 6) згідно блоку 2 алгоритму, який наведено на рис. 4.

В блоці 2 (рис. 6) здійснюється первинне заповнення області пам'яті значеннями координат точок дотику ЛВР від усіх ОС, які знаходяться в просторі Q. В якості первинних значень обрані нульові значення.

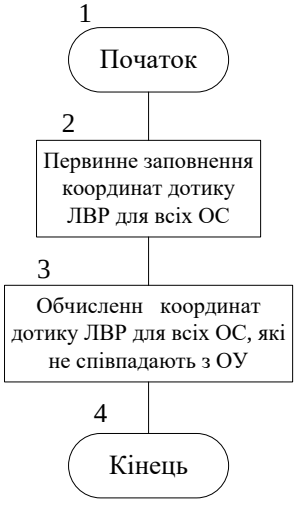


Рис. 6. Схема алгоритму розрахунку перетинів ЛШ

В блоці 3 обчислюються значення координат дотичних ЛВР за умови можливості їх обчислення (відстань між ОУ та ОС більша від припустимої для зближення при взаємному переміщенні).

Завершення функціонування алгоритму забезпечується засобами блоку 4.

В Додатку В.2 наведено текст програмного коду, який реалізує алгоритм, що наведений на рис. 6 (subroutine line()).

Розглянемо алгоритм розрахунку коефіцієнтів та параметрів для вирішуючого правила (рис. 7) згідно блоку 5 алгоритму, який наведено на рис. 4.

В блоці 2 (рис. 7) здійснюється розрахунок модулів векторів швидкості ОУ та ОС.

В блоці 3 аналізується стан переміщення ОУ. В разі переміщення ОУ управління передається блоку 4. В іншому разі управління передається блоку 7.

Обчислення координат та модулів векторів для переміщення ОУ та ОС в точки дотику відповідних ЛВР для векторів переміщення ОУ та ОС здійснюється в блоці 4.

В блоці 5 аналізується стан переміщення ОС. В разі переміщення ОС управління передається блоку 6. В іншому разі управління передається блоку 7.

Рис. 7. Схема алгоритму розрахунку коефіцієнтів вирішуючого правила

Розрахунок значення коефіцієнтів здійснюється в блоці 6.

Блок 7 забезпечує завершення функціонування алгоритму.

В Додатку В.2 наведено текст програмного коду, який реалізує алгоритм, наведений на Рис. 6 (subroutine vv()).

Алгоритми функціонування основних модулів програмних засобів розрахунку ПКУО наведені на рис. 1 - рис. 7.

В Додатку В.2 наведено тексти програмного коду повного проекту розрахунку ПКУО в середовищі розробки MS Visual Studio 2010 з інтегрованими засобами PGI Fortran 12.10 та акселератором.

В.2 - програмні засоби інформаційної технології розрахунку множин опису стану об'єкту управління

```
!
!  ConsoleApp.f90
!
!  Fortran Console Application
!  Generated by PGI Visual Fortran(R)
!  9/7/2013 1:39:35 PM
!

      program prog

      implicit none

      ! Variables
      ! Body
!    программа отладки komby
!    редакция: 07.09.2013
!    ------------------------------------------------
     dimension f(10,2),c(10,2),d(10),ogr(10,2),ogl(10,2),iflag(10),
    ;gr(4,2),buf(2,2)
     amin=-32.
     hy=64./399.
     hx=64./559.
     irez=4
     do 60 i=1,4
     do 60 j=1.2
 60  gr(i,j)=0.
     irez=2
!     type 900
! 900 format(1x,' для просчета контрольного примера введите <1>:'x)
     accept 901,lll
901  format(i1)
     if(lll.ne.1) go to 23
     iflag(1)=1
     d(1)=2.
     c(1,1)=-1.2
     c(1.2)=-11.9
     f(1,1)=22.
     f(1,2)=15.5
     iflag(2)=1
     d(2)=2
     c(2,1)=-12.
     c(2,2)=0.
     f(2,1)=28.7
     f(2,2)=1.6
     iflaqf(3)=1
     d(3)=2.
     c(3,1)=-4.6
     c(3.2)=11.1
     f(3,1)=16.3
     f(3,2)=-11.3
     nr=4
     ogr91,1)=-22.
     ogr(1,2)=16.
     ogr(2,1)=-4.3
     ogr(2,2)=4.3
     ogr(3,1)=17.6
     ogr(3,2)=20.3
     ogr(4,1)=32.
     ogr(4,2)=14.
     nl=3
     ogl(1,1)=-32.
     ogl(1,2)=-32.
     ogl(2,1)=4.
     ogl(2,2)=2.
     ogl(3,1)=32.
     ogl(3,2)-32.
     zon=32.
```

```
      un=12.
      xn=11.9
      vn=-1.2
      gb1=3.141593*270./180.
      ge1=3.141592*90./180.
      g1=31.41592/180.
      t1=6.
      t=0.1
      ge=90.
      ge=270.
      g=10.
      psy=3.141592*354./180.
      umax=30.
      go to 150
   23 type 3
      i0=0
      i1=1
      i2=2
      i3=3
      i4=4
      i5=5
      i6=6
      i7=7
      i8=8
      i10=10
      nn=2
      call term1(i1,i2,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
      call term1(i2,i1,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
      buf(1,1)=0
      buf(1,2)=-32.
      buf(2,1)=0.
      buf(2,2)=32.
      call pict(buf,hy,hx,amin,nn,i1,nn)
      buf(1,1)=-32.
      buf(1,2)=0.
      buf(2,1)=31.
      buf(2,2)=0.
      call pict(buf,hy,hx,amin,amin,nn,i1,nn)
      call term1(17,i1,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
      call term1(18,10,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
    3 format(1x,'собственная скорость ОУ: 'x)
      accept 4,un
      type 80
   80 format(1x,'глубина оценки ситуации zon='x)
      accept 4,zon
    4 format(f8.3)
      type 35
   35 format(1x, ' направление перемещения ОУ: 'x)
      accept 4,psy
      p=psy
      psy=3.141592*psy/180.
      xn=vn*cos(psy)
      yn=vn*si(psy)
      buf(1,1)=xn
      buf92,1)=yn
      buf(2,1)=0.
      buf(2,2)=0.
      call pict(buf,hy,hx,amin,amin,nn,i3,nn)
      ir=5
      call perevd(ir,ir1,ir2,ir3)
      call term1(i4,i1,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
      iz=zon/hy
      call perevd(iz,iz1,iz2,iz3)
      call term1(i4,i1,iy1,iy2,iy3,ix1,ix2,ix3,iz1,iz2,iz3)
      type 5
    5 format(1x,' начало сектра просмотра: 'x)
      accept 4,gb
      type 7
    7 format(1x, ' конец сектра просмотра: 'x)
      accept 4,ge
      type 8
    8 format(1x,' шаг просмотра сектора:  'x)
      accept 4,gb
      type 9
    9 format(1x, 'время упреждения[мин]: 'x)
```

```
      gb1=gb*3.141592/180.
      ge1=ge*3.141592/180.
      g1=g*3.141592/180.
      accept 4,t1
      t=t1/60.
      type 10
   10 format(1x,' ограничения пространства наблюдения обязательны ! ' /1x,
      accept 41,nr
      format(i2)
   40 format(1x, ' !ограничения пространства наблюдения обязательны ! ' /1x,
     *' количество точек правого ограничения: 'x)
      accept 41,nr
   41 format (i2)
      do 43 i=1, nr
      type 44.1
   44 format(1x, 'точка ',i2,' [2f8.3]: ')
      accept 18,ogr(i,1),ogr(i,2)
      ix=abs(ogr(i,1)-amin)/hx
      iy=abs(ogr(i,2)-amin)/hy
      call perevd(ix,ix1,ix2,ix3)
      call prevd(iy,iy1,iy2,iy3)
      if(i.ne.1)go to 62
      call term1(i3,i1,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
      go to 43
   62 call term(i3,i1,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
   43 continue
      type 45
   45 format(1x,' точек левого ограничения <>1): 'x)
      accept 41,nl
      do 47 i=1,nl
      type 44,i
      accept 18,ogl9i,1), ogl(i,2)
      ix=abs(ogl(i,1)-amin)/hx
      iy=abs(ogl(i,2)-amin)/hy
      call perevd(ix,ix1,ix2,ix3)
      call perevd(iy,iy1,iy2,iy3)
      if(i.ne.1) go to 63
      call term1(i3,i1,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
      go to 47
   63 call term1(15,17,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
   47 continue
      type 11
   11 format(1x, 'обстановка наблюдения')
      do 12 i=1,10
      type 13,i
   13 format(1x , 'ОН ',12, '  :')
      type 14
   14 format(1x,' ОН ',12,' :')
      accept 15,iflag(i)
   15 format9i1)
      if(iflag(i).eq.0) go to 12
      type 16
   16 format(1x,' координаты ОН [2f8.3]:  'x)
      accept 18,f(i,1),f(i,2)
      type 17
   17 format(1x,' координаты вектора ист.скор.ОН [2f8.3]: 'x)
      accept 18,c(i,1),c(i,2)
   18 format(2f8.3)
      buf(2,1)=f(i,1)
      buf(2,2)=f(i,2)
      buf(1,1)=f(i,1)+c(i,1)
      buf(1,2)=f(i,2)+c(i,2)
      call pict(buf,hy,hx,amin,amin,nn,i3,nn)
      type 19
   19 format(1x,'допустимая дистанция сближения: 'x)
      accept 4,d(i)
      ir=d(i)/hx
      call perevd(ir,ir1,ir2,ir3)
      call term1(i4,i1,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
   12 continue
  150 call posit(ogr,ogl,nr,nl,gr,f,c,xn,yn,zon,d,iflag,irez)
      call komby(f,zon,ogr,ogl,nr,nl,c,xn,yn,gb1,ge2,un,umax,psy,g1,*t,d,iflag,ierr)
      type 30,ierr
   30 format(1x,'ierr=',i1)
      type 22
```

```
 22 format(1x,'для выхода введите 1' /1x, '?' )
    accept 15,l
    if(l.ne.1)go to 23
    return
    end

    sunroutine
komby(a2,zon,ogr,ogl,nr,nl,c,xn,yn,ab,*ae,vn,vmax,psy,da,tu,ddop,iflag,ierr)
!    п/п постоения области реализуемости маневра с учетеом сектора просмотра и !
!    принципа учета по соблюдению правил
!    для случая равномерного прямолинейного движения он с учетом
!    времени запаздывания на выплнение маневра
!    a2(10,2) - координаты отметок он [отн.север]
!    zon- грубина просмотра
!    c(10,2) - координаты векторов ист.скоростей он
!    [ab,ae] - границы просматриваемого сектора [рад]
!    vn-модуль собственной скорости ОУ
!    [xn,yn]- координаты вектора собственной скорости
!    psy-собственный курс[рад]
!    da-дискретность перебора в интервале [ab,ae] (рад)
!    tu- время упреждения (час)
!    iflag(10)-массив сопровождения он
!    ierr- признак ошибки [0-нет, 1-есть]
!    ddop(10)- допустимая дистанция сближения для он
!    редакция:04.07.13
!
================================================================================
================================
    diension a3(10,2) , a2(10,2) , c(10,2) , iflag(10) , isb(10) , ddop(10),
*l(40)
    logical*1 r1(42) , r2(6)
    data r2/' ','.','-','*','i'/
    ierr=0
!    преобразование пространства и селекция он по пространству
    do 1 i=1,10
    isb(i)=iflag(i)
    if(isb9i).eq.0)go to 1
    asx=(c(i,1)-xn)*tu
    asy=(c(i,2)-yn)*tu
    a3(i,1)=a2(i,1)+asx
    a3(i,2)=a2(i,2)+asy
    az=sqrt(a3(i,1)*a3(i,1)+a3(i,2)*a3(i,2))
    if((az.le.zon).and.(az.ge.ddop(1)))go to 13
    go to 14
 13 call pnt(ogl,ogr,nl,nr,a3(i,1),a3(i,2),ipn)
    if(ipn.eq.o)go to 1
 14 isb(i0=0
  1 continue
!    _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!    вычисления количества интервалов построения <п к у о>
    if(ab.le.ae)go to 10
    n-abss(6.283184-ab)/da+ae/da
    go to 11
 10 n=(ae-ab)/da
 11 if(n.ne.0)go to 3
    type 2
  2 format(10x,'пкуо нельзя построить')
    ierr=1
    go to 100
    _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
  3 type 7
  7 format(20x,'пкуо')
    do 17 i2=1,42
 17 r1(i2)=r2(3)
    type 18,r1
 18 format(1x,42a1)
    do 4 i=1,n
    at=ab+da8(i-1)
    if(at.ge.6.283184)at=at-6.283184
    xxn=vn*cos(at)
    yyn=vn*sin(at)
    call cross1(vmax,xxn,yyn,c,a3,ddop,l,isb,ie)
    if(ie.eq.1)ierr=1
    if(ie.eq.1) go to 4
    do 19 i4=1,40
```

```
      i3=i4+1
      r1(i3)r2(5)
      if(l(i4).eq.0)r1(i3)=r2(2)
   19 continue
      r1(1)=r2(6)
      r1(42)=r2(6)
      ipsp=at*180./3.141592
      type 20,r1,ipsp
   20 format(1x,42a1,3x,'курс=',i3)
    4 continue
      do 21 i=1,42
   21 r1(i)=r2(3)
      type 18,r1
      type 22
   22 format(10x,'10',8x,'20',8x,'30'/10x,'скорости')
  100 return
      end

      subroutine pnt(bl,br,nl,nr,x0,y0,ind)
!     п/п выделения признака расположения вершины между линиями ограничений
!     bl(10,2)- массив координат точек левіх ограничений
!     br(10,2) - массив координат точек правіх ограничений
!     x0,y0 - координаты точки, положения которой относительно ограничений
!     определяется
!     nl-количетсво точек правых ограничений
!     nr-количество точек левых ограничений
!     ind - признак результата :
!           0-точка между ограничениями
!           1-точка за ограничений пространства наблюдения
!     длина звена # 0
!     2 звена не составляют одну прямую
!     min значения: nl=2,nr=2
!     редакция: 24.04.13
!     = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =
      dimension bl(10,2),br(10,2),b(21,2),k(21)
!     создание массива точек замкнутого пространства:
      nlr=nl+nr
      do 1 i=1,nl
      do 1 j=1,2
    1 b(i,j)=bl(i,j)
      do 3 i=nl+1,nlr
      do 3 j=1,2
    3 b(i,j)=br(nlr+1-i,j)
  !   выбор вершин,совпадающих с прямой y=y0,
  !   создание массива к(20): k(j)=0 вершина совпадает,
  !                           k(j)=1 вершина не совпадает
      do 10 i=1,nlr
      k(i)=1
      if(y0.ne.b(i,2))go to 10
      k(i)=0
   10 continue
!     определение отрезков берегов и растояний по y0
!     от переечений до точки 0
      kin=0
   11 if(k(1).ne.0)go to 20
      do 66 j=1,2
      b2=b(1,j)
      do 65 i=1,nlr-1
      i1=i+1
      b(i,j)=b(i1,j)
   65 continue
      b(nlr,j)=b2
   66 continue
      kb=k(1)
      do 67 i=1,nlr-1
      i1=i+1
      k(i)=k(i1)
   67 continue
      k(nlr)=kb
      kin=kin+1
      go to(11,20),kin
   20 do 9 j=1,2
    9 b(nlr+1,j)=b(1,j)
      k(nlr+1)=k(1)
!     сформирован массив в и к , т.о., что к(1)#0 и к(nlr1)#0
```

```
!    n - число '+' пересечений
     n=0
     do 30 i=1,nlr
     if((k(i).eq.1).and.(k(i+1).eq.1))go to 21
     if((k(i).eq.1).and.(k(i+1).eq.0).and.(k(i+2).eq.1))go to 22
     if((k(i).eq.1).and.(k(i+1).eq.0).and.(k(i+2).eq.0))go to 23
     go to 30
!    пересекает ли отрезок b(i,j)-b(i+1,j) прямую y=y0
 21  if(((b(i,2).gt.y0).and.(b(i+1,2).lt.y0)).or.
    *((b(i,2).lt.y0).and.(b(i+1,2).gt.y0)))go to 61
     go to 30
     расстояние от т.пересечения отрезка с y=y0 до т. 0
 61  x=((y0-b(i,2))*b(i,1)-b(i+1,1))-b(i,1)*b(i+1,2)-
    *b(i,2)))/(b(i,2)-b(i+1,2))-x0
 37  if(x)30,41,25
 41  ind=1
     go to 200
 25  n=n+1
     go to 30
 22  d1=b(i,2)-y0
     d2=b(i+2,2)-y0
 38  if(((d1.gt.0.).and.(d2.gt.0.)).or.((d1.lt.0).and.
    *(d2.lt.0.)))go to 30
     x=b(i+1,1)-x0
     go to 37
 23  if(((xo.ge.b(i+1,1)).and.(x0.le.b(i+2,1))),or.
    *((x0.le.b(i+1,1)).and.(x0.ge.b(i+2,1)))go to 41
     d1=b(i,2)-y0
     d2=b(i+3,2)-y0
     go to 38
 30  continue
     анализ числа '+' - n - отклонений
     ind=0
     an=1.
     if(n.eq.0)go to 41
     do 51 i=1,n
 51  an=-an
     if(an.eq.1.)go to 41
200  return
     end

     subroutine cross1(vmax,xn,yn,c,a2,d,lu,iflag,ierr)
!    п/пвычисления интервалов безопасных скоростей
!    .c(i,j) - массив координат векторов он
!           i-номер он
!              ,j=1 - x
!              ,j=2 - y
!     a2(i,j)-массив координат последних засечек он
!             i-номер он
!               ,j=1 - x
!               ,j=2-y
!     d(i) - массив d кр он
!     iflag(10) - признак сопровождения он [ 1- есть, 2- нет]
!     lv(40) - массив опасных скоростей
!     ierr - признак наличия ошибки
!            0 - ошибки нет
!              1 - ошибка есть
!      редакция: 10.02.13
!     ==============================================================
     dimension c(10,1),a2(10,1),d(10),lv(40),iflag(10)
     ierr=0
     k=0
     do 1 i=1,10
  1  k=k+iflag(i)
     i1=0
     if(k.ne.0)go to 10
     ind=1
     kin=1
     l=1
     call aria(vmax,vn1,vn2,lv,ind,l,kin)
     go to 100
 10  do 11 i=1,10
     if(iflag(i).eq.0)go to 11
     i1=i1+1
     call stick(vmax,xn,yn,vvn1,vn2,c(i,1),a2(i,1),a2(i,2),
```

```
     *d(i) , ind)
      call aria(vmax,vn1,vn2,lv,ind,i1,kin)
      if(ind.ne.6)go to 11
      ierr=1
      go to 100
   11 contiinue
  100 return
      end


      subroutine aria(vmax,vn1,vn2,lv,ind,n,k)
!     п/п отыскание пересечений для линейки скоростей
!     vmax - max скорость ОУ
!     vn1,vn2 - точки разбиения интервала из stick
!     lv(40) - массив дробления интервала 0<vn<vmax
!     n - номер (количество он) для линейки скоростей
!     ind - признак исходной информации , получаемой из stick
!       1 - 0<vvn<vmax
!       2-vn2<vn<vmax
!       3 - все скорости опасны
!       4 - 0<vvn<vn1 , vn2<vn<vmax
!       5 - 0<vn<vn1
!       6 - ошибка
!    k - признак существования результата
!       0 - все скорости опасны или был ind=6
!       1 - есть безопасные интервалы скоростей
!    редакция:08.02.13
!    ================================================================
      dimension lv(40),vb(40)
      data vb/1.,2.,3.,4.,5.,6.,7.,8.,9.,10.,11.,12.,13.,14.,15.,
     *16.,17.,18.,19.,20.,21.,22.,23.,24.,25.,26.,27.,28.,29.,
     *30.,31.,32.,33.,34.,35.,36.,37.,38.,39.,40./
     if(n.ne.1)go to 10
     iv-max
     if(iv.ne.0) go to 1
     do 63 i=1,40
   63 lv(i)=0
    3 k=0
     go to 100
    1 do 2 i=1,40
     lv(i)=0
     if(i.le.iv)lv(i)=1
    2 continue
     k=1
     go to 9
   10 if(k.eq.0)go to 100
    9 go to(100,20,50,30,40),ind
     go to 3
   20 do 21 i=1,iv
     if(vn2.gt.vb(i))lv(i)=0
   21 continue
     go to 100
   50 do 51 i=1,40
   51 lv(i)=0
     go to 3
   30 do 31 i=1,iv
     if((vn1.lt.vbi)).and.(vn2.gt.vb(i)))lv(i)=0
   31 continue
     go to 100
   40 do 41 i=1,iv
     if(vn1.lt.vb(i))lv(i)==0
   41 continue
  100  return
     end

     subroutine vv(vvar,xc,yc,xn,yn,x2,y2,xa,ya,rn,rk,ind)
!    п/п вычисления требуемой скорости
!    vn - модуль скорости ОУ
!    xc,yc - координаты вектора он
!    xn,yn - координаты вектора скорости ОУ
!    x2,y2 - координаты засечки (места) он
!    xa,ya - координаты точки пространства
!    vvar - значение рекомендуемой скорости
!    rn - коэффициент направленности vr
!    rk - коэффициент vn
```

```
!  ind - признак количества возможных решений
!      - 0 - решение единственно
!      - 1 - vn//2a, vc#2a
!      - 2 - vn#2a,vc//2a
!      - 3 - vn//2a, vc//2a
!      - 4 - нет решений /vn=0/
!      - 5 - стоящая цель /vc=0/
!  редакция:09.12.13
!  ==================
   sq(a,b)=sqrt(a*a+b*b)
   ind=4
   vvar=0.
   rk=0.
   rn=0.
   vn=sq(xn,yn)
   uc=sq(xc,yc)
   if(vn.eq.0.)go to 2
   x2a=xa-x2
   y2a=ya-y2
   v2a=sq(x2a,y2a)
   s2=(x2a*xn+y2a*yc)/(vn*u2a)
   if(vc.eq.0.)go to 40
   s1=(x2a*xc+y2a*yc)/(vc*u2a)
   as2=abs(s2)
   as1=abs(s1)
   if((as2.ne.1).and.(as1.eq.1.))go to 22
   if((as2.eq.1.).and.(as1.ne.1.))go to 21
   if((as1.eq.1.).and.(as2.eq.1.))go to 10
   go to 60
40 ind=5
   go to 2
21 ind=1
   go to 2
22 ind=2
   rk=0
   rn=s1
   go to 2
10 ind=3
   go to 2
60 ind=0
   if(xa.eq.x2)go to 1
   s=yn*x2a-xn*y2a
   rk=(yc*x2a-xc*y2a)/s
   vvar=rk*vn
   rn=(xc-rk*xn)/x2a
   go to 2
 1 rk=xc/xc
   vvar=vn*rk
   rn=(yc-rk*yn)/y2a
 2 return
   end

   subroutine line(x2,y2,d,r,ind)
 ! п/п вычисления точек касания с окружностью
 ! x2,y2-координаты последней засечки (места) он
 ! ind=0 точка 2 вне окружности
 ! ind=1 точка 2 внутри или на окружности
 ! d - дистанция сближения
 ! r(i,j) - массив результатов
 !          i-номер точки касания
 !          j=1 - x , j=2 - y
 ! редакция:25.11.13
 ! _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
   dimension r(2,2)
   ttt(a,b,c)=a*a*c*c-(a*a+b*b)*(c*c-b*b)
   do 10 i1=1,2
   do 10 i2=1,2
10 r(i1,i2)=0
   ind=1
   f=y2*y2+x2*x2
   if(abs(y2).gt.0.0001)go to 1
   if(abs(x2).gt.0.0001)go to 2
   go to 20
 1 diskr=ttt(x2,y2,d)
   if(diskr.le.0)go to 20
```

```
      ind=0
      yd=y2*d*d
      diskr=sqrt(diskr)*d
      r(1,1)=(xd+diskr)/f
      r(2,1)=(xd-diskr)/f
      r(1,2)=(d*d-x2*r(2,1))/y2
      r(2,2)=(d*d-x2*r(2,1))/y2
      go to 20
    2 diskr=ttt(y2,x2,d)
      if(diskr.le.0)go to 20
      ind=0
      yd=y2*d*d
      diskr=sqrt(diskr)*d
      r(1,2)=(yd+diskr)/f
      r(2,2)=(yd-diskr)/f
      r(1,1)=(d*d-y2*r(1,2))/x2
      r(2,1)=(d*d-y2*r(2,2))/x2
   20 return
      end

      subroutine  comp(xa,ya,xb,yb,xc,yc,ia)
!     п/п вычисления признака взаимного расположения векторов
!     xc,yc - координаты вектора он
!     xa,ya,xb,yb - координаты векторов внешних
!     ограничений области
!     ia - признак расположения вектора в области
!        - 0 - внутри области или на границе области
!        - 1 - вне области
!        - 2 - ошибка в данных
!     редакция п/п:28.12.13
!     ======================
      f(x1,x2,y1,y2)=x1*ya-x2*y1
      ia=0
      d=f(xa,xb,ya,yb)
      if(d.ne.0.)go to 1
      ia=2
      go to 2
    1 al=f(xc,xb,yc,yb)/d
      bl=f(xa,xc,ya,yc)/d
      if((al.lt.0.).or.(bl.lt.0))ia=1
    2 return
      end

      subroutine chang(a,b)
      п/п замены a и в местами
!     редакция:19.01.13
!     ============================
      c=a
      a=b
      b=c
      return
      end


      subroutine posit(ogr,nr,nl,gr,a3,c,xn,yn,zon,ddop,iflag,irez)
!     п/п вывода информации на графический экран
!     ogr(10,2) - пересчитанные правые ограничения
!     ogl(10,2) - пересчитанные левые ограничения
!     nr - к-во точек правых ограничений
!     nl - к-во точек левых ограничений
!     gr(4,2) - координаты ломаной линии маршрута
!     ddop(10) - допустимая дистанция сближений
!     a3(10,2) - координаты отметок он пересчитанные
!     c(10,2) -  координаты векторов ист.скор.он
!     xn,yn - координаты вектора собств.скорости ОУ
!     zon - глубина оценки ситуации
!     iflag(10) - массив признаков слпровождения ОН
!     irez - признак результата из <mprt1>
!            1 - результата нет
!            2 - прямой проход
!            3 - двухелементний маневр
!     редакция:08.12.13
!     ======================================================
      dimension ogr(10,2),ogl(10,2),gr(4,2),a3(10,2),c(10,2),
     * iflag(10),bufr(2,2),ogr1(10,2),ogl1(10,2),ddop(10)
```

```
!  масштабирование экрана видеотерминала
   _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
   tx=(ogl(1,1)+ogr(1,1))/2.
   ty=(ogl(1,2)+ogr(1,2))/2.
   txl=(ogl(1,1)-tx)*1.1
   tyl=(ogl(1,2)-ty)*1.1
   txr=(ogr(1,1)-tx)*1.1
   tyr=(ogr(1,2)-ty)*1.1
   xm1=0.
   ym1=0.
   xm2=0.
   ym2=0.
   do 1 i=1,nr
   if(xm1.le.ogr(i,1))xm1=ogr(i,1)
   if(xm1.le.ogr(i,1))xm2=ogr(i,1)
   if(ym1.le.ogr(i,2))ym1=ogr(i,2)
   if(ym2.ge.ogr(i,2))ym2=ogr(i,2)
   ogr1(i,1)=ogr(i,1)+txr
   ogr1(i,2)=ogr(i,2)+tyr
   if(xm1.le.org1(i,1))xm1=ogr1(i,1)
   if(xm2.ge.ogr1(i,1))xm2=ogr1(i,1)
   if(ym1.le.ogr1(i,2))ym1=ogr1(i,2)
   if(ym2.ge.ogr1(i,2)ym2=ogr1(i,2)
 1 continue
   do 2 i=1,nl
   if(xm1.le.ogl(i,1))xm1=ogr(i,1)
   if(xm2.ge.ogl(i,1))xm2=ogl(i,1)
   if(ym1.le.ogl(i,2))ym1=ogl(i,2)
   if(ym2.ge.ogl(i,2))ym2=ogl(i,2)
   ogl1(i,1)=ogl(i,1)+txl
   ogl1(i,2)=ogl(i,2)+tyl
   if(xm1.le.ogl1(i,1))xm1=ogl1(i,1)
   if(xm2.ge.ogl1(i,1)xm2=ogl1(i,1)
   if(ym1.le.ogl1(i,2))ym1=ogl1(i,2)
   if(ym2.ge.ogl1(i,2))ym2=ogl1(i,2)
 2 continue
   do 3 i=1,10
   if(iflag(i).eq.0)go to 3
   if(xm1.le.a3(i,1))xm1=a3(i,1)
   if(xm2.le.a3(i,2))xm2=a3(i,1)
   if(ym1.le.a3(i,2))ym1=a3(i,2)
   if(ym2.ge.a3(i,2))ym2=a3(i,2)
 3 continue
   if(xm1.lt.zon)xm1=zon
   if(ym1.lt.zon)ym1=zon
   z1=-zon
   if(xm2.gt.z1)xm2=z1
   if(ym2.gt.z1)ym2=z1
   hy=abs(xm1-xm2)/399.
   hx=abs(ym1-ym2)/559.
!  _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!  рисование левой границы
   i1=1
   call term1(i1,il,ix1,ix2,ix3,iy1,iy2,iy3,ir1,ir2,ir3)
   i1=2
   call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
   il=7
   nn=10
   call pict(ogl,hy,hx,xm2,ym2,nl,il,nn)
   call pict(ogl1,hy,hx,xm2,ym2,nl,il,nn)
!  _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!  рисование правой границы
   call pict(ogr,hy,hx,xm2,ym2,nr,il,nn)
   call pict(ogr1,hy,hx,xm2,ym2,nr,il,nn)
!  _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!  нанесение осей
   n=2
   bufr(1,1)=0.
   bufr(1,2)=ym2
   bufr(2,1)=0.
   bufr(2,2)=ym1.
   nn=2
   il=3
   call pict(bufr,hy,hx,xm2,ym2,n,il,nn)
   bufr(1,1)=xm2
```

```
      bufr(1,2)=ym2
      bufr(2,1)=0.
      bufr(2,2)=ym1
      nn=2
      il=3
      call pict(bufr,hy,hx,xm2,ym2,n,il,nn)
      bufr(1,1)=xm2
      bufr(1,2)=0.
      bufr(2,1)=xm1
      bufr(2,2)=0.
      call pict(bufr,hy,hx,xm2,ym2,n,il,nn)
      bufr(1,1)=xm2
      bufr(1,2)=0.
      bufr(2,1)=xm1
      bufr(2,2)=0.
      call pict(bufr,hy,hx,xm2,ym2,n,il,nn)
!     _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!     вычисление зоны решения задачи конфликта
      ir=zon/hx
      call perevd(ir,jr1,jr2,jr3)
!     _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!     рисование объекта управления с вектором и зоной
      ir=5
      il=4
      call perevd(ir,ir1,ir2,ir3)
      bufr(1,1)=xn/2.
      bufr(1,2)=yn/2.
      bufr(2,1)=0.
      bufr(2,2)=0.
      call pict(bufr,hy,hx,xm2,n,il,nn)
      i1=4
      call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,jr1,jr2,jr3)
      call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
!     _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!     рисование встречных ОС с векторами
      j1=7
      j2=0
      call term1(j1,j2,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
      do 4 i=1,10
      if(iflag(i).eq.0)go to 4
      bufr(2,1)=a3(i,1)
      bufr(2,2)=a3(i,2)
      bufr(1,1)=a3(i,1)+c(i,1)/2.
      bufr(1,2)=a3(i,2)+c(i,2)/2.
      call pict(bufr,hy,hx,xm2,ym2,n,il,nn)
      ir=ddop(i)/hx
      call perevd(ir,ir1,ir2,ir3)
      call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
      if(i.lt.10)go to 11
      j1=1
      j2=0
      call term1(j3,j1,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
   12 call term1(j3,j2,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
      go to 4
   11 j2=i
      go to 12
    4 continue
!     _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!     рисование конечной точки
      go to(5,6,7),irez
    5 fx=gr(2,1)
      fy=gr(2,2)
    9 ix=abs(fx-xm2)/hx
      iy=abs(fy-ym2)/hy
      call perevd(ix,ix1,ix2,ix3)
      call perevd(iy,iy1,iy2,iy3)
      i1=3
      call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
      i1=4
      call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
      go to 8
    6 fx=gr(3,1)
      fy=gr(3,2)
      n=3
      go to 9
```

```
 7 fx=gr(4,1)
   fy=gr(4,2)
   n=4
   go to 9
!  _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!  рисование траектории
 8 il=5
   nn=4
   call pict(gr,hy,hx,xm2,ym2,n,il,nn)
   i1=4
   do 10 ir=1,8
   call perevd(ir,ir1,ir2,ir3)
   call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
10 continue
   return
   end

   suberoutine perevd(is,is1,is2,is3)
!  п/п перевода числа is=is*100+is2*10+is3
!  редакция:06.12.13
!  ========================================
   is1=is/100
   is2=(is-is1*100)/10
   is3=is-is1*100-is2*10
   return
   end

   subroutine pict(buf,hx,hy,xmin,ymin,n,il,nn)
!  п/п рисования ломаной линии из массива <buf> с n-1 звеньями
!  hx - шаг разбиения по оси <x>
!  hy - шаг разбиения по оси <y>
!  nn - размерность массива <buf>
!  xmin,ymin - нуль экрана
!  buf(1,1) - массив рисования
!  n - количество элементов <buf>
!  il - тип линии :
!       1 - сплошная тонкая
!       2 - пунктирная [длинный штрих]
!       3 - штрих-пунктирная
!       4 - пунктирная[короткий штрих]
!       5 - точечная[подсвет-гашение]
!       6 - подсвет конечной точки[для векторов]
!       7 - сплошная толстая[для векторов]
!  редакция:08.12.13
!  в п/п ось <x>-ординат ; <y>-абсцис
!  ================================================================
   dimension buf(nn,2)
   i1=3
   ix=abs(buf(1,1)-xmin)/hx
   iy=abs(buf(1,2)-ymin)/hy
   call perevd(ix,ix1,ix2,ix3)
   call perevd(iy,iy1,iy2,iy3)
   call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
   i1=5
   do 1 i=2,n
   ix=abs(buf(i,1)-xmin)/hx
   iy=abs(buf(i,2)-ymin)/hx
   call perevd(ix,ix1,ix2,ix3)
   call perevd(iy,iy1,iy2,iy3)
   call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
 1 continue
   return
   end


   subroutine stick(umax,xn,yn,vn1,vn2,xc,yc,x2,y2,d,ind)
!   п/п вычисления линейки скоростей для одного ОН
!   vmax- max значение модуля скорости ОУ
!   xn,yn - координаты вектора скорости ОУ
!   xc,yc - координаты вектора скорости ОН
!   x2,y2 - координаты последней отметки ОН
!   vn1 - min допустимая скорость
!   vn2 - max допустимая скорость
!   ind - признак типа результата
!         1 - 0<vn<max
```

```
!         2 - vn2<vn<vmax
!         3 - все скорости опасны
!         4 - 0<vn<vn1 и vn2<vn<vmax
!         5 - 0<vn<vn1
!         6 - ошибка в данных
!    d - безопасная дистанция /dkp/
!    редакция:24.01.13.
!    ================================================================
     dimension point(2,2),rn(2),rk(2),kin(2),vvar(20
!         вычисление координат точек касания
     call line(x2,y2,d,point,ipr)
     xa=point(1,1)-x2
     ya=point(1,2)-y2
     xb=point(2,1)-x2
     yb=point(2,2)-y2
     if(ipr.ne.1)go to 1
  6  ind=6
     go to 100
!        определение параметров выхода на окружность d
  1  do 2 i1=1,2
     call vv(vvar(i1),xc,yc,xn,yn,x2,y2,point(i1,1),point(i1,2),
    *rn(i1),rk(i1),kin(i1)
     vn1=vvar(1)
     vn2=vvar(2)
  2  continue
     if((kin(1).eq.5).and.(kin(2).eq.5))go to 5
     if(((kin(1).eq.5).eq.5).and.(kin(2).ne.5)).or.((kin(1).ne.5).and.
    * (kin(2).eq.5)).or.(kin(1).eq.4).or.(kin(2).eq.4).or.((kin(1).
    * eq.3).and.(kin(2).eq.3)).or.((kin(1).eq.1).and.(kin(2).eq.3)).or.
    * ((kin(1).eq.3).and.(kin(2).eq.1)).or.((kin(1).eq.2).and.(kin(2).
    * eq.3)).or.((kin(1).eq.3).and.(kin(2).eq.2)))go to 6
      if(((kin(2).eq.3).and.(kin(2).eq.0)).or.((kin(1).eq.0).and.
    * (kin(2).eq.3)))go to 10
     if(((kin(1).eq.1).and.(kin(2).eq.1)).or.((kin(1).eq.2).and.
    * (kin(2).eq.2)))go to 6
     if(((kin(1).eq.1).and.(kin(2).eq.2)).or.((kin(1).eq.2).and.
    * (kin(2).eq.1)))go to 15
     if(((kin(1).eq.0).and.(kin(2).eq.0))go to 20
     if(((kin(1).eq.0).and.(kin(2).eq.0))go to 25
     if(((kin(1).eq.0).and.(kin(2).eq.0))go to 26
     if(((kin(1).eq.0).and.(kin(2).eq.0))go to 20
     if(((kin(1).eq.0).and.(kin(2).eq.0))go to 20
  5  xnn=-xn
     ynn=-yn
  9  call comp(xa,ya,xb,yb,xnn,ynn,ia)
     if(ia.eq.2)go to 6
     if(ia.eq.1)go to 10
 28  ind=3
     goto 100
 10  ind=1
     go to 100
 15  xnn=xc-xn
     ynn=yc-yn
     go to 9
 25  if(rn(1).levvar(1).0)go to 10
     call comp(xa,ya,xb,yb,xc,yc,ia)
     if(ia.eq.2)go to 6
     if(ia.eq.0)go to 27
 67  if(rk(1).lt.0.)go to 18
 33  if(vvar(1).ge.vmax)go to 10
     vn1=vvar(1)
 41  ind=5
     go to 100
 27  if(rk(1).lt.0)go to 28
 32  if(vvar(1).gt.vmax)go to 28
     vn2=vvar(1)
 46  ind=2
     go to 100
 26  call chang(rn(1),rn(2))
     call chang(rk(1),rk(2))
     call chang(vvar(1),vvar(2))
     go to 25
 20  if((rn(1).gt.0.).and.(rk(1).gt.0).and.(rn(2).gt.0.).*and.(rk(2).gt.0.))go to 40
     call comp(xa,ya,xb,yb,xc,yc,ia)
     if(ia.eq.2)go to 6
```

```
      if((rn(1).gt.0.).and.(rn(2).gt.0.).and.(((rk(1).gt.0.).and.and.
      if((kin(1).eq.5).and.(kin(2).eq.5))go to 5
      if(((kin(1).eq.5).eq.5).and.(kin(2).ne.5)).or.((kin(1).ne.5).and.
     * (kin(2).eq.5)).or.(kin(1).eq.4).or.(kin(2).eq.4).or.((kin(1).
     * eq.3).and.(kin(2).eq.3)).or.((kin(1).eq.1).and.(kin(2).eq.3)).or.
     * ((kin(1).eq.3).and.(kin(2).eq.1)).or.((kin(1).eq.2).and.(kin(2).
     * eq.3)).or.((kin(1).eq.3).and.(kin(2).eq.2)))go to 6
       if(((kin(2).eq.3).and.(kin(2).eq.0)).or.((kin(1).eq.0).and.
     * (kin(2).eq.3)))go to 10
      if(((kin(1).eq.1).and.(kin(2).eq.1)).or.((kin(1).eq.2).and.
     * (kin(2).eq.2)))go to 6
      if(((kin(1).eq.1).and.(kin(2).eq.2)).or.((kin(1).eq.2).and.
      go to 6
   40 if(vn1.gt.vn2)call chang(vn1,vn2)
      if(vn2.gt.vmax0go to 28
      go to 46
   50 if(vn1.gt.vn2)call chang(vvar(1),vvar(2))
      go to 32
   55 if(vn1.lt.vn2)call chang(vvar(1),vvar(2))
      go to 33
   65 xxa=-xa
      yya=-ya
      xxb=-xb
      yyb=-yb
      call comp(xa,ya,xb,yb,xc,yc,ia1)
      call comp(xxa,yya,xxb,yyb,xc,yc,ia2)
      if(((ia1.eq.2).or.(ia2.eq.2)).or.((ia1.eq.0).
      if((ia1.eq.1).and.(ia2.eq.1))go to 10
      if((ia1.eq.1).and.(ia2.eq.0))go to 67
      go to 27
  100  return
       end

         end program prog
```

Додаток Г. Алгоритми і програмні компоненти методу інтегрального усікання варіантів та синтезу стратегій рішень

Г.1 - алгоритми програмних компонент методу інтегрального усікання варіантів та синтезу стратегій рішень

Алгоритми створені та реалізовані в програмному засобі імітаційної моделі згідно співвідношенням, які отримані за результатами дисертаційного дослідження.

Алгоритм методу інтегрального усікання варіантів та синтезу стратегій рішень наведено на рис. 1.

Аналогічно до алгоритму розрахунку інформаційних множин опису стану об'єкту управління (Додаток В), алгоритм передбачає наявність інформації щодо ОС, ОУ, простору Q, мети переміщення ОУ, границі простору $\Gamma_{гр}(Q)$, значень припустимого зближення ОУ з ОС, тощо. Усі похідні дані мають вводитись модулем приймання та перетворення похідних даних СІУ.

Робота алгоритму методу інтегрального усікання варіантів та синтезу стратегій рішень починається з визначення змінних та параметрів (блок 2), які необхідні для функціонування алгоритму та його програмної реалізації в середовищі розробки MS Visual Studio 2010/2012 з інтегрованими засобами PGI Fortran 12.10, засобу розпаралелювання обчислень OpenMP та акселератора Visual PGI Fortran, який є універсальним для мов програмування C/C++ та Visual PGI Fortran.

Робота алгоритму рішення конфлікту вимагає наявності детермінованого визначення коефіцієнтів критерію (функції ціни).

Програмний модуль синтезу стратегій управління ОУ для ІП СІУ при розрахунку ПКУО оформлено у вигляді підпрограми (subroutine mprt ()), текст якої наведено в Додатку Г.2.

В блоці 2 проводиться процедура контролю наявних для обчислення критерію відбору синтезованих стратегій рішення конфлікту при переміщенні ОУ в цільову точку простору рішень Q.

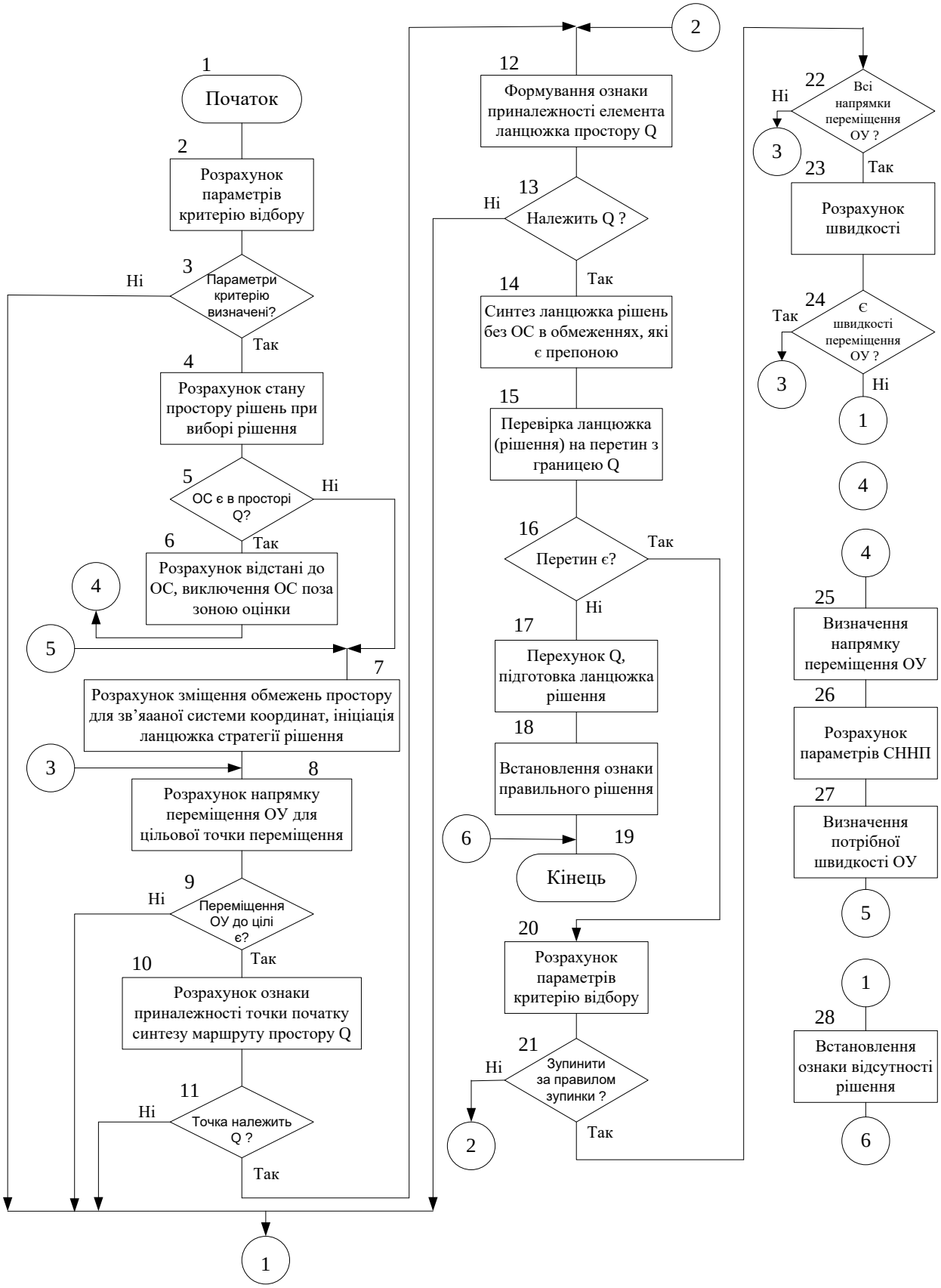


Рис. 1. Схема алгоритму методу інтегрального усікання варіантів та синтезу стратегій рішень.

В блоці 3 здійснюється процедура аналізу ознаки визначеності параметрів для обчислення критерію відбору. В разі наявності значень параметрів, необхідних для обчислення значення критерію, управління передається блоку 4. В іншому випадку управління передається блоку 28 для формування ознаки помилки та завершення роботи алгоритму.

Засоби блоку 4 забезпечують виконання розрахунків, які дозволяють визначити обчислювальну імітаційну модель простору рішень. Крім того, слід враховувати зміни в просторі рішень Q, які обумовлені часом затримки, пов'язаним з обчислювальними процедурами та прийняттям рішення (особливо важливо для технічних ергатичних систем). Перераховуються значення координат границі $\Gamma_{гр}(Q)$, всіх ОС, цільової точки переміщення ОУ, а також похідні дані траєкторії переміщення ОУ, яка відображає геометричне місце положень ОУ для стратегії управління, яка обирається, згідно синтезованому ланцюжку управління.

Засоби блоку 5 забезпечують перевірку наявності ОС в просторі рішень. При наявності ОС в просторі Q управління передається блоку 6. В іншому випадку управління передається блоку 7.

Основні алгоритми, пов'язані з розпізнаванням приналежності підмножин множинам наведені на рис. 2 - 5.

Після перерахунку простору Q в блоці 6 оцінюється відношення ОУ до зон оцінки ситуації та реалізації стратегій рішення. ОС, які не входять до зазначених зон, не враховуються при вирішенні конфлікту. Для зони надзвичайного зближення, синтез стратегій управління, як правило, визначається відповідними інструкціями, настановами, правилами, тощо. Також здійснюється ініціація ланцюжка управління ОУ, як віртуальних даних в пам'яті обчислювального середовища ІП СІУ.

Після виконання алгоритмічної частини блоку 6 управління передається засобам блоку 25, в якому визначається напрями переміщення ОУ в просторі Q.

Основний алгоритм блоку 25 наведено на рис. 6.

В блоці 26 виконуються процедури обчислення та оброблення даних щодо СННП усіх ОС в просторі рішень.

Алгоритм обчислення ЛШ для одного ОС наведено на рис. 4 Додатку В.1.

Основні алгоритми розрахунку та обробки даних щодо СННП наведені на рис. 7 і 8.

Після визначення напрямків переміщення ОУ (блок 25) та простору рішення з врахуванням усіх обмежень (блок 26) в блоці 27 визначаються швидкості переміщення ОУ при синтезі ланцюжків стратегій рішення конфлікту взаємодії з ОС в просторі рішень $G^*_{ріш}$ за умов обмежень та невизначеності.

Алгоритм розрахунку значень коефіцієнтів та параметрів для вирішую чого правила згідно блоку 27 алгоритму наведено на рис. 4 Додатку В.1, а текст програмного коду, який реалізує алгоритм, наведено в Додатку В.2 (subroutine vv()).

Алгоритм визначенння рекомендованої швидкості переміщення ОУ наведено на рис. 9.

Після закінчення функціональних процедур блоку 28 управління передається блоку 7.

В блоці 7 здійснюються процедури перерахунку простору $G_{ріш}$ для зв'язаної з ОУ системи координат, що спрощує подальші обчислення, а також здійснюється ініціація структури ланцюжка для стратегій управління, які в подальшому будуть синтезовані для ОУ.

Засобами блоку 8 здійснюється розрахунок напрямку переміщення ОУ в цільову точку простору $G^*_{ріш}$, який враховує всі обмеження простору $G_{ріш}$. Алгоритм основного функціонального модуля блоку 8 наведено на рис. 12.

В блоці 9 здійснюється аналіз наявності переміщення ОУ шляхом визначення модуля вектору швидкості. Якщо ОУ знаходиться в стані

відсутності руху, управління передається блоку 28. В разі переміщення ОУ управління передається блоку 10.

Засоби блоку 10 здійснюють розрахунок ознаки приналежності точки початку переміщення синтезованому ланцюжку управління та траєкторії переміщення ОУ за траєкторією в просторі $G^*_{piш}$. В блоці 10 застосовуються основні алгоритми блоку 5.

В блоці 11 аналізується приналежності основнох параметрів (координат) ланцюжка (траєкторіЇ) переміщення ОУ простору $G^*_{piш}$. По-перше - це точки початку та закінчення траєкторії переміщення ОУ, а також координати точок зміни стратегії управління ОУ (точки конкатенації елементів ланцюжків рішень) в просторі $G^*_{piш}$.

В разі приналежності точок конкатенації ланцюжків рішень (ламаної лінії траєкторії переміщення в перерізі) простору $G^*_{piш}$ (перерізу простору) управління передається блоку 12. В іншому випадку управління передається блоку 28.

Засоби блоку 12 забезпечують визначення ознаки приналежності синтезованих ланцюжків рішень простору $G^*_{piш}$ або його перерізу площиною.

Основний алгоритм блоку 12 наведено на рис. 4.

Засоби блоку 13 здійснюють аналіз ознаки (ознак) приналежності елемента ланцюжка синтезованого рішення (елемента ламаної лінії траєкторії) простору $G^*_{piш}$. В разі приналежності елемента простору $G^*_{piш}$ управління передається блоку 14. В іншому випадку - блоку 28.

Блок 14 функціонально здійснює синтез ланцюжка стратегії рішення для випадку, коли ОС не накладають обмеження на $G_{piш}$. Тобто $\left(G_{piш} \bigcap_Q G^*_{piш}\right) \subset \emptyset$.

Основні алгоритми функціональних складових блоку 14 наведені на рис. 14 - 20.

Функціональні складові блоку 15 здійснюють перевірку ланцюжка стратегії управління (траєкторії переміщення) ОУ щодо перетину з границею

простору $\Gamma_{гр}(Q)$. Основні алгоритми функціональних складових блоку співпадають з основними функціональними складовими блоків 12 та 14.

За сформованими в блоці 15 ознаками засоби блоку 16 здійснюють передачу управління блоку 17 в разі приналежності ланцюжка стратегій управління (траєкторії переміщення ОУ) простору Q. В іншому разі управління передається блоку 20.

Засоби блоку 17 формують параметри ланцюжка стратегії рішення та здійснюють перерахунок параметрів простору Q з урахуванням переміщення ОУ за умов вибору синтезованої стратегії рішення.

Блок 18 здійснює обчислювальні процедури щодо формування ознаки наявності правильного рішення, обчислення значення критерію відбору та виведення результатів розрахунку параметрів лінцюжка рішень. Для ергатичної СУ передбачається виведення інформації щодо траєкторії та параметрів переміщення для ОПР. При наявності рішення, яке відповідає вимогам критерію відбору або правила зупинки процедура синтезу стратегій рішення завершується і управління передається блоку 19.

В блоці 19 виконуються процедури, які пов'язані з завершенням процесу синтезу стратегій рішення та завершенням функціонування алгоритму.

При передачі управління блоком 16 в блоці 20 здійснюється обчислення значення параметрів критерію, а також обчислюється ознака зупинки процесу синтезу стратегій за правилом зупинки.

Засобами блоку 21 здійснюється аналіз наявності ознаки зупинки обчислень за правилом зупинки при аналізі можливості переміщення ОУ за певним напрямком. За наявності ознаки зупинки управління передається блоку 22. В іншому разі для продовження процесу синтезу альтернативних стратегій рішення управління передається блоку 12.

Засобами блоку 22 здійснюється аналіз можливості подальшого синтезу стратегій рішення в просторі $G^*_{ріш}$. В разі наявності можливості продовження процесу полшуку стратегій рішення за можливим напрямком переміщення в

просторі $G^{*}_{piш}$ управління передається блоку 8. В іншому випадку управління передається блоку 23.

Засобами блоку 23 визначається можливість зміни швидкості переміщення ОУ за напрямком, який аналізується. В разі можливості зміни значення швидкості переміщення ОУ обчислюється значення швидкості та формується ознака наявності можливості зміни швидкості. В разі відсутності можливості зміни швидкості ОУ при переміщенні за аналізуємим напрямком формується ознака неможливості зміни швидкості.

В блоці 24 здійснюється аналіз ознаки зміни швидкості переміщення ОУ, яку було визначено в блоці 23. За ознакою можливості зміни швидкості ОУ управління передається блоку 8, в іншому випадку - блоку 28.

В блоці 28 за відсутністю синтезованих стратегій рішення встановлюється ознака відсутності рішення і управління передається блоку 19.

Таким чином здійснюється синтез стратегій рішень за методом інтегрального усікання варіантів.

Розглянемо алгоритми роботи окремих функціональних складових алгоритму методу інтегрального усікання варіантів.

Алгоритм формування ознаки приналежності точки перерізу простору Q площиною реалізовано згідно функціональності блоку 5 алгоритму рис. 1 наведено на рис. 2.

В блоці 2 здійснюється обчислення координат лінії обмежень для зв'язаної з точкою системою координат. Тобто здійснюється перетворення простору рішення стосовно точки простору.

Блок 3 вміщує засоби обчислення кута, під яким спостерігається простір рішення з точки, яка підлягає аналізу на приналежність, при обігу границі простору рішень з початкової до кінцевої точки ламаної лінії системи обмежень.

В блоці 4 здійснюється розрахунок суми кутів.

Засоби блоку 5 обчислюють значення предикату приналежності точки простору рішень.

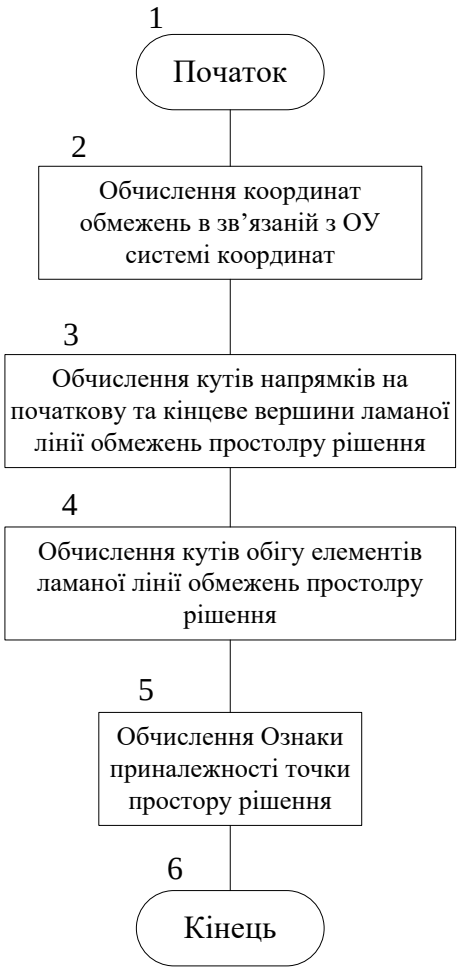


Рис. 2. Схема алгоритму формування ознаки приналежності точки обмеженому простору

В Додатку Г.2 наведено текст програмного коду, який реалізує алгоритм, який представлено на рис. 2 (subroutine raspr()).

Алгоритм доповнення обмежень простору Q при знаходженні цільової точки переміщення ОУ за його межами згідно блоку 5 алгоритму рис. 1 наведено на рис. 3.

В блоці 2 за алгоритмом рис. 3 визначається приналежність точки простору Q.

Засобами блоку 3 аналізується ознака приналежності точки простору Q. В разі приналежності точки простору Q управління передається блоку 6. В іншому випадку - блоку 4.

Доповнення простору в напрямку знаходження точки, положення якої аналізується по відношенню до обмеженого простору, здійснюється засобами блоку 4. Доповнення здійснюється шляхом додавання додаткових двох точок до обмежень простору в напрямку точки, яка знаходиться поза межами простору. Тобто, елемент ламаної лінії обмежень простору, який є найближчим до цільової точки, зміщується за напрямком, визначеним вектором, побудованим з початку координат до цільової точки переміщення ОУ. Довжина вектору зміщення елемента (відрізку) лінії обмежень має бути більшою на довільне значення щодо відстані від точки до точки перетину побудованого вектору з відрізким лінії обмежень.

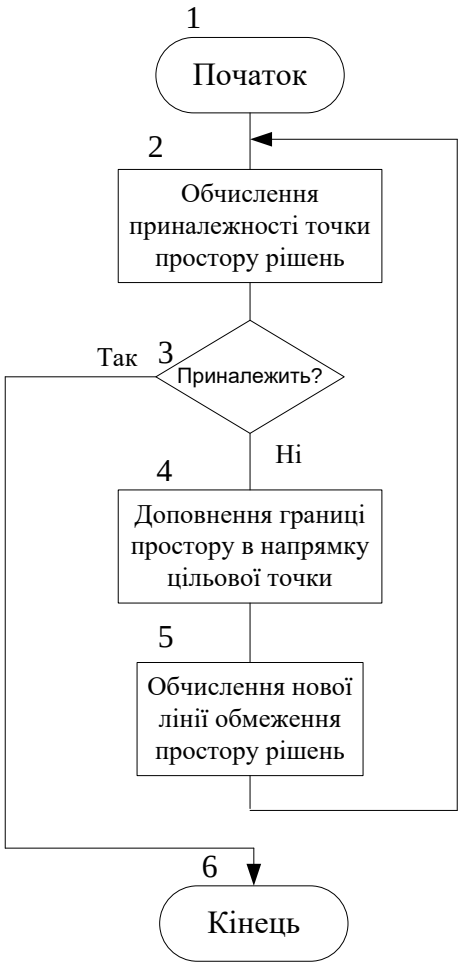


Рис. 3. Схема алгоритму доповнення обмежень простору простору Q

В блоці 5 здійснюється обчислення координат вершин нової лінії обмежень простору простору Q.

Блок 6 вміщує засоби забезпечення завершення функціонування алгоритму.

В Додатку Г.2 наведено текст програмного коду, який реалізує алгоритм, який наведено на рис. 3 (subroutine attb()).

Алгоритм визначення ознаки перетину траєкторії переміщення ОУ за синтезованою стратегією управління з обмеженнями перерізу простору Q згідно блоку 5 алгоритму (рис. 1) наведено на рис. 4.

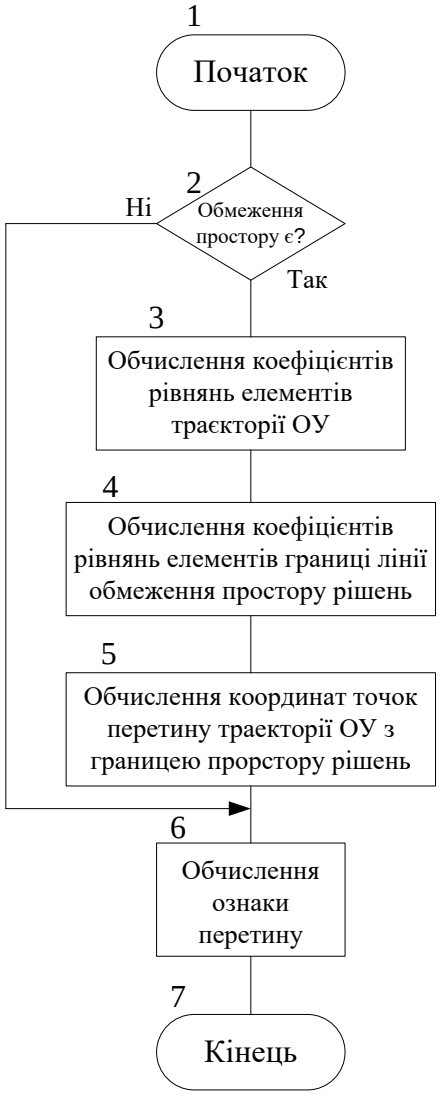


Рис. 4. Схема алгоритму визначення ознаки перетину траєкторії переміщення ОУ з обмеженнями простору рішення

Функціональними засобами блоку 2 визначається наявність обмежень щодо простору Q. При наявності обмежень управління передається блоку 6, в іншому випадку - блоку 3.

В блоці 4 обчислюються коефіцієнти рівнянь траєкторії переміщення ОУ в цільову точку переміщення.

Засобами блоку 5 обчислюються координати точок перетину елементів траєкторії переміщення ОУ з елементами лінії границі обмежень простору рішень.

В блоці 6 обчислюється значення ознаки перетину траєкторії переміщення ОУ з $\Gamma_{гр}(Q)$.

Блок 7 забезпечує завершення функціонування алгоритму.

В Додатку Г.2 наведено текст програмного коду, який реалізує алгоритм, наведений на рис. 4 (subroutine bereg()).

Алгоритм імітації СННП при перетині траєкторії переміщення ОУ за синтезованою стратегією управління з обмеженнями перерізу простору Q згідно блоку 5 алгоритму (рис. 1) наведено на рис. 5.

В блоці 2 здійснюється обчислення координат вершин елементів $\Gamma_{гр}(Q)$ при переході до зв'заної з ОУ системи координат.

Засобами блоку 3 для елементів $\Gamma_{гр}(Q)$, як для СННП, обчислюються відстані.

В блоці 4 проводиться виділення лише тих елементів $\Gamma_{гр}(Q)$, які знаходяться в зоні синтезу стратегій рішень. Тобто елементів границі, які є образами СННП.

Засоби блоку 5 здійснюють обчислення напрямків на кінці виділених елементів $\Gamma_{гр}(Q)$.

Таким чином для функціонування блоку 6 маємо всі похідні дані для формування образу СННП.

Блок 7 вміщує засоби забезпечення завершення функціонування алгоритму.

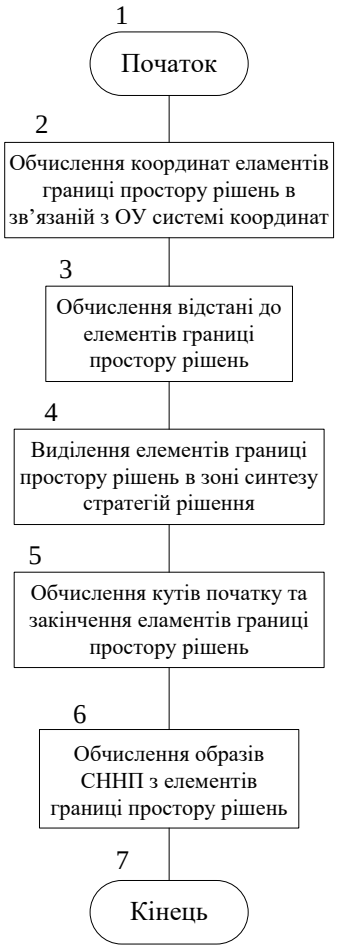


Рис. 5. Схема алгоритму імітації СННП при перетині траєкторії переміщення ОУ з обмеженнями простору рішення

В Додатку Г.2 наведено текст програмного коду, який реалізує алгоритм, наведений на рис. 5 (subroutine afro()).

Алгоритм визначення радіанної міри кута напрямку на точку в перерізі простору Q згідно блоку 25 алгоритму (рис. 1) наведено на рис. 6.

В блоці 2 за знаками координат точки обчислюється чверть координатної площини.

Засоби блоку 3 визначають значення радіанної міри кута при відрахунку від додатнього напряму вісі абсцис в напрямку проти часової стрілки.

Блок 4 вміщує засоби забезпечення завершення функціонування алгоритму.

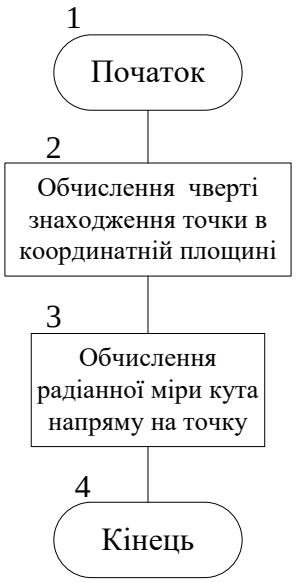


Рис. 6. Схема алгоритму визначення радіанної міри кута напрямку на точку

В Додатку Г.2 наведено текст програмного коду, який реалізує алгоритм, наведений на рис. 6 (subroutine agl()).

Алгоритм обчислення ЛШ для одного ОС згідно блоку 26 алгоритму (рис. 1) наведено на рис. 4 Додатку Г.1, а в Додатку И.2 наведено текст програмного коду, який реалізує алгоритм (subroutine stick()).

Алгоритм визначення координат кінців СННП при переміщенні ОУ в дійсній системі координат згідно блоку 26 алгоритму (рис. 1) наведено на рис. 7.

Засобами блоку 2 здійснюється оцінка ситуації переміщення ОУ та ОС за паралельними напрямками та з рівними швидкостями.

В блоці 3 для уникнення ситуації, що визначає неможливість рішення конфлікту взаємодії ОУ та ОС, швидкість та напрям переміщення ОС змінюється в межах $\pm 5\%$ від значення діапазону зміни відповідного параметру переміщення ОУ.

Засоби блоку 4 здійснюють обчислення координат точок дотику ЛВР з колом області припустимого зближення ОУ з ОС. За відсутністю рішення, що

відповідає випадку знаходження ОС у внутрішній області кола, формується ознака неможливості рішення.

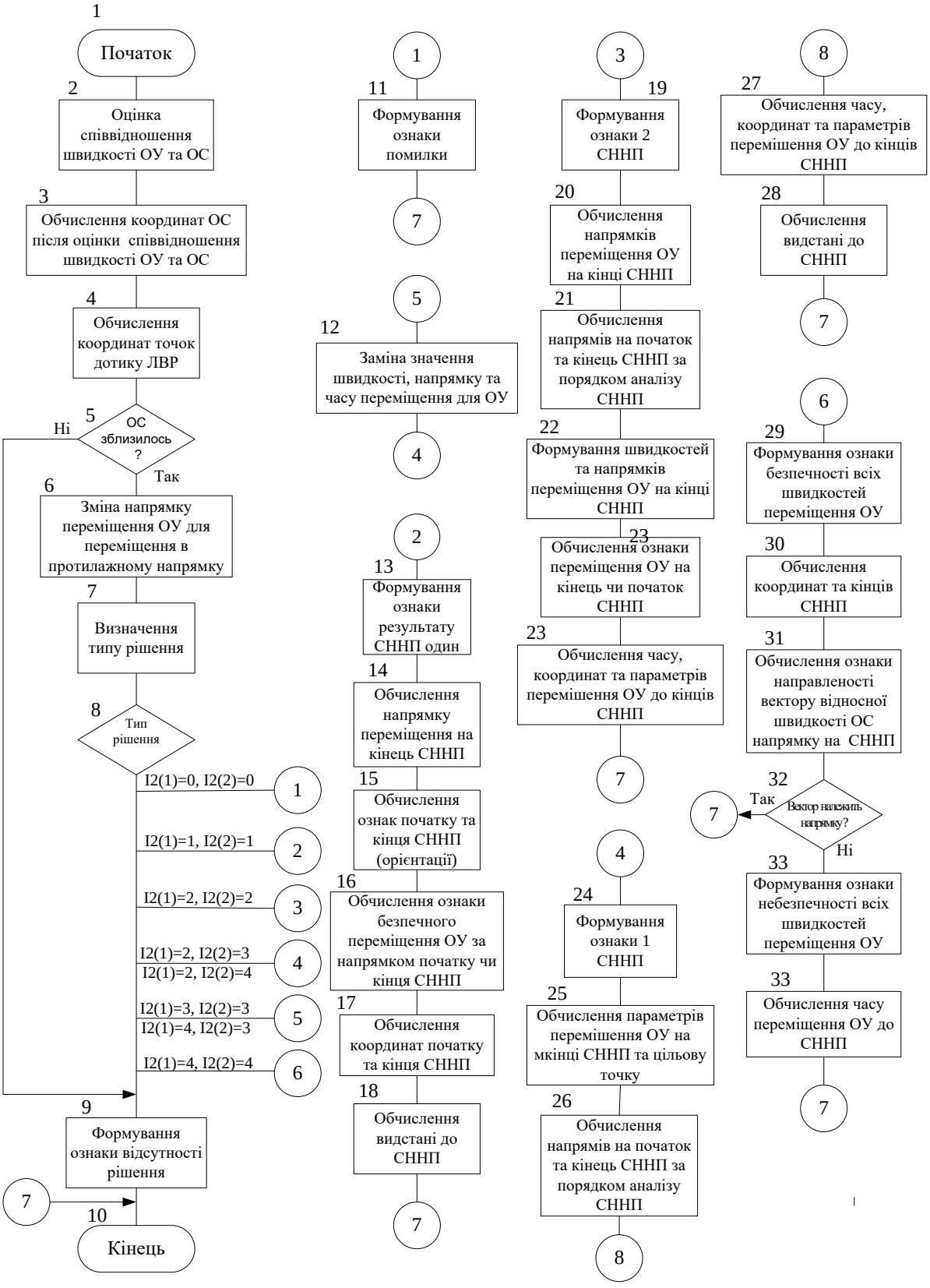


Рис. 7. Схема алгоритму визначення координат кінців СННП при переміщенні ОУ

Аналіз можливості рішення аналізується засобами блоку 5. За наявності ознаки зближення з ОС (відсутність рішення) управління передається засобам блоку 6, в іншому випадку - блоку 9.

В блоці 6 здійснюється перехід від рішення задачі ухиляння на рішення задачі убігання - переслідування. В такому разі напрям вектору переміщення ОУ змінюється на протилежний. При цьому враховується час, необхідний на здійснення дій щодо управління ОУ, за результатами яких напрямок переміщення зміниться на протилежний.

Засоби блоку 7 дозволяють визначити тип рішення, яке синтезується щодо управління ОУ в залежності від результатів функціонування засобів блоків 2 та 6.

В залежності від результатів функціонування засобів блоку 7 в блоці 8 забезпечується передача управління блокам 9, 11, 12, 13, 19, 24, 27 та 29.

При передачі управління блоку 9 функціональними складовими блоку визначається ознака відсутності рішення. В такому разі стратегія управління ОУ не може бути сформована, а опис ситуації згідно Γ_{np}^{i} та Γ_{np}^{0} для ТММ не дозволить синтезувати ланцюжок стратегій рішення $V_{j_k}^{0}$ щодо управління ОУ.

По закінченню функціонування блоку 9 управління передається засобам блоку 10, який забезпечує завершення функціонування алгоритму.

При передачі управління блоку 11 його функціональні засоби забезпечують визначення ознаки помилки щодо результатів попередніх обчислень в блоках 6 та 7, а сама ситуація відсутності управління може бути ознакою відмови в функціонуванні складових СІУ. По завершенню функціонування засобів блоку 11 управління передається блоку 10.

При передачі управління блоку 12 його функціональні засоби забезпечують опрацювання дій, пов'язаних обчисленням значень швидкості, напрямку та часу переміщення для ОУ. Далі управління передається блоку 28.

Засоби блоку 24 визначають ознаку наявності одного СННП для рішення, яке синтезується щодо управління ОУ.

В блоці 25 здійснюються процедури обчислення параметрів з метою забезпечення переміщення ОУ по траєкторії, яка орієнтована на проходження через кінці СННП до цільової точки. Тобто відсікаються варіанти рішень, які не призводять до рішення конфлікту.

Блок 26 своїми засобами забезпечує обчислення параметрів управління, для переміщення ОУ згідно синтезованого напрямку, який не передбачає знаходження в геометричному місці точок на площині (перерізі площиною) простору Q, в якому можливий конфлікт взаємодії ОУ та ОС.

Далі управління передається засобам блоку 27, в якому виконуються обчислювальні процедури щодо визначення часу, координат вершин лінії траєкторії та параметрів переміщення ОУ на кінці СННП в ПР. Тобто здійснюється синтез ланцюжка стратегії безконфліктного управління ОУ.

Функціональними засобами блоку 28 забезпечується обчислення відстані до СННП. Результати цих обчислень є похідними для подальших обчислень, пов'язаних з вибором СННП (ОС), який є найбільш небезпечним і має бути врахованим в ланцюжку стратегії рішення. Слід зазначити, що при виборі напрямку переміщення ОУ деякі СННП можуть бути "затінені" обраним СННП. Тобто рішення стає загальним до певної кількості ОС, які знаходяться в просторі $G^*_{ріш}$.

Далі управління передається засобам блоку 10.

При передачі управління блоку 29 його функціональні засоби за співвідношеннями результатів попередніх обчислень в блоках 6 та 7 забезпечують визначення ознаки безпечності всіх швидкостей, які обираються в ланцюжку стратегії управління ОУ.

Функціональні засоби блоку 30 забезпечують обчислення координат кінців СННП.

Для подальших обчислень засобами блоку 31 формується значення для ознаки направленості вектору відносної швидкості переміщення ОС множині напрямків, які визначені в ПР наявністю СННП для ОС, вектор відносної швидкості переміщення якого аналізується.

Аналіз ознаки, яка обчислена засобами блоку 31, здійснюється функціональними засобами блоку 32. У випадку, коли за ознакою вектор відносної швидкості переміщення ОС належить множині напрямків, які визначені в ПР наявністю СННП для ОС, управління передається блоку 10, в іншому випадку - блоку 33.

Функціональні засоби блоку 33 формують ознаку результату алгоритму, що визначає всі небезпечні швидкості переміщення ОУ в синтезованих ланцюжках стратегії рішення.

Блок 34 забезпечує визначення відстані до СННП аналогічно блоку 28. Після завершення процедур функціонування блоку 34 управління передається блоку 10.

В Додатку Г.2 наведено текст програмного коду, який реалізує алгоритм, який наведено на рис. 7 (subroutine ends()).

Алгоритм згідно функціональності блоку 26 (рис. 1) наведено на рис. 8.

В блоці 2 здійснюється підрахунок кількості ОС в ПР.

Формування ПКУО вільного від конфліктів взаємодії ОУ з ОС здійснюється в блоці 3.

В разі відсутності ОС в просторі Q (блок 4) ПКУО визначається без обмежень і управління передається блоку 7, тобто функціонування програмного модуля завершується.

В разі відсутності ОС в просторі Q (блок 4) ПКУО визначається без обмежень і управління передається блоку 7, тобто функціонування програмного модуля завершується.

При наявності в просторі Q ОС (блок 3) управління передається блоку 5, в якому викликається процедура (підпрограма subroutine stick()), яка

здійснює розрахунок ЛШ для кожного ОС в просторі Q. Схема алгоритму розрахунку ЛШ наведена на рис. 4 Додатку Г.1.

Після розрахунку ЛШ формується інформаційний образ ПКУО (блок 6), в якому здійснюється формування перетинів ЛШ для ОС (subroutine aria()), схема алгоритму якої наведена на рис. 5 Додатку Г.1.

Обчислення кількості та меж діапазонів безпечних швидкостей для ОУ здійснюється функціональними складовими блоку 7.

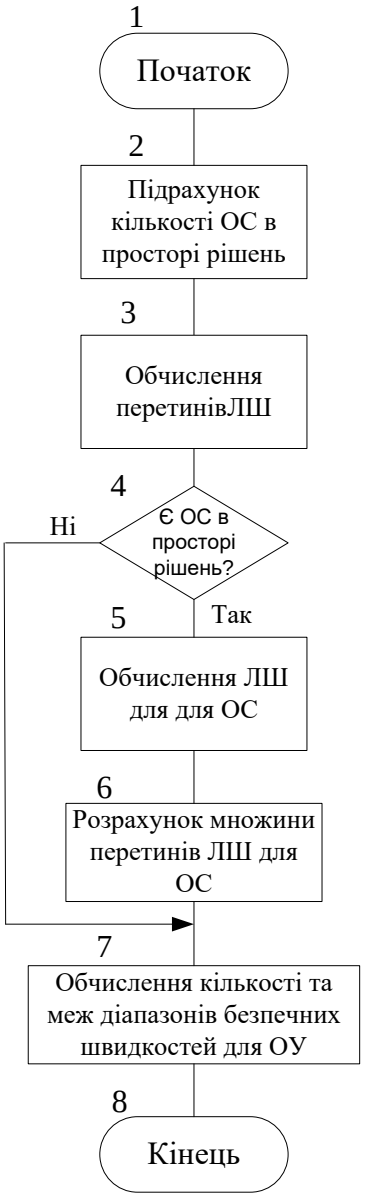


Рис. 8. Схема алгоритму побудови ЛШ при взаємодії ОУ з ОС

Блок 8 забезпечує завершення функціонування алгоритму.

В Додатку Г.2 наведено текст програмного коду, який реалізує алгоритм, представлений на рис. 8 (subroutine cross()).

Алгоритм визначення рекомендованої швидкості переміщення ОУ згідно блоку 27 алгоритму рис. 1 наведено на рис. 9.

В блоці 2 здійснюється аналіз наявності діапазонів припустимих швидкостей для напрямку переміщення ОУ. За наявності діапазонів припустимих швидкостей переміщення управління передається блоку 2, в іншому випадку - блоку 6.

Функціональні засоби блоку 6 забезпечують формування ознаки відсутності рішення та передачу управління блоку 7.

Блок 7 забезпечує завершення функціонування алгоритму.

При передачі управління блоку 3 здійснюється формування ознаки типу рішення та ініціація значень рішення, що синтезується, для початку процедури обчислення.

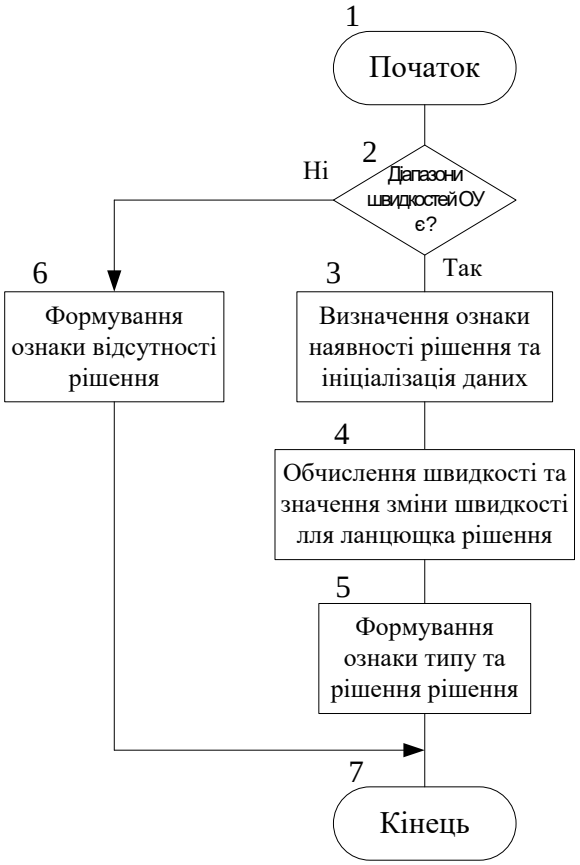


Рис. 9. Схема алгоритму визначення рекомендованої швидкості переміщення ОУ

Функціональні засоби блоку 4 здійснюють обчислення кількості значень безпечних швидкостей переміщення ОУ за напрямком та обчислюють значення зміни швидкості для елемента ланцюжка рішення, яке синтезується. За результатами функціонування блоку 4 визначається тип рішення. По закінченню функціонування блоку 4 управління передається блоку 7.

В Додатку Г.2 наведено текст програмного коду, який реалізує алгоритм, представлений на рис. 9 (subroutine sps()).

Алгоритм визначення напрямку переміщення ОУ щодо синтезу рішення при зміні напрямку для ланцюжка управління ОУ згідно блоку 8 алгоритму рис. 1 наведено на рис. 10.

В блоці 2 формується ознака відсутності помилки.

Функціональні засоби блоку 3 здійснюється аналіз ознаки зміни напрямку переміщення ОУ щодо СННП (кінець або початок). За наявності іншої ознаки зміни напрямку управління передається блоку 4. За значенням ознаки зміни напрямку переміщення управління передається блоку 6 або блоку 7.

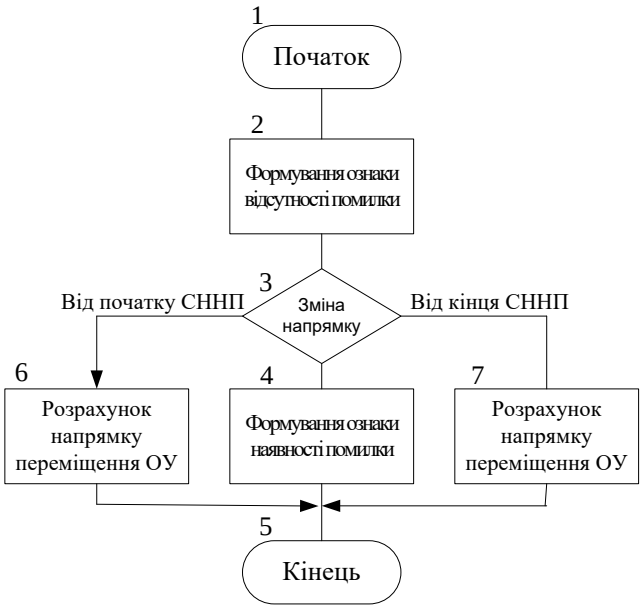


Рис. 10. Схема алгоритму визначення напрямку переміщення ОУ при синтезі рішення при зміні напрямку для ланцюжка управління ОУ

Функціональні засоби блоку 4 формують ознаку наявності помилки та забезпечують передачу управління блоку 5.

Блок 5 забезпечує завершення функціонування алгоритму.

Функціональні засоби блоку 6 забезпечують обчислення значення напрямку переміщення в напрямку додатнього відрахунку кутів (проти годинникової стрілки) від початкового напрямку переміщення ОУ на кут в проміжку $\left[0,-\frac{\pi}{2}\right]$. При цьому враховується значення зміни напрямку переміщення ОУ на кут, який перевищує значення 2π. Після закінчення обчислювальних процедур блоку 6 управління передається блоку 5.

Функціональні засоби блоку 7 забезпечують обчислення значення напрямку переміщення в напрямку від'ємного відрахунку кутів (за годинниковою стрілкою) від початкового напрямку переміщення ОУ на кут в проміжку $\left[0,\frac{\pi}{2}\right]$. При цьому враховується значення зміни напрямку на кут, який є меншим за 0. Після закінчення обчислювальних процедур блоку 6 управління передається блоку 5.

В Додатку Г.2 наведено текст програмного коду, який реалізує алгоритм, що представлений на рис. 10 (subroutine react()).

Алгоритм порівняння та формування множини альтернативних стратегій управління ОУ згідно блоку 14 алгоритму рис. 1 наведено на рис. 11.

В блоці 2 формується ознака відсутності умов завершення процесу ситнтезу стратегій за правилом зупинки та здійснюється обчислення значення критерію вибору стратегії рішення.

Функціональними засобами блоку 4 здійснюється аналіз ознаки першої синтезованої стратегії рішення. В разі першої синтезованої стратегії управління передається блоку 4. В іншому випадку - блоку 5.

В блоці 5 встановлюється ознака того, що рішення вже було синтезовано, та управління передається блоку 6.

Функціональні засоби блоку 5 порівнюють значення критерію з раніш обраним значенням. Якщо обчислене значення критерію є меншим за існуюче, управління передається блоку 6, в іншому випадку - блоку 9.

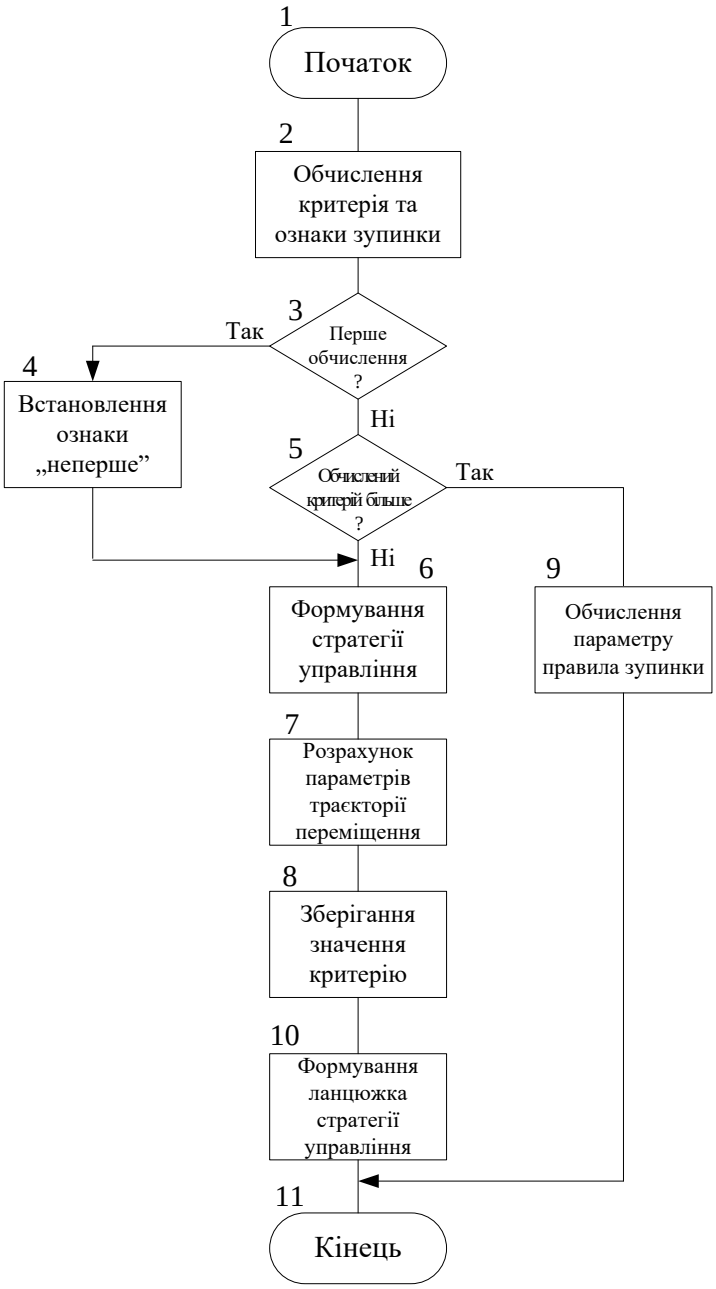


Рис. 11. Схема алгоритму визначення напрямку переміщення ОУ при синтезі рішення при зміні напрямку для ланцюжка управління ОУ

В блоці 9 для продовження пошуку стратегії оптимального за критерієм рішення обчислюється значення параметра правила зупинки і управління передається блоку 11.

Блок 11 забезпечує завершення функціонування алгоритму.

Блок 6 забезпечує формування ланцюжка стратегії управління ОУ.

Блок 7 виконує процедури обчислення параметрів траєкторії переміщення ОУ в ПР.

Блок 9 забезпечує збереження обчисленого значення критерію в разі, якщо його значення менше від попередньо збереженого.

Блок 7 виконує процедури обчислення параметрів траєкторії переміщення ОУ.

Блок 9 забезпечує збереження обчисленого значення критерію в разі, якщо його обчислене значення менше від попередньо збереженого.

Функціональні засоби блоку 10 забезпечують формування ланцюжка стратегії управління та обчислюють значення елементів траєкторії переміщення ОУ п просторі $G^{*}_{piш}$.

В Додатку Г.2 наведено текст програмного коду, який реалізує алгоритм, що представлений на рис. 11 (subroutine zapl1()).

Алгоритм розрахунку місця знаходження ОУ в ПР при одночасній зміні напрямку та швидкості переміщення ОУ згідно блоку 14 алгоритму рис. 1 наведено на рис. 12.

В блоці 2 формується ознака відсутності помилки при функціонуванні алгоритму розрахунку місця знаходження ОУ в ПР при одночасній зміні напрямку та швидкості переміщення ОУ.

Функціональні засоби блоку 3 обчислюють координати направляючого вектору напрямку переміщення ОУ, який буде по закінченні процесу управління ОУ згідно синтезованого елемента ланцюжка стратегії рішення (траєкторії переміщення).

В блоці 4 здійснюється аналіз значення часу, впродовж якого має бути здійснено зміну швидкості переміщення ОУ з одночасною зміною напрямку переміщення. Якщо значення часу здійснення функцій управління ОУ не більше значення часу на рішення задачі синтезу стратегій управління ОУ та прийняття рішення, управління передається блоку 5, в іншому випадку - блоку 9.

В блоці 9 формується ознака наявності помилки при функціонуванні алгоритму розрахунку місця знаходження ОУ в ПР при одночасній зміні напрямку та швидкості переміщення ОУ та управління передається блоку 8.

Блок 8 забезпечує завершення функціонування алгоритму.

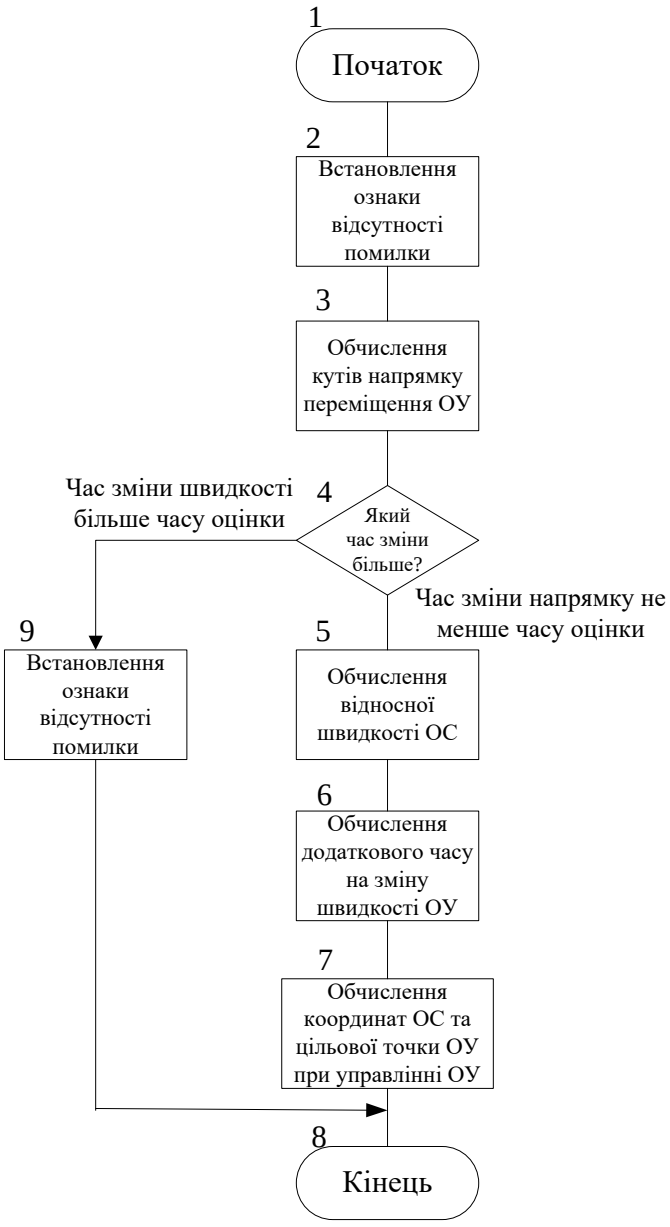


Рис. 12. Схема алгоритму розрахунку місця знаходження ОУ в ПР при одночасній зміні напрямку та швидкості переміщення ОУ

Функціональні засоби блоку 5 забезпечують обчислення відносної швидкості переміщення ОС при переміщенні ОУ за напрямком до цльової точки в ПР.

В блоці 6 здійснюється обчислення значення додаткового часу, який залишається після зміни швидкості ОУ згідно стратегії за елементом лінцюжка управління та часом, який передбачено на здійснення обчислювальних операцій по синтезу стратегії рішення щодо управління ОУ та прийняття рішення.

Блок 7 забезпечує обчислення модуля середньої швидкості переміщення ОУ під час реалізації управління, а також координати ОС в зв'язаній системі координат після реалізації управління ОУ згідно елементу ланцюжка стратегії управління ОУ (траєкторії переміщення).

В Додатку Г.2 наведено текст програмного коду, який реалізує алгоритм, що представлений на рис. 12 (subroutine sharp()).

Алгоритм розрахунку координат кінця СННП при переміщенні ОУ за умов лінійного прогнозу переміщення ОС в дійсній системі координат згідно блоку 14 алгоритму рис. 1 наведено на рис. 13.

В блоці 2 здійснюється обчислення значення часу переміщення ОС до точки зближення з ОУ на відстань d^i.

Блок 3 забезпечує обчислення координат кінця СННП при переміщенні ОУ за умов лінійного прогнозу переміщення ОС в дійсній системі координат.

Рис. 13. Схема алгоритму розрахунку координат кінця СННП при переміщенні ОУ за умов лінійного прогнозу переміщення ОС в дійсній системі координат

Блок 4 забезпечує завершення функціонування алгоритму.

В Додатку Г.2 наведено текст програмного коду, який реалізує алгоритм, що представлений на рис. 13 (subroutine dvi()).

Алгоритм формування ознаки перетину напрямків переміщення ОУ з ОС з початку чи кінця ОС згідно блоку 14 алгоритму рис. 1 наведено на рис. 14.

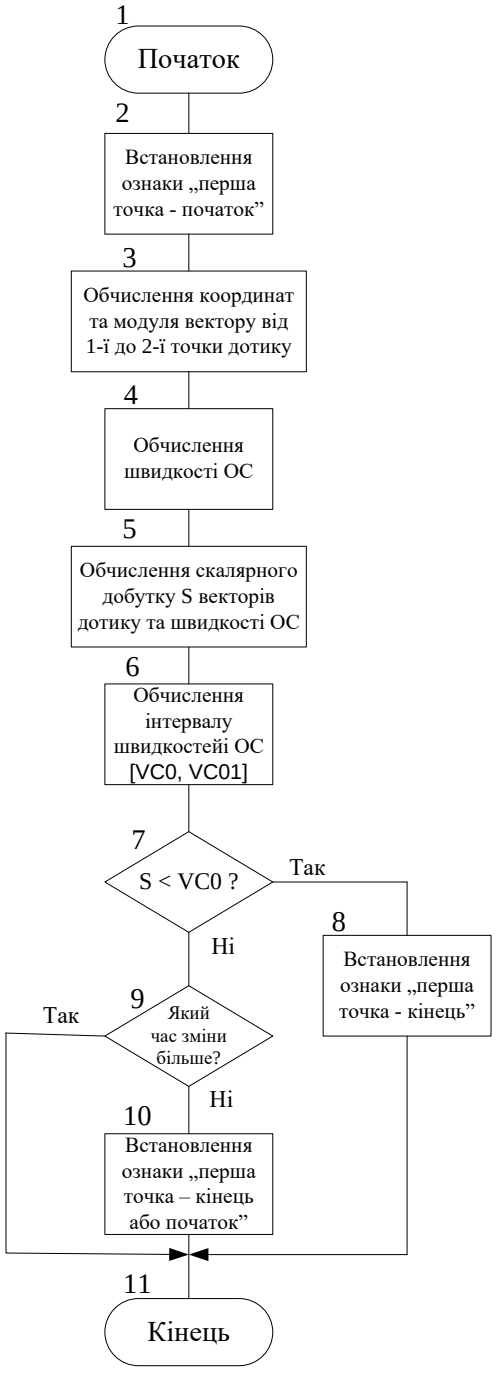


Рис. 14. Схема алгоритму формування ознаки перетину напрямків переміщення ОУ з ОС з початку чи кінця ОС

Блок 4 вміщує засоби обчислення модуля вектору швидкості ОС.

В блоці 2 формується попереднє значення ознаки перетину напрямків переміщення ОУ з ОС з початку ОС.

Функціональні засоби блоку 3 обчислюють значення координат вектору, який направлений з першої до другої точки дотику ЛВР.

Блок 5 дозволяє обчислити значення скалярного добутку вектору, який було визначено в блоці 3 та вектору швидкості ОС на модуль вектору швидкості ОС. Таким чином, обчислене значення параметру дозволяє визначити ознаку зближення ОУ з ОС.

В блоці 6 здійснюється обчислення границь інтервалу для подальшого порівняння значення швидкості і визначення ознаки зближення ОУ з ОС.

Функціональність блоків 7, 8, 9 та 10 дозволяє визначити ознаку перетину напрямків переміщення ОУ і ОС як "перша точка - кінець ОС" (блоки 7, 8) та "перша точка - кінець/початок ОС" (випадок невизначеності типу зближення).

Блок 11 забезпечує завершення функціонування алгоритму.

В Додатку Г.2 наведено текст програмного коду, який реалізує алгоритм, що представлений на рис. 14 (subroutine thick()).

Алгоритм визначення початку та кінця СННП при взаємодії ОУ з ОС згідно блоку 14 алгоритму рис. 1 наведено на рис. 15.

В блоці 2 пірівнюються числові зачення напрямків на початок та кінець СННП.

Блок 3 забезпечує формування напрямку, який був кінцем СННП, як початок, а напрямку, який був початком СННП, як кінець.

Блок 4 дозволяє визначити приналежність напрямку переміщення множині напрямків, які включають можливість перетину з СННП. В разі приналежності напрямку, який аналізується, множині напрямків на СННП управління передається блоку 5, в іншому випадку - блоку 6.

Блоки 5 та 6 забезпечують формування вихідних значень за результатами функціонування алгоритму та визначають ознаку приналежності напрямку, який аналізується, множині напрямків на СННП.

Рис. 15. Схема алгоритму визначення початку та кінця СННП при взаємодії ОУ з ОС

Блок 7 забезпечує завершення функціонування алгоритму.

В Додатку Г.2 наведено текст програмного коду, який реалізує алгоритм, щопредставлений на рис. 15 (subroutine beg()).

Алгоритм обчислення кутового значення напрямку переміщення в градусах згідно блоку 14 алгоритму рис. 1 наведено на рис. 16.

В блоці 2 здійснюється аналіз значення швидкості переміщення. Якщо значення швидкості дорівнює нулю, управління передається блоку 5, в іншому випадку - блоку 3.

В блоці 3 визначається значення направляючих синуса і косинуса вектору швидкості об'єкту.

Функціональні засоби блоку 4 дозволяють визначити кутову міру вектору швидкості в градусах при відрахунку в додатньому напрямку від вісі абсцис (протилежно напрямку годинникової стрілки).

Рис. 16. Схема алгоритму обчислення кутового значення напрямку переміщення в градусах

Блок 5 забезпечує завершення функціонування алгоритму.

В Додатку Г.2 наведено текст програмного коду, який реалізує алгоритм, що представлений на рис. 16 (subroutine ancl()).

Алгоритм обчислення ознаки взаємного розташування векторів згідно блоку 14 алгоритму рис. 1 наведено на рис. 17.

В блоці 2 визначається ознака "належить області обмежень".

Функціональні засоби блоку 3 обчислюють матрицю векторів, які замикають відрізок лінії обмежень.

В разі отримання нульового значення матриці, яка обчислена в блоці 3, управління передається блоку 9, в іншому випадку - блоку 5.

Засоби блоку 9 визначають ознаку "помилка в похідних даних" та передають управління блоку 8.

Блок 8 забезпечує завершення функціонування алгоритму.

Функціональні засоби блоку 5 обчислюють значення матриць напрямків на обмеження та вектору швидкості ОС та часток від ділення цих значень на значення матриці векторів, які замикають відрізок лінії обмежень (блок 3).

В блоці 6 здійснюється аналіз знаків значень часток, які були обчислені в блоці 5.

Рис. 17. Схема алгоритму обчислення ознаки взаємного розташування векторів

Якщо будь-яка частка є від'ємною, функціональні засоби блоку 7 встановлюють ознаку знаходження вектору швидкості ОС поза множиною обмежень. По закінченню функціонування блоку 7 управління передається блоку 8.

В Додатку Г.2 наведено текст програмного коду, який реалізує алгоритм, що представлений на рис. 17 (subroutine comp()).

Алгоритми функціонування основних модулів програмних засобів методу інтегрального усікання варіантів наведені на рис. 1 - 17.

В Додатку Г.2 наведено тексти програмного коду повного проекту методу інтегрального усікання варіантів та синтезу стратегій управління ОУ за умов

обмежень та невизначеності в середовищі розробки MS Visual Studio 2010 з інтегрованими засобами PGI Fortran 12.10 та акселератором.

Г.2 - програмні засоби інформаційної технології методу інтегрального усікання варіантів та синтезу стратегій рішень

```
!
!  ConsoleApp.f90
!
!  Fortran Console Application
!  Generated by PGI Visual Fortran(R)
!  9/7/2013 1:39:35 PM
!
      program prog

      implicit none

      ! Variables
      ! Body


!Программа вызова  <MPRT1>
!вариант программы для оу
!редакция:31.10.13
!_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
 dimension ogr(10,2),ogl(10,2),iflag(10),rt(3),rv(3),rp(3),
*ddop(10),gr(4,2),c(10,2),a2(10,2),buf(2,2)
 data vn,psy,tup,px,py,zon/12.,5.619922,0.1,28.,-3.2,32./
 data nl,ogl(1,1),ogl(1,2),ogl(2,1),ogl(2,2),ogl(3,1),
:ogl(3,2)/3,-32.,-32.,4.,2.,32.,-32./
 data nr,ogr(1,1),ogr(1,2),ogr(2,1),ogr(2,2),ogr(3,1),
:ogr(3,20,ogr(4,1),ogr(4,2)/4,-22.,16.,-4.3,4.3,17.6,20.3,
:32.,14./
3 format(1x,'скорость и направление перемещения оу '/1x,
 *'?'*)
  accept 9,ivn,ipsy,
  xn=vn*cos(psy)
  yn=vn*sin(psy)
  xmin=-32.
  ymin=-32.
  i1=1
  call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
  i1=2
  call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
  hy=64./339.
  hx=64./559.
  nn=2
  n=2
  il=1
  buf(1,1)=0.
  buf(1,2)=-32.
  buf(2,1)=0.
  buf(2,2)=32.
  call pict(buf,hy,hx,xmin,ymin,n,il,nn)
  buf(1,1)=-32.
  buf(1,2)=0.
  buf(2,1)=31.
  buf(2,2)=0.
  call pict(buf,hy,hx,xmin,ymin,n,il,nn)
  i1=7
  i2=0
  call term1(i1,i2,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
  i1=10
  call term1(i1,i2,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
  buf(1,1)=xn
  buf(1,2)=yn
  buf(2,1)=0.
  buf(2,2)=0.
  call pict(buf,hy,hx,xmin,ymin,n,il,nn)
  ir=5
  call perevd(ir,ir1,ir2,ir3)
  i1=4
  call term1(i1,il,iy1,iy2,iy2,ix1,ix2,ix3,ir1,ir2,ir3)
  type 5
5 format(1x,'наибольшая собственная скорость оу[f8.3]: '*)
  accept 6,vmax
```

```
6 format(f8.3)
  type 24
24 format(1x',глубина оценки ситуации [f8.3] <единиц расстояния>: '*)
   accept 6,zon
   ir=zon/hx
   call perevd(ir,ir1,ir2,ir3)
   call term1(i1,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
   type 8
8 format(1x,'целевая точка маршрута [пеленг, дальность]'/1x,
 *' [2f8.3]: '*)
 accept 9,p,d
9 format (2f8.3)
  p1=3.141592*p/180.
  px=d*cos(p1)
  py=d*sin(p1)
  ix=abs((px-xmin)/hy)
  iy=abs((py-ymin)/hx)
  call perevd(ix,ix1,ix2,ix3)
  call perevd(iy,iy1,iy2,iy3)
  i1=3
  call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
  i1=4
  il=1
  do 60 i=1,8
  call perevd(i,ir1,ir2,ir3)
  call term1(i1,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
60 continue
   type 10
10 format(1x,'коэффициенты критерия отбора [4f8.3]: '/1x,'?'*)
   accept 11,rk1,rk2,rk3,rk4
11 format (4f8.3)
   type 12
12 format(1x'вершин левой границы: '*)
   accept 4,nl
4 format(i2)
  do 13 i=1,nl
  type 14,i
14 format(1x,'точка ',12,' [2f8.3]: '*)
   accept 9,ogl(i,1),ogl(i,2)
   ix=abs((ogl(i,1)-xmin)/hy)
   iy=abd((ogl(i,2)-ymin)/hx)
   call perevd(ix,ix1,ix2,ix3)
   call perevd(iy,iy1,iy2,iy3)
   if(i.eq.1) go to 61
   i1=3
   go to 62
61 i1=5
62 call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
13 continue
   type 15
15 format(1x,'вершин правой границы: '*)
   accept 4,nr
   do 16 i=1,nr
   type 14,i)
   accept 9,ogr(i,1),org(i,2)
   ix=abs((ogr9i,1)-xmin)/hy)
   iy=abs((ogr(i,2)-ymin)/hx
   call perevd(ix,ix1,ix2,ix3)
   call perevd9iy,iy1,iy2,iy3)
   if(i.eq.1) go to 63
   i1=3
   go to 64
63 i1=5
64 call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
16 continue
   type 17
17 format(1x,'данные о объектах наблюдения')
   do 18 i=1,10
   type 20
20 format(1x,'iflag='*)
   accept 4,iflag(i)
   if(iflag(i).eq.0) go to 18
   type 19,i
19 format(1x,'цель ' ,12,' пеленг, дальность [2f8.3]: ' /1x, '?'*)
   accept 9,p,d
```

```
      p1=p*3.141592/180
      a2(i,1)=d*cos(p1)
      a2(i,2)=d*sin(p1)
      buf(2,1)=a2(i,1)
      buf(2,2)=a2(i,2)
      type 21
   21 format(1x,'допустимая <dkr> [f8.3]: '*)
      accept 6,ddop(i)
      type 23
   23 format(2x,'отн.скорость и направление перемещения он [2f8.3]: '*)
      accept 9,vr,pr
      pr=pr*3.141592/180.
      c(i,1)=vr*cos(pr)+vn*cos(psy)
      c(i,2)=vr*sin(pr)+vn*sin(psy)
      buf(1,1)=buf(2,1)+c(i,1)
      buf(1,2)=buf(2,2)+c(i,2)
      call pict(buf,hy,hx,xmin,ymin,n,il,nn)
      ir=ddop(i)/hx
      call perevd(ir,ir1,ir2,ir3)
      i1=4
      call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
   18 continue
  150 irez=4
       do 151 i=1,4
       do 151 j=1,2
  151 gr(i,j)=0.
       call posit(ogr,ogl,nr,nl,gr,a2,c,xn,yn,zon,ddop,iflag,irez)
       type 152
  152 format(1x,' для повторного воода данных введите *)
       accept 4,l1l
       if(l1l.eq.1) go to 46
       type 153
  153 format(1x,'учет он вне пространства-ввод <1>: '*)
       accept 4,isud
       call mprt(ogr,ogl,nr,nl,iflag,zon,psy,rt,rv,rp,ddop,vn,vmax,
      *tup,irez,gr,c,px,py,a2,rk1,rk2,rk3,rk4,ierr,isud)
       if(ierr.eq.0)type 25
   25 format(1x,'ошибки нет')
      if(ierr.eq.1)type 26
   26 format(1x,'обнаружена ошибка')
      go to 100
   40 call rezlt2(gr,rt,rp,rv,irez)
  100 type 45
   45 format(1x,'для выхода введите <1>: '*)
       accept 4,l
       if(l.ne.1)go to 46
       stop end



       subroutine mprt(ogr,ogl,nr,nl,iflag,zon,psy,rt,rv,rp,
      *ddop,vn,vmax,tup,irez,gr,c,px,py,a2,rk1,rk2,dk3,dk4,ierr,isud)
!      п/п расчета траектории перемещения [ вариант для оу ]
!      ogr(10,2),ogl(10,2) - ограничения пространства наблюдения [отн.сев.]
!      nr,nl - к-во точек массивов <ogr> и <ogl>
!      iflag(10) - массив признаков сопровождения
!      zon - глубина оценки ситуации [единицы расстояния]
!      isud - не учитывать(0), учитывать(1) он вне пространства
!      psy - собственный курс (направление перемещения) [рад.]
!      pt(3),rv(3),rp(3) - массивы времен, скоростей и направлений
!      doop(10) - массив допустимых дистанций сближения по он
!      vn - модуль скорости оу
!      vmax- наибольшая скорость оу [единицы скорости]
!      tup - время упреждения [часов]
!      irez - тип полчаемого результата:
!                1 - результата нет
!                2 - прямой приход
!                3 - двухэлементный маневр
!      gr(4,2) - массив координат точек, определяющих траекторию (маршрут)
!      c(10,2) - массив векторов ист.скоростей он
!      px,py - координаты целевой точки
!      a2(10,2) - масиив координат отметок ос [отн.север]
!      rk1,rk2,rk3,rk4 - коэффициенты критерия отбора
!      ierr - признак ошибки [0 - нет, 1 - есть]
!      редакция: 19.10.13.
```

```
!   ==========================================================================
    dimension ogr(10,2),ogl(10,2),iflag(10),rt(3),rp(3),
   * ddop(10),gr(4,2),c(10,2)a3(10,2),a4(10,2)isb(100,a2(10,2),
   * ogr1 (10,2),ogl1(10,2),kr(9),kl(9),lv(40),iw(40,2),
   * tv(2,2,2),tvt(2,2,2),ets(2,2,2)
   * a(2,2),t(2,2),dl(2)
    type 925
 925 format(30x, ' ! внимание! ' /15x, ' ориентировочное',
   ::' промежуточное?????????????????????????? ,')
      nold=0
       nrn=nr
       tnul=0.
       iend=0
       itl=0
       ld=0
       n=0
       nld=0
       dbsy=0,853758
       irez=1
       ierr=0
       xn=vn*cos(psy)
       yn=vn*sin(psy)
       dk=rk1+rk2
       if(dk.eq.00go to 300
       dk1=rk1/dk
       dk2=rk2/dk
!    _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!    пересчет внешней обстановки
 530 do 1 i=1,10
     isb(i)=iflag(i)
       if(iflag(i).eq.0)go to 1
       if(isud.eq.1) go to 535
       a2(i,1)=a2(i,1)+(c(i,1) - xn)*tup
       a3(i,2)=a2(i,2)+(c(i,2)-  yn)*tup
       az=sqrt(a2(i,1)*a2(i,1)+a2(i,2))*a2(i,2))
       if(az.gt.zon)isb(i)=0
 535 id=id+isb(i)
   1 continue
     xnt=xn*tup
       ynt=yn*tup
       if(iend.eq.q)go to 550
       gr(1,1)=0.
       gr(1,2)=0.
       gr(2,1)=xnt
       gr(2,2)=ynt
       gr(3,1)=px
       ggr(3,2)=py
       gr(4,1)=px
       gr(4,2)=py
       s1=sqrt(xnt*xnt+ynt*ynt)
       sp=sqrt(px*px+py*py)
       px1=px-xnt
       py1=py-ynt
!    пересчет берегов
 520 do 6 i=1,nr
     ogr1(i,1)=ogr(i,1)-xnt
       ogr1(i,2)=ogr(i,20=ynt
   6 continue
     do 7 i=1,nl
       ogl1(i,1)=ogl(i,1)-xnt
       ogl1(i,2)=ogl(i,2)-ynt
   7 continue
     if(iend.eq.1)go to 540
!  _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!  расчет требуемого направления (курса) перемещения
   call agl(px1,py1,ang,ia1)
   if(ia1.eq.1)go to 300
!  - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   go to 10
11 if(id.ne.0)go to 2

!  выход при отсутствии судов

 108 irez=2
     nm=2
```

```
      rv(1)=vn
      rv(2)=vn
      rv(3)=vn
      go to 12
 14   nom=3
      go to 42
!     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
!     пересчет координат перед выходом
!     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
!     формирование рузельтата
  50  rt(1)=tup*60.
      do 4 i=1,nm
       irez2=i+1
       ax=gr(irez2,2)-gr(i,1)
       ay=gr(irez2,2)-gr)i,2)
       rt(i)=sqrt(ax8ax+ay*ay)/rv(i)
       call agl (ax,ay,rp(i),ia1)
       if(ia1.eq.1)go to 300
   4  continue
      iend=1
       tp1=tup
       tup=0.05
       go to 530
 550  tup=tp1
      go to 520
 540  type 930
 930  format(1x,'для ввыода на графический экран введите <1>:'*)
      accept 931,ipra
 931  format(i10
      if(ipra,ne.1)go to 500
       call posit(ogr1,ogl1,nr,nl,gr,a3,c,xn,yn,zon,ddop,iflag,irez)
       go to 500
!     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
!     проверка принадлежности (x0,y0), (xnt,ynt), (px,py)
!                          пространству решений
!
  10  call raspr(ogl,ogr,nl,nr,xnt,ynt,inp,ie1)
      call attb(ogr,ogl,px,py,nr,nl,inp1)
       call attb(ogr,ogl,gr(1,1),gr(1,2),nr,nl,inp2)
       if((inp.ne.0).or.(inp1.ne.0).or.(inp2.ne.0))go to 570
       go to 11
!     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
!     проверка рекомендуемой линии траектории на принадлежность
!     пространству решений [отсутствие он или прямой проход]
  12  do 13 i=1,nm
      i1=i+1
       call bereg(ogr,ogl,nr,nl,gr(i,1),gr(i,2),gr(i1,1),gr(i1,2),iber,
     *kr,kl)
      if(iber.eq.0) go to 13
       call afro(ogr,ogl,gr(i,1),gr(i,2),gr(i1,1),gr(i1.2),nr,nl,kr,kl,
     *ea1,ea2,dal,ierr)
      if(ierr.eq.1) go to 300
       if(itl.ne.0)go to 165
       go to 14
  13  continue
      if(itl.ne.0)go to 42
       go to 50
 570  nr=nrn
      nl=nln
       go to 500
!     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
!     отыскание прохода для случая отсутствия он или прямого
!     прохода по причине пересечения с границей пространства наблюдения
!                     фрагмент из <kriter>
  42  go to(90,91,91),nom
  90  l=2
  16  nld=nld+1
      nv=1
 310  an=react(ang,ea1,ea2,dpsy,l,nv,ier1)
      if(ier1.eq.1)go to 300
       if(ier1.eq.2)go to 350
       delp1=dpsy*nv
       idt=1
       xxn=vn*cos(an)
       yyn=vn*sin(an)
```

```
      call cross(vmax,xxn,yyn,c,a3,ddop,lv,iw,nw,isb,ier1)
      if(ier1.eq.1)go to 305
      if(nw.eq.0)go to 305
      call sps(vn,nw,iw,vtek1,dv1,vtek2,dv2,vd1,vd2,vd3,vd4,nsps)
      if(nsps.eq.0)go to 305
 15   dtidt=0.05*idt
     xxnd=xxn*dtidt
      yynd=yyn*dtidt
      vtek=vtek1
      do 100 i=1,10
      if(isb(i).eq.0)go to 100
      a4(i,1)=a3(i,1)+(c(i,1)-xxn)*dtidt
      a4(i,2)=a3(i,2)+(c(i,2)-yyn)*dtidt
 100  continue
     px2=px1-xxnd
      py2=py1-yynd
!     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
!     проверка предлагаемой линии траектории на пересечение с
!              границами пространства наблюдения

     call bereg(ogr1,ogl1,nr,nl,tnul,tnul,xxnd,yynd,iber1,kr,kl)
      call bereg(ogr1,ogl1,nr,nl,xxnd,yynd,px1,py1,iber2,kr,kl)
      if((iber1.eq.0).and.(iber2.eq.0))go to 103
!     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
!     наращивание времени прогноза
  105 idt=idt+1
      n=n+1
!     правило останова
      nind=n-nold
       if((nold.ne.0).and.(nind.ge.100))go to 540
       if(idt.le.15)go to 15
!     наращивание величины угла
  305 nv=nv+1
      go to 310
!     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
!     обработка случая непересечения с границами
  103 if(id.ne.0)go to 104
      nm=3
       gr(3,1)=xnt+xxnd
       gr(3,2)=ynt+yynd
       irez=3
       go to 50
!     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
!     обработка случая наличия сопровождаемых он [id#0]
  104 call agl(px2,py2,an3,ia1)
      nm=3
       if(ia1.eq.q)go to 300
  140 x2n=vtek*cos(an3)
      y2n=vtek*sin(an3)
       call cross(vmax,x2n,y2n,c,a4,ddop,lv,iw,isb,ier1)
       if(ier1.eq.q)go to 105
       if(nw.eq.0)go to 105
       call sps(vtek,nw,iw,vtek3,dv3,vtek4,dv4,vd5,vd6,vd7,vd8,n2ps2)
!      - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
!     блок коммутации переходов
      if(ld.eq.2)go to 142
       if(ld.eq.3) go to 145
       if(ier1.eq.1)go to 300
       if(ier1.eq.2)go to 350
       delp1=dpsy*nv
       idt=1
       xxn=vn*cos(an)
       yyn=vn*sin(an)
       call cross(vmax,nw,iw,vtek3,dv3,vtek4,vd5,vd6,vd7,vd8,nsps2)
!     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
_
!     блок коммутации переходов
       if(ld.eq.2)go to 142
       if(ld.eq.3)go to 145
       if((ld.eq.5).and.(nsps2.eq.0)) go to 170
       if(ld.eq.5)go to 145
!     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
_
!       формирование маршрута по критерию отбора
!       соответствует случаю nsps=1, nsps2 = 1
```

```
   145 rtek=vtek3
       if((ld.eq.3).or.(ld.eq.5))go to 143
        delp2=abs(an3 - an)
        if(delp2.gt.3.141592)delp2 = abs(6.283184-delp2)
        s2=sqrt(xxnd*xxnd+yynd*yynd)
        s3=sqrt(px2*px2+py2*py2)
        rj3=abs(1.-sp/(s1+s2+s3))
!     - - -
```

```
!    - - -
```

```
      nom1=nom
      dal1=dal
      go to 108
  165 if(dal1.gt.dal)go to 14
      ea1=ta1
      ea2=ta2
      nom=nom1
      nm=3
      go to 42
  300 ierr=1
  500 return
      end


      subroutine rezlt2(g,rt,rp,rv,irez)
!     программа выдачи результатов расчета траектории (маршрута)
!     g(4,2) - координаты вершин графа
!     rt(3) - массив времен движения по ветвям графа
!     rp(3) - массив курсов движения по ветвям графа
!     rv(3) - массив скоростей движения по ветвям графа
!     irez - признак результата:
!                 1 - результата нет
!                 2 - одноэлементный маневр
!                 3 - двухэлементный маневр
!     редакция: 25. 02. 13.
!
!=======================================================================
      dimension g(4,2),rt(3),rp(3),rv(3)
      type 3
    3 format(10x,'режим синтеза траектрии (прокладки маршрута)'
      go to(50,30,40),irez
   26 ipsy=180.*rp(1)/3.141592
      ivn=rv(1)
      type 62,ipsy,ivn
   62 format(10x,' направление (курс): ',i3,'  скорость:   ',i2,
     *' единиц скорости')
      itup=60.*rt(1)-3.
      type 60,itup
   60 format(10x,' запас времени на маневр: ',12,' мин.')
      type 66,g(2,1),g(2,2)
   66 format(10x,' начало таектории (маршрута) [' ,f5.1,',',f5.1,'] ')
      go to 50
   30 type 31,g(3,1),g(3,2)
   31 format(10x,' координаты целевой точки: [',f5.1,',',f5.1,']')
      irp=rp(2)*180./3.141592
      irv=rv(2)
      irt=rt(2)*60.
   42 type 32,irp,irv,irt
   32 format(10x,'курс: ',i3,' скорость: ',i2,' единиц скорости время ',
     *'перемещения: ',i3,' минут')
      go to 26
   40 type 31, g(4,1),g(4,20
      irp=rp(2)*180./3.141592
      irv=rv(2)
      irt=rt(2)*60.
      type 32,irp,irv,irt
      type 41, g(3,1),g(3,2)
   41 format(10x,' координаты точки изменения направления: [',f5.1,',',f5.1,'j')
      irp=rp(3)*180./3.141592
      irv=rv(3)
      irt=rt(3)*60.
      go to 42
   50 return
      end


      subroutine sharp(x2,y2,xc,yc,vn1,vn2,pe,xr,yr,t1,t2,ind)
      п/п просчета положения судна при переходе с начального курса
!     к курсу pe при одновременном изменении скорости от vn1 до vn2
!     x2,y2 - координаты последней отметки он (цели) [единиц расстояния]
!     xc,yc - координаты вектора ист.скорости оу [единиц скорости]
!     vn1 - скорость оу в начале маневра [единиц скорости]
!     vn2 -скорость оу в тконце маневра [единиц скорости]
!     предполагается, что направление оу (курс) изменяется мгновенно
```

```
!     pe - направление (курс) в конце маневра [рад]
!     xr,yr - координаты он (цели) в конце маневра [отн.<север>]
!     t1 - время выполнения маневра скоростью [час]
!     t2 - время упреждения для пересчета обстановки [час]
!     ind - признак ошибки [0- нет, 1 - есть]
!     редакция: 01. 06. 13.
!
================================================================================
============
      ind=0
       cp=cos(pe)
       sp=sin(pe)
       if(t1.gt.t2)go to 1
       vx2=xc-vn2*cp
       vy2=yc-vn2*sp
       t=t2-t1
       vs=(vn2+vn1)/2.
       vx=xc-vs*cp
       vy=yc-vs*sp
       xr=x2+vx*t1+vx2*t
       yr=y2+vy*t1+vy2*t
       go to 2
   1  ind=1
   2  return
      end


       subroutine ends(x2,y2,ddop,xc1,yc1,vn,tv,tvt,ets,a,t,dal,ind,nk)
!     п/п вычисления ккоординат концов секторов в ист.движении
!     x2,y2 - координаты последеней засечки он (цели)
!     ddop1 - допустимая дистанция сближения
!     xc1,yc1 - координаты вектора цели
!     ind1=1 - точка(x2,y2) внутри или на окружности
!     ind=0 - ошибки в данных нет/точка (x2,y2) вне окружности/
!     ind2=0 - погрешности данных измерений или ручного ввода
!     ind2=1 - vr>0, vr<0
!     ind2=2 - vr>0, vr>0
!     ind2=3 - vr<0, vr<0
!     ind2=4 - существует 1 вектор vr или vr man
!     ind - признак характера решения
!         - 0 - ошибка(t.2 внутри или на окружности)
!         - 1 - 1 сектор
!         - 2 - 2 сектора
!         - 3 - все скорости опасны
!         - 4 - все скорости безопасны
!     nk - признак расхождения поносу или корме
!        1 - н-к-к-н
!        2 -к-н-н-к
!        3 -нк-нк-нк-нк
!        4 - к-к
!     vn1 - модуль вектра скорости оу (единиц скорости)
!     rez - массив промежуточный для хранения vn.повер. (единиц скорости)
!     tv - массив результатов вычисления vn.повер. (единиц скорости)
!     vt - массив промежуточный для хранения  v отн. повер. (единиц скорости)
!     tvt - массив результатов для хранения v отн. повер. (единиц скорости)
!     ets - координаты сектора в абс.системе координат (единиц расстояния)
!     a(i,j) - массив углов (град)
!            i - номер сектора
!            j=1 - начало сектора
!            j=2 - конец сектора
!     t(i,j) - массив времени выхода на конец сектора (час)
!            i - номер сектора
!            j=1 - начало сектора б j=2 - конец сектора
!     dal(i) - массив дальностей до сектора (единиц расстояния)
!            i - номер сектора
!     редакция: 27.12.13
!
================================================================================
====================
      dimension rez(2,2),vt(2,2),r(2,2),tv(2,2,2),tvt(2,2,2)
     *ets(2,2,2),a(2,2),t(2,2),dal(2),i2(2)
      sq(as,bs)sqrt(as*as+bs*bs)
       nk=4
!     проверка условия: 0.95*vn<vc<1.05*vn
      xc=xc1
```

```
      yc=yc1
      vc=sq(xc,yc)
      vn0=0.95*vn
      vn01=1.05*vn
      c=0.
      if((vn0.le.vc).and.(vc.lt.vn))go to 170
      if((vn.le.vc).and.(vc.lt.cn01))go to 180
      go to 140
 170 c=vn/vc
     xc=0.95*c*xc
      yc=0.95*c*yc
      go to 140
 180 c=vn/vc
     xc=1.05*c*xc
      yc=1.05*c*yc
!    обнуление массивов dal,a,tv,tvt,ets
 140 do 130 k1=1,2
     dal(k1)=0.
      do 130 k2=1,1
      a(k1,k2)=0.
      t(k1,k2)=0.
      do 130 k3=1,2
      tv(k1,k2,k3)=0.
      tvt(k1,k2,k3)=0.
 130 ets(k1,k2,k3)=0.
!    вычисление координат точек касания с окружностью d
     call line(x2,y2,doop,r,ind1)
      if(ind1.eq.1)go to 111
!    получение исходных данных о секторе
     do 2 k3=1,2
      call rotat(vn,xc,yc,x2,y2,r(k3,1),r(k3,2),rez,vt,ind2)
      i2(k3)=ind2
      do 2 k1=1,2
      do 2 k2=1,2
      tv(k3,k1,k2)=rez(k1,k2)
      tvt(k3,k1,k2)=vt(k1,k2)
   2 continue
     if((i2(1).eq.0.).or.(i2(2).eq.0.))go to 111
      if((i2(1).eq.1.).and.(i2(2).eq.1.))go to 10
      if((i2(1).eq.2.).and.(i2(2).eq.2.))go to 20
      if((i2(1).eq.2.).and.(i2(2).eq.3.))go to 40
      if((i2(1).eq.3.).or.(i2(2).eq.2.))go to 1
      if((i2(1).eq.2.).or.(i2(2).eq.4.))go to 40
      if((i2(1).eq.4.).or.(i2(2).eq.2.))go to 1
      if((i2(1).eq.4.).or.(i2(2).eq.4.))go to 50
      ind=4
      go to 200
 111 ind=0
     go to 200
!    замена местами массивов r,tv и tvt для т.1 и 3
   1 do 3 k1=1,2
     bf=r(1,k1)
      r(2,k1)=bf
      do 3 k2=1,2
      bf=tv(1,k1,k2)
      tv(1,k1,k2)=tv(2,k1,k2)
      tvt(2,k1,k2)=bf
   3 continue
     go to 40
!    расчет 1 сектора 1-3 обл.1
  10 ind=1
     b1=ancl(vn,tv(1,1,1),tv(1,1,2))
      b2=ancl(vn,tv(2,1,1),tv(2,1,2))
      b=ancl(vn,tv(1,2,1),tv(1,2,2))
      b0=b1
      call beg(2,b1,b2,b,a(1,1),a(1,2))
      if(bo.eq.a(1,1))go to 11
      do 12 k1=1,2
      bf=tv(1,1,k1)
      tv(1,1,k1)=tv(2,1,k1)
      tv(2,1,k1)=bf
      bf=tvt(1,1,k1)
      tvt(1,1,k1)=tvt(2,1,k1)
      tvt(2,1,k1)=bf
      bf=r(1,k1)
```

```
      r(1,k1)=r(2,k1)
      r(2,k1)=bf
   12 continue
      call thick(r,xc,yc,nk)
   11 do 13 k1=1,2
      call dvi(ets(k1,1,1),ets(k1,1,2),tp,r(k1,1),x2,y2,
     *tvt(k1,1,1),tvt(k1,1,2),tv(k1,1,1),tv(k1,1,2))
      t(1,k1)=tp+0.05
   13 continue
   15 do 14 k1=1,2
      d1=sq9ets(1,k1,1),ets(1,k1,2))
       d2=sq(ets(2,k1,1),ets(2,k1,2))
       dal(k1)=d2
       if(d1.le.d2)dal(k1)=d1
   14 continue
      go to 200
!     расчет секторов 1-3 и 1'-3'
   20 ind=2
      b1=ancl(vn,tv(1,1,1),tv(1,1,2))
       b2=ancl(vn,tv(1,2,1),tv(1,2,2))
       b3=ancl(vn,tv(2,1,1),tv(2,1,2))
       b4=ancl(vn,tv(2,2,1),tv(2,2,2))
       b01=b1
       b03=b3
       call beg(2,b1,b2,b3.a1,a2)
       call beg(2,b3,b4,b1,a3,a4)
       if(a1.eq.b01)go to 21
       k2=1
   24 do 22 k1=1,2
      bf=tv(k1,1,k1)
       tv(k2,1,k1)=tv(k2,2,k1)
       tv(k2,2,k1)=bf
       bf=tvt(k2,1,k1)
       tvt(k2,1,k1)=tvt(k2,2,k1)
       tvt(k2,2,k1)=bf
   22 continue
      go to(21,23),k2
   21 if(a3.eq.b03)go to 23
      k2=2
       go to 24
   23 a(1,1)=a4
      a(1,2)=a1
       a(2,1)=a2
       a(2,2)=a3
       call thick(r,xc,yc,ink)
       if(ink.eq.1)nk=2
       if(ink.eq.2)nk=1
       nk=3
       do 25 k1=1,2
       bf=tv(1,1,k1)
       tv(1,1,k1)=tv(2,2,k1)
       tv(2,2,k1)=tv(2,1,k1)
       tv(2,1,k1)=bf
       bf=tvt(1,1,k1)
       tvt(1,1,k1)=tvt(2,2,k1)
       tvt(2,2,k1)=tvt(2,1,k1)
       tvt(2,1,k1)=bf
   25 continue
      call dvi(ets(1,1,1)ets(1,1,2),tp,r(2,1),r(2,2),x2,y2,
     *tvt(1,1,1),tvt(1,1,2),tv(1,1,1),tv(1,1,2))
      t(1,1)=tp+0.05
      call dvi(ets(2,1,1),ets(2,1,2),tp,r(1,1),r(1,2),x2,y2,
     *tvt(2,1,1),tvt(2,1,2),tv(2,1,1),tv(2,1,2))
      t(1,2)=tp+0.05
      do 26 k1=1,2
       call dvi(ets(k1,2,1),ets(k1,2,2),tp,r(k1,1),r(k1,2),x2,y2,
     *tvt(k1,2,1),tvt(k1,2,2),tv(k1,2,1),tv(k1,2,2))
      t(2,k1)=tp+0.5
   26 continue
      go to 15
!     расчет сектора 1-1' (kl=1 т.принадлежит интервалу)
   40 ind=1
      b1=ancl(vn,tv(1,1,1),tv(1,1,2))
       b2=ancl(vn,tv(1,2,1),tv(1,2,2)
       b=(b1+b2)/2
```

```
      aa=b*3.141592/180.
      xp=vn*cos(aa)
      yp=vn*sin(aa)
      call poin(xp,yp,xc,yc,x2,y2,ddop,kl)
      bo=b1
      call beg(kl,b1,b2,b,a(1,1),a(1,2))
      if(bo.eq.a(1,1))go to 41
      do 42 k1=1,2
      bf=tv(1,1,k1)
      tv(1,1,k1)=tv(1,2,k1)
      tv(1,2,k1)=bf
      bf=tvt(1,1,k1)
      tvt(1,1,k1)=tvt(1,2,k1)
      tvt(1,2,k1)=bf
   42 continue
   41 do 46 k1=1,2
      tv(2,1,k1)=tv(1,2,k1)
      tvt(2,1,k1)=tvt(1,2,k1)
   46 continue
      do 43 k1=1,2
      call dvi(ets(k1,1,1),ets(k1,1,1),tp,r(1,1),r(1,2),x2,y2,
     *tvt(k1,1,1),tvt(k1,1,2),tv(k1,1,1),tv(k1,1,2))
      t(1,k1)=tp+0.5
   43 continue
      go to 15
!     расчет опасных и безопасных скоростей обл.7
   50 ind=4
      x2a=r(1,1)-x2
      y2a=r(1,2)-y2
      x2b=r(2,1)-x2
      y2b=r(2,2)-y2
      call com;p(x2a,y2a,x2b,y2b,xc,yc,ia)
      ind=3
      xy2=sq(x2,y2)
      dal(1)=xy2*vn/(vc+vn)
  200 return
      end

      subroutine beg(ind,b1,b2,b,an,ak)
!     п/п определение начала и конца сектора
!     b1,b1 - курсы концов секторов [град]
!           0<b1<360
!           0<b2<360
!     b - курс точки относительно сектора (b1,b2) [град]
!     an, ak - углы (курсы) начала и конца сектора соответственно
!     ind - признак режима
!           1 - точка в принадлежит сектору
!           2 - точка не принадлежит сектору
!     условия: b1 # b2, b2 # b, b1 # b
!     редакция: 8.05.13
!     ====================================
      if(b1.lt.b2)go to 10
      call chang(b1,b2)
   10 if((b1.lt.b).and.(b.lt.b2))go to 20
      an=b2
      ak=b1
      go to 30
   20 an=b1
      ak=b2
   30 if(ind.eq.2)call chang(an,ak)
      return
      end

      subroutine dvi(xa,ya,t,xk,yk,x2,y2,xr,yr,xn,yn)
!     п/п вычисления координат конца сектора
!     абсолютная система координат
!     xa,ya - координаты конца сектора /ист.движ./
!     t - время для выхода на конец сектора (час.)
!     xk,yk - координаты точки касания /отн.движ./
!     x2,y2 - координаты последней отметки он (цели)
!     xr, yr - координаты вектора отн.скор.
!     xn,yn - координаты вектора собств.скор. оу
!     редакция:12.12.13
!     ====================
      t=sqrt(((xk-x2)*(xk-x2)+(yk-y2)*(yk-y2))/(xr*xr+yr*yr))
```

```
      xa=xn*t
      ya=yn*t
      return
      end

      subroutine chang(a,b)
!     п/п замены a и b местами
!     редакция:19.01.13
!     ============================
      c=a
      a=b
      b=c
      return
      end

      subroutine d2(a,b,x,ind)
!     a - матрица коэффициентов
!     в - вектор правых частей
!     x - вектор решений
!     ind - признак наличия решений
!         1 - корни есть
!         0 - система равносильн.или несовместных уравнений
!     редакция: 16.04.13
!     ==========================================================
      dimension a(2,2),b(2),x(2)
      ind=1
      x(1)=0
      x(2)=0
      d=a(1,1)*a(2,2)-a(2,1)*a(1,2)
      if(d)1,2,1
    1 d1=b(1)*a(2,2)-b(2)*a(1,2)
      d2=a(1,1)*b(2)-a(2,1)*b(1)
      x(1)=d1/d
      x(2)=d2/d
      go to 3
    2 ind=0
    3 return
      end

      subroutine rotat(vn,xc,yc,x2,y2,xa,ya,rez,ut,ind)
!     п/п вычисления координат вектора vn при повороте
!     назначение параметров:
!       vn - модуль вектра собственной скорости/ единиц расстояния /сек
!       xc,yc - координаты вектора он (цели)
!       x2,y2 - координаты последней засечки он (цели)
!       xa,ya - координаты точки касания
!       rez(i,j) - координаты вектора un
!                i=1 - первая вершина, i=2 - вторая вершина
!                j=1 -x, j=2 -y
!    ind - индикатор признака типа результата
!          - 0 - погрешности измерений
!          - 1 - v отн. > 0, v отн. < 0
!          - 2 - v отн. > 0, v отн. > 0
!          - 3 - v отн. < 0, v отн. < 0
!    vt(i,j) - координаты вектора отн.скор.после поворота
!    редакция п/п: 15. 09. 13.
!    ==========================================================
     dimension rez(2,2), vt(2,2)
     ft(bx,by,cx,cy)=bx*cx+by*cy
     do 70 j1=1,2
     do 70 j2=1,2
     rez(j1,j2)=0.
  70 vt(j1,k2)=0.
     x2a=ya-y2
     x2a=xa-x2
     xz=x2+xc
     yz=y2+y2
     if(xa.eq.x2)go to 20
     yx=y2a/x2a
     a=1.+yx*yn
     a2=2.*a
     b=2.*(yx*(yz-y2-yx*xz)-x2)
     c=x2*y2+(yz-y2-yx*xz)*(yz-y2-yx*xz)-vn*un
     d=b*b-4.*a*c
     if(d.gt.0.)go to 30
```

```
 45 ind=4
    go to 100
 30 sqd=sqrt(d)
    rez(1,1)=-(b+sqd)/a2
    rez(1,2)=yz+yx*(rez(1,1)-xz)
    rez(2,1)=(sqd-b)/a2
    rez(2,2)=yz+yx*(rez(2,1)-xz)
  2 rez(1,1)=-x2+rez(1,1)
    rez(1,2)=-y2+rez(1,2)
    rez(2,1)=-x2+rez(2,1)
    rez(2,2)=-y2+rez(2,2)
    vt(1,1)=xc-rez(1,1)
    vt(1,2)=yx-rez(1,2)
    vt(2,1)=xc-rez(2,1)
    vt(2,2)=yc-rez(2,2)
    ft1=ft(vt(1,1),vt(1,2),x2a,y2a)
    ft2=ft(vt(2,1),vt(2,2),x2a,y2a)
    if((ft1.gt.0.).and.(ft2.gt.0.))go to 31
    if((ft1.gt.0.).and.(ft2.lt.0.))go to 34
    if((ft1.lt.0.).and.(ft2.gt.0.))go to 33
    if((ft1.lt.0.).and.(ft2.lt.0.))go to 35
    ind=0
    go to 100
 35 ind=3
    go to 100
 34 ind=1
    go to 100
 33 do 32 ll=1,2
    bf=rez(1,ll)
    rez(1,ll)=rez(2,ll)
    rez(2,ll)=bf
    bf=vt(1,ll)
    vt(1,ll)=vt(2,ll)
    vt(2,ll)=bf
 32 continue
    go to 34
 31 ind=2
    go to 100
 20 d=vn*vn-(x2-xz)*(x2-xz)
    if(d.gt.0.)go to 50
    go to 45
 50 sqd=sqrt(d)
    rez(1,1)=xz
    rez(1,2)=y2-sqd
    rez(2,1)=xz
    rez(2,2)=y2+sqd
    go to 2
100 return
    end


    subroutine comp(xa,ya,xb,yb,xc,yc,ia)
!   п/п вычисления признака взаимного расположения векторов
!   xc,yc - координаты вектора он (цели)
!   xa,ya,xb,yb - координаты вектора внешних
!   ограничений области
!   ia - признак расположения вектора в области
!     - 0 - внутри области или на границе области
!     - 1 - вне области
!     - 2 - ошибка в данных
!   редакция п/п:28.12.13
!   ================================
    f(x1,x2,y1,y2)=x1*y2-x2*y1
    ia=0
    d=f(xa,xb,ya,yb)
    if(d.ne.0.)go to 1
    ia=2
    go to 2
  1 al=f(xc,xb,yc,yb)/d
    bl=f(xa,xc,ya,yc)/d
    if((al.lt.0.).or.(bl.lt.0.)) ia=1
  2 return
    end
```

```
      subroutine sps(vn,nw,iw,vtek1,dv1,vtek2,dv2,vd1,vd2,
     *vd3,vd4,num)
!     п/п определения рекомендуемой скорости движения
!     vn - текущее значение собственной скорости движения
!     nw - к-во диапазонов безопасных скоростей
!     iw(40,2) - границы диапазонов безопасных скоростей
!     vtek1,vtek2 - рекомендуемые скорости движения
!     dv1,dv2 - модули величин изменения скорости
!     num- к-во рекомендуемых скоростей [ <1>, <2> ]
!     ========================================================
      dimension iw(40,2)
       if(nw.eq.0)go to 50
       num=1
       dv1=0.
       vtek=vn
       dv2=0.
       vtek2=0.
       vmin=iw(1,1)
       if(vn.lt.vmin)go to 30
       do 1 i=1,nw
       vmin=iw(i,1)
       vmax=iw(i,2)
       ind=1
       if((vmin.lt.vn).and.(vmax.gt.vn))go to 10
       i1=i+1
       ind=2
       if(i1.gt.nw)go to 1
       vmin1=iw(i1,1)
       if((vmax.le.vn).and.(vmin.ge.vn))go to 20
    1 continue
      vtek1=iw(nw,2)
       i=nw
   31 dv1=abs(vn-vtek1)
      go to 10
   30 dv1=abs(vmax-vn)
      dv2=abs(vmin1-vn)
       num=2
       vtek1=vmax
       vtek2=vmin1
   10 go to(35,36),ind
   35 vd1=iw(i,1)
      vd2=iw(i,2)
       go to 42
   36 vd3=iw(i1,1)
      vd4=iw(i1,2)
       go to 35
   50 num=0
   40 return
      end


      subroutine line(x2,y2,d,r,ind)
!     п/п вычисления точек касания с окружностью
!     x2,y2-координаты последней засечки он (цели)
!     ind=0 точка 2 вне окружности
!     ind=1 точка 2 внутри или на окружности
!     d - дистанция сближения
!     r(i,j) - массив результатов
!              i - номер точки касания
!              j=1 - x, j=2 -y
!     редакция:25.11.13
!     --------------------------
      dimension r(2,2)
       ttt(a,b,c)=a*a*c*c-(a*a+b*b)*(c*c-b*b)
       do 10 i1=1,2
       do 10 i2=1,2
   10 r(i1,i2)=0
      ind=1
       f=y2*y2+x2*x2
       if(abs(y2).gt.0.0001)go to 1
       if(abs(x2).gt.0.0001)go to 2
       go to 20
    1 diskr=ttt(x2,y2,d)
      if(diskr.le.0)go to 20
       ind=0
```

```
      xd=x2*d*d
      diskr=sqrt(diskr)*d
      r(1,1)=(xd+diskr)/f
      r(2,1)=(xd-diskr)/f
      r(1,2)=(d*d-x2*r(1,1))/y2
      r(2,2)=(d*d-x2*r(2,1))/y2
      go to 20
    2 diskr=ttt(y2,x2,d)
      if(diskr.le.0)go to 20
      ind=0
      yd=y2*d*d
      diskr=sqrt(diskr)*d
      r(1,2)=(yd+diskr)/f
      r(2,2)=(yd-diskr)/f
      r(1,1)=(d*d-y2*r(1,2))/x2
      r(2,1)=(d*d-y2*r(2,2))/x2
   20 return
      end

      subroutine poin(xp,yp,xc,yc,x2,y2,d,ind)
!     п/п определения пересечения vr с окружностью радиуса d
!     для линейки скоростей
!     xp,yp - координаты скорости оу
!     xc,yc - координаты вектора скорости цели
!     x2,y2 - координаты последней засечки он (цели)
!     d - допустимая дистанция сближения
!     ind - признак результата
!           2 - vr не пересекает окружность d
!           1 - vr пересекает или касается окружности d
!     редакция:29.05.13
!     ========================================
      ind=2
      x1=0.
      y1=0.
      xr=xc-xp
      yr=yc-yp
      xyr=xr*xr+yr*yr
      if(xyr.eq.0.)go to 50
      if(xr.eq.0.)go to 39
      yx=yr/xr
      a=1.+yx*yx
      b=yx*(y2-yx*x2)*(y2-yx*x2)-d*d
      dis=b*b-a*c
      if(dis.lt.0.)go to 50
      diskr=sqrt(dis)
      x1=(-b+diskr)/a
      y1=y2+yx*(x1-x2)
   40 x12=x1-x2
      y12=y1-y2
      ft=x12*xr+v12*yr
      if(ft.le.0.)go to 50
      ind=1
      go to 50
   30 dis=d*d-x2*y2
      if(dis.lt.0.)go to 50
      x1=x2
      y1=sqrt(dis)
      go to 40
   50 return
      end

      subroutine agl(x,y,a,ierr)
!     п/п высчисления угла (пеленга) на он (цель) по координатам отметки
!     x, y - координаты отметки он (цели) - единиц расстояния
!     a - искомый угол (рад)
!     ierr - признак ошибки (0 - нет, 1 - есть)
!     редакция:16.06.13
!     ========================================
      s=sqrt(x*x+y*y)
      if(s.le.0.) go to 10
      ierr=0
      call qwart(x,y,ind)
      x=x/s
      c=sqrt(1-c*c)
      c=arcsin (c)
```

```
      go to(1,2,3,4,5,6,7,8), ind
      go to 10
    1 a=c
      go to 20
    2 a=3,141592-c
      go to 20
    3 a=3.141592+c
      go to 20
    4 a=6.283184-c
      go to 20
    5 a=0.
      go to 20
    6 a=3,141592
      go to 20
    7 a=1.1570796
      go to 20
    8 a=4.712398
      go to 20
   10 ierr=1
   20 return
      end

      function ancl(v,x,y)
!     п/п функция вычисления напрвления (курса)
!     v- скорость
!     x,y, - координатi вектора скорости
!     редакция:20.11.13
!     ==================
      if(v.eq.9)go to 1
       s=y/v
       c=x/v
       a=abs(s)
       ancl=180.*arcsin(a)/3.141592
       if((s.gt.0.).and.c.gt.0.))go to 1
      if((s.gt.0.).and.c.lt.0.))go to 2
       if((s.lt.0.).and.(c.lt.0.))go to 3
       if((s.lt.0.).and.(c.gt.0.))go to 4
       if((s.eq.0.).and.(c.gt.0.))ancl=0.
       if((s.gt.0.).and.(c.eq.0.))ancl=90.
       if((s.eq.0.).and.(c.lt.0.))ancl=180.
       if((s.lt.0.).and.(c.eq.0.))ancl=270.
       go to 1
    2  ancl=180.-ancl
       go to 1
    3  ancl=180.+ancl
       go to 1
    4  ancl=360.-ancl
    1  return
       end

       function arcsin(x)
       real*8 a(8),8
       data a/1.570796305d0,-0.2145988016d0,0.08897898774d0,-.05017430
      :46d0,0.030891881do,-0.0170881256d0,0.006670091d0,-0.0012624911d0/
       e=1
       arcsin=0.
       if(x.eq.0.)goto100
       if(abs(x).gt.1.)go to 10
       if(x.ge.0)go to 1
       e=-1.
       x=-x
    1  s=a(1)
       p=x
       do 2 i=2,8
       s=s+a(i)*p
    2  p=p*x
       arcsin=(1.570796-sqrt(1.-x)*s)*e
  100  return
   10  type 3
    3  format(' arcsin_ _ _ _ _ _ _ _! arg!>1 !')
       stop
       end

       subroutine thick(r,xc,yc,ind)
!     п/п выработки признака расхождения (начало/конец он «голова-хвост»)
```

```
!     r(1,1), r(1,2) - координаты 1-ой точки касания
!     r(2,1), r(2,2) - координаты 2-ой точки касания
!     xc, yc - координаты вектора он (цели)
!     ind - признак результата
!         1 - 1-ая точка начало (голова)
!         2 - 1-ая точка конец (хвост)
!         3 - 1-ая точка нк
!     редакция:21.11.13
!     ================
      dimension r(2,2)
       ind=1
       rx=r(2,1)-r(1,1)
       ry=r(2,2)-r(1,2)
       rxy=abs(sqrt(rx*rx+ry*ry))
       vc=abs(sqrt(xc*xc+yc*yc))
       s=(rx*xc+ry*yc)/rxy
       vc0=-0.02*vc
       vc01=-vc0
       if(s.lt.vc0)go to 3
       if(s.gt.vc01)go to 1
       go to 2
    3 ind=2
      go to 1
    2 ind=3
      go to 1
    1 return
      end

       subroutine vv(vvar,xc,yc,xn,yn,x2,y2,xa,ya,rn,rk,ind)
!     п/п вычисления требуемой скорости
!     vn - модуль скорости оу
!     xc, yc - координаты вектора он (цели)
!     xn, yn - координаты вектора скорости оу
!     x2, y2 - координаты последней засечки он (цели)
!     xa, ya - координаты точки пространства
!     vvar - значение рекомендуемой скорости
!     rn - коэффициент направленности vr
!     rk - коэффициент vn
!     ind - признак количества возможных решений
!         - 0 - решение единственно
!         - 1 - vn/2a, vc#2a
!         - 2 - vn#2a, vc/2a
!         - 3 - vn/2a, vc//2a
!         - 4 - нет решений /vn=0/
!         - 5 - стоящая цель /vc=0/
!    редакция:09.12.13
!    ====================
     sq(a,b)=sqrt(a*a+b*b)
     ind=4
     vvar=0.
     rk=0.
     rn=0.
     if(vn.eq.0.)go to 2
     x2a=xa-x2
     y2a=ya-y2
     v2a=sq(x2a,y2a)
     s2=(x2a*xn+y2a*yn)/(vn*v2a)
     if(vc.eq.0.)go to 40
     s1=(x2a*xc+y2a*yc)/(vc*v2a)
     as2=abs(s2)
     as1=abs(s1)
     if((as2.ne.1).and.(as1.eq.1.))go to 22
     if((as2.eq.1.).and.(as1.ne.1.))go to 21
     if((as1.eq.1.).and.(as2.eq.1.))go to 10
     go to 60
 40  ind=5
     go to 2
 21  ind=1
     go to 2
 22  ind=2
     rk=0.
     rn=s1
     go to 2
 10  ind=3
     go to 2
```

```
 60 ind=0
    if(xa.eq.x2)go to 1
    s=yn*x2a-xn*y2a
    rk=(yc*x2a-xc*y2a)/s
    vvar=rk*vn
    rn=(xc-rk*xn)/x2a
    go to 2
  1 rk=xc/xn
    vvar=vn*rk
    rn=(yc-rk*yn)/y2a
  2 return
    end

    subroutine raspr(ogl,ogr,nl,nr,x0,y0,ind,ierr)
!   п/п формирования признака принадлежности множеству
!   ogl(10,2)-координаты левых ограничений
!   ogr(10,2)-координаты правых ограничений
!   nl - к-во точек левых ограничений
!   nr - к-во точек правых ограничений
!   x0,y0 - координаты анализируемой точки
!   ind - признак результата
!             0 - точка принадлежит множеству
!             1 - точка вне множества
!   ierr - признак ошибки
!             0 - ошибки нет
!             1 - ошибки есть
!   редакция:27.11.13
!   ================================================================
    dimension x(20), y(20), ogl(10,2),ogr(10,2)
    do 10 i=1,nl
    x(i)=ogl(i,1)-x0
    y(i)=ogl(i,2)-y0
 10 continue
    do 11 i=1,nr
    i1=nl+i
    i2=nr+1-i
    x(i1)=ogr(i2,1)-x0
    y(i1)=ogr(i2,2)-y0
 11 continue
    ierr=0
    n=nl+nr
    n1=n-1
    call agl(x(1),y(1),a1,ie1)
    call agl(x(n),y(n),a2,ie2)
    if((ie1.eq.1).or.(ie2.eq.1))ierr=1
    phi=a1-a2
    if(abs(phi).gt.3.141593)phi=phi-6.283185*abs(phi)/phi
    do 4 i=1,n1
    call agl(x(i+1),y(i=1+,a1,ie1)
    call agl(x(i),y(i),a2,ie2)
    if((ie1.eq.1).or.(ie2.eq.1))ierr=1
    dphi=a1-a2
    if(abs(dphi).gt.3.141593)dphi=dphi-6.283185*abs(dphi)/dphi
  4 phi=phi+dphi
    if(abs(phi).lt.0.001)ind=1
    if(abs(phi).gt.6.28)ind=0
    return
    end

    subroutine chain(a,x,n)
!   a - имя сортируемого массива
!   x - новое значение переменной
!   n - размерность массива a
!   п/п перестановки
!   редакция:10.04.13.
!   ==========================
    dimension a(1)
    b=x
    do 1 i=n,1, -1
    c=a(i)
    a(i)=b
    b=c
  1 continue
    return
    end
```

```
      subroutine koef(xa,ya,xb,yb,a,b,c)
!     п/п вычисления коэффициентов прямой a*x+b*y=c
!     xa,ya, - координаты начала
!     xb,yb - координаты конца
!     редакция: 20.04.13
!     ================================================
      a=yb-ya
      b=xa-xb
      c=ya*xb-yb*xa
      return
      end


      subroutine qwart(x,y,ind)
!     п/п определения четверти координатной плоскости
!     x,y - координаты вектора [точки] на плоскости
!     ind - номер четверти плоскости [n-x,w-y]
!       5 - вверх по x
!       6 - вниз по x
!       7 - вправо
!       8 - влево
!     [отсчет углов по часовой стрелке от оси x]
!     редакция: 31.03.13.
!     ============================================
      if((x.gt.0.).and.(y.gt.0.))ind=1
      if((x.lt.0.).and.(y.gt.0.))ind=2
      if((x.gt.0.).and.(y.lt.0.))ind=3
      if((x.gt.0.).and.(y.lt.0.))ind=4
      if((x.gt.0.).and.(y.eq.0.))ind=5
      if((x.lt.0.).and.(y.eq.0.))ind=6
      if((x.eq.0.).and.(y.gt.0.))ind=7
      if((x.eq.0.)and.(y.lt.0.))ind=8
      return
      end

      subroutine shift(xp,yp,xa,ya,x0,y0,a)
!     п/п вычисления координат точки при повороте осей и смеш.
!     xp,yp - координаты точки после преобразования
!     xa,ya - координаты точки до преобразования
!     x0,y0 - координаты нового начала координат
!     a - угол поворота осей /рад/
!     редакция: 07.02.13
!     - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      ca=cos(s)
      sa=sin(a)
      xa0=xa-x0
      ya0=ya-y0
      xp=xa0*ca+ya0*sa
      yp=ya0*ca-xa0*sa
      return
      end

      subroutine bereg(br,bl,nr,x1,y1,x2,y2,ind,kr,kl)
!     п/п выделения признака пересечения ветви графа
!     с линиями ограничений пространства решения
!     br(10,2) - координаты вершин ломаной правыой границы
!     nr - к-во вершин левой границы
!     x1,y1 - координаты начала елемента графа расхождения
!     x2,y2 - координаты конца елемента графа расхождения
!     ind - признак пересечения линий ограничений
!         0 - пересечения с линиями ограничений нет
!         1 - пересечение с линией правой линии ограничений
!         2 - пересечение с линией левойлинией ограничений
!         3 - пересечения с линиями обеих ограничений
!     kr(9) - признак пересечения со звном i[kr(i)=1]
!         0 - пересений нет[пр.берег]
!     kl(9) - аналогично kr по левой линии ограничений
!     редакция: 15.05.13
!     ====================
      dimension br(10,2), bl(10,2), kr(9),kl(9), a(2,2),
     *b(2), x(2)
      s(tx,ty,qx,qy,rx,ry)=(tx-rx)*(qx-rx)+(ty-ry)*(qy-ry)
      do 1 i=1,9
```

```
      kr(i)=0
      kl(i)=0
    1 continue
      k1=0
      k2=0
      if((nr.le.1).and.(nl.le.1))go to 56
      call koef(x1,y1,x2,y2,ag,bg,cg)
      cg=-cg
      if(nr.le.1)go to 55
      nr1=nr-1
      do 2 i=1,nr1
      i1=i+1
      call koef(br(i,1),br(i1,1),br(i1,2),ar,bbr,cr)
      cr=-cr
      call bufwr(ag,bg,ar,bbr,cg,cr,a,b)
      call d2(a,b,x,inr)
      if(inr.eq.0)go to 2
      s1=s(br(i,1),br(i,2),br(i1,1),br(i1,2),x(1),x(2))
      s2=s(x1,y1,x2,y2,x(1),x(2))
      if((s1.le.0.).and.(s2.le.0.))kr(i)=1
      k1=k1+kr(i)
    2 continue
   55 if(nl.le.1)go to 56
      nl1=nl-1
      do 2 i=1,nl1
      i1=i+1
      call koef(bl(i,1),bl(i,2),bl(i1,1),bl(i1,2),al,bbl,cl)
      cl=-cl
      call bufwr(ag,bg,al,bbl,cg,cl,a,b)
      call d2(a,b,x,inl)
      if(inl.eq.0)go to 3
      s1=s(bl(i,1),bl(i,2),bl(i1,1),bl(i2,2),x(1),x(2))
      if((s1.le.0.).and.(s2.le.0.))kl(i)=1
      k2=k2+kl(i)
    3 continue
   56 ind=3
      if((k1.eq.0).and.(k2.eq.0))ind=0
      if((k1.ne.0).and.(k2.eq.0))ind=1
      if((k1.eq.0).and.(k2.ne.0))ind=2
    3 continue
   56 ind=3
      if((k1.eq.0).and.(k2.eq.0))ind=0
      if((k1.n2.0).and.(k2.eq.0))ind=1
      if((k1.eq.0).and.(k2.ne.0))ind=2
      return
      end

      subroutine bufwr(a11,a12,a21,a22,b1,b2,a,b)
!     п/п заполнения буфера коэффициентов для окружности d2
!     aij - коэффициенты матрицы левой части (a)
!     b1 - коэфиициенты вектора левых частей (b)
!     редакция:03.04.13.
!     ===========================================
      dimension a(2,2),b(2)
      a(1,1)=a11
      a(1,2)=a12
      a(2,1)=a21
      a(2,2)=a22
      b(1)=b1
      b(2)=b2
      return
      end

      subroutine cross(vmax,xn,yn,c,a2,d,lv,iw,nw,iflag,ierr)
!     п/п вычисления интервалов безопасных скоростей
!     c(i,j) - массив координат векторов ОН (целей)
!              i - номер ОН (цели)
!               ,j=1 - x
!               ,k=2 - y
!     a2(i,j)- массив координат последних засечек ОН (целей)
!            i - номер ОН (цели)
!                ,j=1 - x
!                ,j=2 - y
!     d(i) - массив d kp ОН (целей)
!     iflag(10) - признаки сопровождения ОН (целей) [1 - есть, 2 - нет]
```

```
!   lv(40) - массив опасных скоростей
!   iw(40,2) - массив диапазонов безопасных скоростей
!   nw - количество диапазонов безопасных скоростей
!   ierr - признак наличия ошибки
!          0- ошибки нет
!          1 -ошибка есть
!   редакция:10.02.13
!   ================================================================
    dimension c(10,2),a2(10,2),d(10),lv(40),iw(40,2),iflag(10)
    ierr=0
    k=0
    do 1 i=1,10
  1 k=k+iflag(i)
    i1=0
    if(k.ne.0)go to 10
    ind=1
    kin=1
    l=1
    call aria(vmax,vn1,vn2,lv,ind,l,kin)
    go to 100
 10 do 11 i=1,10
    if(iflag(i).eq.0)go to 11
    i1=i1+1
    call stick(vmax,xn,yn,vn1,vn2,c(i,1),c(i,2),a2(i,1),a2(i,2),
   * d(i),ind)
    call aria(vmax,vn1,vn2,lv,ind,i1,kin)
    if(ind.ne.6)go to 11
    ierr=1
    go to 100
 11 continue
100 call iar(lv,iw,nw)
    return
    end

    subroutine aria(vmax,vn1,vn2,lv,ind,n,k)
!   п/п отыскание пересечений для линейки скоростей
!   vmax - max скорость ОУ
!   vn1,vn2 - точки разбиения интервала из stick
!   lv(40) - массив дробления интервала 0<vn<max
!   n - номер (количество ОН - целей) для линейки скоростей
!   ind - признак исходной информации, получаемой из stick
!     1 - 0<vn<vmax
!     2 - vn2<vn<vmax
!     3 - все скорости опасны
!     4 - 0<vn<vn1, vn2<vn<vmax
!     5 - 0<vn<vn1
!     6 - ошибка
!   k - признак существования результата
!     0 - все скорости опасны или был ind=6
!     1 - есть безопасные интервалы скоростей
!   редакция: 08.02.13.
!   ================================================================
    dimension lv(40), vb(40)
    data vb/1.,2.,3.,4.,5.,6.,7.,8.,9.,10.,11.,12.,13.,14.,15.,
   *16.,17.,18.,19.,20.,21.,22.,23.,24.,25.,26.,27.,28.,29.,
   *30.,31.,32.,33.,34.,35.,36.,37.,38.,39.,40./
    if(n.ne.1)go to 10
    iv=vmax
    if(iv.ne.0)go to 1
    do 63 i=1,40
 63 lv(i)=0
  3 k=0
    go to 100
  1 do 2 i=1,40
    lv(i)=0
    if9i.le.iv)lv(i)=1
  2 continue
    k=1
    go to 9
 10 if(k.eq.0)go to 100
  9 go to(100,20,50,30,40),ind
    go to 3
 20 do 21 i=1,iv
    if(vn2.gt.vb(i))lv(i)=0
 21 continue
```

```
      go to 100
   50 do 51 i=1,40
   51 lv(i)=0
      go to 3
   30 do 31 i=1,iv
      if((vn1.lt.vb(i)).and.(vn2.gt.vb(i)))lv(i)=0
   31 continue
      go to 100
   40 do 41 i=1,iv
      if(vn1.lt.vb(i))lv(i)=0
   41 continue
  100 return
      end

      subroutine iar(lv,iw,nw)
!     п/п вычисления границ и к-ва диапазонов безопасных скоростей
!     lv(40) - массив констант [0 - опас.скор.,1 - безопас.скор.]
!     iw(40,2) - массив интервалов безопасных скоростей
!     iw(i,j) - i- номер диапазона, j - 1 - начало, 2 - конец
!     nw - к-во интервалов безопасных скоростей
!     редакция: 02.4.13
!     ==================
      dimension lv(40), iw(40,2)
      nw=0
      ind=0
      do 1 i=1,40
      if((lv(i).eq.1).and.(ind.eq.0))go to 2
      if((lv(i).eq.0).and.(ind.eq.1))go to 3
      go to 1
    2 nw=nw+1
      iw(nw,1)=i
      ind=1
      go to 1
    3 iw(nw,2)=i-1
      ind=0
    1 continue
      if((nw.ne.0).and.(ind.eq.1))iw(nw,2)=40
      return
      end


      subroutine stick(vmax,xn,yn,vn1,vn2,xc,yc,x2,y2,d,ind)
!     п/п вычисления линейки скоростей для одного ОН (цели)
!     vmax - max значение модуля скорости ОУ
!     xn, yn - координаты вектора скор.ОУ
!     xc,yc - координтаы вектора скорости ОН (цели)
!     x2, y2 - координаты последней отметки ОН (цели)
!     vn1 - min допустимая скорость
!     vn2 - max допустимая скорость
!     ind - признак типа результата
!             1 - 0<vn<vmax
!             2 - vn2<vn<vmax
!             3 - все скорости опасны
!             4 - 0<vn<vn1 и vn2<vn<vmax
!             5 - 0<vn<vn1
!             6 - ошибка в данных
!     d - безопасная дистанция /dkp/
!     редакция: 24.01.13.
!     ===================================================================
      dimension point(2,2),rn(2),rk(2),kin(2),vvar(2)
!       вычисление координат точек касания
      call line(x2,y2,d,point,ipr)
      xa=point(1,1)-x2
      ya=point(1,2)-y2
      xb=point(2,1)-x2
      yb=point(2,2)-y2
      if(ipr.ne.1)go to 1
    6 ind=6
      go to 100
!        определение параметров выхода на окружность d
    1 do 2 i1=1,2
      call vv(vvar(i1),xc,yc,xn,yn,x2,y2,point(i1,1),point(i1,2)?
     *rn(i1), rk(i1), kin(i1))
      vn1=vvar(1)
      vn2=vvar(2)
```

```
 2 continue
   if((kin(1).eq.5).and.(kin(2).eq.5))go to 5
   if(kin91).eq.5).and.(kin92).ne.5)).or.((kin(1).ne.5).and.
  *(kin(2).eq.5)).or.(kin(1).eq.4).or.(kin(2).eq.4).or.((kin(1).
  *eq.3).and.(kin(2).eq.3)).or.((kin(1).eq.1).and.(kin92).eq.3)).or.
  *((kin(1).eq.3).and.(kin(2).eq.1)).or.((kin(1).eq.2).and.(kin(2).
   if(((kin(1).eq.3).and.(kin(2).eq.0)).or.((kin10.eq.0).and.
  *(kin(2).eq.3))go to 10
   if(((kin(1).eq.1).and.(kin(2).eq.1)).or.((kin(1).eq.2).and.
  *(kin(2).eq.1)))go to 15
   if((kin(1).eq.0).and.(kin(2).eq.0))go to 20
   if((kin(1).eq.0).and.(kin(2).eq.1))go to 25
   if((kin(1).eq.1).and.(kin(2).eq.0))go to 26
   if((kin(1).eq.0).and.(kin(2).eq.2))go to 20
   if((kin(1).eq.2).and.(kin(2).eq.0))go to 20
 5 xnn=-xn
   ynn=-yn
 9 call comp(xa,ya.xb,yb,xnn,ynn,ia)
   if(ia.eq.2)go to 6
   if(ia.eq.1)go to 10
28 ind=3
   go to 100
10 ind=1
   go to 100
15 xnn=xc-xn
   ynn=yc-yn
   go to 9
25 if(rn(1).le.0)go to 10
   call comp(xa,ya,xb,yb,xc,yc,ia)
   if(ia.eq.2)go to 6
   if(ia.eq.0)go to 27
67 if(rk(1).lt.0.)go to 10
33 if(vvar(1).ge.vmax)go to 10
   vn1=vvar(1)
41 ind=5
   go to 100
27 if(rk(1).lt.0.)go to 28
32 if(vvar(1).gt.vmax)go to 28
   vn2=vvar(1)
46 ind=2
   go to 100
26 call chang(rn91),rn(2))
   call chang(rk(1),rk(2))
   call chang(vvar91),vvar(2))
   go to 25
20 if((rn(1).gt.0.).and.(rk(1).gt.0.).and.(rn(2).gt.0.).
  *and.(rk(2).gt.0.))go to 40
   call comp(xa,ya,xb,yb,xc,yc,ia)
   if(ia.eq.2)go to 6
   if((rn(1).gt.0.).and.(rn(2).gt.0.).and.(((rk(1).gt.0.).and.
  *(rk(2).le.0.)).or.((rn(1).lt.0.).and.(rk(2).gt.0.))))go to 45
   if(((rn(1).gt.0.).and.(rn(2).gt.0.).and.(rk(1).le.0.).and.
  *(rk(2).le.0.)).or.((rn(1).lt.0.).and.(rn(2).lt.0.).or.
  *((ia.eq.1).and.(((rn(1).gt.0.).and.(rk(1).lt.0.)).or.
  *le.0.).and.(rn(2).lt.0.)).or.((rn(1).lt.0.).and.(rk(1).le.0.).
  *and.(rn(2).gt.0.).and.(rk(2).lt.0.)).or.((rn(1).gt.0.).
  *and.(rk(1).lt.0.).and.(rn(2).lt.0.).and.(rk(2).gt.0.)).or.
  *((rn(1).lt.0.).and.(rk(1).gt.0.).and.(rn(2).gt.0.).and.
  *(rk(2).lt.0.)))))go to 10
   if((ia.eq.0).and.(((rn(1).gt.0.).and.(rk91).ge.0.).and.
  *(rn(2).lt.0.).and.(rk(2).gt.0.)).or..((rn(1).lt.0.).and.
  *(rk(1).ge.0.).and.(rn(2).gt.0.).and.(rk(2).gt.0.))))
  *go to 55
   if((ia.eq.0.).and.((n(1).gt.0.).and.(rk(1).le.0.).and.
  *(rn(2).lt.0.).and.(rk(2).lt.0.)).or.((rn(1).lt.0.).and.(rk(1).
  *lt.0.).and.(rn(2).gt.0.).and.(rk(2).le.0.))))go to 28
   if((rn(1).gt.0.).and.(rk(1).gt.0.).and.(rn(2).lt.0.).and.
  *(rk(2).lt.0.)).or.((rn(1).lt.0.).and.(rk(1).lt.0.).and.(rn(2).
  *gt.0.).and.(rk(2).gt.0.)))go to 55
   if((rn(1).eq.0.).and.(rn(2).eq.0.))go to 65
   go to 6
40 if(vn1.gt.vn2)call chang(vn1,vn2)
   if(vn1.gt.vmax)go to 10
   if(vn2.ge.vmax)go to 41
   ind=4
```

```
      go to 100
   45 if(vn1.gt.vn2)call chang(vn1,vn2)
      if(vn2.gt.vmax)go to 28
      go to 46
   50 if(vn1.gt.vn2)call chang(vvar(1),vvar(2))
      go to 32
   55 if(vn1.lt.vn2)call chang9var(1),vvar(2))
      go to 33
   65 xxa=-xa
      yya=-ya
      xxb=-xb
      yyb=-yb
      call comp(xa,ya,xb,yb,xc,yc,ia1)
      call comp(xxa,yya,xxb,yyb,xc,yc,ia2)
      if((ia1.eq.2).or.(ia2.eq.2.or.((ia1.eq.0).
     *and.(ia2.eq.0)))go to 6
      if((ia1.eq.1).and.(ia2.eq.1))go to 10
      if((ia1.eq.1).and.(ia2.eq.0))go to 67
      go to 27
  100 return
      end

      function react(psy,an,ak,dpsy,l,n,ierr)
!     п/п наращивания величины угла для вычисления оптимального
!                           маневра
!     react - искомое значение направления (курса) [рад.]
!     psy - текущее опорное направление (курс) [рад.]
!     an,ak - угол начала и окончания сектора [рад.]
!     dpsy - модуль приращения угла [рад.]
!     l - признак направления приращения угла:
!                 1 - от начала сектора
!                 2 - от конца сектора
!     n - номер вхождения в программу
!     ierr - признак ошибки [0 - нет, 1 - есть, 3 - конец приращений]
!     редакция: 25.01.13
!     ================================================================
      ierr=0
      go to(1,2),l
      ierr=1
      go to 100
!     дополнение угла от начала сектора во вннешнюю сторону
    1 an1=psy-1.570796
      if(an1.lt.0.)an1=6.283184+an1
      k=(an-an1)/dpsy
      if(n.gt.k)go to 20
      react=an-n*dpsy
      if(react.lt.0.)react=6.283184+react
      go to 100
   20 ierr=2
      go to 100
!     дополнение гла от конца сектора во внешнюю сторону
    2 an2=psy+1.570796
      if((ak.ge.0.).and.(ak.le.1.570796))go to 3
    4 k=(an2-ak)/dpsy
      ifn.gt.k)go to 20
      react=ak+n*dpsy
      if(react.ge.6.283184)react=react-6.283184
      go to 100
    3 if((psy.ge.0.).and.(psy.le.1.570796))go to 4
      an2=an2-6.283184
      go to 4
  100 return
      end

        subroutine zapl1(rv,rp,rt,dk1,dk2,dk3,dk4,vn,spy,vtek,
     *an,rtek,an3,rj3,usm,gr31, gr32,dv1,dv2,dp1,dp2,gr,nnew,
     *nold,istop,ld)
!       п/п сравнения и формирования массива альтернатив
!         dp1, dp2 - изменения направления (курса) на этапах 2 и 3
!       rv(3) - массив скоростей
!       dv1,dv2 - изменения скорости на этапах 2 и 3
!       dk1,dk2, dk3,dk4 - коэффициенты критерия отбора
!       istop - признак останова п/п [0 - продолжение, 1- останов]
!       rp(3) - массив направлений перемещения (курсов)
!       usm - значение критерия
```

```
!       nnew - номер итерации вхождения
!       nold -номер итерации последнего запоминания
!       редакция: 29.10.13.
!       ==================================================================
        dimension rv(3),rp(3),gr(4,2),rt(3)
         v34=2.4674
         istop=0
         rj1=dk1*dv1*dv1/(vn*vn)+(1.-dk1)*dp1*dp1/v34
         rj2=dk2*dv2*dv2/(vtek*vtek)+(1.-dk2)*dp2*dp2/v34
         rjs=dk3*rj1+(1.-dk3)*rj2+dk4*rj3
         if(ld.ne.0)go t 4
         ld=1
         type 10
    10 format(25x,'промежуточные решения')
       go to 3
      4 if(rks.ge.usm)go to 1
      3 nold=nnew
         gr(3,1)=gr31
         gr(3,20=gr32
         rv(1)=vn
         rv(2)=vtek
         rv(3)=rtek
         rp(1)=psy
         rp(2)=an
         rp(3)=an3
         usm=rjs
         rt(1)=sqrt(gr(2,1)*gr(2,1)+gr(2,2)*gr(2,2))/rv(1)
         rt(3)=sqrt((gr(4,1)-gr31)*(gr(4,1)-gr31)+(gr(4,2)-gr32)*
       *(gr(4,2)-gr32))/rv(3)
         rt(2)=sqrt((gr31-gr(2,1))*gr31-gr(2,1)+(gr32-gr(2,2))*
       *(gr32-gr(2,2)))/rv(2)
         type 15
    15 format(10x,'****')
       do 20 j=1,3
         irp=180.*rp(j)/3.141592
         irv=rv(j)
         irt=60.*rt(j)
         type 11,j,irp,irv,irt
    11 format(1x,i1,' элемент траектории: направление ',i3,' , скорость ',i3,
      :' ,время ', i3)
    20 continue
       go to 2
      1 is=nnew-nold
         if(is.ge.200)istop=1
      2 return
         end

         subroutine afor(psy,psyb,psye,psyh)
!       п/п формирования углов начала и конца запретных направлений (курсов)
!       при решении конфликта без секторов опасных направлений перемещения
!       входы в п/п:  psy - текущее направление (курс) [рад. ]
!                     psyh - смещение по направлению перемещения (курсу) [0,п/2, ]
!       выходы в п/п:[spyb,psye]
!       редакция: 10.06.13
!       ==========================================================
        psyb=psy-psyh
         psye=psy+psyh
         if(psyb.lt.0.)psyb=6.283184+psyb
         if(psyb.eq.6.283184)psyb=0
         if(psye.ge.6.283184)psye=psye-6.283184
         return
         end

         subroutine afro(ogr,ogl,x0,y0,px,py,nr,nl,kr,kl,ea1,ea2,dal,ierr)
!       п/п имитации секторов опасных курсов при отсутствии прямого
!                           прохода по <irez=0>
!       [px,py] - координаты целевой точки перемещения
!       ogr(10,2), ogl(10,2) - координаты вершин границ пространства решений
!       [x0,y0] - координаты исходной точки имитации сектора
!       kr(9),kl(9)-выходы п/п <bereg>
!       [ea1,ea2] - грницы сектора опасных направлений перемещения
!       dal - среднне расстояние до имитируемого сектора
!       ierr признак ошибки [0 - нет, 1 - есть]
!       редакция 27.10.13.
!       ==================================================================
```

```
      dimension ogr(10,2),ogl(10,2),kr(9),kl(9)
       irk=0
       do 1 i=1,9
       i1=i+1
       ax1=ogr(i1,1)-xo
       ay1=ogr(i1,2)-y0
       ax2=ogr(i,1)-x0
       ay2=ogr(i,2)-y0
       bx1=ogl(i1,1)-x0
       by1=ogl(i1,2)-y0
       bx2=ogl(i,1)-x0
       by2=ogl(i,2)-y0
       if(kr(i),eq,0)go to 3
       if(irk,eq,0)dal=(sqrt(ax1*ax1+ay1*ay1)+sqrt(ax2*ax2+ay2*ay2))/2.
       d=(sqrt(ax1*ax1+ay1*ay1)+sqrt(ax2*ax2+ay2*ay2))/2
       if(d.gt.dal)go to 3
       dal=0
       irk=1
       x1=ax1
       y1=ay1
       x2=ax2
       y2=ay2
    3  if(kl(i).eq.0)go to 1
       if(irk.eq.0)dal=(sqrt(bx1*bx1+by1*by1)+sqrt(bx2*bx2+by2*by2))/2.
       d=(sqrt(bx1*bx1+by1*by1)+sqrt(bx2*bx2+by2*by2))/2.
       if(d.gt.dal)go to 1
       x1=bx1
       y1=by1
       x2=bx2
       y2=by2
       dal=d
       irk=2
    1  continue
       if(irk.ne.0)go to 15
    2  ierr=1
       go to 50
   15  call agl(x1,y1,ea1,ia1)
       call agl(x2,y2,ea2,ia2)
       if((ia1.eq.1).or.(ia2.eq.1))go to 2
       xp=px-x0
       yp=py-y0
       vek1=x1*yr-y1*xr
       vek2=xp*y2-yp*x2
       if((vek1.ge.0.).and.(vek.2.ge.0.))go to 50
       ea=ea1
       ea1=ea2
       ea2=ea
   50  return
       end


       subroutine posit(ogr,ogl,nr,nl,gr,a3,c,xn,yn,zon,ddop,iflag,irez)
!      п/п вывода  информации на графический экран
!      ogr(10,2) - пересчитанные правые ограничения
!      ogl(10,2) - пересчитанные левые ограничения
!      nr - к-во точек девых ограничений
!      gr(4,2) - координаты ломаной линии траектории
!      ddop(10) - допустимая дистанция сближения
!      a3(10,2) - координаты отметок ОН (пересчитанные)
!      c(10,2) - координаты векторов ист.скор. ОН (целей)
!      xn,yn - координаты вектора собств.скорости ОУ
!      zon - глубина оценки ситуации
!      iflag(10) - массив признаков сопровождения ОН (целей)
!      irez - признак результата из <mprt1>
!                 1 - результата нет
!                 2 - прямой проход
!                 3 - двухэлементный маневр
!      редакция:08.12.13.
!      ==========================================================================
       dimension ogr(10,2), ogl(10,2),gr(4,2),a3(10,2),c(10,2),
      *iflag(10),bufr(2,2),ogr1(10,2),ogl1(10,2),ddop(10)
!       масштабирование графического экрана
!      _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
       tx=(ogl(1,1)+ogr(1,1))/2
       ty=(ogl(1,2)+ogr(1,2))/2.
```

```
      txl=(ogl(1,1)-tx)*1.1
      tyl=(ogl(1,2)-ty)*1.1
      txr=(ogr(1,1)-tx)*1.1
      tyr=(ogr(1,2)-ty)*1.1
      xm1=0.
      ym1=0.
      xm2=0.
      ym2=0.
      do 1 i=1,nr
      if(xm1.le.ogr(i,1))xm1=ogr(i,1)
      if(xm2.ge.ogr(i,1))xm2=ogr(i,1)
      if(ym1.le.ogr(i,2))ym1=ogr(i,2)
      if(ym2.ge.ogr(i,2))ym2=ogr(i,2)
      ogr1(i,1)=ogr(i,1)+txr
      ogr1(i,2)=ogr(i,2)+tyr
      if(xm1.le.ogr1(i,1))xm1=ogr(i,1)
      if(xm2.ge.ogr1(i,1))xm2=ogr1(i,1)
      if(ym1.le.ogr1(i,2))ym1=ogr1(i,2)
      if(ym2.ge.ogr1(i,2))ym2=ogr1(i,2)
    1 continue
      do 2 i=1,nl
      if(xm1.le.ogl(i,1))xm1=ogl(i,1)
      if(xm2.ge.ogl(i,1))xm2=ogl(i,1)
     if(vm1.le.ogl(i,2))ym1=ogl(i,2)
      if(ym2.ge.ogl(i,2))ym2=ogl(i,2)
      ogl1(i,1)=ogl(i,1)+txl
      ogl1(i,2)=ogl(i,2)+tyl
      if(xm1.le.ogl1(i,1))xm1=ogl1(i,1)
      if(xm2.ge.ogl1(i,1))xm2=ogl1(i,1)
      if(ym1.le.ogl1(i,2))ym1=ogl1(i,2)
      if(ym2.ge.ogl1(i,2))ym2=ogl1(i,2)
    2 continue
      do 3 i=1,10
     if(iflag(1).eq.0)go to 3
      if(xm1.le.a3(i,1))xm1=a3(i,1)
      if(xm2.ge.a3(i,1))xm2=a3(i,1)
      if(ym1.le.a3(i,2))ym1=a3(i,2)
      if(ym2.ge.a3(i,2))ym2=a3(i,2)
    3 continue
      if(xm1.lt.zon)xm1=zon
      if(ym1.lt.zon)ym1=zon
      z1=-zon
      if(xm2.gt.z1)xm2=z1
      if(ym2.gt.z1)ym2=z1
      hy=abs(xm1-xm2)/399.
      hx=abs(ym1-ym2)/599.
!     _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!     рисование левой границы пространства решений
      i1=1
      call term1(i1,il,ix1,ix2,ix3,iy1,iy2,iy3,ir1,ir2,ir3)
      i1=2
      call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
      il=7
      nn=10
      call pict(ogl,hy,hx,xm2,ym2,nl,il,nn)
      call pict(ogl1,hy,hx,xm2,ym2,nl,il,nn)
!     _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!     рисование правой границы пространства решений
     call pict(ogr,hy,hx,xm2,ym2,nr,il,nn)
      call pict(ogr1,hy,hx,xm2,ym2,nr,il,nn)
!     _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!     нанесение осей
      n=2
      bufr(1,1)=0.
      bufr(1,2)=ym2.
      bufr(2,1)=0.
      bufr(2,2)=ym1
      nn=2
      il=3
      call pict(bufr,hy,hx,xm2,ym2,n,il,nn)
      bufr(1,1)=xm2
      bufr(1,2)=0.
      bufr(2,1)=xm1.
      bufr(2,2)=0.
      call pict(bufr,hy,hx,xm2,ym2,n,il,nn)
```

```
!       _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!       вычисление зоны решения задачи конфликта
        ir=zon/hx
         call perevd(ir,jr1,jr2,jr3)
!       _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!       рисование ОУ с вектором и зоной
        ir=5
         il=4
         call perevd(ir,ir1,ir2,ir3)
         bufr(1,1)=xn/2.
         bufr(1,2)=yn/2.
         bufr(2,1)=0.
         bufr(2,2)=0.
         call pict(bufr,hy,hx,xm2,ym2,n,il,nn)
         i1=4
         call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,jr1,jr2,jr3)
         call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
!       _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!       рисование ОН с векторами
        j1=7
         j2=0
         call term1(j1,j2,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
         j1=0
         j2=0
         call term1(j1,j2,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
         do 4 i=1,10
         if(iflag(i).eq.0)go to 4
         bufr(2,1)=a3(i,1)
         bufr(2,2)=a3(i,2)
         bufr(1,1)=a3(i,1)+c(i,1)/2.
         bufr(1,2)=a3(i,2)+c(i,2)/2.
         call pict(bufr,hy,hx,xm2,ym2,n,il,nn)
         ir=ddop(i)/hx
         call perevd(ir,ir1,ir2,ir3)
         call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
         if(i.lt.10)go to 11
         j1=1
         j2=0
         call term1(j3,j1,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
    12 call term1(j3,j2,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
       go to 4
    11 j1=i
       go to 12
     4 continue
!       _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!       рисование конечной точки
       go to(5,6,7),irez
     5 fx=gr(2,1)
        fy=gr(2,2)
     9 ix=abs(fx-xm2)/hx
        iy=abs(fy-ym2)/hy
       call perevd(ix,ix1,ix2,ix3)
        call perevd(iy,iy1,iy2,iy3)
        i1=3
        call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
        i1=4
        call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
        go to 8
     6 fx=gr(3,1)
        fy=gr(3,2)
        n=3
        go to 9
     7 fx=gr(4,1)
        fy=gr(4,2)
        n=4
        go to 9
!       _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
!       рисование траектории
     8 il=5
        nn=4
        call pict(gr,hy,hx,xm2,ym2,n,il,nn)
        i1=4
        do 10 ir=1,3
        call perevd(ir,ir1,ir2,ir3)
        call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
```

```
   10 continue
      return
       end

      subroutine perevd(is,is1,is2,is3)
!     п/п перевода числа is=is1*100+is2*10+is3
!     редакция: 06.12.13.
!     ========================================
      is1=is/100
       is2=(is-is1*100)/10
       is3=is-is1*100-is2*10
       return
       end


         subroutine pict(buf,hx,hy,xmin,ymin,n,il,nn)
!        п/п рисования ломаной линии из массива <buf> с n-1 звеньями
!        hx - шаг разбиения по оси <x>
!        hy - шаг разбиения по оси <y>
!        nn - размерность массива <buf>
!        xmin, ymin - нуль экрана
!        buf(1,1) - массив рисования
!        n - количество элементов <buf>
!        il - тип линии:
!                   1 - сплошная тонкая
!                   2 - пунктирная [длинный штрих]
!                   3 - штрих-пунктирная
!                   4 - пунктирная [короткий штрих]
!                   5 - точечная [подсвет-гашение]
!                   6 - подсвет конечной точки [для векторов]
!                   7 - сплошная толстая [для векторов]
!        редакция:08.12.13.
!        в п/п ось <x>-ординат; <y>-абсцисс
!        ============================================================
         dimension buf(nn,2)
           i1=3
           ix=abs(buf(1,1)-xmin)/hx
           iy=abs(buf(1,2)-ymin)/hy
           call perevd(ix,ix1,ix2,ix3)
           call perevd(iy,iy1,iy2,iy3)
           call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
           i1=5
           do 1 i=2,n
           ix=abs(buf(i,1)-xmin)/hx
           iy=abs(buf(i,2)-ymin)/hy
           call perevd(ix,ix1,ix2,ix3)
           call perevd(iy,iy1,iy2,iy3)
           call term1(i1,il,iy1,iy2,iy3,ix1,ix2,ix3,ir1,ir2,ir3)
         1 continue
           return
            end

           subroutine attb(ogr1,ogl1,px,py,nr,nl,ierr)
!        п/п продолжение ограничений для включения точки (px,py)
!        org1(10,2),
!        (px,py) - координаты включаемой точки
!        ierr - признак результат (0-есть, 1-нет)
!        редакция:21.02.13
!        _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
         dimension ogr(10,2), ogl(10,2), ogr1(10,2), ogl1(10,2)
           ierr=0
           call raspr(ogl1,ogr1,nl,nr,px,py,inr,ie)
           if((inp.eq.0).and.(ie.eq.0))go to 20
           do 30 i=1,nr
           do 30 j=1,2
        30 ogr(i,j)=ogr1(i,j)
            do 31 j=1,nl
            do 31 j=2
         31 ogl(i,j)=ogl1(i,j)
           nl1=nl+1
            nr1=nr+1
            if((nl.ge.10).or.(nr.ge.10))go to 3
            xl=ogl(1,1)-ogl(nl,1)
            yl=ogl(1,2)-ogl(nl,2)
            xr=ogr(1,1)-ogr(nr,1)
```

```
      yr=ogr(1,2)-ogr(nr,2)
      sl=px*xl+py*yl
      sr=px*xr+py*yr
      s=sl*sr
      if(s.ge.0.)go to 1
    3 ierr=1
      type 10
   10 format(1x,'ошибка в задании ограничений')
      go to 20
    1 if((sl.ge.0.).and.(sr.ge.0.))go to 2
      xm=1.1*(px-(ogl(nl,1)+ogr(nr,1))/2.)
      ym=1.1*(py-(ogl(nl,2)+ogr(nr,2))/2.)
      ogl(nl1,1)=ogl(nl,1)+xm
      ogl(nl1,2)=ogl(nl,2)+ym
      ogr(nr1,1)=ogr(nr,1)+xm
      ogr(nr1,2)=ogr(nr,2)+ym
      go to 4
    2 xm=1.1*(px-(ogl(1,1)+ogr(1,1))/2.)
      ym=1.1*(py-(ogl91,2)+ogr(1,2))/2.)
      do 6 i=1,nr
      n1=nr1-i
      n2=n1+1
      ogr(n2,1)=ogr(n1,1)
      ogr(n2,2)=ogr(n1,2)
    6 continue
      ogr(1,1)=ogr(1,1)+xm
      ogr(1,2)=ogr(1,2)+ym
      do 7 i=1,nl
      n1=nl1-i
      n2=n1+1
      ogl(n2,1)=ogl(n1,1)
      ogl(n2,2)=ogr(n2,1)
    7 continue
      ogl(1,1)=ogl(1,1)+xm
      ogl(1,2)=ogl(1,2)+ym
    4 call raspr(ogl,ogr,nl1,nr1,px,py,inr,ie)
      if((inr.ne.0).or.(ie.ne.0))go to 3
      do 32 i=1,nr1
      do 32 j=1,2
   32 ogr1(i,j)=ogr(i,j)
      do 33 i=1,nl1
      do 33 j=1,2
   33 ogl1(i,j)=ogl(i,j)
      nr=nr1
      nl=nl1
   20 return
      end

end program prog
```

Додаток Д. Алгоритми і програмні компоненти семантичного аналізу і евристичного пошуку

Д.1 - алгоритми програмних компонент семантичного аналізу повідомлень системи інтелектуального управління

За правилом формування лексем для вхідної мови, яка описується КВ-граматикою $\Gamma_{пр}$, алгоритм формування таблиці ідентифікаторів (рис. 1) є похідним щодо функціонування лексичного сканера похідної мови опису стану взаємодії ОУ та ОС.

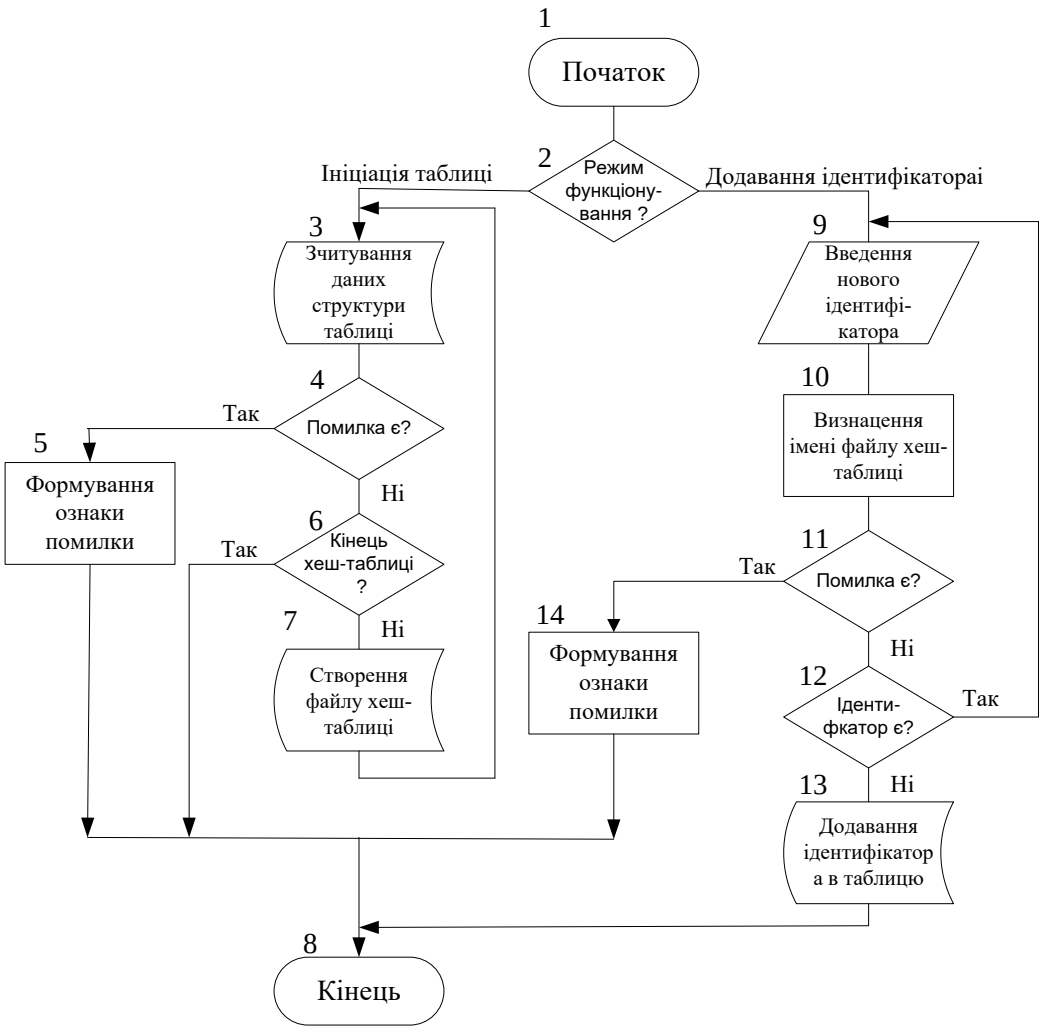


Рис. 1. Схема алгоритму формування таблиці ідентифікаторів

В блоці 2 здійснюється аналіз значення режиму функціонування алгоритму формування таблиці лексем. В разі ініціації таблиці лексем управління передається блоку 3, в іншому випадку - блоку 8.

Засобами блоку 3 зчитуються з файлу дані хеш-таблиці ідентифікаторів. Виходячи з таблиці 3.1, для хеш-таблиці ідентифікаторів достатнім є хешування за кодом першого символу ідентифікатора. Тобто достітнім буде використання символів алфаваіту англійської мови. За результатом функціонування блоку 3 створюється в пам'яті (оперативній, жорсткому магнітному диску, змінній на флеш носії, тощо) набір файлів, які мають назву, яка вміщує первинний символ ідентифікатора.

В разі відсутності похідного фалу чи відсутності можливості створення файлу хеш-таблиці ідентифікаторів блок 4 передає управління блоку 5, в іншому випадку - блоку 6.

Блок 5 формує ознаку помилки та передає управління блоку 8.

Блок 8 забезпечує завершення функціонування алгоритму.

Засоби блоку 6 забезпечують процедуру аналізу ознаки закінчення похідних даних, які використовуються для синтезу таблиці ідентифікаторів семіотичної моделі опису процесу взаємодії ОУ та ОС. В разі закінчення потоку вихідних даних управління передається блоку 8, в іншому випадку - блоку 7.

В блоці 7 забезпечуються процедури наповнення даними відповідних хеш-файлів таблиці ідентифікаторів. блок 7 передає управління блоку 3 для продовження процесу створення таблиці ідентифікаторів.

Засоби блоку 9 забезпечують введення ідентифікатора з вхідного потоку даних з метою доповнення відповідного хеш-файлу таблиці ідентифікаторів.

Функціональні елементи блоку 10 забезпечують формування ознаки (назви) хеш-файлу, який відповідає введеному ідентифікатору за прийнятим методом обчислення значення хеш-функції.

Блок 11 своїми функціональними засобами дозволяє визначити появу помилки при опрацюванні хеш-функцій та хеш-файлів при обчисленні таблиці ідентифікаторів.

При виявленні непередбачуваної або помилкової ситуації блок 11 передає управління блоку 14, в іншому випадку - блоку 12.

Блок 14 забезпечує формування ознаки помилки при опрацюванні хеш-функцій та хеш-файлів та передає управління блоку 8.

Функціональні засоби блоку 12 забезпечують процес аналізу наявності ідентифікатора в обраному хеш-файлі. В разі наявності ідентифікатора в хеш-файлі управління передається блоку 9, в іншому випадку - блоку 13.

Засоби блоку 13 забезпечують функції доповнення відповідного хеш-файлу таблиці ідентифікаторів новим значенням після чого управління передається блоку 14.

Функціональні засоби блоку 14 здійснюють аналізх похідного потоку даних на ознаку завершення. В разі, якщо похідний поток даних закінчено, управління передається блоку 8, в іншому випадку - блоку 9.

В Додатку Д.2 наведено текст програмного коду, який реалізує алгоритм, представлений на Рис. 4.29. Алгоритм реалізовано в середовищі розробки MS Visual Studio 2010 на мові програмування VC++ з інтегрованими засобами акселератора.

Наступним етапом здійснення семіотичного аналізу інформації щодо опису стану ПР з метою синтезу стратегій рішень по управлінню ОУ є лексичний аналіз опису поточної ситуації взаємодії об'єктів в ПР.

ЛА будує лексеми згідно співвідношень (3.18), (3.23), (3.24) з формального опису поточного стану ПР, який синтезується в автоматичному або автоматизованому режимі з використанням мови $L(\Gamma_{пр})$ згідно співвідношенню (3.21) та множини нетермінальних символів згідно (3.15) і (3.22).

Алгоритм функціонування лексичного сканера наведено на рис. 2.

Блок 2 здійснює ініціалізацію таблиць ключових слів, роздільників та присвоєння. Таблиця ключових слів визначається КВ-граматикою $\Gamma_{пр}$, яка за співвідношенням (3.15) вміщує набір нетермінальних символів згідно (3.22). В цьму блоці також створюються шаблони для опису роздільників, знаків присвоєння, а також лексем, ключовитх слів, ідентифікаторів та змінних.

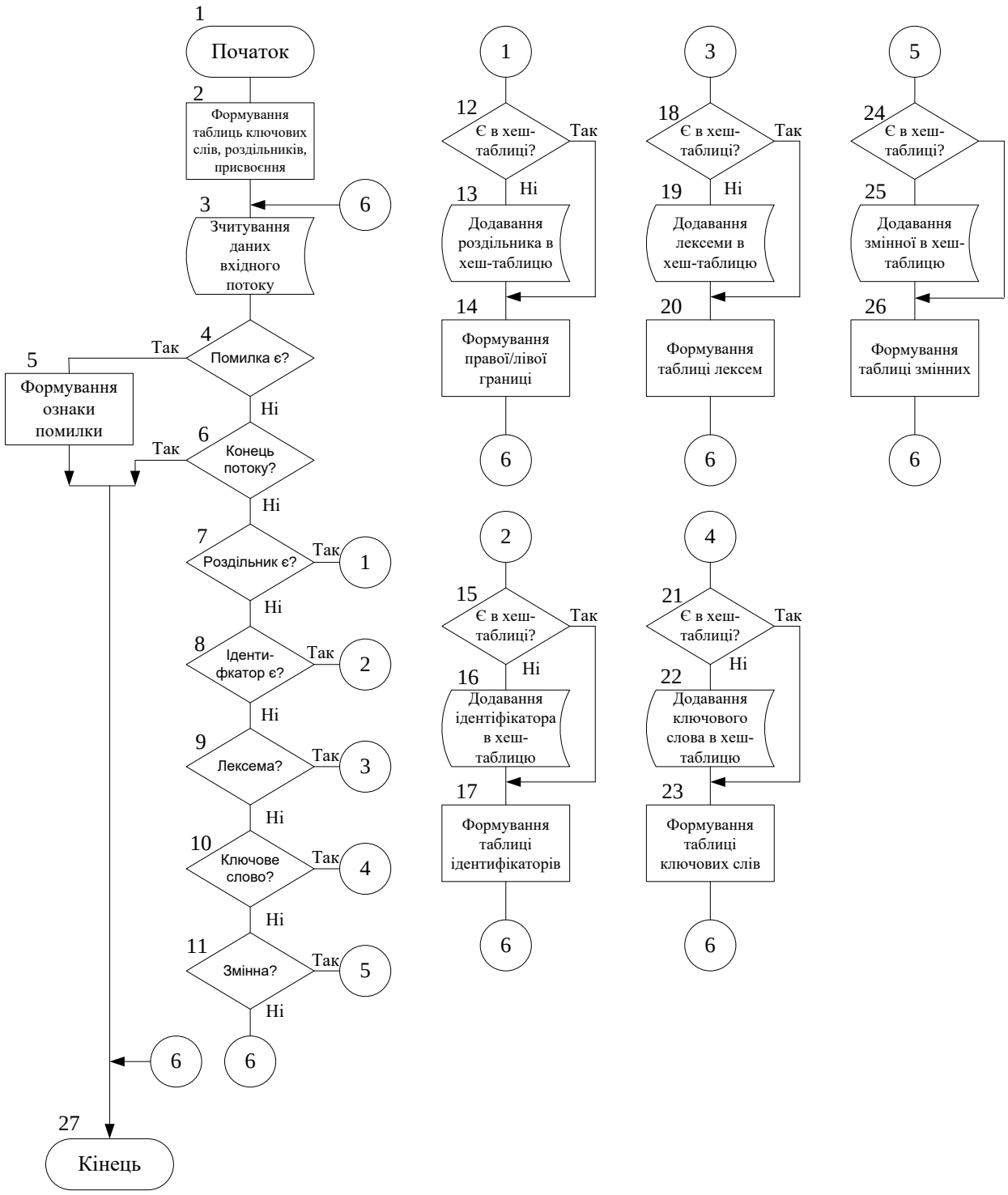


Рис. 2. Схема алгоритму функціонування лексичного сканера

Блок 3 забезпечує зчитування вхідного потоку похідних даних щодо опису взаємодії ОУ з ОС в ПР згідно співвідношень (3.1) - (3.4) для формування ланцюжків (3.5).

Блок 4 забезпечує аналіз правильності похідного потоку даних (ланцюжків опису взаємодії ОУ з ОС в ПР). В разі визначення помилкової

послідовності (неприпустимого ланцюжка) управління передається блоку 5, в іншому випадку - блоку 6.

Блок 5 здійснює формування ознаки наявності помилки в потоці похідних даних та передає управління блоку 27.

Блок 27 забезпечує завершення функціонування алгоритму.

Засоби блоку 6 аналізують забезпечують формування та аналіз події, полв'язаної з завершенням потоку похідних даних щодо опису стану взаємодії ОУ з ОС в ПР. В разі закінчення потоку похідних даних управління передається блоку 27, в іншому разі - блоку 7.

Блок 7 аналізує ознаку зчитування роздільника у вхідному потоці. В разі виявлення роздільника управління передається блоку 12, в іншому разі - блоку 8.

Блок 8 аналізує ознаку зчитування ідентифікатора у вхідному потоці. В разі виявлення роздільника управління передається блоку 15, в іншому разі - блоку 9.

Блок 9 аналізує ознаку зчитування лексеми у вхідному потоці. В разі виявлення роздільника управління передається блоку 18, в іншому разі - блоку 10.

Блок 10 аналізує ознаку зчитування ключового слова у вхідному потоці. В разі виявлення роздільника управління передається блоку 21, в іншому разі - блоку 11.

Блок 11 аналізує ознаку зчитування змінної у вхідному потоці. В разі виявлення роздільника управління передається блоку 24, в іншому разі - блоку 27.

Засоби блоку 12 аналізують наявність відповідної хеш-таблиці та здійснюють пошук виявленого у вхідному потоці роздільника. В разі наявності роздільника в хеш-таблиці управління передається блоку 14, в іншому випадку - блоку 13.

Блок 13 забезпечує процедуру додавання нового роздільника з вхідного потоку похідних даних у відповідну хеш-таблицю.

Засоби блоку 14 забезпечують формування правої та лівої границь мовної конструкції вхідного потоку, аналіз якої виконується, для подальшої побудови дерева структури повідомлення при здійсненні синтаксичного аналізу.

Після завершення функціонування блоку 14 управління передається блоку 27.

Засоби блоку 15 аналізують наявності відповідної хеш-таблиці та здійснюють пошук виявленого у вхідному потоці ідентифікатора. В разі наявності ідентифікатора в хеш-таблиці управління передається блоку 17, в іншому випадку - блоку 16.

Блок 16 забезпечує процедуру додавання нового ідентифікатора з вхідного потоку похідних даних у відповідну хеш-таблицю.

Засоби блоку 17 забезпечують формування змін до таблиці ідентифікаторів з вхідного потову похідних даних, аналіз якого виконується, для подальшої побудови дерева структури повідомлення при здійсненні синтаксичного аналізу.

Після завершення функціонування блоку 17 управління передається блоку 27.

Засоби блоку 18 аналізують наявності відповідної хеш-таблиці та здійснюють пошук виявленого у вхідному потоці лексеми. В разі наявності лексеми в хеш-таблиці управління передається блоку 19, в іншому випадку - блоку 20.

Блок 19 забезпечує процедуру додавання новоі лексеми з вхідного потоку похідних даних у відповідну хеш-таблицю.

Засоби блоку 20 забезпечують формування змін до таблиці лексем з вхідного потову похідних даних, аналіз якого виконується, для подальшої побудови дерева структури повідомлення при здійсненні синтаксичного аналізу.

Після завершення функціонування блоку 20 управління передається блоку 27.

Засоби блоку 21 аналізують наявності відповідної хеш-таблиці та здійснюють пошук виявленого у вхідному потоці ключового слова. В разі наявності ключового слова в хеш-таблиці управління передається блоку 23, в іншому випадку - блоку 22.

Блок 22 забезпечує процедуру додавання нового ключового слова за переліком, який було сформовано в блоці 2, з вхідного потоку похідних даних у відповідну хеш-таблицю.

Засоби блоку 23 забезпечують формування змін до таблиці ключових слів з вхідного потову похідних даних, аналіз якого виконується, для подальшої побудови дерева структури повідомлення при здійсненні синтаксичного аналізу.

Після завершення функціонування блоку 23 управління передається блоку 27.

Засоби блоку 24 аналізують наявності відповідної хеш-таблиці та здійснюють пошук виявленої у вхідному потоці змінної в мовній конструкції повідомлення згідно таблиці 3.1. В разі наявності змінної в мовній конструкції повідомлення в хеш-таблиці управління передається блоку 26, в іншому випадку - блоку 25.

Блок 25 забезпечує процедуру додавання нового значення змінної з вхідного потоку похідних даних у відповідну хеш-таблицю.

Засоби блоку 26 забезпечують формування змін до таблиці змінних з вхідного потову похідних даних, аналіз якого виконується, для подальшої побудови дерева структури повідомлення при здійсненні синтаксичного аналізу.

Після завершення функціонування блоку 23 управління передається блоку 27.

В Додатку Д.2 наведено текст програмного коду, який реалізує алгоритм в середовищі розробки Borland Visual C++.

Наступним етапом здійснення семіотичного аналізу інформації щодо опису стану ПР з метою синтезу стратегій рішень по управлінню ОУ є синтаксичний аналіз опису поточної ситуації взаємодії об'єктів в ПР.

СА будує лексеми згідно співвідношень (3.18), (3.23), (3.24) з формального опису поточного стану ПР, який синтезується в автоматичному або автоматизованому режимі з використанням мови $L(\Gamma_{пр})$ згідно співвідношенню (3.21) та множини нетермінальних символів згідно (3.15) і (3.22), а також враховуючи співвідношення (3.4)- (3.12).

Алгоритм функціонування синтаксичного сканера наведено на рис. 3.

СА є суто програмним засобом і використовує технології формування та обробки інформаційних структур в запам'ятовуючих пристроях сучасних обчислювальних систем. При функціонуванні СА застосовуються технології створення та оброблення програмних стеків (списків). Це дозволяє реалізувати основні принципи застосування автоматних граматик, наприклад, у вигляді КА з магазинною пам'яттю.

При функціонуванні СА в якості похідних даних використовуються таблиці, сформовані за результатами роботи ЛА, які знаходяться в запам'ятовуючих пристроях обчислювального середовища, яке здійснює обробку мовних структур опису взаємодії ОУ та ОС в ПР. В такому разі семіотичний аналіз похідного опису взаємодії ОУ з ОС є багатопрохідним.

Формально низхідний недетермінований СА має вигляд

$$A = (F^0, T^0, P^0, C^0 \subset \mathrm{Q}), \qquad (4.1)$$

а мова $L(A)$ є мовою, яка припускається аналізатором A згідно співвідношення (4.1) може бути описана для множини термінальних символів T^* для будь-якого об'єкту в ПР у вигляді

$L(A) = \{a\ O\ T^*\} \mid A$ *може побудувати виведення* a *в* $\Gamma_{пр}$, $\forall\ a \in A$.

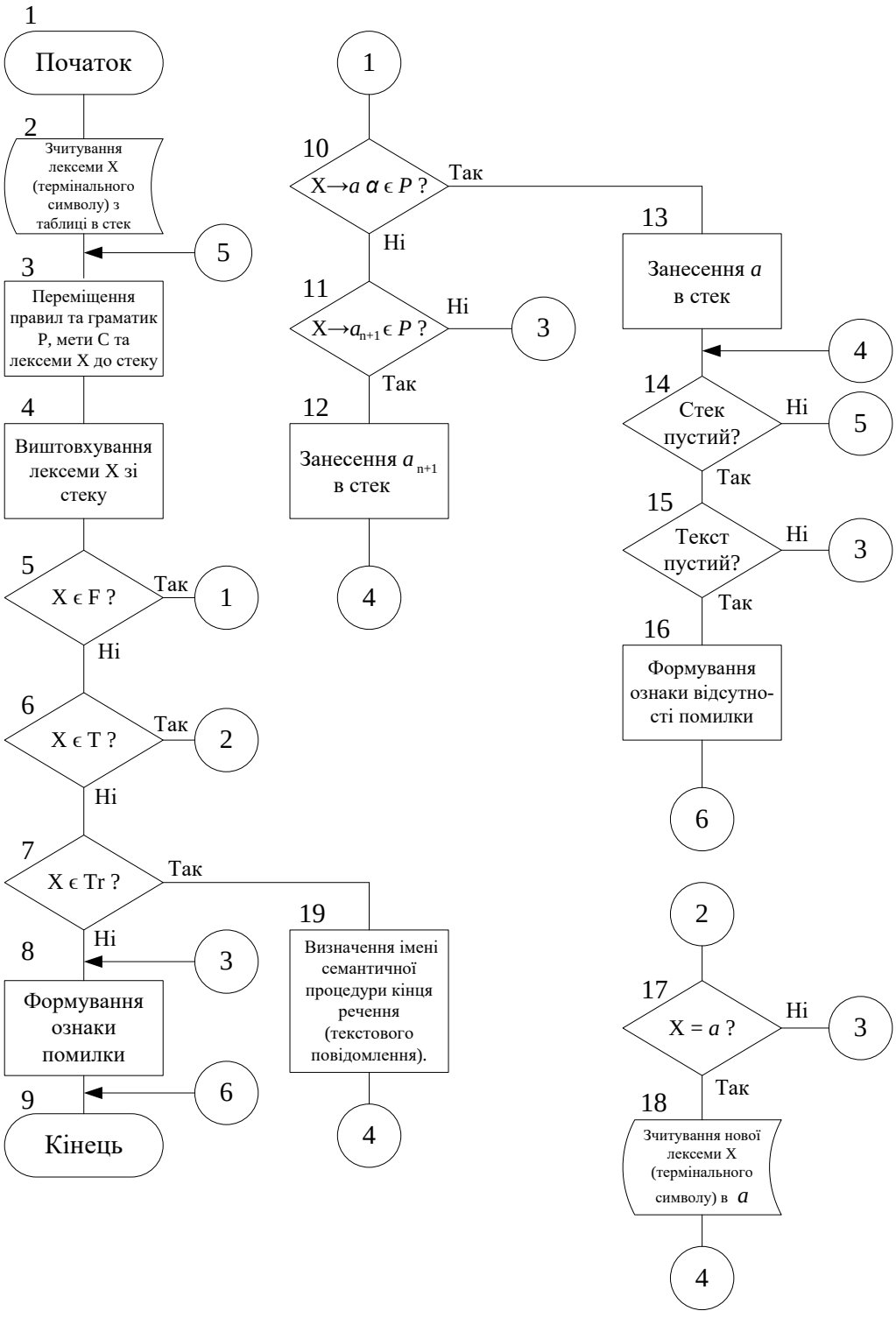


Рис. 3. Схема алгоритму функціонування синтаксичного сканера

При цьому для кожної лексеми X_i щодо кожного лівого нетермінального символа застосовується на кожному кроці її довільне нетермінальне правило.

В такому разі передбачається, що термінальні символи - це лексеми, що читаються ЛА за викликом з синтаксичного аналізатора.

Tr - це імена семантичних процедур (символів із спеціального алфавіту - таблиці символів семантичних прооцедур), які можна включати в праві частини правил у будь-якій кількості. При цьому також передбачається, що для забезпечення правильності роботи СА після успішного прочитання синтаксичним аналізатором останнього термінального символу програма читання (ЛА) повинна видати спеціальний символ кінець-тексту (по кінцю файлу), який не належить термінальному алфавіту T.

В такому разі алгоритм функціонування СА починається з зчитування засобами блоку 2 лексеми з похідного потоку мовних структур опису стану взаємодії об'єктів в ПР, які визначені КВ-граматикою $\Gamma_{пр}$ за співвіднеошенням (3.3).

Засоби блоку 3 здійснюють формування стеку, в якому здійснюється посилання на клас об'єкту типу структура, в якому знаходяться посилання на правила P^0 та сформовані ЛА таблиці термінальних T^0 та нетермінальних F^0 символів для лексеми X,

Засоби блоку 4 забезпечують "виштовхування" значення лексеми зі стеку при побудові аналізі ланцюжків опису стану взаємодії ОУ з ОС в ПР, а також ситнтез формального коду, який є похідним при синтезі ланцюжків управління ОУ. Процес "виштовхування" здійснюється шляхом запису нуля в адресу посилання на елемент структури, яка аналізується, а вивільнення пам'яті забезпечується інструментальними можливостями технології об'єктно-орієнтованого програмування шляхом використання поліморфізму та перевантаження елементів структури при програмній реалізації алгоритму.

Функціональні складові блоку 5 забезпечують процедуру аналізу приналежності лексеми, яка була "виштовхнутою" з програмного стеку СА (X), множині нетермінальних символів (F^0) згідно співвідношенню (4.1). В разі приналежності лексеми, яка аналізується, множині нетермінальних символів управління передається блоку 10, в іншому випадку - блоку 6.

Засоби блоку 10 здійснюють перетворення введеного значення лексеми X для ланцюжка $a \in A$ за умови того, що елемент a_i належить $a \in A$ та визначають належність правилу α є P^0. В разі приналежності ланцюжка введення для лексеми X умовам правил P^0 управління передається блоку 13, в іншому разі - блоку 11.

Блок 11 здійснює аналіз приналежності введеної лексеми кінцевого елементу a_{n+1} ланцюжка a правилам P^0, які визначають кінець речення повідомлення або кінець вхідного потоку повідомлень. Якщо $a_{n+1} \in P^0$, що відповідає лексемі, яка визначає кінець речення повідомлення або кінець вхідного потоку повідомлень, управління передається блоку 12, в іншому випадку - блоку 8.

Блок 8 забезпечує формування ознаки наявності помилки у вхідному потоці повідомлень.

Блок 9 забезпечує завершення функціонування алгоритму.

Функціональні засоби блоку 12 забезпечують занесення ознаки кінця речення повідомлення або кінця вхідного потоку повідомлень до моделі вихідного дерева повідомлень, що відображає результат функціонування СА. Виявлена лексема з ознакою кінця речення повідомлення або кінця вхідного потоку повідомлень заноситься до стеку у вигляді посилання на конструктор елемента структури лексеми вхідного повідомлення. Управління передається блоку 14.

Блок 13 забезпечує занесення лексеми вхідного потоку повідомлень до стеку у вигляді посилання на конструктор елемента структури лексеми вхідного повідомлення що відображає результат функціонування СА. Управління передається блоку 14.

Функціональні засоби блоку 14 аналізують стек на наявність в ньому елементів. Якщо стек пустий, управління передається блоку 15, в іншому випадку - блоку 3 для введення наступної лексеми вхідного потоку повідомлень.

Блок 15 здійснює аналіз наявності лексеми у вхідному потоці повідомлень після виявлення ознаки пустоти стеку. Якщо текст не є пустим, тобто у вхідному потоці наявна лексема, управління передається блоку 8 для формування ознаки помилки, в іншому випадку управління передається блоку 16.

Засоби блоку 16 забезпечують формування ознаки відсутності помилки та передають управління блоку 9.

Блок 6 забезпечє процедуру аналізу приналежності лексеми, яка була "виштовхнутою" з програмного стеку СА (X), множині термінальних символів (T^0) згідно співвідношенню (4.1). В разі приналежності лексеми, яка аналізується, множині термінальних символів управління передається блоку 17, в іншому випадку - блоку 7.

Функціональні засоби блоку 17 забезпечують аналіз приналежності лексеми X зі стеку ланцюжку $a \in A$. В разі належності лексеми зі стеку ланцюжку СА управління передається блоку 18, в іншому випадку - блоку 8.

Блок 18 забезпечує зчитування нової лексеми X (термінального символу) вхідного потоку повідомлень в ланцюжок a та передає управління блоку 14.

Функціональні засоби блоку 7 аналізують приналежністі лексеми X таблиці символів імен семантичних процедур спеціального алфавіту Tr символи із спеціального алфавіту Tr. В разі істиності виразу $X \in Tr$ управління передається блоку 19, в іншому випадку - блоку 8.

Блок 19 визначає ім'я семантичної процедури кінця речення (текстового повідомлення) по таблиці імен семантичних процедур та передає управління блоку 14.

В Додатку Д.2 наведено текст програмного коду, який реалізує алгоритм, представлений на Рис. 4.31. Алгоритм реалізовано в середовищі розробки Borland Visual C++.

Д.2 - програмні засоби інформаційної технології семантичного аналізу повідомлень системи інтелектуального управління

Таблиця ідентифікаторів

```
// Table_ident.cpp : Defines the entry point for the console application.
//

#include "stdafx.h"
#include "Formuls.h"

using namespace std;

void main()
{
	system("cls");
	setlocale( LC_ALL,"" );
	int sw = 3;
	Formuls Tmp;
	while (sw != 0)
	{
		cout <<  "\n <----Пожалуйста, введите номер подзадачи---->\n0- Завершение
программы\n1 - Инициация структуры файдов таблиц идентификаторов и хэш-таблицы"
			<< "\n2 - Ввод идентификатора для для анализа наличия в файлах-
таблицах\nВедите номер выполнгяемой операции -----> ";
		cin >> sw;
		if (sw == 0) //finish operation
			{
				cout << "Программа завершена" << endl;
				system("PAUSE");
			}
		if (sw == 1) //Инициация структуры файдов - таблица идентификаторов и
инициация хэш-таблицы
			{
				Tmp.InForm1();
			}
		if (sw == 2)//Ввод идентификатора для для анализа наличия в файлах-
таблицах
			{
				Tmp.InForm2();
			}
				if ((sw << 0) && (sw >> 2))  //error data
			{
				system("cls");
				cout << "\n!!!!!!!!!!! Несуществующая операция !!!!!!!!!!!"
<< endl;
				sw = 4;
			}
	}
}

#include "StdAfx.h"
#include "Formuls.h"


Formuls::Formuls(void)
{
}
```

```
Formuls::~Formuls(void)
{
}

void Formuls::InForm1()
{
    system("cls");
    setlocale( LC_ALL,"" );
    cout << "Метод Инициация структуры файдов-таблиц идентификаторов и хэш-таблицы" <<
endl;
//  Первичная инициация файлов-таблиц идентификаторов
    ifstream fs("Data_t.txt");
    num1 = 0;
    if (!fs)
          {
                cout << "\nintput file Data_t.txt is not open\n";
         }
    while (!fs.eof())
          {
                fs >> str;
                str1 = "File_";
                str1 = str1 + str[0];
                ofstream ofs(str1);
                if (!ofs)
                     {
                          cout << "\noutput file " << str1 << " is not open\n";
                     }
                ofs << num1 << endl;
                ofs.close();
          }
    fs.close();
//  Инициация файлов-таблиц идентификаторов первичными значениями идентификаторов
    num = 512;
    ifstream fs2("Data_ident.txt");
    string *bufer =  new string[num];
    if (!fs2)
          {
                cout << "\noutput file Data_ident.txt is not open\n";
         }
    while (!fs2.eof())
    {
          fs2 >> str;
                str1 = "File_";
                str1 = str1 + str[0];
                ifstream fs1(str1);
                if (!fs1)
                     {
                          cout << "\noutput file " << str1 << " is not open\n";
                     }
                k=0;
                fs1 >> num1;
                bufer[k] = str;
                while (!fs1.eof())
                {
                          fs1 >> buf;
                          if(buf != str)
                          {
                              k++;
                              bufer[k] = buf;
```

```
                        }
                }
                fs1.close();
                num1 = k;
                ofstream fs3(str1);
                if (!fs3)
                      {
                            cout << "\noutput file " << str1 << " is not open\n";
                      }
                fs3 << num1 << endl;
                for(i=0; i<num1; i++)
                {
                      fs3 << bufer[i] << endl;
                }
                fs3.close();
    }
    fs2.close();
}

void Formuls::InForm2()
{
       system("cls");
       setlocale( LC_ALL,"" );
       cout << "\nМетод Ввод идентификатора для для анализа наличия в файлах-
таблицах\nВведите имя идентификатора :   ";
       cin >> str;
       num = 512;
       string *bufer =  new string[num];
       str1 = "File_";
       str1 = str1 + str[0];
       ifstream fs1(str1);
       if (!fs1)
          {
                cout << "\noutput file " << str1 << " is not open\n";
          }
       k=0;
       i=0;
       cout << "\nСуществующая таблица идентификаторов для идентификатора ---> " <<
str << " <---" << endl;
       fs1 >> num1;
       bufer[k] = str;
       while (!fs1.eof())
       {
          fs1 >> buf;
          if(i<num1)cout << buf << endl;
          i++;
          if(buf != str)
                {
                      k++;
                      bufer[k] = buf;
//                    if(k<num1)cout << buf << endl;
                }

       }
       fs1.close();
       if(num1 == k)
       {
           cout << "\nИдентификатор ---> "<< str << " <--- присутствует в таблице
идентификаторов" << endl;
       }
       else
```

```
      {
         cout << "\nИдентификатор ---> "<< str << " <--- добавлен в таблицу
идентификаторов" << endl;
         cout << "\nНовая таблица идентификаторов для идентификатора ---> " << str <<
" <---" << endl;
         num1 = k;
         ofstream fs3(str1);
         if (!fs3)
              {
                   cout << "\noutput file " << str1 << " is not open\n";
              }
         fs3 << num1 << endl;
         for(i=0; i<num1; i++)
         {
              fs3 << bufer[i] << endl;
              cout << bufer[i] << endl;

         }
         fs3.close();
      }
}

#pragma once
#include "stdafx.h"
#include "math.h"
#include <iostream>
#include <string>
#include <windows.h>
#include <iomanip>
#include <fstream>
#include <ostream>
#include <stdlib.h>
//#include <afx.h>

using namespace std;


class Formuls
{
public:
    string str, str1, buf;
    int i, j, k, num, num1, num2, id;
//    bool ind;
    Formuls(void);
    ~Formuls(void);
    virtual void InForm1();
    virtual void InForm2();
};
```

Лексичний аналізатор

```
//---------------------------------------------------------------------------
//unit1.cpp

#include <vcl.h>
#pragma hdrstop

#include "Unit1.h"
//---------------------------------------------------------------------------
#pragma package(smart_init)
```

```
#pragma resource "*.dfm"
TForm1 *Form1;
//---------------------------------------------------------------------------
__fastcall TForm1::TForm1(TComponent* Owner)
      : TForm(Owner)
{
}
//---------------------------------------------------------------------------

void __fastcall TForm1::Button2Click(TObject *Sender)
{
  if (!OpenDialog1->Execute()) return;

  Edit1->Text = OpenDialog1->FileName;
  Memo1->Lines->Clear();
  Memo1->SetFocus();

  iNumber = 1;

  TStringList *list = new TStringList;
  list->LoadFromFile(OpenDialog1->FileName);

  for(int i=0; i < list->Count; i++)
  {
     // вывод строк в мемо
     Memo1->Lines->Add(list->Strings[i]);

     // обработка строки анализатором
     analyzer(list->Strings[i]);
  }
  delete list;

}
//---------------------------------------------------------------------------

void __fastcall TForm1::Button1Click(TObject *Sender)
{
     exit(0);
}


//---------------------------------------------------------------------------
void __fastcall TForm1::FormCreate(TObject *Sender)
{
     strKey_word = new TStringList;
     strKey_word->Add("if");
     strKey_word->Add("then");
     strKey_word->Add("else");
     strKey_word->Add("or");
     strKey_word->Add("xor");
     strKey_word->Add("and");

     InitForm();
}
//---------------------------------------------------------------------------

void TForm1::InitForm()
{
```

```
        PageControl1->ActivePage =  TabSheet1;
        PageControl1->Enabled = true;

        StringGrid1->ColWidths[0]= int(StringGrid1->Width*0.08);
       StringGrid1->ColWidths[1]= int(StringGrid1->Width*0.44);
       StringGrid1->ColWidths[2]= int(StringGrid1->Width*0.44);

       StringGrid1->Cells[0][0]="№ п/п";
       StringGrid1->Cells[1][0]="Лексема";
       StringGrid1->Cells[2][0]="Значение";

       //Шаблон вывода
       //  AddStrGrid(1, "1", "Ключевое слово", "if");
        // AddStrGrid(2, "2", "Переменная", "22");
        // AddStrGrid(3, "3", "Переменная", "322");
}

void TForm1::analyzer(String str)
{

       //убираем вначале и в конце пробелы
       str = str.Trim() ;

       // позиция первого разделителя
       int iPos = FirstPostIndex(str);

       while(iPos == 1)
       {
         str = str.Delete(1,1);
         iPos = FirstPostIndex(str);
       }



       if(iPos != 0)
       {
             String strFirstPost;

             strFirstPost = str.SubString(0,iPos-1);
             str = str.SubString(iPos,str.Length());
             // проверка на знак =

            if(strFirstPost.Pos("=") != 0)
            {
              String strSub;
              strSub = strFirstPost.SubString(0,strFirstPost.Pos("=")-1);
              strFirstPost = strFirstPost.SubString(strFirstPost.Pos("="),strFirstPost.Length());

              AddStrGrid(iNumber, (String)iNumber, FindType(strSub), strSub);
              iNumber++;

               AddStrGrid(iNumber, (String)iNumber, FindType("="), "=");
               iNumber++;

               strFirstPost = strFirstPost.SubString(strFirstPost.Pos("=")+1,strFirstPost.Length());
                AddStrGrid(iNumber, (String)iNumber, FindType(strFirstPost), strFirstPost);
               iNumber++;
```

```
            }
             else
            {
             AddStrGrid(iNumber, (String)iNumber, FindType(strFirstPost), strFirstPost);

             iNumber++;
             str = str.Trim() ;
             }
            // проверка на обработку последнего одиночного символа
             if(str.Length() != 1)
              analyzer(str);
             else
              AddStrGrid(iNumber, (String)iNumber, FindType(str), str);

      }
}

void TForm1::AddStrGrid(int iNum, String strNumber, String strType, String strValue)
{
     StringGrid1->Cells[0][iNum]=strNumber;
     StringGrid1->Cells[1][iNum]=strType;
     StringGrid1->Cells[2][iNum]=strValue;
}

int TForm1::FirstPostIndex(String str)
{
       int iPos = str.Length();
      if( (str.Pos("(") < iPos) & (str.Pos("(") != 0))
            iPos = str.Pos("(");
      if( (str.Pos(")") < iPos) & (str.Pos(")") != 0))
            iPos = str.Pos(")");
       if( (str.Pos(" ") < iPos) & (str.Pos(" ") != 0))
            iPos = str.Pos(" ");
       if( (str.Pos(";") < iPos) & (str.Pos(";") != 0))
            iPos = str.Pos(";");
       return iPos;
}

String TForm1::FindType(String str)
{
      int val=0;
      String strType = "Переменная";
      if(strKey_word->IndexOf(str) != -1)
             strType = "Ключевое слово";
      if(TryStrToInt(str,val))
              strType = "Константа";
      if(str == "=")
              strType = "Знак присвоения";
      return  strType;
}
void __fastcall TForm1::PageControl1Change(TObject *Sender)
{
}
//---------------------------------------------------------------------------

//---------------------------------------------------------------------------
//unit1.h
#ifndef Unit1H
```

```
#define Unit1H
//---------------------------------------------------------------------------
#include <Classes.hpp>
#include <Controls.hpp>
#include <StdCtrls.hpp>
#include <Forms.hpp>
#include <ComCtrls.hpp>
#include <Dialogs.hpp>
#include <Menus.hpp>
#include <Grids.hpp>
//---------------------------------------------------------------------------
class TForm1 : public TForm
{
__published:       // IDE-managed Components
      TPageControl *PageControl1;
      TButton *Button1;
      TTabSheet *TabSheet1;
      TTabSheet *TabSheet2;
      TGroupBox *GroupBox1;
      TEdit *Edit1;
      TButton *Button2;
      TMemo *Memo1;
      TOpenDialog *OpenDialog1;
      TStringGrid *StringGrid1;
      void __fastcall Button2Click(TObject *Sender);
      void __fastcall Button1Click(TObject *Sender);
      void __fastcall FormCreate(TObject *Sender);
private:  // User declarations
public:
      int iNumber;

     TStringList *strKey_word;
       __fastcall TForm1(TComponent* Owner);
      void InitForm();
     void analyzer(String);
     void AddStrGrid(int, String, String, String);
      int FirstPostIndex(String);
      String FindType(String);

};
//---------------------------------------------------------------------------
extern PACKAGE TForm1 *Form1;
//---------------------------------------------------------------------------
#endif
//---------------------------------------------------------------------------
//unit1.dfm

object Form1: TForm1
  Left = 406
  Top = 193
  Width = 725
  Height = 599
  Caption = #1051#1072#1073#1086#1088#1072#1090#1086#1088#1085#1072#1103'
'#1088#1072#1073#1086#1090#1072
  Color = clBtnFace
  Font.Charset = DEFAULT_CHARSET
  Font.Color = clWindowText
  Font.Height = -11
```

```
Font.Name = 'MS Sans Serif'
Font.Style = []
OldCreateOrder = False
OnCreate = FormCreate
PixelsPerInch = 96
TextHeight = 13
object PageControl1: TPageControl
 Left = 8
 Top = 4
 Width = 689
 Height = 505
 ActivePage = TabSheet1
 Enabled = False
 TabIndex = 0
 TabOrder = 0
 object TabSheet1: TTabSheet
  Caption = #1048#1089#1093#1086#1076#1085#1099#1081' '#1092#1072#1081#1083
  object GroupBox1: TGroupBox
   Left = 8
   Top = 8
   Width = 665
   Height = 465
   Caption = #1048#1089#1093#1086#1076#1085#1099#1077' '#1076#1072#1085#1085#1099#1077
   TabOrder = 0
   object Edit1: TEdit
    Left = 8
    Top = 24
    Width = 537
    Height = 21
    ReadOnly = True
    TabOrder = 0
    OnClick = Button2Click
   end
   object Button2: TButton
    Left = 554
    Top = 20
    Width = 96
    Height = 25
    Caption = #1047#1072#1075#1088#1091#1079#1080#1090#1100' '#1092#1072#1081#1083
    TabOrder = 1
    OnClick = Button2Click
   end
   object Memo1: TMemo
    Left = 8
    Top = 52
    Width = 641
    Height = 405
    ReadOnly = True
    ScrollBars = ssBoth
    TabOrder = 2
   end
  end
 end
 object TabSheet2: TTabSheet
  Caption = #1058#1072#1073#1083#1080#1094#1072' '#1083#1077#1082#1089#1077#1084
  ImageIndex = 1
  object StringGrid1: TStringGrid
   Left = 16
```

```
      Top = 8
      Width = 665
      Height = 457
      ColCount = 3
      FixedCols = 0
      RowCount = 1000
      TabOrder = 0
      ColWidths = (
        64
        228
        289)
      RowHeights = (
        24
        24
        24
        24
        24
        24
        24
        24
        24
        24
        24)
     end
    end
  end
  object Button1: TButton
    Left = 8
    Top = 519
    Width = 689
    Height = 33
    Cursor = crHandPoint
    Caption = #1047#1072#1074#1077#1088#1096#1080#1090#1100' '#1088#1072#1073#1086#1090#1091' '#1089'
'#1087#1088#1086#1075#1088#1072#1084#1084#1086#1081
    TabOrder = 1
    OnClick = Button1Click
  end
  object OpenDialog1: TOpenDialog
    Left = 656
    Top = 16
  end
end
//---------------------------------------------------------------------------
//unit1.ddp
                                                DIAGRAM.PORTFOLIOA@
```

Untitled1 Untitle

Синтаксичний аналізатор

```
//---------------------------------------------------------------------------
//main.cpp
#include <vcl.h>
#include <stdio.h>
#pragma hdrstop
#include "Main.h"
//---------------------------------------------------------------------------
#pragma package(smart_init)
#pragma resource "*.dfm"
TfrmMain *frmMain;
```

```
String *input_line;
String stack[256];
String terminal[32];
String rules[10];
String input_rules;
int **mega_matrix;
int all_rules=0;
__fastcall TfrmMain::TfrmMain(TComponent* Owner)
   : TForm(Owner)
{
   mmInput->Lines->LoadFromFile("input.txt");
   TLexTable *LexTable;
   LexTable = new TLexTable("input.txt");

   input_line = LexTable->GetModLines();
   input_line[LexTable->ModLength] = "end";

   AddTerminal("precedence_matrix.txt");
   all_rules = AddRules();
   int count_stack;
   stack[0] = "begin";


   int stack_index = 0;
   int lines_index = 0;
   stack[0]="begin";
   int cmp_rez;
   int left;
   int fl = 0;
   while(1)
   {
      if((input_line[lines_index]=="end") && (stack[1]=="S") && (stack[2]=="")) break;
      cmp_rez = cmp_substr(stack[first_terminal(stack_index)],input_line[lines_index]);
      if(cmp_rez==1 || cmp_rez==0)
      {
         sdvig(stack_index,lines_index);
         lines_index++;
         stack_index++;
      }
      if(cmp_rez==2)
      {
         left = package(stack_index,all_rules);
         if(left == -1)  {fl=1; break; }
         else stack_index= stack_index - left;

      }
      if(cmp_rez==-1) {fl=1; break; }
   }
   if (fl==1) {
      Edit1->Text= "Неправильная конструкция";
   }
   else Edit1->Text=input_rules;

   TTreeNode *Node1;
   tree->Items->Clear();
   tree->Items->Add(NULL, "S");
   Node1 = tree->Items->Item[0];

   for(int j = StrToInt(input_rules.Length()); j>0 ;j--)
   {
      int n = tree->Items->Count;
      for (int i=n-1;i>=0 ;i--)
      {
         if(tree->Items->Item[i]->HasChildren == false)
         if(tree->Items->Item[i]->Text =="S")
```

```
        {
          String t = rules[StrToInt(input_rules.SubString(j,1))];
          AddChild(rules[StrToInt(input_rules.SubString(j,1))],i);
          break;
        }
      }
    }
}
void TfrmMain::AddTerminal(String filename)
{
   FILE *input;
   char *temp;
   temp = new char[32];
   input = fopen(filename.c_str(),"r");
   int i=0;
   while (strcmp(temp,"end"))
   {
      fscanf(input,"%s",temp);
      terminal[i]=temp;
      i++;
   }
   terminal[i]="#";
   AddMatrix(input,i);
   fclose(input);
}
void TfrmMain::AddMatrix(FILE *input,int n)
{
   int i=0;
   char *temp;
   temp = new char [32];
   mega_matrix = new int *[n];
   for (int i = 0; i < n; i++)
        mega_matrix[i] =new int[n];

   while(!feof(input))
   {
      fscanf(input,"%s",temp);
      for(int j=0;j<n;j++) {
         fscanf(input,"%s",temp);
         if(temp[0]=='<')   mega_matrix[i][j] =  1;
         if(temp[0]=='>')   mega_matrix[i][j] =  2;
         if(temp[0]=='=')   mega_matrix[i][j] =  0;
         if(temp[0]=='-')   mega_matrix[i][j] = -1;
      }
      i++;
   }
}
int TfrmMain:: index_terminal(String trm)
{
   int count=0;
   if(trm=="begin") trm = "end";
   while(terminal[count]!="#")  {
      if(trm==terminal[count]) return count;
      count++;
   }
   return -1;
}
int TfrmMain::first_terminal(int count)
{
   for(int i=count;i!=0;i--)
   {
      if(index_terminal(stack[i])!=-1) return i;
   }
}
int TfrmMain::cmp_substr(String sub1, String sub2)
```

```
{
   if(index_terminal(sub1)>=0&&index_terminal(sub2)>=0)
      return mega_matrix[index_terminal(sub1)][index_terminal(sub2)];
   else return -1;
}
void TfrmMain::sdvig(int count_s,int count_l)
{
   stack[count_s+1]=input_line[count_l];
}
int TfrmMain::AddRules()
{
   char *temp = new char [32];
   FILE *input = fopen("rules.txt","r");
   int i=0;
   while(!feof(input))
   {
      fgets(temp,256, input);
      rules[i] = String(temp);
      rules[i].SetLength(rules[i].Length()-1);
      while(rules[i].Pos(" ")) {rules[i].Delete(rules[i].Pos(" "),1);}
      i++;
   }
   return i;
}
int TfrmMain::cmp_rules(String rls,int n)
{
   for(int i=0;i<n;i++)
   {
      if(rls==rules[i])   return i;
   }
   return -1;
}
int TfrmMain::package(int count_st,int n)
{
   String trm,tV,Vt,VVt,V;
   int ftrm = first_terminal(count_st);
   trm = stack[ftrm];
   tV = trm + stack[ftrm+1];
   Vt = stack[ftrm-1] + trm;
   V =  stack[ftrm-1] + trm + stack[ftrm+1];
   VVt = stack[ftrm-2]+ stack[ftrm-1]+trm;

   if(cmp_rules(trm,n)>=0)
   {
      int rez = cmp_rules(trm,n);
      stack[ftrm]="S";
      input_rules+=IntToStr(rez);
      return 0;
   }

   if(cmp_rules(tV,n)>=0)
   {
      int rez = cmp_rules(tV,n);
      stack[ftrm]="S";
      stack[ftrm+1]="";
      input_rules+=IntToStr(rez);
      return 1;
   }
   if(cmp_rules(Vt,n)>=0)
   {
      int rez = cmp_rules(Vt,n);
      stack[ftrm-1]="S";
      stack[ftrm]="";
      input_rules+=IntToStr(rez);
      return 1;
```

```
    }
    if(cmp_rules(V,n)>=0)
    {
       int rez = cmp_rules(V,n);
       stack[ftrm-1]="S";
       stack[ftrm]="";
       stack[ftrm+1]="";
       input_rules+=IntToStr(rez);
       return 2;
    }

    if(cmp_rules(VVt,n)>=0)
    {
       int rez = cmp_rules(VVt,n);
       stack[ftrm-2]="S";
       stack[ftrm-1]="";
       stack[ftrm]="";
       input_rules+=IntToStr(rez);
       return 2;
    }
    return -1;
}
void __fastcall TfrmMain::btnCloseClick(TObject *Sender)
{
    Close();
}
void TfrmMain::AddChild(String child,int ind_item)
{
    int lenght = child.Length();
    char *rezult_child;
    String temp_string="";
    rezult_child = new char [lenght];
    TTreeNode *Node1;
    Node1 = tree->Items->Item[ind_item];
    strcpy(rezult_child,child.c_str());

    for(int i=0;i<lenght;i++)
    {
       temp_string = temp_string + rezult_child[i];
       if(temp_string=="S")
       {
          tree->Items->AddChild(Node1,temp_string);
          temp_string="";
       }

       if(index_terminal(temp_string)>=0)
       {
          tree->Items->AddChild(Node1,temp_string);
          temp_string="";
       }
    }

}
//---------------------------------------------------------------------------

//---------------------------------------------------------------------------
//main.h
//---------------------------------------------------------------------------

#ifndef MainH
#define MainH
//---------------------------------------------------------------------------
#include <Classes.hpp>
#include <Controls.hpp>
#include <StdCtrls.hpp>
```

```
#include <Forms.hpp>
#include <Dialogs.hpp>
#include <ExtCtrls.hpp>
#include <Grids.hpp>
#include <ComCtrls.hpp>
//---------------------------------------------------------------------------
class TLexTable
{
private:
   char h(char ch)
   {
      if (ch=='n') return 'F';
      if (ch=='o') return 'A';
      if (ch=='x') return 'C';
      if (ch=='a') return 'I';
      if (ch==':') return 'P';
      if (ch==' ') return 'h';
      if (ch==';') return 'S';
      if (ch=='(') return 'T';
      if (ch==')') return 'U';
      if ((ch=='0')||(ch=='1')) return 'L';
      if (((ch>='a')&&(ch<='z'))||((ch>='A')&&(ch<='Z'))||(ch=='_')) return 'O';
      if (ch=='{') return 'M';
      return 'e';
   }
   char A(char ch)
   {
      if (ch=='r') return 'B';
      if (((ch>='a')&&(ch<='z'))||((ch>='A')&&(ch<='Z'))||(ch=='_')||((ch>='0')&&(ch<='9'))) return 'O';
      if (ch==' ') return 's';
      return 'e';
   }
   char B(char ch)
   {
      if (ch==' ') return 's';
      if (((ch>='a')&&(ch<='z'))||((ch>='A')&&(ch<='Z'))||(ch=='_')||((ch>='0')&&(ch<='9'))) return 'O';
      return 'e';
   }
   char C(char ch)
   {
      if (ch=='o') return 'D';
      if (((ch>='a')&&(ch<='z'))||((ch>='A')&&(ch<='Z'))||(ch=='_')||((ch>='0')&&(ch<='9'))) return 'O';
      if (ch==' ') return 's';
      return 'e';
   }
char D(char ch)
{
   if (ch=='r') return 'E';
   if (((ch>='a')&&(ch<='z'))||((ch>='A')&&(ch<='Z'))||(ch=='_')||((ch>='0')&&(ch<='9'))) return 'O';
   if (ch==' ') return 's';
   return 'e';
}
char E(char ch)
{
   if (ch==' ') return 's';
   if (((ch>='a')&&(ch<='z'))||((ch>='A')&&(ch<='Z'))||(ch=='_')||((ch>='0')&&(ch<='9'))) return 'O';
   return 'e';
}
char F(char ch)
{
   if (ch=='o') return 'G';
   if (((ch>='a')&&(ch<='z'))||((ch>='A')&&(ch<='Z'))||(ch=='_')||((ch>='0')&&(ch<='9'))) return 'O';
   if ((ch==' ')||(ch==';')) return 's';
   return 'e';
}
```

```
char G(char ch)
{
   if (ch=='t') return 'H';
   if (((ch>='a')&&(ch<='z'))||((ch>='A')&&(ch<='Z'))||(ch=='_')||((ch>='0')&&(ch<='9'))) return 'O';
   if (ch==' ') return 's';
   return 'e';
}
char H(char ch)
{
   if (ch==' ') return 's';
   if (((ch>='a')&&(ch<='z'))||((ch>='A')&&(ch<='Z'))||(ch=='_')||((ch>='0')&&(ch<='9'))) return 'O';
   return 'e';
}
char I(char ch)
{
   if (ch==' ') return 's';
   if (ch=='n') return 'J';
   if (((ch>='a')&&(ch<='z'))||((ch>='A')&&(ch<='Z'))||(ch=='_')||((ch>='0')&&(ch<='9'))) return 'O';
   return 'e';
}
char J(char ch)
{
   if (ch=='d') return 'K';
   if (((ch>='a')&&(ch<='z'))||((ch>='A')&&(ch<='Z'))||(ch=='_')||((ch>='0')&&(ch<='9'))) return 'O';
   if (ch==' ') return 's';
   return 'e';
}
char K(char ch)
{
   if (ch==' ') return 's';
   if (((ch>='a')&&(ch<='z'))||((ch>='A')&&(ch<='Z'))||(ch=='_')||((ch>='0')&&(ch<='9'))) return 'O';
   return 'e';
}
char L(char ch)
{
   if (ch==' ') return 's';
   return 'e';
}
char M(char ch)
{
   if (ch=='}') return 'N';
   if (((ch>='a')&&(ch<='z'))||((ch>='A')&&(ch<='Z'))||(ch=='_')||((ch>='0')&&(ch<='9'))||(ch==' ')) return 'M';
   return 'e';
}
char N(char ch)
{
   if (ch==' ') return 's';
   return 'e';
}
char O(char ch)
{
   if (ch==' ') return 's';
   if (((ch>='a')&&(ch<='z'))||((ch>='A')&&(ch<='Z'))||(ch=='_')||((ch>='0')&&(ch<='9'))) return 'O';
   return 'e';
}
char P(char ch)
{
   if (ch=='=') return 'R';
   return 'e';
}
   char R(char ch)
   {
      if (ch==' ') return 's';
      return 'e';
   }
```

```
   char S(char ch)
   {
     if (ch==' ') return 's';
     return 'e';
   }
   char T(char ch)
   {
     if (ch==' ') return 's';
      return 'e';
   }
   char U(char ch)
   {
     if (ch==' ') return 's';
     return 's';
   }
   char e(char ch)
   {
      return 'e';
   }
int        intLex[250];
AnsiString  ansiLex[250];
AnsiString  ModString;
int Length;
public:
   int ModLength;
   TLexTable(AnsiString FileName)
   {
      FILE *f;
      f = fopen(FileName.c_str(),"r");
      char ch=' ';
      char state='h';
      int type = 0;
      AnsiString lex="";
      int i=0;

      while (!feof(f))
      {
         fscanf(f,"%c",&ch);
         if (ch=='\n') ch=' ';

         switch (state){
            case 's':
            {
               i++;
               ansiLex[i] = lex.Trim();
               intLex[i]  = type;

               lex="";
               fpos_t filepos;
               fgetpos(f,&filepos);
               filepos--;
               fsetpos(f,&filepos);
               ch=' ';
               state='h';

            }break;
            case 'h':state=h(ch);break;
            case 'A':{
               type = 1;
               state=A(ch);
            }break;
            case 'B':{
               type = 2;
               state=B(ch);
            }break;
```

```
case 'C':{
   type = 1;
   state=C(ch);
}break;
case 'D':{
   type = 1;
   state=D(ch);
}break;
case 'E':{
   type = 3;
   state=E(ch);
}break;
case 'F':{
   type=1;
   state=F(ch);
}break;
case 'G':{
   type = 1;
   state=G(ch);
}break;
case 'H':{
   type = 4;
   state=H(ch);
}break;
case 'I':{
   type=1;
   state=I(ch);
}break;
case 'J':{
   type=1;
   state=J(ch);
}break;
case 'K':{
   type=5;
   state=K(ch);
}break;
case 'L':{
   type = 6;
   state=L(ch);
}break;
case 'M':state=M(ch);break;
case 'N':{
   type=7;
   state=N(ch);
}break;
case 'O':{
   type=1;
   state=O(ch);
}break;
case 'P':state=P(ch);break;
case 'R':{
   type=8;
   state=R(ch);
}break;
case 'S':{
   type=9;
   state=S(ch);
}break;
case 'T':{
   type=10;
   state=T(ch);
}break;
case 'U':{

   type=11;
```

```
            state=U(ch);
          }break;
          case 'e':
          {
            i++;
            ansiLex[i] = lex.Trim();
            intLex[i]  = 12;
            lex="";
            state=h(ch);
          }break;
        }
        AnsiString t(ch);
        lex+=t;
      };
      Length = i;
    }
/*  AnsiString GetModString()
    {
      AnsiString tmp="";
      for (int i=0;i<=Length;i++)
      {
        switch (intLex[i]){
          case  1:{tmp += "a ";}break;
          case  2:{tmp += "OR ";}break;
          case  3:{tmp += "XOR ";}break;
          case  4:{tmp += "NOT ";}break;
          case  5:{tmp += "AND ";}break;
          case  6:{tmp += "a ";}break;
//          case  7:{tmp = "êîìåíòàðèé";}break;
          case  8:{tmp += ":= ";}break;
          case  9:{tmp += "; ";}break;
          case 10:{tmp += "( ";}break;
          case 11:{tmp += ") ";}break;
//          case 12:{tmp = "îøèáêà";}break;
          default: tmp += "";
        }
      }
      return tmp;
    }
*/

    AnsiString *GetModLines()
    {
      AnsiString *ModLines;
      ModLines = new AnsiString[Length+1];

      int j=0;
      for (int i=1;i<=Length;i++)
      {
        switch (intLex[i]){
          case  1:{ModLines[j]="b"; j++;}break;
          case  2:{ModLines[j]="or";j++;}break;
          case  3:{ModLines[j]="xor";j++;}break;
          case  4:{ModLines[j]="not";j++;}break;
          case  5:{ModLines[j]="and";j++;}break;
          case  6:{ModLines[j]="b";j++;}break;
          case  8:{ModLines[j]=":=";j++;}break;
          case  9:{ModLines[j]=";";j++;}break;
          case 10:{ModLines[j]="(";j++;}break;
          case 11:{ModLines[j]=")";j++;}break;
        }
        ModLength = j;
      }
      return ModLines;
    }
```

```
  void Output(TStringGrid *sgGrid)
  {
    sgGrid->RowCount = Length+1;
    sgGrid->Cells[0][0] = "Ëåêñåìà";
    sgGrid->Cells[1][0] = "Òèï ëåêñåìû";

    for (int i=1;i<=Length;i++)
    {
      sgGrid->Cols[0]->Add(ansiLex[i]);
      AnsiString tmp="";
      switch (intLex[i]){
        case  1:{tmp = "èäåíòèôèêàòîð";}break;
        case  2:{tmp = "OR";}break;
        case  3:{tmp = "XOR";}break;
        case  4:{tmp = "NOT";}break;
        case  5:{tmp = "AND";}break;
        case  6:{tmp = "êîíñòàíòà";}break;
        case  7:{tmp = "êîìåíòàðèé";}break;
        case  8:{tmp = "îïåðàòîð ïðèñâàèâàíèÿ";}break;
        case  9:{tmp = "òî÷êà ñ çàïÿòîé";}break;
        case 10:{tmp = "îòêðûâàþùàÿ êðóãëàÿ ñêîáêà";}break;
        case 11:{tmp = "çàêðûâàþùàÿ êðóãëàÿ ñêîáêà";}break;
        case 12:{tmp = "îøèáêà";}break;
      }
      sgGrid->Cols[1]->Add(tmp);
    }
  }
};

class TfrmMain : public TForm
{
__published:          // IDE-managed Components
  TPanel *Panel1;
  TMemo *mmInput;
  TPanel *Panel2;
  TTreeView *tree;
  TPanel *Panel3;
  TLabel *Label1;
  TEdit *Edit1;
  TButton *btnClose;
  TSplitter *Splitter1;
  void __fastcall btnCloseClick(TObject *Sender);
private:   // User declarations
  void AddTerminal(String);
  void AddMatrix(FILE*,int);
  int index_terminal(String);
  int first_terminal(int);
  int cmp_substr(String,String);
  void sdvig(int,int);
  int AddRules();
  int cmp_rules(String,int);
  int package(int,int);
  void AddChild(String,int);
public:                 // User declarations
  __fastcall TfrmMain(TComponent* Owner);
};
//---------------------------------------------------------------------------
extern PACKAGE TfrmMain *frmMain;
//---------------------------------------------------------------------------
#endif

//---------------------------------------------------------------------------
//main.dfm
object frmMain: TfrmMain
```

```
 Left = 288
 Top = 80
 Width = 797
 Height = 544
 Caption = #1040#1083#1075#1086#1088#1080#1090#1084'
'#1083#1077#1082#1089#1080#1095#1077#1089#1082#1086#1075#1086' '#1089#1082#1072#1085#1077#1088#1072' '#1080'
'#1089#1077#1084#1072#1085#1090#1080#1095#1077#1089#1082#1086#1075#1086'
'#1072#1085#1072#1083#1080#1079#1072#1090#1086#1088#1072
 Color = clBtnFace
 Font.Charset = DEFAULT_CHARSET
 Font.Color = clWindowText
 Font.Height = -11
 Font.Name = 'MS Sans Serif'
 Font.Style = []
 OldCreateOrder = False
 PixelsPerInch = 96
 TextHeight = 13
 object Splitter1: TSplitter
  Left = 465
  Top = 0
  Width = 0
  Height = 506
  Cursor = crHSplit
 end
 object Panel1: TPanel
  Left = 0
  Top = 0
  Width = 465
  Height = 506
  Align = alLeft
  TabOrder = 0
  object mmInput: TMemo
   Left = 1
   Top = 1
   Width = 463
   Height = 447
   Align = alClient
   Font.Charset = RUSSIAN_CHARSET
   Font.Color = clWindowText
   Font.Height = -11
   Font.Name = 'Fixedsys'
   Font.Style = []
   ParentFont = False
   TabOrder = 0
  end
  object Panel3: TPanel
   Left = 1
   Top = 448
   Width = 463
   Height = 57
   Align = alBottom
   ParentShowHint = False
   ShowHint = False
   TabOrder = 1
   object Label1: TLabel
    Left = 136
    Top = 2
    Width = 94
    Height = 15
    Caption = #1062#1077#1087#1086#1095#1082#1072' '#1074#1099#1074#1086#1076#1072
    Font.Charset = RUSSIAN_CHARSET
    Font.Color = clWindowText
    Font.Height = -12
    Font.Name = 'Arial'
    Font.Style = []
```

```
      ParentFont = False
    end
    object btnClose: TButton
      Left = 382
      Top = 16
      Width = 75
      Height = 25
      Cursor = crHandPoint
      HelpType = htKeyword
      HelpKeyword = #1042#1099#1093#1086#1076' '#1080#1079' '#1087#1088#1086#1075#1088#1072#1084#1084#1099
      Caption = #1047#1072#1074#1077#1088#1096#1080#1090#1100
      DragCursor = crArrow
      TabOrder = 1
      OnClick = btnCloseClick
    end
    object Edit1: TEdit
      Left = 38
      Top = 20
      Width = 291
      Height = 21
      TabOrder = 0
    end
  end
 end
 object Panel2: TPanel
  Left = 465
  Top = 0
  Width = 316
  Height = 506
  Align = alClient
  TabOrder = 1
  object tree: TTreeView
    Left = 1
    Top = 1
    Width = 314
    Height = 504
    Align = alClient
    Indent = 19
    TabOrder = 0
  end
 end
end
//---------------------------------------------------------------------------
//main.dfm
```

DIAGRAM.PORTFOLIOA@

Д.3 - програмні засоби інформаційної технології евристичного пошуку

```
/*****************************************************************************
main.pro
                Copyright (c) 2012 Unregistered
*****************************************************************************/

implement main
    open core

constants
    className = "main".
    classVersion = "".

class facts  - date
    xy: (string, integer, integer).
    op: (string, string).

clauses
    classInfo(className, classVersion).

clauses

    cmp(t(A, _, _), t(B, _, _)) = greater() :- A > B, !.
    cmp(t(A, _, _), t(B, _, _)) = equal() :- A=B, !.
    cmp(_, _) = less().

    init("a").

goalReached(t(_,_,[r(_,M)|_])):-
    theGoal(R),
    R=M.
    theGoal("s").

 reconsult(FileName) :-
     retractFactDB( date),
     file::consult(FileName, date).

not_in_circle(Stt, Path):-
    not(list::isMember(r("", Stt), Path)).

getXY(M, X, Y) :-
    xy(M, X, Y), !.

cost(No, NoFilho) = C:-
    if getXY(No, XN, YN), getXY(NoFilho, XF, YF)
    then
    C = math::sqrt(((XN-XF)*(XN-XF)) + ((YN - YF)*(YN - YF)))
    else
    C= 0.0
    end if.
```

```
hn(N) = HN :-
   theGoal(S),
   if getXY(S, XS, YS), getXY(N, XN, YN)
   then
   HN= math::sqrt(((XN-XS)*(XN-XS)) + ((YN - YS)*(YN - YS)))
   else
   HN= 0.0
   end if.

search([T|Queue],S):-
if goalReached(T) then S= T
else
Extension= [E || toDaughter(T,E)],
NewQueue= list::append(Queue,Extension),
BestFirst= list::sortBy(cmp,NewQueue),
search(BestFirst, S)
end if.

toDaughter(t(_F,G,[r(B,N)|Path]),t(F1,G1,[r(Op, Child),r(B,N)|Path])):-
op(N, Child),
Op= string::format("Объект перемещается  из пункта % в пункт %", N, Child),
not_in_circle(Child, Path),
G1 = G + cost(N, Child),
F1 = G1 + hn(Child).

prtSolution( t(_,_,T)):-
foreach X= list::getMember_nd(list::reverse(T)) do
stdio::write("",X),
stdio::nl
end foreach.

   solve():-
    stdIO::write("\nИспользовать ортограф (да) ориентированный граф (нет) ? :  "),
      X = stdio::readLine(),
      if (X="нет") then reconsult("data2.txt") else reconsult("data1.txt") end if,
     if init(E),
      search([t(hn(E),0,[r("",E)])],S),stdIO::write("\nОбъект начинает движение из пункта : ",E,"\n")
         then   prtSolution(S)
         else
          stdio::write("Нет решения") end if.

   run():-
      console::init(),
      console::clearOutput(),
      console::setTextAttribute(3,4),
      console::setConsoleTitle("Эвристический поиск при синтезе решения конфликта (метод ИУВ)"),
      console::setTextAttribute(11),
      stdio::write("Эвристический поиск\n"),
      stdio::nl,
      stdIO::write("\nДвухмерное проостранство поиска\n"),
      stdio::nl,
     % stdIO::write("\nИспользовать ортограф (да) или ориентированный граф (нет) ? :  "),
```

```
      %X = stdio::readLine(),
     % if (X="нет") then reconsult("data2.txt") else reconsult("data1.txt") end if,
      solve(),
      stdIO::write("\nПродолжить расчеты (да/нет) ? :  "),
      X = stdio::readLine(),
      if (X="да") then solve () else  succeed() end if,
      succeed(),
   fail.

   run():-
      console::setTextAttribute(13),
      stdio::write("\nДля продолжения нажмите клавишу ENTER ---> "),
      _ = stdio::readLine(),
     succeed().

/*****************************************************************************
main.ph
                Copyright (c) 2012 Unregistered
*****************************************************************************/

#requires @"main.pack"
% publicly used packages
#include @"pfc\core.ph"

% exported interfaces

% exported classes
#include @"main.cl"

/*****************************************************************************
main.pack
                Copyright (c) 2012 Unregistered
*****************************************************************************/

#include @"main.ph"

#include @"main.ph"

% privately used packages
#include @"pfc\fileSystem\fileSystem.ph"
#include @"pfc\string\string.ph"
#include @"pfc\math\math.ph"
#include @"pfc\list\list.ph"
#include @"pfc\application\Exe\exe.ph"
#include @"pfc\console\console.ph"
#include @"pfc\exception\exception.ph"

% private interfaces

% private classes

% implementations
#include @"main.pro"
```

```
/*****************************************************************************
main.cl
                Copyright (c) 2012 Unregistered
*****************************************************************************/
class main
   open core

domains
   item=r(string, string).
   it= item*.
   node=t(real, real, it).
   tree=node*.tle= string.

predicates
   classInfo : core::classInfo.

   getXY:(string, integer, integer) determ (i, o, o).
   cost: (string, string) -> real.
   hn: (string) -> real.
   not_in_circle: (string, it) determ (i,i).
   theGoal: (string) procedure (o).
   toDaughter: (node, node) nondeterm (i,o).
   init:(string) procedure (o).
   goalReached:(node) determ.
   search: (tree, node) determ (i,o).
   prtSolution: (node) procedure (i).
   solve: () procedure.
   reconsult : (string) procedure (i).
   cmp:(node, node) -> compareResult.

predicates
   run : core::runnable.

end class main
```

---

data1.txt

```
op("a", "b" ).
op("b", "a" ).
op("a", "c" ).
op("c", "a" ).
op("b", "m").
op("m", "b" ).
op("m", "f").
op("f", "q").
 op("f", "q").
 op("q", "p").
 op("p", "n").
 op("p", "s").
 op("b", "c").
```

```
 op("c", "d").
 op("d", "q").
 op("d", "n").
 op("n", "d").
 op("d", "g").
 op("n", "h").
 op("n", "s").
 op("h", "g").

xy("a", 2, 4).
 xy("b", 5, 6).
 xy("c", 4, 2).
 xy("d", 7, 4).
 xy("f", 7, 8).
 xy("g", 8, 2).
 xy("h", 10, 1).
 xy("m", 9,6).
 xy("n", 11, 3).
 xy("p", 12, 6).
 xy("q", 11, 7).
 xy("s", 13, 2).
```

---

data2.txt

```
op("a", "b" ).
op("b", "m").
op("m", "f").
op("f", "q").
op("f", "q").
op("q", "p").
op("p", "n").
op("b", "c").
op("d", "q").
op("d", "g").
op("n", "h").
op("n", "s").
op("h", "g").

xy("a", 2, 4).
xy("b", 5, 6).
xy("c", 4, 2).
xy("d", 7, 4).
xy("f", 7, 8).
xy("g", 8, 2).
xy("h", 10, 1).
xy("m", 9,6).
xy("n", 11, 3).
xy("p", 12, 6).
xy("q", 11, 7).
xy("s", 13, 2).
```