

МОДЕЛЮВАННЯ ВЗАЄМОДІЇ У ГРІ МОБІЛЬНИХ АГЕНТІВ

© Кравець П.О., Маланюк Ю.В., 2008

Розглянуто питання організації командної взаємодії у грі мобільних агентів на прикладі системи гри у футбол. Введено фактор старіння інформації при навчанні мобільних агентів. Розроблено інформаційні, алгоритмічні та програмні засоби моделювання мультиагентної системи гри у футбол.

The questions of command interaction organization in game of the mobile agents are considered. The factor of the information aging for learning the mobile agents is offered. The information, algorithmic and the software tools for modeling of multiagent system of game in football is developed.

Вступ

Функціональна складність сучасних організаційних та інформаційних систем, їх динамічність, багатопараметричність, просторова розподіленість призводить до зниження ефективності централізованого керування у зв'язку зі зростанням накладних видатків на передавання потоків службової інформації. Підвищення ефективності функціонування таких систем досягається введенням різноманітних схем децентралізованого керування. Системи з децентралізованим керуванням будуються на основі автономних, адаптивних до середовища агентів (активних елементів), які досягають системних цілей за рахунок локального обміну інформацією та координації дій [1, 2].

Агент – це обчислювальна одиниця (комп'ютерна програма або робот) з елементами штучного інтелекту та раціональною поведінкою в навколишньому середовищі. Агент отримує інформацію про навколишнє середовище та діє у ньому за допомогою підсистеми рецепторів для сприйняття інформації та ефекторів для реалізації керуючих дій. Гнучкість та раціональність поведінки агента досягається за рахунок розв'язування ним локальних задач, планування, прийняття рішень та навчання [3].

Поведінка агента може бути змінена впливом інших агентів або людей. Колектив інтелектуальних агентів, які розв'язують спільну системну задачу децентралізованими методами, називається мультиагентною системою (МАС). Мультиагентні системи зародились на перетині теорії систем та розподіленого штучного інтелекту. Вони є відкритими, активними, динамічними, основаними на знаннях системами, в яких головна увага приділяється процесам взаємодії агентів з метою виникнення системи з новими якостями [1–5].

Ключовим моментом взаємодії у МАС є координація дій для досягнення цілей та розв'язання задач у ситуації кооперації або суперництва [6, 7]. У ситуації кооперації агенти намагаються об'єднати свої зусилля, щоб досягти групою результатів, яких вони не могли б досягти поодиночки. У ситуації суперництва агенти намагаються досягти найкращих індивідуальних або групових результатів.

У більшості випадків задачі МАС формулюються в умовах обмежених ресурсів. Для оволодіння ресурсами агенти можуть бути мобільними, тобто здатними мігрувати у складному середовищі в умовах невизначеності [8]. Щоб виконати завдання, агент взаємодіє з середовищем та координує свої дії з іншими агентами. Координація дій, як правило, реалізується у межах команди агентів [1, 9]. Команда агентів має спільну мету, якої досягають в умовах невизначеності через локальні навички та взаємодію членів команди. Прийняття рішень в МАС через кооперативну або антагоністичну взаємодію агентів вивчається методами теорії ігор [10 – 13].

Незважаючи на значний період та досвід дослідження МАС, питання ефективної взаємодії та координації дій агентів у межах коаліцій гравців є недостатньо вивченими та висвітленими у науковій літературі. Тому важливим є кожен крок у дослідженні цієї проблеми.

Метою цієї роботи є розроблення моделі взаємодії та координації мобільних агентів на прикладі командної гри у футбол.

Координація дій агентів

Моделі, методи та алгоритми взаємодії мультиагентних систем ґрунтуються на соціальних моделях [1 – 3]. Агенти у мультиагентних системах діють у навколишньому середовищі. При значній чисельності агентів модель навколишнього середовища має передбачати інфраструктуру для колективної взаємодії агентів. Ця інфраструктура має містити протоколи спілкування та взаємодії агентів між собою.

Протоколи спілкування дають змогу агентам обмінюватися та розуміти повідомлення. Протоколи взаємодії забезпечують спілкуватися у вигляді структурованого обміну повідомленнями. Наприклад, протоколи спілкування можуть специфікувати такі типи повідомлень, які один агент може передати іншому: пропозиція напрямку дій; згода або незгода з напрямком дій; відмова напрямку дій; корегування напрямку дій; вироблення іншої позиції щодо напрямку дій.

Агенти взаємодіють між собою з метою наближення до їх цілей або цілей системи, у якій вони функціонують. Взаємодія може дати змогу агентам координувати свої дії і поведінку, забезпечуючи їх більшу цілеспрямованість.

Координація – це властивість системи агентів виконувати деякі дії у спільному навколишньому середовищі. Ступінь координації – це міра усунення непотрібної активності у системі шляхом зменшення розбіжностей дій, уникнення тупикових ситуацій та циклічностей, а також підтримка прийняттого ступеня безпеки. Кооперація – це координація між неантагоністичними агентами. Переговори (домовленості) – це координація простих самозацікавлених, конкуруючих між собою агентів. Щоб кооперуватися успішно, кожен агент має підтримувати модель інших агентів, а також розробляти модель майбутніх взаємодій. Це визначає товариськість агентів (здатність до спілкування) [6, 7].

Систематика різних способів координації поведінки та активності агентів зображена на рис. 1.

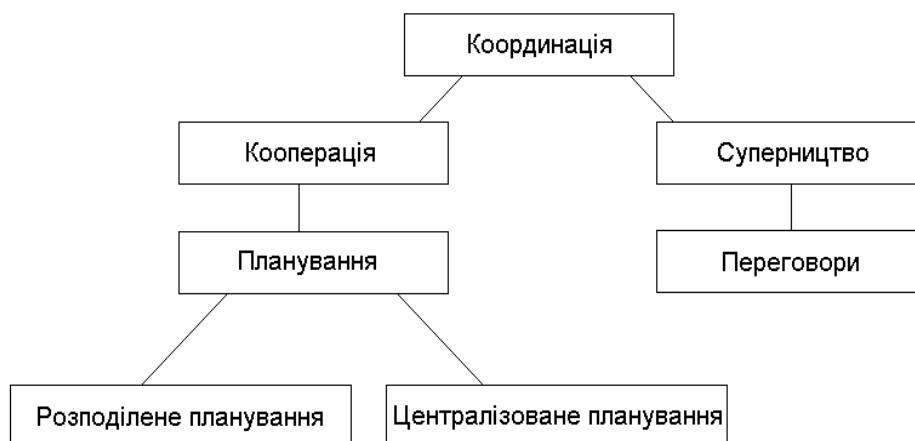


Рис. 1. Способи координації поведінки агентів

Цілеспрямованість означає, наскільки вдало система поводить себе як єдине ціле. Проблема мультиагентних систем полягає у підтримуванні глобальної цілеспрямованості без явного глобального контролю. Агенти мають мати можливість самі визначати цілі, які вони розділяють з іншими агентами, визначати спільні задачі та уникати непотрібних конфліктів. Цілеспрямованості МАС можна досягти за рахунок розподілу ролей агентів та їх підпорядкування.

Спілкування агентів здійснюється на основі комунікаційного протоколу. Він має бути лаконічним, універсальним та спільним для всіх агентів. Типовим протоколом є мова запитів знань та маніпуляції KQML (knowledge query and manipulation language) [14].

Поряд із спілкуванням, агенти потребують опису об'єктів навколишнього світу. Опис натуральною мовою є непрактичним через те, що може містити багатозначність і є надзвичайно складним для реалізації. Символьна логіка є загальним математичним апаратом для опису об'єктів, дій, ситуацій. Проста логіка (наприклад, числення предикатів першого ступеня) була визнана спроможною описати майже все, що цікавить людину або іншого інтелектуального агента, зокрема прості факти, визначення, абстракції, правила виведення, обмеження і навіть метазнання (знання про знання).

Як стандарт спілкування між елементами експертних систем, баз даних, інтелектуальних агентів тощо був запропонований Стандарт обміну знаннями (Knowledge Interchange Format, KIF) [15], який використовує префіксну форму числення предикатів першого ступеня з розширенням означень та правил виведення з неповною інформацією. Формат спілкування побудовано так, що він є зрозумілий як для комп'ютерних систем, так і для людей. Більше того, цей стандарт був розроблений як посередник у перекладах різними мовами.

Протоколи взаємодії керують спілкуванням – обміном серії повідомлень між агентами. Протоколи взаємодії розробляються для окремих агентів або систем агентів. У випадках, коли агенти мають конкуруючі цілі, тобто вони зацікавлені в успіху тільки для себе, задача протоколів полягає у максимізації корисності агентів. У випадку спільних цілей задача протоколів полягає у покращенні загальної продуктивності агентів без порушення їх автономності.

Рішення, яке потребує колективного зусилля, отримується через планування розподілу робіт між агентами. Для цього агенти наділяються певною спеціалізацією щодо можливості виконання робіт. Спеціалізація агентів здійснюється призначенням ролей. Послідовності реалізації ролей досягають механізмом планування, або побудови сценарію поведінки агентів. В загальному, розгортання сценарію залежить від передісторії МАС.

Алгоритми прийняття рішень у МАС

Вироблення та прийняття рішень у МАС здійснюються в умовах невизначеності і тому, як правило, ґрунтується на різноманітних навчальних алгоритмах. Розробник МАС повинен правильно вибрати машинний алгоритм навчання, оскільки поведінка системи може бути непередбачуваною або небажаною.

У загальному процес прийняття поточного рішення полягає у визначенні набору виходів системи до отриманого набору входів. У складній динамічній системі прийняття рішення є ситуативним, оскільки залежить від стану середовища та поточних станів інших агентів.

Пошукові алгоритми. Агенти часто діють у просторовому середовищі і для розв'язання цілої низки задач вони потребують алгоритмів пошуку. Більшість пошукових алгоритмів МАС будується на основі адаптивних методів „спроб та помилок” [16] з можливим залученням арсеналу систем штучного інтелекту.

Задачі, які розв'язуються пошуковими алгоритмами, можна поділити на три типи:

- задачі знаходження шляху;
- задачі задоволення обмежень;
- ігри двох (або більше) агентів.

Особливістю децентралізованої реалізації пошукових алгоритмів є проведення незалежних (паралельних) обчислень та обмін поточними результатами у межах локальних коаліцій агентів.

Генетичні алгоритми. При використанні генетичних алгоритмів розв'язування поставленої задачі досягається за рахунок еволюційних законів. У МАС кожен агент приймає рішення на основі розв'язування деякої локальної задачі. Вважатимемо, що агент з кращою пристосованістю дає більше значення цільової функції. Тоді задача максимізації цільової функції зводиться до пошуку найбільш пристосованого агента. Не можна помістити у систему відразу усіх агентів, що відповідають всім розв'язкам задачі. Натомість розглянемо багато поколінь, що змінювалися. Якщо ввести в дію природний відбір і генетичне спадкоємство, то отриманий світ підкорятиметься

законам еволюції. Відповідно до визначення пристосованості, ціллю цієї штучної еволюції буде створення якнайкращих рішень. Очевидно, еволюція — нескінченний процес, в процесі якого пристосованість агентів поступово підвищується. Примусово зупинивши цей процес через достатньо довгий час після його початку і вибравши найбільш пристосованого агента в поточному поколінні, ми отримаємо не абсолютно точний, але близький до оптимального розв'язок задачі.

Еволюція агентів здійснюється на основі операції схрещування хромосом (числових векторів, які є параметрами системи прийняття рішень), мутації окремих генів хромосом та механізму відбору хромосом, які забезпечують кращі значення цільової функції [17, 18].

Ігрові алгоритми. Теорія ігор надає математичні засоби для визначення можливих оптимальних стратегій прийняття рішень. Ігрові моделі функціонування МАС дають можливість дослідити кооперативні та антагоністичні фактори взаємодії агентів [10–13]. Суть гри зводиться до наступного. Кожен агент має вектор чистих стратегій – сукупність варіантів рішень (варіантів впливу на середовище). Вибір варіантів рішень здійснюється на основі векторів змішаних стратегій гравців, яким визначають умовні імовірності вибору чистих стратегій. Вибір стратегій здійснюється гравцями одночасно та незалежно. Після завершення вибору рішень усіма гравцями кожен агент отримує значення поточного виграшу, яке є функцією спільного вибору гравців або їх локальних коаліцій. Гравці можуть обмінюватися між собою інформацією про поточні виграші. Потім гравці аналізують стан гри на основі інформації про поточні виграші і змінюють стратегію своєї поведінки у вигляді перерозподілу ймовірностей векторів змішаних стратегій. Гра повторюється до певного моменту часу або до наперед заданих станів гравців [19].

Постановка ігрової задачі

Дослідження координації дій агентів виконаємо на основі моделі гри у футбол [20, 21]. Вхідними даними для цієї системи є:

- розмір поля, який становить: 120 x 80 одиниць;
- кількість гравців (агентів): 22 (по 11 за кожен команду);
- межі переміщення кожного гравця на полі;
- порядковий номер кожного гравця, його швидкість, точність, сила, тип (нападник, захисник і т.п.);
- базові правила гри для кожного гравця.

Поле гри є прямокутною сіткою позицій агентів. У грі беруть участь дві команди агентів: сині та червоні. Кожен гравець має свій номер. Гравця із м'ячем позначають жовтою міткою. Гра починається з введення м'яча червоними. Якщо м'ячем не володіє жоден з гравців, то м'яч зображається жовтим колом.

Для заборони надмірного групування гравців навколо м'яча кожен гравець має свої межі переміщення, перетнути які він не може.

Якщо в одній клітці опиняються гравець з м'ячем і суперник, то суперник відбирає м'яч лише в тому випадку, якщо його сила є більшою. Якщо сила обох гравців є однаковою, то імовірність залишення м'яча у кожного з них дорівнює 0,5.

М'яч рухається прямолінійно. Якщо гравець пасує м'яч гравцю з координатами (x, y) , то м'яч буде рухатися тільки до точки з координатами (x, y) і потім зупиниться в цій точці. Під час пасу м'яч можуть перехопити як гравець цієї ж команди, який знаходиться ближче до м'яча, так і суперник. Ні м'яч, ні гравець не можуть вийти за межі поля.

Якщо гравець володіє м'ячем, то він рухається в бік воріт суперника доти, поки відстань від нього до найближчого суперника є більшою від заданої величини (5 одиниць), або поки гравець знаходиться на закріпленій за ним ділянці поля. Якщо ж гравець із м'ячем підходить надто близько до суперника, то, щоб не втратити м'яч, він робить пас одному із гравців своєї команди, який знаходиться не більше, ніж на заданій відстані (65 одиниць).

Якщо відстань від гравця з м'ячем до воріт суперника є меншою або дорівнює заданій (15 одиниць), то гравець робить удар по воротах. Ворота червоних та синіх знаходяться зліва та справа відповідно. Ширина воріт 10 одиниць. Воротами вважається ділянка поля зі значенням Y -

координати від 35 до 45. X-координата для воріт червоних становить 0, а для воріт синіх – 119. Якщо м'яч потрапляє в цю ділянку, то така ситуація вважається голом, незалежно від того, чи м'яч потрапив туди після удару, чи разом із гравцем (гравець може завести м'яч у ворота). У змодельованій системі після голу на екран виводиться відповідний напис, робиться пауза у 2 секунди та збільшується лічильник забитих м'ячів відповідної команди. Після голу всі гравці займають свої початкові позиції і м'яч віддається команді, яка пропустила м'яч.

Якщо гравець не володіє м'ячем, але м'яч знаходиться у товаришів по команді, то він рухається в бік воріт суперника. Якщо гравець не володіє м'ячем, і м'яч знаходиться у суперника, то він рухається до гравця-суперника, який володіє м'ячем.

Метод навчання гравців

Цей метод навчання ґрунтується на методі спроб і помилок. Тобто гравець (агент) запам'ятовує всі невдалі ситуації, які призвели до небажаного результату та уникає їх надалі. Всі ситуації, крім невдалих, вважаються прийнятними до виконання.

На рис. 2 зображені три гравці, відстані між ними (a, b, c) та напрям передачі м'яча. Гравець з номером I передає м'яч гравцю по команді з номером K. Гравець-суперник з номером J намагається перехопити м'яч.

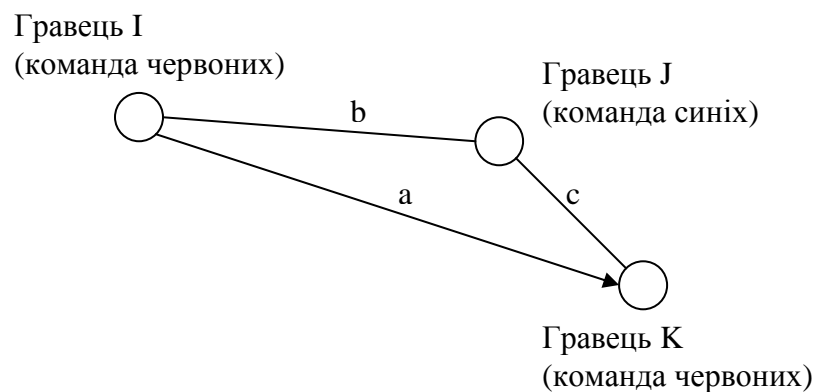


Рис. 2. Варіант розміщення гравців

Кожен гравець має свою власну базу правил та знань. До цієї бази записується тільки інформація про невдалі передачі м'яча. На початок гри гравець має порожню базу правил та знань і в процесі гри навчається. Чим довше грає гравець, тим більше він „знає”. Навчання відбувається так. Номери всіх гравців, які володіли м'ячем, записуються у масив, який опрацьовується як черга. Він є спільним для всіх гравців. Кожен гравець, який робить пас, запам'ятовує номер гравця, якому він пасує та координати всіх гравців на певний момент, крім своєї і того гравця, якому пасує, для того, щоб потім можна було обчислити відстані, якщо пас буде невдалий (на цей момент невідомо, чи пас вдалий чи ні, тому потрібно запам'ятовувати координати гравців). У результаті, якщо м'яч потрапляє не до того гравця, якому він призначався, то така ситуація вважається невдалою і потрапляє до бази правил та знань. Якщо м'яч перехоплює „свій”, то такий пас все одно вважається невдалим, оскільки на місці гравця по команді міг опинитися суперник. Якщо в базі вже існує схожа ситуація, то повторний запис не робиться. Схожою вважається ситуація, коли модуль різниці між відповідними відношеннями в базі правил та відношеннями, якими робить гравець на даний момент, не перевищує 0,1.

База правил та знань являє собою масив змінної довжини (мовою Java – колекція ArrayList), у якому кожному запису відповідає об'єкт, що містить 3 числа, які характеризують невдалу ситуацію:

- відношення відстаней: b/a ;
- відношення відстаней: c/a ;
- відношення швидкостей гравця J та гравця K.

При пасі гравець робить перевірку поточної ситуації на полі зі всіма правилами, що записані в базі. Якщо, згідно з визначеними правилами, вдалий пас зробити неможливо, то гравець робить удар м'ячем у будь-яку точку на половину поля суперника. Звичайно, м'яч можуть перехопити, поки він дійде до точки призначення.

Результати моделювання

Гра симулюється у ігровому циклі. Тривалість одного такту циклу становить 100 мс і може бути зміненою. Інтерфейс програми та ігрові обчислення виконуються в окремих потоках.

Програма написана мовою Java (JDK 1.5 - Java Development Kit) у візуальному середовищі програмування Eclipse 3.2, добре пристосованому для моделювання розподілених систем [22]. Програма може бути виконана у будь-якій операційній системі, де встановлена віртуальна машина Java (JRE – Java Runtime Environment). Інтерфейс написано з використання бібліотеки Swing. Загальний вигляд інтерфейсу зображено на рис. 3.

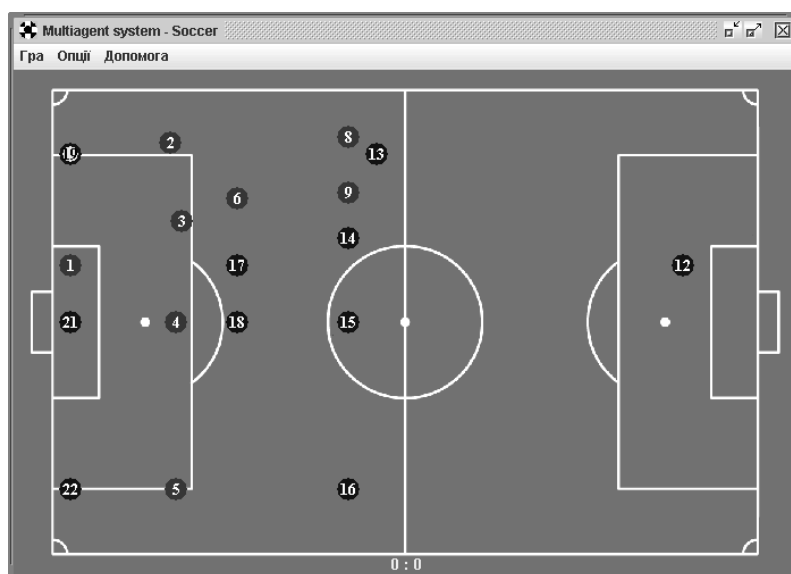


Рис. 3. Загальний вигляд мультиагентної системи «Гра у футбол»

Результатом виконання програми є динаміка кількості забитих м'ячів, кількості всіх пасів, кількості вдаливих пасів, кількості випадкових ударів та кількості топологічних станів гравців, які визначають базу знань гри. Результати виводяться на консоль гравця, номер якого задається параметром командного рядка.

На рис. 4 зображено поведінку одного із гравців в різні ігрові такти. Як видно з рисунка, в процесі гри збільшується кількість точних пасів гравців із збільшенням кількості правил, тобто гравці навчаються з часом займати вигідні або результативні позиції. Але настає такий момент, коли в базі є вже достатня кількість правил, згідно з якими не можна віддавати пас у цій ситуації. Тоді гравець, згідно із своєю закладеною поведінкою, робить удар м'ячем на половину поля суперника. Це призводить до зростання кількості випадкових передач м'яча під час гри.

Для уникнення такої ситуації, коли гравець переставав робити паси, було введено фактор старіння інформації у базі знань, згідно з яким більший обмежувальний вплив на реалізацію поточного ходу гравця мають нещодавно записані у базу дані про невдалі ходи. Номер правила в базі знань вибирається випадково, причому імовірність застосування раніше зробленого невдалого ходу є більшою, ніж новішого.

Отримані для моделі гри зі старінням інформації результати подано на рис. 5.

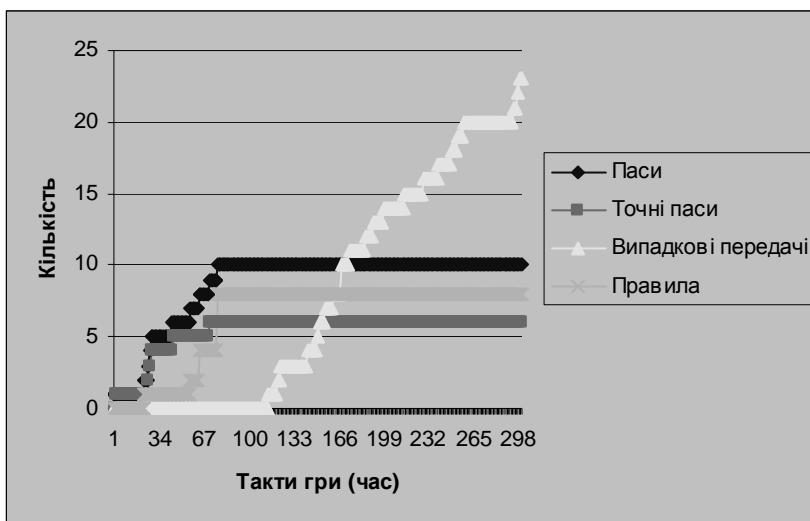


Рис. 4. Результат навчання гравця

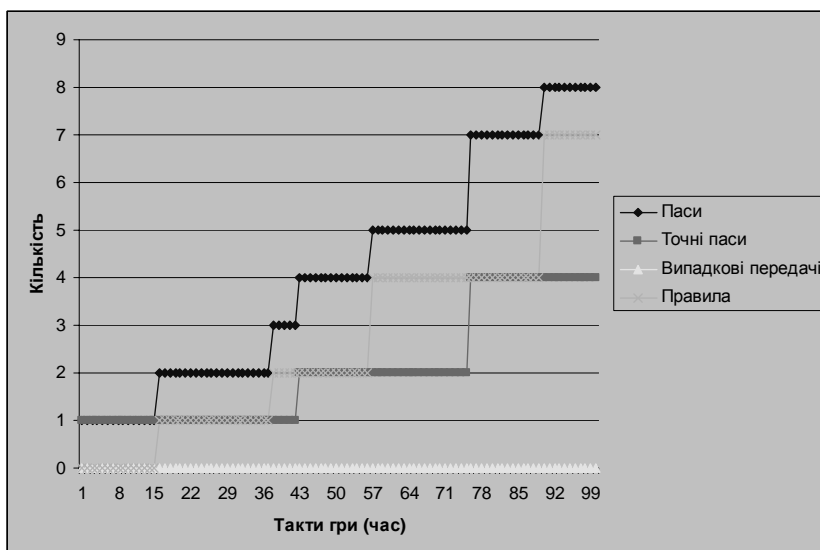


Рис. 5. Результат навчання гравця з врахуванням старіння знань

Як видно з наведених на рис. 5 графіків, кількість нецільових випадкових передач для моделі гри зі старінням інформації є мінімальною. В результаті база знань не переповнюється, гра розгортається у нормальному режимі, кількість пасів та точних передач зростає.

Спроектвана система демонструє основні принципи роботи мультиагентної системи, побудованої на основі взаємодії мобільних, інтелектуальних та самонавчальних агентів. Мобільних, оскільки агенти переміщуються (змінюють свої координати) у межах ігрового поля. Інтелектуальних, оскільки агенти приймають рішення згідно із встановленими правилами. Самонавчальних, оскільки в процесі гри агенти «запам'ятовують» всі невдалі передачі м'яча і використовують здобуті знання для подальшої гри.

Змодельована система може бути корисною для побудови схожих мобільних систем, таких як військові навчання та розвідування простору в умовах невизначеності.

Висновки

У цій роботі розглянуто питання командної взаємодії мобільних агентів на прикладі моделі гри у футбол. Розроблено алгоритм та програмні засоби моделювання командної гри агентів з формуванням навчальної бази правил. Запропоновано модель старіння інформації у базі правил, що

дало змогу уникнути тупикових станів навчання гравців у зв'язку з формуванням повної множини правил при обмеженій дискретизації простору гри.

Розроблені алгоритмічні та програмні засоби моделювання поведінки команд агентів можуть бути використані для керування МАС в інших предметних областях.

1. Gerhard Weiss and Sandip Sen, editors. *Adaptation and Learning in Multiagent Systems*. Springer Verlag, Berlin, 1996.
2. Wooldridge M. *An Introduction to Multiagent Systems*. John Wiley & Sons (Chichester, England), 2002.
3. Бурков В.Н., Новиков Д.А. *Теория активных систем: состояние и перспективы*. – М.: СИНТЕГ, 1999.
4. Котенко И.В. *Модели и алгоритмы обеспечения гарантированного времени решения задач системами, основанными на знаниях* // Международная конференция по мягким вычислениям и измерениям. SMC'2000: Сборник докладов. – СПб, 2000.
5. Тарасов В.Б. *Эволюционная семиотика и нечеткие многоагентные системы – основные теоретические подходы к построению интеллектуальных организаций* // Информационные технологии и вычислительные системы. – 1998. – №1. – С.54–68.
6. Vittikh V.A., Skobelev P.O. *Multy-agent systems for modeling of self-organization and cooperation processes*. // *Proceedings of the XIII International Conference on the Application of Artificial Intelligence in Engineering*. Galway. 1998, pp. 91 – 96.
7. Городецкий В.И. *Многоагентные системы: основные свойства и модели координации поведения* // Информационные технологии и вычислительные системы. – 1998. – №1. – С.22–34.
8. Gray R., Kotz D., Cybenko G. and Rus D. *Mobile agents: Motivations and state-of-the-art systems* // *Thayer School of Engineering / Department of Computer Science*. – Dartmouth College. - Hanover, New Hampshire 03755. - April 19, 2000.
9. Grosz B., Kraus S. *Collaborative plans for complex group actions* // *Artificial Intelligence*, Vol.86, 1996.
10. Воробьев Н.Н. *Основы теории игр: Бескоалиционные игры*. – М.: Наука, 1984.
11. Доманский В.К. *Стохастические игры* // *Математические вопросы кибернетики*. – 1988. – № 1. – С. 26–49.
12. Jos'e M. Vidal. *Learning in Multiagent Systems: An Introduction from a Game-Theoretic Perspective*. - University of South Carolina, Computer Science and Engineering, Columbia, SC 29208. - In Eduardo Alonso, editor, *Adaptive Agents: LNAI 2636*. Springer Verlag, 2003.
13. Fudenberg D., Levine D.K. *The Theory of Learning in Games*. MIT Press, 1998.
14. *Knowledge Query and Manipulation Language*, http://en.wikipedia.org/wiki/Knowledge_Query_and_Manipulation_Language.
15. *Knowledge Interchange Format*, <http://logic.stanford.edu/kif/kif.html>.
16. Радмизин Л.А., Рина К.К., Тарасенко Г.С. *Адаптация случайного поиска*. – Рига: Зинатне, 1973.
17. Holland J.H. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, 1975.
18. Goldberg D.E. *Genetic Algorithms in Search, Optimization and Mashine Learning*. – Addison-Vesley, 1989.
19. Назин А.В., Позняк А.С. *Адаптивный выбор вариантов: Рекуррентные алгоритмы*. – М.: Наука, 1986.
20. Stankevitch L.A. *A cognitive agent for soccer game* // *Proceeding of First Workshop of Central and Eastern Europe on Multi-agent Systems*. CEEMAC'99. S-Petersburg, 1999.
21. Stone P. *Layered learning in multiagent systems. A winner approach to Robotic Soccer*. The MIT Press, 2000.
22. Bellifemine, Fabio. *Developing multi-agent systems with JADE* / Fabio Bellifemine, Giovanni Caire, Dominic Greenwood. John Wiley&Sons, Ltd, 2007.