

**Б. А. САЛТАН, Д. А. СОБКО, І. В. КУЛАКОВСЬКА**

### **УДОСКОНАЛЕННЯ ПРИЙНЯТТЯ РІШЕНЬ З ВИБОРУ ІГРОВОЇ ЗБРОЇ КОМП'ЮТОРНИМ БОТОМ З ВИКОРИСТАННЯМ АЛГОРИТМУ НАВЧАННЯ НЕЙРОННОЇ МЕРЕЖІ FALCON**

Пропонується розроблена математична модель алгоритму вибору зброї ботом в комп'ютерній грі жанру шутер з використанням сценарного аналізу, що наближає поведінку штучного інтелекту (ШІ) до людської у ситуаціях, пов'язаних із вибором засобу ураження та прогнозуванням переміщення супротивника, що дозволяє гравцю отримати від гри досвід високої якості. Розроблений інструментарій зорієнтований на подальшу розробку та оптимізацію ігрового процесу, який покликаний покращити інтерактивність та ступінь реалізму у комп'ютерних іграх. Наведені проєктні рішення щодо розробленої моделі; проводиться узагальнення проведеної роботи та отриманих результатів, визначаються перспективи подальшого навчання ШІ з використанням нейронних мереж. Математична модель, що побудована в даній роботі, дає змогу визначати параметри керування, а разом з ними і стратегії, які регулюють рівень складності гри і дають змогу гравцеві отримати перемогу при правильному виборі його власної стратегії. Проаналізовано кілька підходів до вирішення поставленої задачі, а саме із застосуванням логічного висновку, досвіду та гібридного підходів, сценаріїв, дерев класифікації та регресії, алгоритму FALCON.

**Ключові слова:** математична модель, комп'ютерна гра, параметри керування, алгоритми вибору, сценарний аналіз, логічний висновок, досвід та гібридний підхід; сценарії; дерева класифікації та регресії; алгоритм FALCON.

**Б. А. САЛТАН, Д. А. СОБКО, И. В. КУЛАКОВСКАЯ**

### **СОВЕРШЕНСТВОВАНИЕ ПРИНЯТИЯ РЕШЕНИЙ ПО ВЫБОРУ ИГРОВОГО ОРУЖИЯ КОМПЬЮТОРНЫМ БОТОМ С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМА ОБУЧЕНИЯ НЕЙРОННОЙ СЕТИ FALCON**

Предлагается разработанная математическая модель алгоритма выбора оружия ботом в компьютерной игре жанра шутер с использованием сценарного анализа, которая приближает поведение искусственного интеллекта (ИИ) к человеческому в ситуациях, связанных с выбором средства поражения и прогнозированием перемещения противника, а также позволяет игроку получить от игры опыт высокого качества. Разработанный инструментальный ориентирован на дальнейшую разработку и оптимизации игрового процесса, который призван улучшить интерактивность и степень реализма в компьютерных играх. Приведены проектные решения по разработанной модели; проводится обобщение проведенной работы и полученных результатов, определяются перспективы дальнейшего обучения ИИ с использованием нейронных сетей. Математическая модель, построенная в данной работе, позволяет определять параметры управления, а вместе с ними и стратегии, которые регулируют уровень сложности игры и позволяют игроку одержать победу при правильном выборе его собственной стратегии. Проанализированы несколько подходов к решению поставленной задачи, а именно с применением логического вывода, опыта и гибридных подходов, сценариев деревьев классификации и регрессии, алгоритма FALCON.

**Ключевые слова:** математическая модель, компьютерная игра, параметры управления, алгоритмы выбора, сценарный анализ, логический вывод, опыт и гибридный подход; сценарии; деревья классификации и регрессии; алгоритм FALCON.

**B. A. SALTAN, D. A. SOBKO, I. V. KULAKOVSKA**

### **IMPROVING COMPUTER BOT WEAPON CHOICE DECISIONS USING FALCON NEURAL NETWORK LEARNING ALGORITHM**

The paper deals with the research and development of computer games using neural networks. A mathematical model of the algorithm for selecting a weapon by a bot in a shooter computer game using scenario analysis is proposed. The model approximates the behavior of the artificial intellect (AI) to human in situations involving choosing the means of defeat and predicting the movements of the enemy, which allows the player to obtain high quality game experience. The developed toolkit is focused on the further development and optimization of the game process, aimed at improving interactivity and degree of realism in computer games. The design decisions concerning the developed model are given; the generalization of the work performed and the results obtained, the prospects for further study of AI using neural networks are determined. The mathematical model built in the given work allows one to determine the control parameters, and with them the strategies that regulate the level of complexity of the game and allow the player to win in case of the right choice of his own strategy. Several approaches to solving the problem are analyzed, namely, the once involving logical conclusion, experience and hybrid approaches; scripts; classification and regression trees; FALCON algorithm.

**Key words:** mathematical model, computer game, control parameters, choice algorithms, scenario analysis, logical conclusion, experience and hybrid approach; scripts; classification and regression trees; FALCON algorithm.

**Вступ.** XXI століття – це час комп'ютерних технологій, які щодня розвиваються і використовуються як для праці, так і для відпочинку. Окрему нішу займають комп'ютерні ігри, тобто комп'ютерні програми або частини комп'ютерних програм, що служать для організації ігрового процесу (*геймплея*).

*Шутер* (від англ. shoot – *стріляти*) – популярний жанр відео ігор, ігровий процес яких заснований на боях в режимі реального часу з використанням, як правило, вогнепальної зброї. Програмування ігрового штучного інтелекту в шутерах зводиться, головним чином, до розробки моделі поведінки ботів. *Ботами* прийнято називати ворожих до гравця персонажів, які контролюються комп'ютером. Для того, щоб ігровий процес шутера був цікавим і реалістичним, боти повинні бути запрограмовані таким чином, щоб, з одного боку, їх було складно переграти, але, з іншого боку, у людини залишалися шанси на перемогу. В ідеалі, боти повинні діяти подібно персонажам, контрольованим реальними людьми.

У зв'язку з цим, виникає ряд непростих завдань, таких як реалізація тактики командного ведення бою ботами; моделювання прицілювання, стрільби, переміщення ботів по карті; реалізація алгоритму вибору зброї ботом в залежності від ігрової ситуації. У даній роботі розглядається рішення останньої з перерахованих завдань.

**Аналіз останніх досліджень та публікацій.** Інформаційна база у сфері математичних моделей для комп'ютерних ігор жанру шутер майже відсутня. Аналіз джерел, знайдених в мережі Інтернет([1] – [6]), показав, що данні моделі представлені тільки у формі опису на форумах для гравців.

Основною метою роботи є дослідження математичної моделі алгоритму вибору стратегій ігрової програми, які визначають рівень складності гри і гарантують певний відсоток вигравів гравцеві, за умови, коли він визначить вірну стратегію гри для себе, з використанням сценарного аналізу, нейронних мереж, їх застосування в алгоритмі вибору зброї ботом в тривимірному шутері від першої особи, тестування реалізованих алгоритмів і виявлення переваг і недоліків кожного підходу.

Задача штучного інтелекту для ігор полягає в імітації поведінки об'єктів реального світу. Для цього слід почати розгляд штучного інтелекту з базових компонентів – від низькорівневих правил і алгоритмів до більш високого рівня, на якому працює тактичний і стратегічний ШІ. При цьому, слід домогтися високої ефективності роботи системи ШІ, оптимізувати її для використання на комп'ютерах. Це означає, що потрібні нові наукові і практичні розробки, які б дозволили більш широко використовувати технологію створення засобів ШІ.

**Постановка задачі.** В ході вивчення предметної області, було проаналізовано ряд матеріалів, які стосуються деталей використання сценарного аналізу, штучного інтелекту в відео іграх, в основному, в шутерах від першої особи. У багатьох сучасних іграх поведінка ботів характеризується високою передбачуваністю. Це пов'язано з тим, що розробники наголошують на ефективності застосовуваних алгоритмів, не замислюючись безпосередньо про модель прийняття рішень ботом. Так, в шутерах Left 4 Dead і Killzone для розрахунку траєкторії переміщення ботів використовується алгоритм  $A^*$ . При цьому, моделі поведінки ботів засновані на використанні кінцевих автоматів. В результаті, поведінка ботів повністю детермінована, на відміну від поведінки ігрових персонажів, контрольованих реальними людьми [1]. Безумовно, такий підхід до створення ігрового штучного інтелекту істотно знижує інтерес користувачів до проекту.

Крім передбачуваності прийнятих ботом рішень, недоліком такого підходу є велика кількість параметрів, які необхідно підбирати (в описаному вище прикладі таким параметром є *enemy.distance* – відстань між ботом і ворогом). Для того, щоб наблизити поведінку бота до реальності, потрібно враховувати багато факторів, і налаштування експертної системи стає складним завданням. Пропонується використовувати генетичні алгоритми для вирішення цього завдання.

Навчання з підкріпленням – сукупність методів машинного навчання, в ході яких агент навчається, взаємодіючи з деяким навколишнім середовищем. Агент взаємодіє з оточенням (*Environment*), застосовуючи різні дії (*Actions*) в залежності від значень певних параметрів (*State*) і отримуючи нагороду (*Reward*). Мета таких методів полягає в навчанні агента вибирати дії, ґрунтуючись на поточному стані навколишнього середовища, таким чином, щоб отримати максимальну нагороду [2]. У термінах поставленого завдання, використання методів навчання з підкріпленням в алгоритмі вибору зброї дозволяє боту протягом гри накопичувати досвід використання кожної зброї в різних ігрових ситуаціях і діяти відповідно із зібраними знаннями, постійно модифікуючи тактику вибору зброї з метою збільшення одержуваної нагороди.

*Штучна нейронна мережа* (ШНМ) – математична модель, побудована за принципом організації мереж нервових клітин живого організму. Нейронні мережі широко застосовуються в різних областях. Зокрема, вони активно використовуються в аналізі даних, наприклад, при вирішенні задач класифікації даних, кластеризації даних, апроксимації неперервних функцій.

Залежно від характеру навчання, ШНМ прийнято розділяти на ті, що навчаються з учителем і навчаються без учителя. Також, останнім часом, як правило, окремо виділяють нейронні мережі, які навчаються з підкріпленням. Кожен з перерахованих типів ШНМ призначений для вирішення певного класу задач. Навчання з учителем застосовується, коли заздалегідь відомі вихідні значення мережі при певних вхідних значеннях. Нейронні мережі даного типу підходять для вирішення таких завдань, як класифікація даних. Серед алгоритмів навчання таких мереж найбільш відомий *метод зворотного поширення помилки* (*backpropagation*) [3]. Нейронні мережі, які навчаються без вчителя, має сенс використовувати в тих випадках, коли немає ніякої апріорної інформації про те, якими мають бути вихідні значення мережі. Такі нейронні мережі, як правило, застосовуються для вирішення завдань кластеризації даних. У даній роботі була використана нейромережева модель FALCON, яка навчається з підкріпленням.

**Модифікації алгоритму навчання FALCON.** В ході роботи над проектом було прийнято рішення внести певні зміни в оригінальний алгоритм навчання FALCON з метою підвищення якості навчання нейронної мережі. Для нейрона когнітивного поля були визначені два цілочисельних параметра.

Перший параметр спочатку ініціалізується нулем. Кожен раз, коли нейрон вибирається для навчання і вибору дії, в разі, якщо застосована дія принесла позитивну нагороду, значення цього параметра збільшується на 1. В іншому випадку (якщо нагорода нульова), значення параметра зменшується на 1. Якщо значення параметра стає негативним, то такий нейрон підлягає негайному видаленню з когнітивного поля мережі. Даний параметр дозволяє боту не запам'ятовувати непотрібну інформацію і видаляти застарілі знання про навколишнє середовище.

Другий параметр – локальний лічильник випадків помилкового використання нейрона. Він збільшується в тих випадках, коли активність нейрона призводить до нульової нагороди, і обнуляється, коли бот отримує позитивну нагороду. Якщо значення параметра перевищує деяку константу (значення константи визначається користувачем нейронної мережі, в ході тестування було прийнято рішення зробити цю константу рівною трьом), тобто, якщо кілька разів поспіль активність нейрона призвела до отримання нульової нагороди, то відповідний нейрон видаляється з когнітивного поля. Даний параметр дозволяє боту швидко реагувати на зміни, що відбуваються в навколишньому середовищі (наприклад, на зміну тактики супротивника).

Під час «військових дій» можна зіштовхнутися із багаточисленними і різноманітними сценаріями, такими як рейди по тилах супротивника або засідки, лобові атаки і флангові напади, маневрена війна і позиційні бої, і т.п. Солдати, опинившись в певній ситуації, безперечно, прагнуть знайти найкращі інструментальні засоби для виконання свого бойового завдання, що дозволяє збільшити ймовірність успіху або просто забезпечити шанси на виживання.

До категорії стратегій високого рівня відносяться навички вибору зброї, які доповнюють навички володіння зброєю. Вибір зброї вдає із себе задачу, при вирішенні якої в процесі гри виникає багато варіантів. Під час бою найчастіше виявляється успішним один-єдиний якісний спосіб поведінки, незалежно від контексту. А коли доводиться маніпулювати самою зброєю, боту доводиться динамічно враховувати набагато більше факторів.

При вирішенні завдання вибору зброї може використовуватися навчання. При цьому виникають цікаві тенденції в розвитку проекту гри або навіть засобів ШІ. Наприклад, ми можемо визначити, чи є вибір зброї цілком виправданим і, в разі необхідності, модифікувати логіку поведінки гри.

Вихідні дані:

- розроблювані боти повинні бути здатними до стрільби, вони не зобов'язані мати ідеальну поведінку;
- для вибору зброї в грі абсолютно необхідна підтримка декількох видів зброї, як завжди, таку можливість має надавати ігрова машина;
- передбачається, що середовище дозволяє учасникам битви приймати вельми різноманітні розташування, що може приводити до того, що бойова ситуація кожен раз буде складатися по-різному.

**Вибір зброї із застосуванням сценаріїв.** Безумовно, засоби сценарної підтримки не є результатом досліджень виключно в області ШІ, але неможливо заперечувати й те, що ці засоби дуже широко використовуються в додатках ШІ, і особливо в комп'ютерних іграх [5].

Необхідність в застосуванні зручних мов сценаріїв не вичерпується традиційними підходами до створення ігрових засобів ШІ; в міру покращення додатків ШІ сценарна підтримка в них не виключається, а просто починає обслуговувати інші потреби (відмінні від потреб реалізації форм поведінки). Системи, які діють на основі застосування сучасних методів ШІ, виграють за рахунок використання мов сценаріїв також і в тому, що ці мови повністю забезпечують взаємодію основних компонентів. Рішення можуть бути отримані логічним шляхом на підставі вихідних фактів. Ми будемо керуватися даними про властивості зброї (наприклад, про ступінь завданої шкоди і темп стрільби) для вибору найбільш підходящої зброї. Застосовується декларативний підхід для визначення властивостей зброї. Це дозволяє надати засобам ШІ інформацію про скорострільність і очікувану шкоду.

Головна мета розробників комп'ютерних ігор – створити програму, яка б мала кілька рівнів складності, ігровий процес, та викликала високий рівень зацікавленості гравця і дозволяла б йому при виборі правильних стратегій поведінки досягати перемоги. Крім того, перемога гравця не повинна бути досить легкою. Отже, при розробці програмного коду комп'ютерної гри необхідно мати досить гнучкий алгоритм підбору стратегій, що гарантують певний процент позитивних результатів для гравця. Цей відсоток визначає рівень складності гри. Саме проблемі математичного моделювання алгоритму вибору стратегій комп'ютерної програми у відповідності до рівня складності гри присвячена дана робота. Вибір ботом зброї залежить від так званих *параметрів керування*. Розв'язання задачі визначення параметрів вибору базується на *математичній теорії ігор*.

Засоби ШІ повинні прийняти до розгляду зазначені факти і виробити рішення, що стосується того, наскільки підходить той чи інший вид зброї в поточній ситуації. Результатом застосування функції придатності стає одичиничне значення, яке представляє рівень придатності зброї. В даному випадку до складу характеристик входять компоновка середовища, відстані між учасниками гри, поточний рівень життєздатності та інші. Критерії прийняття рішення про те, який вид зброї повинен використовуватися, фактично представляють собою результати аналізу комбінацій цих характеристик [6].

В основі системи голосування лежить така ідея, що характеристики використовуються для вироблення рішення про те, яка властивість зброї в найбільшій мірі потрібна для їх реалізації, і сформована при цьому оцінка розглядається як голос на користь цієї властивості (рис. 1). На відміну від тієї системи голосування, яка застосовується в людському суспільстві, в даній системі голосування можна простіше передбачити можливість подавати більше ніж один голос, і навіть є можливість використовувати дробові значення кількості голосів.

Після цього, як показано нижче, отримані голоси застосовуються для масштабування властивостей зброї. Якщо за якусь властивість подано багато голосів, то це означає, що властивість є важливою, тому її числове значення множиться на більше число; і навпаки, отримання меншої кількості голосів свідчить про те, що властивість не є надто важливою. Маємо формулу

$$f(w) = P_i(w) * V_i + P_n(w) * V_n. \quad (1)$$

В наведеній формулі  $f(w)$  позначає придатність зброї  $w$ ,  $P_i(w)$  позначає  $i$ -у властивість зброї, що розглядається (наприклад, швидкість та збиток від пострілу), а  $V_i$  показує загальну кількість голосів, поданих за цю властивість.

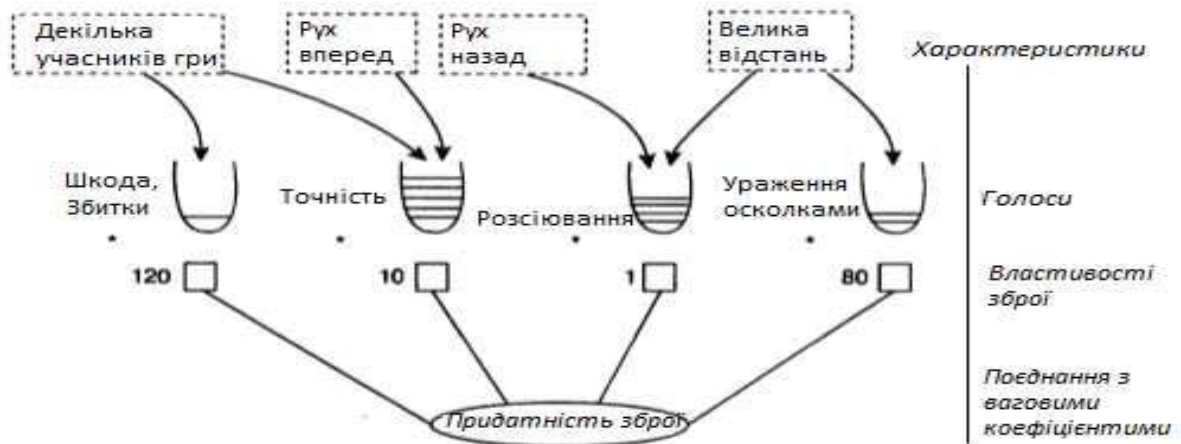


Рис. 1 – Схема організації системи голосування, що застосовується, в якій свій вклад в визначення кількості голосів вносять характеристики.

Значення кількості голосів множаться на значення властивості зброї і обраховується зважена сума, що визначає придатність зброї. Перш за все необхідно залучити до роботи експерта, щоб він визначив важливість властивостей в різних ситуаціях. Перш ніж почнеться голосування, необхідно виявити можливих кандидатів. У цьому розділі використовуються всі вказані на рис. 1 властивості зброї та до них додаються дві зазначені нижче властивості.

- **Точність стрільби.** Визначається як величина, зворотна розсіюванню.
- **Потенціал ураження.** Визначається як максимальна шкода з розрахунку на постріл, але використовується інше визначення вагового коефіцієнта, для того щоб цей фактор став менш значущим.

Таблиця 1 – Уточнення значимості характеристик зброї з допомогою голосування

Характеристика	Кількість голосів
Декілька гравців	Шкода, +4
Велика відстань	Точність, +2
Середня відстань	Кількість пошкоджень за секунду, +1; точність, +1
Коротка відстань	Кількість пошкоджень за секунду, +2
Рух уперед	Точність, +1
Рух назад	Розсіювання, +1
Рух в обмеженому просторі	Ураження осколками, +3
Напад із засади	Точність, +2
Пошук	Шкода, +1
Втеча	Потенційна ймовірність поразки, +аммо (кількість боєприпасів)

На наступному етапі необхідно вибрати характеристики ситуації, що враховуються при проведенні голосування. Рішення про використання тих чи інших характеристик засноване, головним чином, на інтуїції, а їх відносна значимість повинна уточнюватися на етапі проведення експериментів, як показано в табл. 1; деякі характеристики володіють голосами, величина яких перевищує одиницю, а інші характеристики віддають голоси на користь інших характеристик. Дані останнього рядка таблиці застосовуються незалежно від того, яку величину мають всі інші характеристики, а сам голос визначається пропорційно кількості боєприпасів.

Після збору всіх значень голосів отримані дані множаться на значення кожної властивості зброї. Сума зважених значень властивостей являє собою кінцеву величину придатності зброї. Зброя з найбільшим значенням придатності вибирається як найкраща.

Привабливою особливістю дерев рішень є те, що вони можуть знайти дуже широку сферу застосування. З іншого боку, існує велика кількість способів, що дозволяють забезпечити вибір зброї. Кількість можливих варіантів спільного застосування дерев рішень і таких способів ще більше. Проте, як показано на рис. 2, найбільш підходящими являються чотири варіанти, перераховані нижче.

1. Навчання вибору відповідної зброї.
2. Навчання оцінці придатності зброї.

3. Визначення за допомогою навчання властивостей зброї.
4. Навчання оцінці важливості властивостей.

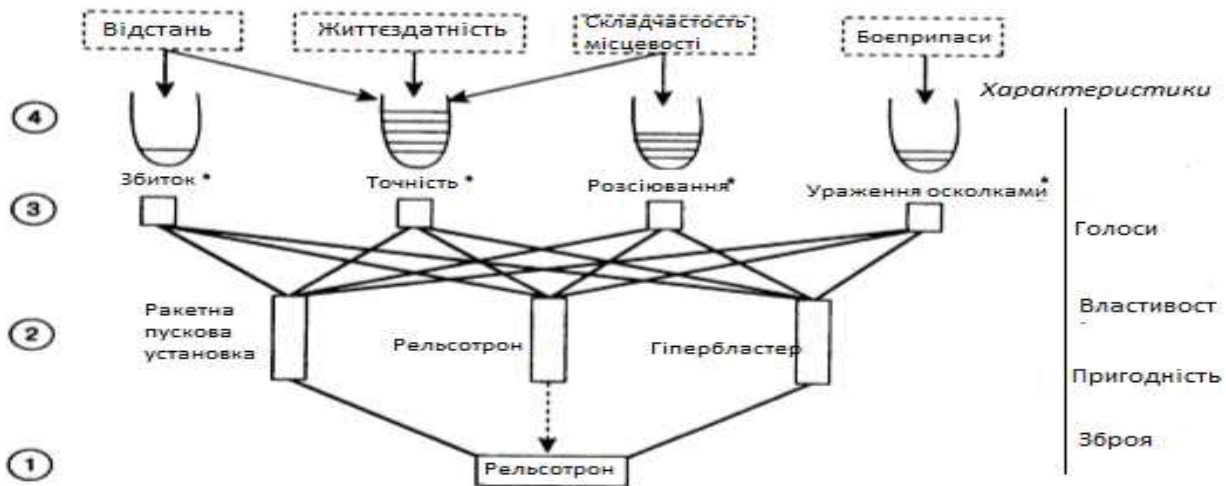


Рис. 2 – Концепції і процеси, пов'язані з прийняттям рішень по вибору зброї.

**Навчання вибору відповідної зброї** полягає в тому, що дерево рішень застосовується для отримання інформації про зброю, найкращу в кожній ситуації. Такий підхід рівносильний формуванню відображення з множини характеристик середовища на тип зброї. В цьому підході зосереджується все, що потрібно для вибору зброї, але використання єдиного компонента системи ШІ для рішення лише завдання визначення найкращої зброї призводить до виникнення на практиці багатьох проблем. Перелічимо деякі з них.

- Дерево рішень повертає тільки одну пропозицію на вибір зброї, тому не може застосовуватися в тих умовах, коли необхідна зброя недоступна.
- Дерево рішень повертає єдиний варіант і не дозволяє зрозуміти хід процесу, в якому цей варіант був сформований.
- Засоби ШІ повинні мати в своєму розпорядженні якийсь спосіб самостійно визначати найкращу зброю для контролю над навчанням дерева рішень.

Очевидно, що для рішення вказаних проблем можуть бути запропоновані три різних способи, що дозволяють більшою чи меншою мірою знайти вихід з положення. По-перше, можна продублювати дерева рішень, щоб можна було використовувати різні дерева в залежності від того, які види зброї є в наявності, але для реалізації цього методу потрібно більше пам'яті, а процес навчання сповільнюється. По-друге, можна вказати наявні види зброї в якості додаткових вхідних даних, але це призводить до *комбінаторного вибуху обсягу простору задачі*, тому спроба наближеного формування дерева рішень може викликати помилки. По-третє, можна зберігати результат вибору зброї, отриманий з допомогою дерева рішень, у вигляді списку видів зброї, ранжований за пріоритетами, але для цього потрібна адаптація дерева рішень, щоб його можна було використовувати в умовах наявності багатовимірних змінних відгуку [6]. Ці способи є застосовними, але не ідеальними. Більш прийнятний підхід може передбачати навчання оцінці придатності зброї в кожній конкретній ситуації.

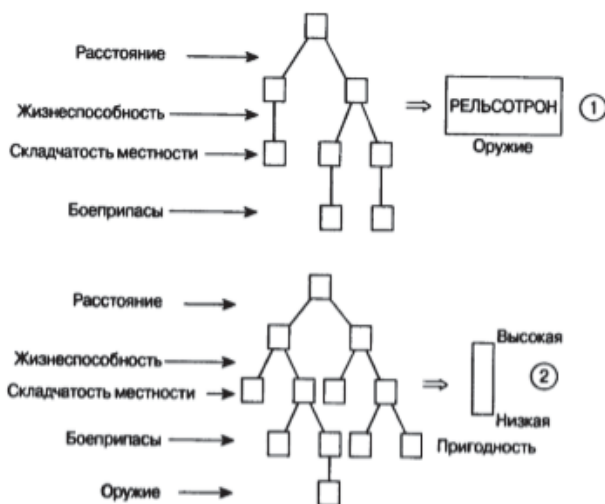


Рис. 3 – Варіанти використання дерев рішень.

**Навчання оцінці придатності зброї.** Для кожного виду зброї може бути передбачено окреме дерево рішень, що дозволяє оцінювати його придатність на підставі інформації про поточну ситуацію. У грі Quake 2 для цього потрібно дев'ять дерев рішень (оскільки зброя, що застосовується за замовчанням виключається з розгляду). При використанні даного підходу навички вибору зброї структуруються стосовно окремих видів зброї, тому навчання оцінці придатності зброї можна легко забезпечити за модульним принципом.

Проте, в зв'язку з необхідністю використовувати кілька дерев рішень замість одного, відбувається збільшення витрат, пов'язаних зі споживанням додаткової пам'яті і збільшенням обсягу коду. Замість цього можна було б використовувати позначення типу зброї в якості змінної прогнозування в дереві рішень (рис. 3), щоб мати можливість застосовувати

для оцінки придатності всіх видів зброї тільки одне велике дерево. Цей варіант більш зручний, оскільки необхідно лише одне дерево, для котрого може знадобитися менше ресурсів, ніж для окремих дерев. Найбільша проблема, з якою доводиться стикатися при використанні даного підходу, полягає в тому, що в деяких ситуаціях для контролю над навчанням доводиться обчислювати значення придатності. В деяких випадках доводиться вирішувати цю задачу окремо, дозволяючи засобам навчання дерева рішення формувати загальну поведінку.

Це можна зробити найпростіше шляхом повторного використання системи голосування. Для цього функціонування дерева рішень повинно бути засноване точно на таких же характеристиках, як і функціонування системи голосування. У такому випадку для навчання дерева рішень можуть використовуватися остаточні результати голосування. Дерево рішень, сформоване в результаті навчання, буде здатне ефективно представляти наближений результат голосування, тому не потрібно кожний раз проходити через весь процес голосування.

**Визначення за допомогою навчання властивостей зброї.** Даний підхід заснований на формуванні в результаті навчання дерева рішень, діючого на більш низькому рівні, що дозволяє виявляти різні властивості зброї виходячи з характеристик поточної ситуації. Такі властивості, як темп стрільби і скорострільність вражаючих елементів, не потребують навчання (принаймні не вимагають навчання за допомогою дерев рішень), оскільки являють собою постійні значення. З іншого боку, такі властивості, як максимальна шкода або оцінювана шкода, заподіяна в секунду, багато в чому залежать від поточної ситуації.

Перевага даного підходу полягає в тому, що його застосування приводить до отримання точних показників для властивостей зброї, на яких були засновані сценарні методи тактичних рішень. Дерева рішень дозволяють встановити те, наскільки отримані статистичні дані залежать від навичок бота, а також від різних ситуацій. З іншого боку, дерево рішень не представляє собою автономний метод, і при його використанні доводиться враховувати аспекти функціонування засобів ШІ. Якщо ми приймемо цей підхід, то отримаємо можливість ще раз скористатися всією системою голосування, яка, до того ж, буде вдосконалена в результаті навчання [7].

**Використання FALCON для вибору зброї.** На полі станів нейронної мережі State field було вирішено подавати значення наступних параметрів навколишнього середовища:

- distance – відстань між ботом і супротивником;
- velocity – модуль проекції вектора швидкості супротивника на площину, перпендикулярну лінії прицілу бота.

Всі дані величини нормовані, і лежать на відрізку  $[0; 1]$ . Разом з кожним параметром  $s_i$  подається зворотна величина  $1 - s_i$ . Це викликано особливостями формул, які використовуються в алгоритмі навчання мережі. Також, було визначено набір зброї, доступної для вибору ботом протягом гри:

1. автомат – зброя, ефективна на середній дистанції. Постріл виконується чергою з трьох патронів.

- шкоди (від одного патрона): 8;
- гранична дистанція 1500;
- затримка між пострілами: 0.5 с.

2. дробовик – зброя, вкрай ефективна на короткій дистанції. Постріл проводиться вісьмома дробинками, кожна з яких завдає ворогові невеликої шкоди:

- шкоди (від однієї дробинки): 5;
- гранична дистанція: 700;
- затримка між пострілами: 2.0 с.

3. гвинтівка – далекобійна зброя, що завдає великої шкоди одиночними пострілами. Ефективна на дальній дистанції:

- шкоди (від одного патрона): 15;
- гранична дистанція 1500;
- затримка між пострілами: 0.5 с.

Дії  $a_i$  відповідає використання зброї з індексом  $i$ . Одним з найбільш нетривіальних питань при використанні нейронної мережі, яка навчається з підкріпленням, є підбір функції нагороди. Очевидно, що нагорода, одержувана ботом після використання зброї, повинна залежати, в першу чергу, від шкоди, завданої ворогу. При цьому, бот повинен мати можливість отримати досить велику нагороду в будь-якій ігровій ситуації. Крім того, значення нагороди повинно належати відрізку  $[0; 1]$ , як будь-яке значення, яке подається на вхід FALCON. У підсумку, було прийнято рішення розраховувати значення нагороди наступним чином:

$$r = (1 + 9 * distance) * damage . \quad (2)$$

Тут  $distance$  – нормована відстань між ботом і його супротивником;  $damage$  – нормована величина шкоди, завданої противнику при пострілі.

Таким чином, при обчисленні нагороди величина шкоди множиться на лінійну функцію від відстані між ботом і його ворогом (конкретні значення параметрів лінійної функції були підібрані, виходячи з характеристик зброї, описаних вище). Це дозволяє вирівняти корисність від використання далекобійної зброї на дальній дисти-

нції, і зброї, призначеної для ближнього бою, на ближній. В іншому випадку, в процесі навчання можуть виникнути проблеми, так як зброї істотно розрізняються за абсолютними величинами нанесення шкоди.

**Результати тестування** нейронної мережі виводяться в спеціальний текстовий файл, який містить інформацію про те, скільки разів було вибрано кожную зброю в процесі гри, середні значення параметрів навколишнього середовища, при яких вибиралась зброя, кількість успішних застосувань зброї (застосування зброї вважається успішним, якщо була отримана позитивна нагорода), а також, середнє значення нагороди, отриманої при використанні зброї. Для наочності отримані результати наведені у вигляді таблиць.

В ході тестування проекту було проведено порівняльний аналіз роботи нейронної мережі, яка навчається за модифікованим алгоритмом, і роботи мережі, яка навчається за базовим алгоритмом (без модифікацій). Таблиці з результатами наведені нижче. В обох випадках нейронна мережа починала навчатися "з нуля"; тестування проводилося протягом 2 хвилин.

Таблиця 2 – Результати (базовий алгоритм):

Зброя	Число застосувань	Середня дистанція	Середня швидкість противника	Число успішних застосувань	Середня нагорода
Автомат	34	0.29	0.21	28	0.44
Дробовик	27	0.26	0.28	13	0.24
Гвинтівка	41	0.35	0.12	39	0.57

Таблиця 3 – Результати (модифікований алгоритм):

Зброя	Число застосувань	Середня дистанція	Середня швидкість противника	Число успішних застосувань	Середня нагорода
Автомат	27	0.28	0.17	22	0.45
Дробовик	25	0.18	0.24	18	0.36
Гвинтівка	48	0.39	0.21	44	0.6

Результати тестування показали, що в обох випадках нейронна мережа працює адекватно. Даний висновок можна зробити, виходячи з досить високої частки успішних застосувань зброї, а також, з середніх значень параметрів зовнішнього середовища: на ближній дистанції бот вважав за краще використовувати дробовик, на далекій – гвинтівку, на середній – автомат. У другій таблиці можна спостерігати більш чіткий поділ випадків застосування зброї в залежності від дистанції, і більш високі середні значення отриманої нагороди. Крім того, бот, тактика вибору зброї якого була заснована на нейронній мережі, навчений за базовим алгоритмом, за 2 хвилини не зміг навчитися правильно використовувати дробовик: з 27 випадків його застосування успішними виявилися лише 13; в разі застосування модифікованого алгоритму ця проблема не спостерігається: 25 випадків застосування дробовика, 18 з яких – успішні.

В ході роботи безпосередньо над алгоритмом вибору зброї було вирішено модифікувати ваги обраного нейрона тільки в разі отримання позитивної нагороди, тобто у разі попадання в супротивника з обраної зброї. Модифікування вагових коефіцієнтів нейрона в разі отримання нульової нагороди заважає стабільному навчанню нейронної мережі, оскільки ймовірність промаху є завжди, навіть при оптимальному використанні зброї. Своєчасні видалення нейронів стабільно приносять нульову нагороду та забезпечують бажані параметри.

Складність однієї ітерації використання FALCON становить  $O(n^2)$ , де  $n$  – кількість нейронів когнітивного поля мережі. Якщо в когнітивному полі накопичується занадто багато нейронів, мережа починає працювати повільно, що не дозволяє використовувати її в режимі реального часу. Для вирішення цієї проблеми був введений параметр, який відповідає за максимальну можливу кількість нейронів когнітивного поля. У разі, якщо число нейронів перевищує даний параметр, то нейрони когнітивного поля упорядковані відповідно до зменшення щодо середньої нагороди, отриманої під час їх активності, і половина нейронів (з меншим значенням середньої нагороди) видаляється. Це дозволяє зберігати компактність структури нейронної мережі на протязі її використання, не втрачаючи при цьому важливих знань, отриманих при взаємодії з навколишнім середовищем.

Зміни, внесені в алгоритм навчання нейронної мережі FALCON в ході роботи, мають перевагу, тому що істотно підвищують якість навчання мережі. Під час проведення дослідження було проаналізовано кілька підходів до вирішення поставленої задачі, а саме із застосуванням логічного висновку, досвіду та гібридного підходів, сценаріїв, дерев класифікації та регресії, алгоритму FALCON. Виявлені основні переваги та недоліки кожного із підходів. Алгоритм FALCON було модифіковано, що дозволило покращити результати його застосування, у порівнянні з базовим алгоритмом.

**Перспективи дослідження.** Проведені дослідження показали, що нейронні мережі, які навчаються з підкріпленням, можуть успішно застосовуватися при розробці ігрового штучного інтелекту. Основна перевага даного підходу перед традиційними методами реалізації моделей поведінки неігрових персонажів полягає в схожості поведінки створюваних таким чином ботів з персонажами, контрольованими реальними людьми: такі боти

здатні в процесі гри модифікувати тактику своєї поведінки, підлаштовуючись під зміни навколишнього середовища. Крім того, людина, що грає проти таких ботів, в силу недетермінованості моделі поведінки останніх, не може передбачати їх дії. Все це істотно підвищує реалістичність гри. Ще одна перевага даного підходу полягає в тому, що нейронна мережа може застосовуватися для підбору параметрів зброї, таких як пошкодження, що наноситься, точність, скорострільність та інших. Якщо в ході тестування гри бот занадто рідко (або занадто часто) вибирає якусь зброю, це означає, що набір зброї бота погано збалансований і параметри повинні бути модифіковані.

Варто додати, що боти, поведінка яких заснована на навчанні з підкріпленням, мають хороший потенціал для успішного проходження *тесту Тьюрінга для комп'ютерних ігор* – випробування, в ході якого судді в процесі гри визначають, ким контролюється той чи інший персонаж: людиною або комп'ютером.

**Висновки** Досліджено математичну модель алгоритму вибору ігрової зброї комп'ютерним ботом і реалізовано в програмі з використанням сценарного аналізу, дерев рішень, навченої нейронної мережі FALCON, а також їх застосування в алгоритмі вибору зброї ботом в тривимірному шутері від першої особи, тестування реалізованих алгоритмів і виявлення переваг і недоліків кожного підходу.

Проведені дослідження показали, що нейронні мережі, які навчаються з підкріпленням, можуть успішно застосовуватися при розробці ігрового штучного інтелекту. Основна перевага даного підходу перед традиційними методами реалізації моделей поведінки неігрових персонажів полягає в схожості поведінки створюваних таким чином ботів з персонажами, контрольованими реальними людьми: такі боти здатні в процесі гри модифікувати тактику своєї поведінки, підлаштовуючись під зміни навколишнього середовища. Крім того, людина, що грає проти таких ботів, в силу недетермінованості моделі поведінки останніх, не може передбачати їх дії. Все це істотно підвищує реалістичність гри. Ще одна перевага даного підходу полягає в тому, що нейронна мережа може застосовуватися для підбору параметрів зброї, таких як пошкодження, що наноситься, точність, скорострільність та інших. Якщо в ході тестування гри бот занадто рідко (або занадто часто) вибирає якусь зброю, це означає, що набір зброї бота погано збалансований і параметри повинні бути модифіковані.

#### Список літератури

1. Abu-Mostafa Y. S., Jaques J. N. St. Information capacity of the Hopfield model // *IEEE Trans. Inform. Theory.* – 1985. – V. 31. – P.461.
2. Ackley D. H., Hinton G. E., Sejnowski T. J. A learning algorithm for Boltzmann machines // *Cognit. Sci.* – 1985. – V. 9. – N. 1. – P. 147 – 169.
3. Amari S. Field theory of self-organizing neural networks // *IEEE Trans. Syst., Man, Cybern.* – 1983. – V. 13. – P. 741.
4. Takefuji D. Y. A new model of neural networks for error correction // *Proc. 9th Annu Conf. IEEE Eng. Med. and Biol. Soc., Boston, Mass., Nov. 13 – 16. – 1987. – V. 3. – New York, N. Y. – 1987. – P. 1709 – 1710.*
5. Тэнк Д. У., Хопфилд Д. Д. Коллективные вычисления в нейроноподобных электронных схемах // *В мире науки.* – 1988. – N 2. – С. 44 – 53.
6. Цуприков С. Нейронные вычисления берутся на вооружение // *Computerworld.* – Moscow, 1985. – N. 7. – С. 57 – 58.
7. Методы обучения нейросистем [WWW document]. – Режим доступа : [http://zdo.vstu.edu.ru/umk/html/manual/L5\\_6.html](http://zdo.vstu.edu.ru/umk/html/manual/L5_6.html). – Дата обращения : 18 февраля 2018.

#### References (transliterated)

1. Abu-Mostafa Y. S., Jaques J. N. St. Information capacity of the Hopfield model. *IEEE Trans. Inform. Theory.* 1985, vol. 31, p. 461.
2. Ackley D. H., Hinton G. E., Sejnowski T. J. A learning algorithm for Boltzmann machines. *Cognit. Sci.* 1985, vol. 9, no. 1, pp. 147–169.
3. Amari S. Field theory of self-organizing neural networks. *IEEE Trans. Syst., Man, Cybern.* 1983, vol. 13, p. 741.
4. Takefuji D. Y. A new model of neural networks for error correction. *Proc. 9th Annu Conf. IEEE Eng. Med. and Biol. Soc.* Boston, Mass., Nov. 13 – 16, 1987, vol. 3, New York, N. Y., 1987, pp. 1709–1710.
5. Tenk D. U., Hopfield D. D. Kollektivnyie vychisleniya v neyronopodobnykh elektronnykh skhemakh [Collective computations in neural-like electronic circuits]. *V mire nauki* [In the world of science]. 1988, no. 2, pp. 44–53.
6. Tsuprikov S. Neyronnye vychisleniya berutsya na vooruzhenie [Adopting neural computations]. *Computerworld.* Moscow, 1985, no. 7, pp. 57–58.
7. *Metody obucheniya neyrosistem* [WWW document] [Neural network learning methods]. Available at [http://zdo.vstu.edu.ru/umk/html/manual/L5\\_6.html](http://zdo.vstu.edu.ru/umk/html/manual/L5_6.html) (accessed 18 February 2018).

Надійшла (received) 21.02.2018

#### Відомості про авторів / Сведения об авторах / Information about authors

**Салтан Борис Андрійович (Салтан Борис Андреевич, Saltan Borys Andriyovych)** – магістр, Чорноморський національний університет ім. Петра Могили, м. Миколаїв; тел.: (068) 083-40-55; e-mail: [bs.arezius@gmail.com](mailto:bs.arezius@gmail.com).

**Собко Дмитро Анатолійович (Собко Дмитрий Анатольевич, Sobko Dmytro Anatoliyovych)** – магістр, Чорноморський національний університет ім. Петра Могили, м. Миколаїв; тел.: (063) 657-29-41; e-mail: [sobkod23@gmail.com](mailto:sobkod23@gmail.com).

**Кулаковська Інесса Василівна (Кулаковская Инесса Васильевна, Kulakovska Inessa Vasyilivna)** – канд. фіз.-мат. наук, завідувач кафедри інтелектуальних інформаційних систем, Чорноморський національний університет ім. Петра Могили, м. Миколаїв; тел.: (066) 646-48-47, (067) 698-93-57; e-mail: [kulaknic@gmail.com](mailto:kulaknic@gmail.com); (ORCID: 0000-0002-8432-1850).