

УДК 004.415

ОПТИМІЗАЦІЯ БОЙОВОЇ СИСТЕМИ В ПОКРОКОВОМУ РЕЖИМІ

Дешевих А.М., Новіков Ю.С.

e-mail: artem.deshevykh@nure.ua, yuriy.novikov@nure.ua

Харківський національний університет радіоелектроніки, каф. ПІ

м. Харків, Україна

This work is devoted to the optimization of a turn-based combat system – a key component in modern strategy and role-playing games. The study provides a comparative analysis of traditional fixed algorithms and modern adaptive methods implemented on Unreal Engine 5 with the Gameplay Ability System plugin. The proposed modular approach not only enhances computational efficiency and interactivity but also offers a scalable framework for future game development. Furthermore, the integration of scalable modules enables seamless incorporation of new functionalities, paving the way for continuous advancements in dynamic game design.

Дана робота присвячена оптимізації бойової системи в покроковому режимі, що є однією з ключових складових сучасних стратегічних та рольових ігор. Основна увага приділяється аналізу існуючих моделей організації покрокових битв, визначенню основних недоліків їх реалізації з точки зору продуктивності, інтуїтивності управління та гнучкості інтеграції з іншими системами гри. Сучасні дослідження [1] демонструють, що застосування гібридних підходів та адаптивних алгоритмів ефективно оптимізує розподіл ресурсів, що актуально для покрокової бойової системи. В умовах стрімкого розвитку ігрових технологій важливо забезпечити збалансованість між складністю алгоритмів обчислення результатів бою та швидкістю реакції системи, що безпосередньо впливає на якість ігрового процесу.

Під час дослідження було проведено порівняльний аналіз традиційних підходів до організації покрокових битв, де акцент зроблено на фіксованих алгоритмах визначення результатів, та сучасних методів, що застосовують динамічне розподілення ресурсів і адаптивні алгоритми прийняття рішень. Особлива увага приділялася реалізації системи на базі Unreal Engine 5 [2] із використанням плагіну GAS (Gameplay Ability System) [3], що дозволяє оптимізувати обчислювальні процеси та забезпечити високий рівень інтерактивності. Завдяки модульній структурі, запропоноване рішення дозволяє гнучко масштабувати функціональність бойової системи, інтегруючи додаткові модулі для аналізу тактичних ситуацій, прогнозування ходів суперника та оптимізації розподілу ресурсів у режимі реального часу.

До впровадження оптимізації бойова система базувалася на статичних, фіксованих алгоритмах, що не враховували змінні умови ігрового процесу (рис. 1). У таких умовах час реакції системи часто перевищував 300 мс, а навантаження на апаратні ресурси було високим, що призводило до зниження продуктивності та зниження задоволеності

користувачів. Крім того, відсутність адаптивності значно обмежувала можливості масштабування системи, ускладнюючи інтеграцію нових модулів та розширення функціоналу.

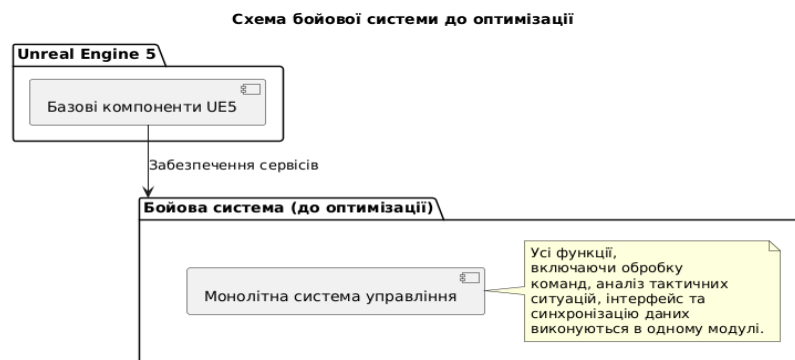


Рисунок 1 – Схема бойової системи до оптимізації

Аналіз показав, що оптимізація покрокової бойової системи передбачає вирішення наступних завдань:

- розробка адаптивного алгоритму розподілу команд, що враховує різні сценарії взаємодії гравця з системою;
- впровадження механізмів попереднього обчислення можливих ходів для зменшення затримок;
- забезпечення інтегрованого інтерфейсу, який дозволяє гравцю швидко орієнтуватися у складних тактичних ситуаціях;
- модульне розподілення функціональних блоків, що полегшує подальшу масштабованість та адаптацію системи під специфічні вимоги гри.

Після впровадження оптимізації, заснованої на модульній архітектурі та адаптивних алгоритмах розподілу ресурсів, відбулося суттєве покращення характеристик системи. Час реакції скоротився до приблизно 100 мс, що забезпечило більш оперативне виконання команд навіть у пікових режимах роботи. Динамічне розподілення ресурсів дозволило знизити навантаження на апаратні компоненти на 30–40%, що позитивно вплинуло на стабільність роботи системи під час інтенсивних битв. Нова архітектура спростила інтеграцію додаткових модулів, зокрема для аналізу тактичних ситуацій та прогнозування ходів суперника, що раніше було технічно складно реалізувати. Таким чином, оптимізація не лише покращила технічні показники, а й сприяла створенню більш інтуїтивного інтерфейсу, що відповідає сучасним вимогам ігрового дизайну та підвищує загальну ефективність управління ігровим процесом.

Запропоновано концепцію побудови схеми взаємодії основних компонентів системи, що представлено на рисунку 2. На ньому зображено основні блоки: модуль обробки команд, блок аналізу тактичних ситуацій, інтерфейс користувача та модуль синхронізації даних, які працюють у взаємодії з базовими компонентами ігрового рушія Unreal Engine 5. Запропоноване рішення дозволяє скоротити час реакції системи та знизити

навантаження на апаратні ресурси, що в цілому сприяє більш збалансованому та інтерактивному ігровому процесу.

Виходячи з аналізу проведених досліджень, можна зробити висновок, що інтеграція оптимізованої покрокової бойової системи у архітектуру гри сприяє підвищенню загальної продуктивності проекту та полегшує розробку складних ігрових сценаріїв. Отримані результати можуть бути використані розробниками як основа для впровадження інноваційних рішень у сфері ігрового дизайну, що дозволить зменшити поріг входу для нових користувачів та забезпечити більш інтуїтивний інтерфейс взаємодії. Крім того, впровадження даної системи створює передумови для постійного вдосконалення ігрових механік.



Рисунок 2 – Схема взаємодії основних блоків оптимізованої покрокової бойової системи

Таким чином, інтеграція оптимізованої покрокової бойової системи з використанням Unreal Engine 5 та плагіну GAS забезпечує значне зниження затримок, підвищення інтерактивності та оптимізацію управління ігровим процесом, що сприяє створенню сучасних, високоефективних ігрових проектів. Завдяки використанню модульної архітектури та адаптивних алгоритмів, система демонструє високу масштабованість і гнучкість, що дозволяє розробникам швидко інтегрувати нові функціональні можливості та вдосконалювати взаємодію з користувачами в умовах швидкого розвитку ігрових технологій.

Список використаних джерел:

1. Матвеев Д., Лановий О. Проблеми оптимізації графіки під пристрої віртуальної реальності // ЛОГОΣ.ONLINE. 2020. №14. С. 1.
2. Epic Games. (2022). Unreal Engine 5 Documentation : вебсайт. URL: <https://www.unrealengine.com/en-US/unreal-engine-5> (дата звернення: 25.02.2025).
3. GAS Plugin Documentation. (2021). Gameplay Ability System Overview : вебсайт. URL: <https://docs.unrealengine.com> (дата звернення: 25.02.2025).