

ДОСЛІДЖЕННЯ МЕТОДІВ ОПТИМІЗАЦІЇ ПЕРЕДАЧІ ДАНИХ У МУЛЬТИПЛЕЄРНИХ ІГРАХ

Яковенко Д.О., Шубін І.Ю.

Кафедра інфокомунікаційної інженерії ім. В.В. Поповського,
Харківський національний університет радіоелектроніки,
Україна

E-mail: dmytro.iakovenko1@nure.ua,
igor.shubin@nure.ua

Abstract

This paper investigates data transfer optimization in modern low-latency multiplayer online games, driven by demands for high concurrency and performance. The core technical challenge is minimizing network traffic while maintaining gameplay quality, addressing issues like latency, jitter, and packet loss inherent in the client-server model. The study aims to analyze contemporary optimization methods and develop an effective approach within the Unity engine. Common strategies reviewed include State/Delta Synchronization, Interest Management, data compression (e.g., bit-packing), and client-side prediction. Experiments conducted using Unity (Netcode for GameObjects) validated the effectiveness of these methods. Results confirm that a combined model integrating Interest Management, delta-compression, prediction, and bit-packing significantly reduces traffic by 59%–78% compared to full synchronization. These findings offer practical solutions for improving the performance and scalability of real-time networked applications, particularly on resource-limited platforms.

У сучасній ігровій індустрії мультиплеєрні ігри утворюють один із найдинамічніших сегментів ринку програмних продуктів. Масштабована мережна взаємодія, підтримка великої кількості одночасних користувачів та забезпечення низької затримки – ключові вимоги до сучасних систем онлайн-геймінгу. Однією з найважливіших технічних проблем у таких системах є оптимізація передачі даних між клієнтами та сервером. Ефективність мережних алгоритмів безпосередньо впливає на якість ігрового процесу, реалістичність, конкурентність, а також на ресурсні витрати інфраструктури.

Актуальність теми обумовлена різким збільшенням кількості гравців у реальному часі, розвитком кіберспортивних дисциплін, а також поширенням мобільних платформ з обмеженими пропускними та обчислювальними ресурсами. Оптимізація мережної взаємодії дозволяє мінімізувати затримки, підвищити точність синхронізації ігрових об'єктів та покращити роботу ігор у нестабільних мережних умовах.

Метою роботи є аналіз сучасних методів оптимізації передачі даних у мультиплеєрних іграх та розробка ефективного підходу, який забезпечує зменшення обсягу мережних повідомлень і покращення продуктивності ігрового додатку на рушії Unity.

У сучасних мультиплеєрних іграх застосовуються різні схеми синхронізації: клієнт-серверна модель, peer-to-peer та гібридні рішення. Найпоширенішою є клієнт-серверна архітектура, у якій основна логіка обробки станів об'єктів виконується на сервері, а клієнти надсилають лише інформацію про дії користувача або власні локальні зміни. Такий підхід забезпечує більшу безпеку та передбачуваність, однак створює низку технічних викликів: висока латентність, джитер (коливання затримки), втрата пакетів, характерна для мобільних мереж та Wi-Fi, а також надмірна кількість мережного трафіку, який суттєво обмежує масштабованість системи. Особливо критичними ці проблеми стають у fast-paced іграх, таких як шутери або спортивні симулятори, де навіть кілька мілісекунд затримки можуть вплинути на результат.

У технічній літературі, документації Unity, аналітичних оглядах та дослідженнях (зокрема роботах McShaffry, Lengyel, матеріалах щодо ECS та OpenArchive) розглядаються основні принципи

побудови оптимізованих мережних систем. Серед відомих підходів виокремлюють стиснення даних, предиктивну інтерполяцію, корекцію руху, пакетну передачу станів, delta-compression, interest management та інші методи. Однак питання комплексного поєднання цих технік у межах єдиного проєкту, а також адаптації їх під різні мережні умови і специфіку ігрової механіки часто залишається недостатньо дослідженим. Саме ця проблема й визначає науково-практичний напрям даної роботи.

Найпоширеніші стратегії синхронізації:

State Synchronization – передавання повного стану об'єкта; підходить для невеликої кількості об'єктів, але генерує великий трафік.

Event Synchronization – передавання лише подій; підходить для fast-paced ігор.

Delta Synchronization – надсилаються лише зміни станів; значно зменшує обсяг даних.

Методи оптимізації передачі даних:

Interest Management – оптимізація, що дозволяє надсилати гравцю тільки важливі для нього дані (видимість, відстань, зони активності). Перевагами є зменшення кількості пакетів та підвищення масштабованості серверу.

Стиснення і структурована передача даних методами біт-пакування (bit-packing), використання фіксованої точності чисел, delta-компресія координат, агрегація кількох апдейтів в один пакет, що може зменшити трафік у 2-5 разів.

Інтерполяція та екстраполяція руху на стороні клієнта, прогнозуючи рух об'єкта за попередніми даними, що дозволяє зменшити частоту надсилання пакетів та приховати джитер та лаги користува-чів.

Для аналізу ефективності методів оптимізації було створено тестовий мультиплеєрний ігровий додаток на рушії Unity з використанням C#, Netcode for GameObjects та Unity Transport, було реалі-зовано декількох моделей передачі даних, та проведення вимірювання, результати яких представлено у табл. 1, табл. 2 та табл. 3.

Таблиця 1. Обсяг мережного трафіку при різних методах.

Метод	К-сть об'єктів: 100	К-сть об'єктів: 200	К-сть об'єктів: 300	Середній трафік (кБ/с)	Зниження трафіку (%)
Повна синхронізація	145	287	426	286	0%
Delta-синхронізація	58	115	173	115	60%
Bit-packing	42	83	128	84	71%
Інтерполяція + предикція	51	101	146	99	65%
Interest Management	41	67	93	67	77%
Комбінована модель	29	47	66	47	84%

Таблиця 2. Затримка оновлення стану (latency) при різних мережних умовах.

Умови	Full Sync	Delta	Bit-pack	Interpolation	Interest Mgmt
0% loss, jitter < 5мс	62 ms	48 ms	45 ms	41 ms	39 ms
5% loss, jitter ~ 30мс	118 ms	92 ms	87 ms	74 ms	68 ms
12% loss, jitter 80–120мс	194 ms	151 ms	138 ms	121 ms	109 ms

Таблиця 3. Розмір одного мережного пакета при різних методах.

Передані дані (приклад од-ного об'єкта)	Без оптимізації	Delta	Bit-pack- ing	Інтерполяці я	Combined
Позиція (x,y,z)	36 B	14 B	8 B	14 B	6 B
Ротація (quaternion)	48 B	21 B	12 B	21 B	9 B
Лінійна швидкість	16 B	8 B	4 B	8 B	3 B
Стан анімації	24 B	12 B	6 B	12 B	5 B
Службові поля	12 B	9 B	6 B	9 B	4 B
Загалом	136 B	64 B	36 B	64 B	27 B

Важливо відзначити, що використання окремих методів оптимізації, навіть достатньо ефективних у певних сценаріях (наприклад, лише delta-compression або виключно агресивна компресія координат), не дозволяє досягти потрібного результату в умовах сучасних мультиплеєрних систем. Кожен із цих методів вирішує лише частину загальної проблеми: delta-compression зменшує обсяг переданих даних, але не усуває проблему надлишкової інформації, якщо клієнту все одно надсилаються стани непотрібних об'єктів; інтерполяція покращує плавність руху, але не впливає на пропускну здатність; interest-management зменшує кількість повідомлень, проте не гарантує мінімальний розмір кожного переданого пакета. Тому саме інтеграція кількох методів — їх взаємне підсилення — дає змогу сформувати збалансовану й високоефективну систему. Поєднання interest-management з delta-compression та інтерполяцією руху перетворюється на комплексну модель, яка оптимізує і сам обсяг даних, і частоту їх відправлення, і якість відображення на клієнті.

У дослідженні проаналізовано й експериментально підтверджено ефективність сучасних методів оптимізації передачі даних у мультиплеєрних іграх. Застосування комбінованої моделі, що включає interest-management, delta-compression, прогнозування руху та біт-пакування, дозволило скоротити трафік на 60–84%, зменшити затримки оновлення стану, підвищити стабільність ігрового процесу в умовах нестабільних мереж, збільшити масштабованість серверної частини.

Отримані результати мають практичне значення для розробників мережних ігор як на Unity, так і на інших рушіях, які використовують схожі архітектурні принципи. Особливо актуальними вони є для мобільних платформ, де обмежені пропускну здатність, енергоспоживання та обчислювальна потужність. Комбінований підхід до оптимізації дає змогу забезпечити стабільний ігровий досвід навіть за несприятливих мережних умов, що є необхідною вимогою сучасних проєктів.

Література

1. Cascading impact of lag on Quality of Experience in cooperative multiplayer games / E. Howard et al. 2014 13th annual workshop on network and systems support for games (netgames), Nagoya, Japan, 4–5 December 2014. 2014. DOI: <https://doi.org/10.1109/netgames.2014.7008965> (date of access: 10.11.2025).
2. Bartak R. Data transfer optimization - going beyond heuristics. 13th international workshop on advanced computing and analysis techniques in physics research, Jaipur, India, 22 February 2010. Trieste, Italy, 2011. DOI: <https://doi.org/10.22323/1.093.0009> (date of access: 10.11.2025).
3. A survey and analysis of techniques for player behavior prediction in massively multiplayer online role-playing games / B. Harrison et al. IEEE transactions on emerging topics in computing. 2015. Vol. 3, no. 2. P. 260–274. DOI: <https://doi.org/10.1109/tetc.2014.2360463> (date of access: 10.11.2025).
4. An active self-optimizing multiplayer gaming architecture / V. Ramakrishna et al. Cluster computing. 2006. Vol. 9, no. 2. P. 201–215. DOI: <https://doi.org/10.1007/s10586-006-7564-2> (date of access: 10.11.2025).
5. Game-o-Meta: trusted federated learning scheme for P2P gaming metaverse beyond 5G networks / P. Bhattacharya et al. Sensors. 2023. Vol. 23, no. 9. P. 4201. DOI: <https://doi.org/10.3390/s23094201> (date of access: 10.11.2025).
6. Liu C., Li K., Li K. A game approach to multi-servers load balancing with load-dependent server availability consideration. IEEE transactions on cloud computing. 2018. P. 1. DOI: <https://doi.org/10.1109/tcc.2018.2790404> (date of access: 10.11.2025).
7. Melo-López C. A., Salcedo-Parra O. J., Gallego-Torres R. A. Networks and their traffic in multiplayer games. Revista científica. 2016. Vol. 1, no. 24. P. 100. DOI: <https://doi.org/10.14483/10.14483/udistrital.jour.rc.2016.24.a10> (date of access: 10.11.2025).
8. Carlini E., Ricci L., Coppola M. Integrating centralized and peer-to-peer architectures to support interest management in massively multiplayer on-line games. Concurrency and computation: practice and experience. 2014. Vol. 27, no. 13. P. 3362–3382. DOI: <https://doi.org/10.1002/cpe.3289> (date of access: 10.11.2025).
9. Rao N. S. V., Ma C. Y. T., He F. Game strategies for data transfer infrastructures against ml-profile exploits. IEEE transactions on machine learning in communications and networking. 2024. P. 1. DOI: <https://doi.org/10.1109/tmlcn.2024.3417889> (date of access: 10.11.2025).

10.Prodan R., Nae V. Prediction-based real-time resource provisioning for massively multiplayer online games. Future generation computer systems. 2009. Vol. 25, no. 7. P. 785–793. **DOI:** <https://doi.org/10.1016/j.future.2008.11.002> (date of access: 10.11.2025).