

3차원 GIS 시뮬레이션의 시각적 연속성 향상을 위한 분산 데이터 처리 구조 연구

조 장 원¹ · 배 중 환² · 유 현 배³ · 박 선 회^{4*}

¹(주)유토비즈 대리

²(주)유토비즈 대표이사

³나사렛대학교 오웬스교양대학 교수

⁴(주)유토비즈 연구소장

A Study on Distributed Data Processing Structures for Enhancing Visual Continuity in 3D GIS Simulations

Jang-Won Jo¹ · Jong-Hwan Bae² · Hyun-Bae You³ · Seon-Hui Bak^{4*}

¹Assistant Manager, Utobiz Co., Ltd., Daejeon 34013, Korea

²Chief Executive Officer, Utobiz Co., Ltd., Daejeon 34013, Korea

³Professor, Department of VR Media Contents, Nazarene University, Daejeon 34503, Korea

⁴Director of Research Institute, Utobiz Co., Ltd., Daejeon 34013, Korea

[요 약]

3차원 GIS 시뮬레이션 환경에서는 동적 객체의 상태 데이터를 안정적으로 처리하고 가시화하는 과정이 중요하다. 그러나 단일 처리 구조에서는 객체 수가 증가함에 따라 연산 부하와 네트워크 병목 현상이 발생하게 된다. 이는 객체 움직임의 연속성을 저해하여 시각적 표현의 자연성을 떨어뜨리는 요인이 된다. 본 연구에서는 객체 상태 데이터의 처리 부하를 분산하는 구조를 설계하였다. 제안된 구조는 객체 움직임 연산을 중앙 서버에서 분리하여 분산 서버에서 병렬 처리하도록 구성하였다. 실험 결과, 중앙 집중형 구조의 평균 FPS는 18 FPS까지 감소한 반면, 제안 구조는 57 FPS를 유지하였다. 또한 평균 객체 상태 반영 지연은 55ms에서 17ms로 감소하였으며, 프레임 안정성은 58%에서 91%로 향상되었다.

[Abstract]

In 3D GIS simulation environments, it is critical to reliably process and visualize the state data of dynamic objects. However, in a single processing structure, increase in the number of objects introduces computational load and network bottlenecks, hindering the continuity of object movement and degrading the naturalness of visual representations. We designed a structure that distributes the processing load of object state data. The proposed structure is configured to object movement calculations from the central server and process them in parallel on distributed servers. Results showed that while the average FPS of the centralized structure decreased to 18 FPS, the proposed structure maintained 57 FPS. Furthermore, the average object state reflection latency decreased from 55ms to 17ms, and frame stability increased from 58% to 91%.

색인어 : 3차원 GIS 시뮬레이션, 분산 데이터 처리, 동적 객체, 시각적 연속성, 병렬 처리 구조

Keyword : 3D GIS, Distributed Data Processing, Dynamic Objects, Visual Continuity, Parallel Processing Architecture

<http://dx.doi.org/10.9728/dcs.2026.27.7.1965>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 04 May 2026; Revised 29 May 2026

Accepted 06 July 2026

*Corresponding Author; Seon-Hui Bak

Tel:

E-mail: sadal@hanmail.net

I. 서 론

지리정보시스템(Geographic Information System, GIS)은 인간 생활에 필요한 지리정보 등을 컴퓨터 데이터로 변환하여 가상공간에 데이터를 구축하여 보다 효율적으로 활용하기 위한 정보시스템이다[1]. 최근 GIS는 정보 통신 기술(Information and Communication Technology, ICT)의 발전으로 건물 정보 모델링(Building Information Modeling, BIM) 기술과 융합하여 핵심 기술로 평가되고 있다[2]. 특히 3차원 GIS의 경우 재난 대응, 도시 및 시설물 관리 등 실제 문제 해결을 위한 도구로 적극 활용되고 있다. 그러나 이러한 활용이 확대됨에 따라 시스템 성능 문제 또한 함께 제기되고 있다. 선행연구로 진행되었던, 재난정보 3차원 GIS 서비스를 위한 시각화 플랫폼의 경우에도 계산적 부하에 따라 처리 지연이 발생할 수 있음이 보고된 바 있다[3]. 실제로 3차원 GIS 시뮬레이션 환경에서는 다수의 동적 객체가 동시에 존재하며, 이들의 상태 데이터를 지속적으로 갱신하고 시각화해야 하는데 이 과정에서 처리가 늦어지면 객체의 움직임이 부자연스럽게 표현되는 문제가 발생한다. 기존의 중앙 집중형 처리 구조는 구현과 관리가 비교적 용이하다는 장점이 있으나, 객체 수가 증가할수록 연산 부하가 특정 서버에 집중되는 한계를 가진다. 특히 다수의 이벤트가 동시에 발생하는 상황에서는 네트워크 병목 현상까지 겹치며 전체 시스템의 응답성이 급격히 저하되는 경향을 보인다. 이러한 문제는 단순한 성능 저하를 넘어 사용자 경험에도 직접적인 영향을 미친다. 이러한 문제를 해결하기 위해 선행 연구에서는 스마트시티 환경에서 대규모 데이터를 처리하기 위한 분산 병렬 처리 구조가 제안된 바 있다[4]. 그러나 이러한 연구는 주로 빅데이터 처리 효율성에 초점을 두고 있으며, 3차원 GIS 시뮬레이션 환경에서 다수의 동적 객체 상태 데이터를 연속적으로 가시화하기 위한 구조적 접근은 상대적으로 부족하다. 이에 본 연구에서는 객체 상태 데이터 처리 구조를 개선하기 위해 중앙 집중형 처리 방식에서 벗어나 분산 서버 기반 병렬 처리 구조를 설계하였다. 제안된 구조는 객체의 움직임 연산을 분리하여 병렬 처리함으로써 처리 안정성을 향상시키고, 시각적 연속성을 개선하는 것을 목표로 한다. 제안 구조의 성능을 검증하기 위해 본 연구에서는 사람 객체를 실험 대상으로 선정하였다. 사람 객체는 머리, 몸통, 팔, 다리 등 다수의 신체 파트로 구성되며, 각 파트의 상태 정보가 지속적으로 갱신되는 특성을 가진다. 이러한 특성은 다수의 동적 객체가 동시에 이동하는 GIS 기반 시뮬레이션 환경의 특성을 효과적으로 반영할 수 있으며, 스마트시티, 재난 대응 훈련, 군중 시뮬레이션과 같은 응용 환경을 모사하는 데 적합하다. 따라서 본 연구에서는 사람 객체를 활용하여 객체 상태 데이터 처리 과정에서 발생하는 연산 부하를 분석하고, 제안된 분산 서버 기반 병렬 처리 구조의 성능 및 시각적 연속성 향상 효과를 검증하였다.

II. 3차원 GIS 시뮬레이션에서 동적 객체 처리 방식

3차원 GIS 시뮬레이션 환경은 지형 및 시설물 등의 공간 정보를 3차원 지리정보 데이터로 구축하고, 가상현실(Virtual Reality, VR)기반 시각화 기술과 연계하여 표현된다[5]. 또한 정적인 공간정보를 표현하는 것을 넘어 다수의 객체 위치 및 상태정보가 동적으로 변화하는 특성이 있다. 이러한 환경에서는 객체의 이동, 상태 변화, 이벤트 발생 등의 갱신된 정보가 시각화 시스템에 연속적으로 반영되어 자연스럽게 표현되어야 한다. 특히 객체 수가 증가할수록 상태 정보 처리에 필요한 연산 복잡도가 증가하게 되며, 중앙 집중형 처리 방식에서는 시스템 병목현상이 발생하게 된다. 선행 연구인 도시 교통 디지털 트윈 시스템 연구에서도 다수의 차량 객체를 동시에 처리하는 과정에서 계산 부하 문제가 주요 성능 저하 요인으로 보고된 바 있다[6]. 이와 같은 문제를 해결하기 위해서는 동적 객체의 상태 정보를 효율적으로 처리함과 동시에 시각적 연속성을 유지할 수 있어야 한다. 따라서 동적 객체 상태 데이터가 생성되고 처리되어 시각화 시스템에 반영되는 전반적인 처리 과정에 대한 구조적 설계가 필요하다.

2-1 객체 상태 갱신 방식

동적인 객체의 상태 정보는 위치, 속도, 방향, 속성 변화 등 다양한 요소로 구성되고, 상태 정보는 시뮬레이션의 시간 흐름에 따라 지속적으로 갱신되고, 갱신된 데이터는 시각화 시스템에 전달되어 객체의 움직임과 상태 변화를 반영한다. 또한 네트워크를 통해 다른 사용자와 공유되어 동일한 상태를 유지하도록 동기화될 수 있다[7]. 객체 상태 갱신은 일반적으로 일정한 시간 간격으로 수행되는 주기적 갱신 방식과 객체 상태 변화가 발생할 때만 정보를 갱신하는 이벤트 기반 갱신 방식으로 구분된다. 주기적 갱신 방식은 처리 구조가 단순하고 동기화에 유리하지만, 객체 수가 증가할 경우 연산 부하가 커지게 된다. 이벤트 기반 갱신 방식은 실제 변화가 발생한 경우에만 처리하여 연산 효율성을 높일 수 있으나, 이벤트가 집중될 경우 순간적인 부하가 증가하게 된다. 이러한 방식은 이산사건 시뮬레이션(Discrete-Event Simulation) 이론과 밀접한 관련이 있다[8]. 대규모 동적 객체를 포함하는 3차원 GIS 시뮬레이션 환경에서는 이러한 갱신 과정에서 발생하는 처리 부하가 시스템 성능에 직접적인 영향을 미치므로, 갱신 방식에 따른 처리 구조의 특성 분석이 필요하다.

2-2 주기적 갱신 기반 처리 구조

주기적 갱신 기반 처리 구조는 일정한 시간 간격마다 모든 객체의 상태 정보를 갱신하는 방식이다. 모든 객체가 동일한 갱신 주기에 따라 상태 정보를 갱신하므로 시뮬레이션 시간 관리와 상태 동기화가 용이하다. 그러나 객체 수가 증가할수록 동일 시간 내 처리해야 할 데이터량이 급격히 증가하여 연

산 부하가 커진다[6]. 특히 중앙 서버 기반 구조에서는 모든 객체 상태 데이터를 단일 서버가 처리하므로 부하 집중 현상이 발생하기 쉽다. 더불어 갱신 주기가 짧을수록 객체 움직임의 연속적인 표현이 가능하지만, 처리 부담이 증가하게 된다. 반대로 갱신 주기가 길어지면 처리 부하는 감소하지만, 객체 움직임이 불연속적으로 표현된다.

2-3 이벤트 기반 처리 구조

이벤트 기반 처리 구조는 객체의 상태 변화가 발생한 경우에만 정보를 갱신하는 방식이다. 객체 이동, 방향 변화, 충돌 등 특정 이벤트 발생 시에만 연산이 수행되므로 불필요한 데이터 처리 과정을 줄일 수 있다. 주기적 갱신 방식이 객체 수와 관계없이 모든 객체를 동일한 주기로 처리하는 구조라면, 이벤트 기반 처리 방식은 실제 변화가 발생한 객체에 한정하여 처리 자원을 할당한다는 점에서 차이를 지닌다. 이러한 처리 방식은 시스템 상태 변화가 특정 사건 발생 시점에서만 이루어진다고 보는 이산사건 시뮬레이션(Discrete-Event Simulation) 이론과 밀접한 관련이 있다. 이산사건 시뮬레이션은 시간의 연속적 흐름보다 사건(Event)이 발생하는 불연속적 시점을 중심으로 시스템을 모델링하는 기법으로, 사건 유형과 그 상호 관계를 기반으로 시스템의 동적 행태를 분석하는 데 효과적이다. 이러한 특성으로 인해 복잡한 프로세스 분석이 필요한 다양한 분야에서 널리 활용되고 있다[7]. 즉, 이벤트 발생 시점에만 상태 정보를 갱신하는 방식은 불필요한 연산을 최소화하고 처리 효율성을 향상시킬 수 있다는 장점을 가진다. 그러나 다수의 객체에서 이벤트가 동시에 발생하면 처리 요청이 특정 시점에 집중되어 순간적인 연산 부하가 증가할 수 있다. 특히 충돌, 경로 변경 등 복합 이벤트 발생 시 처리량이 급증하여 시스템 안정성이 저하된다. 또한 이벤트 감지를 위한 추가 로직이 필요하므로 시스템 설계 복잡성이 증가하는 단점이 존재한다.

2-4 기존 구조의 한계 분석

본 연구의 대상인 3차원 GIS 시뮬레이션 시스템은 객체 상태 데이터의 수집, 연산 및 갱신 과정이 중앙 서버에 집중되는 단일 처리 구조로 운영 되어왔다. 이러한 구조는 시스템 구성이 단순하고 관리가 용이하다는 장점이 있으나, 동적 객체 수가 증가할 경우 중앙 서버에 연산 부하가 집중되는 구조적 한계를 가진다. 특히 모든 객체 상태 데이터를 단일 서버에서 처리하는 환경에서는 객체 수 증가에 비례하여 상태 갱신을 위한 데이터 처리량이 급격히 증가하게 된다. 이러한 처리 지연은 객체 상태 정보의 갱신 주기를 불안정하게 만들어 시각화 과정에서 객체 움직임이 불연속적으로 표현되는 원인이 된다. 또한, 모든 데이터가 중앙 서버를 경유하는 구조적 특성으로 인해 네트워크 트래픽이 특정 구간에 집중되는 병목 현상이 발생한다. 다수의 객체에서 상태 갱신 요청이 동시

에 발생할 경우 데이터 전송이 늦어지게 되며, 이는 시스템 전반의 응답성 저하로 이어질 수 있다. 이벤트 기반 처리 방식이 적용되더라도, 군집 이동이나 대규모 경로 변경과 같이 다수의 객체에서 이벤트가 동시 발생하는 상황에서는 처리 요청이 중앙 서버에 집중되어 순간적인 부하 급증 현상이 발생한다. 결과적으로 중앙 서버 기반 단일 처리 구조는 객체 수 증가 및 이벤트 집중 상황에 유연하게 대응하기 어렵고, 대규모 동적 객체를 포함하는 3차원 GIS 시뮬레이션 환경에서 안정적인 운영을 보장하는 데 구조적 한계를 가진다. 다음 표 1은 기존 구조와 제안 구조를 요약한 비교표이다. 기존 디지털 트윈 기반 분산 처리 연구는 주로 대규모 데이터의 저장 및 전송 효율 향상에 초점을 두고 있으며, ECS(Entity Component System) 기반 구조는 객체 구성요소 단위의 연산 최적화를 목적으로 한다. 그러나 이러한 연구들은 실시간 시각화 환경에서 다수의 동적 객체 상태를 처리하고, 객체 움직임의 시각적 연속성을 향상시키는 문제에 대해서는 상대적

표 1. 기존 중앙집중식 구조와 제안 분산 병렬 구조의 비교

Table 1. Comparison between conventional centralized and proposed distributed parallel architectures

Category	Conventional Centralized Architecture	Proposed Distributed Parallel Architecture
Processing Structure	A single central server processes all object state data	Main engine and distributed engines share processing tasks
Computation Method	Object state computations are concentrated on one server	Computation load is distributed across multiple engines per client
Scalability	Performance degrades as the number of objects and users increases	Horizontal scalability achieved by adding distributed engines
Load Distribution	Computational load is concentrated on the central server	Workloads are distributed to mitigate processing bottlenecks
Synchronization Method	Batch synchronization managed by a central server	Main engine manages global synchronization after parallel processing
Object Processing Unit	Object-level batch processing	Fine-grained processing at body-part level
Network Structure	All data traffic passes through the central server	Network traffic is distributed across multiple processing nodes
Performance	FPS drops significantly as object count increases	Stable FPS maintained even with increasing object count
Visual Continuity	Object motion may appear discontinuous	Improved visual continuity of object movements
System Stability	Performance degradation under burst event loads	Parallel processing mitigates sudden load spikes

으로 제한적으로 다루고 있다. 이에 본 연구에서는 객체 상태 데이터를 신체 파트 단위로 분리하여 분산 서버 환경에서 병렬 처리하고, 이를 실시간 시각화 시스템과 연계하는 구조를 제안한다. 이를 통해 객체 상태 데이터 처리 부하를 분산하고 상태 갱신 효율을 향상시켜, 대규모 동적 객체 환경에서의 시각적 연속성 향상을 목표로 한다.

III. 분산 데이터 처리 구조 설계

3-1 분산 처리 구조 설계 목표

구조 설계의 목표는 다수의 동적 객체로 인해 발생하는 중앙 서버의 처리 부하 집중 문제를 완화하고, 객체 상태 정보의 안정적인 갱신과 시각적 연속성을 확보할 수 있는 데이터 처리 구조를 구현하는 데 있다. 기존 단일 처리 구조는 객체 상태 데이터의 수집, 연산 및 갱신 과정이 중앙 서버에 집중되면서 객체 수가 증가할 경우 연산 부하와 네트워크 트래픽이 급격히 증가하는 한계를 보였다. 이를 해결하기 위해 객체 상태 데이터 처리 과정을 분리하여 연산 부하를 복수의 서버로 분산하여 병렬 처리할 수 있는 구조를 설계하고자 한다. 이를 통해 시각적 불연속성을 최소화하고, 안정적인 처리 성능을 유지하는 것을 목표로 한다.

3-2 데이터 처리 흐름

제안하는 시스템의 데이터 처리 흐름은 중앙 집중형 구조와 달리 메인 엔진과 분산 엔진 간 역할을 분리한 계층적 처리 구조로 구성된다. 클라이언트가 시스템에 접속하면 분산 엔진 내에 해당 클라이언트와 대응되는 Rx 인스턴스가 동적으로 생성되며, 객체 상태 데이터의 수집 및 연산 처리를 담당한다. 분산 엔진은 클라이언트로부터 수신된 객체 상태 정보를 다리, 몸통, 팔, 머리 파트 단위로 나누어 병렬 연산을 수행한다. 각 신체 파트는 독립적인 연산 단위로 처리되며, 파트별 위치 좌표와 회전 값을 개별적으로 계산한다. 연산 결과는 클라이언트에 전달하여 객체 움직임에 반영되며, 동시에 일정 주기로 메인 엔진에 전송된다. 메인 엔진은 분산 엔진으로부터 수집된 최신 객체 상태 정보를 기준으로 전체 객체 정보를 통합하고 동기화를 수행한다. 동기화가 완료된 데이터는 시각화 시스템으로 전송되어 최종적으로 객체 애니메이션에 반영된다. 이러한 처리 흐름을 통해 연산 부하를 분산시키고, 다수의 클라이언트가 접속하는 환경에서도 안정적인 연속성을 확보할 수 있다.

3-3 제안 구조 설명

본 연구에서는 중앙 서버에 집중되었던 처리 부하를 완화하기 위해 분산 서버 기반 병렬 처리 구조를 제안한다. 제안

구조는 객체 상태 데이터 처리 과정을 기능 단위로 분리하고, 각 기능을 서로 다른 서버에서 수행하도록 구성함으로써 중앙 집중형 구조의 한계를 개선하는 것을 목표로 한다. 서버는 두 가지 종류로 객체 연산 처리 서버와 상태 갱신 관리 서버로 구성된다. 객체 연산 처리 서버는 수신받은 상태 데이터를 기반으로 위치를 계산하고 상태 변화 처리와 같은 이벤트 연산을 수행하는 역할을 담당한다. 이를 통해 연산 집약적인 처리 과정을 전달하여 중앙 서버에 집중되던 연산 부하를 분산하도록 하였다. 상태 갱신 관리 서버는 객체 연산 처리 서버로부터 전달받은 연산 결과를 통합 관리하며, 갱신된 객체 상태 정보를 시각화 시스템에 전달하는 역할을 수행한다. 또한, 객체 상태 데이터의 동기화 및 갱신 주기 관리를 담당하여 시각화 과정에서 객체 상태 정보가 안정적으로 반영되도록 한다. 이와 같이 연산 처리 기능과 상태 관리 기능을 서버 간에 분리함으로써 객체 수 증가로 인한 연산 부하가 중앙 서버와 같이 집중되는 현상을 방지할 수 있도록 부하를 분산하였다. 이처럼 전체 시스템의 처리 효율성을 향상시키기 위해 병렬 처리 방식을 적용하였다.

IV. 구조 구현 및 성능 검증

4-1 구현 환경 및 방법

분산 처리 구조를 구현하기 위해 시각화 시스템과 데이터 처리 시스템을 분리하여 구성하였다. 시각화 시스템은 언리얼 엔진을 사용하여 구현하였다. 언리얼 엔진은 전반적인 개발 환경을 제공하는 통합 엔진으로 게임뿐만 아니라 영화, 시뮬레이션, 건축 등의 다양한 분야에서 활용되고 있으며, 최적화 면에서 뛰어난 성능을 보여주는 게임 엔진이다[9]. 또한 고해상도 3차원 공간 표현과 객체 애니메이션 처리가 가능하여 대규모 동적 객체를 포함하는 GIS 시뮬레이션 환경에 적합하다. 본 연구에서는 제안한 분산 처리 구조를 기반으로 메인 엔진과 분산 엔진을 구현하였으며, 플레이어와 객체의 상태

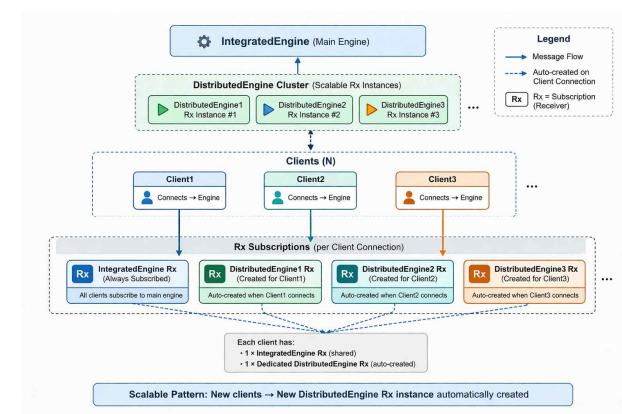


그림 1. 3차원 GIS 시뮬레이션 분산 처리 구조

Fig. 1. 3D GIS simulation distributed processing structure

정보를 다리, 몸통, 팔, 머리 파트로 구성하여 실험을 수행하였다. 분산 처리 구조는 그림 1과 같다.

4-2 시험 시나리오 구성

제안 구조의 성능 검증을 위해 플레이어 객체의 신체 파트 단위 연산이 동시에 발생하는 3차원 GIS 시뮬레이션 환경을 시험 시나리오로 구성하였다. 시험 환경은 실제 사용자 동작이 반영되는 상황을 가정하여 객체 세부 구성 요소에서 발생하는 상태 갱신 부하를 분석하는 것을 목표로 하였다. 본 실험에서는 실제 사용자 환경과 유사한 부하 상황을 재현하는 것을 목표로 하였다. 단순히 객체 수를 증가시키는 방식 대신, 플레이어 객체를 신체 파트 단위로 분할하여 보다 복잡한 연산이 동시에 발생하도록 구성하였다. 이러한 설정은 실제 환경에서 발생할 수 있는 연산 집중 현상을 보다 현실적으로 반영하기 위한 것이다. 또한, 다수의 사용자가 동시에 이동, 방향 전환, 자세 변화 등을 수행하는 상황을 가정하여 특정 시점에 상태 갱신 요청이 집중되도록 설계하였다. 이는 분산 처리 구조가 순간적인 부하 증가 상황에서도 안정적으로 동작하는지를 확인하기 위함이다.

4-3 성능 평가 기준

성능 평가는 시각화 시스템의 처리 성능을 정량적으로 비교하기 위해 프레임률(Frame Per Second, FPS)을 주요 평가 지표로 설정하였다. FPS는 초당 화면에 표시되는 프레임 수를 의미하며, 사용자 체감 성능을 직접적으로 나타내는 대표적인 지표이다. 3차원 GIS 시뮬레이션 환경에서는 다수의 객체 상태 갱신이 동시에 발생할 경우 데이터 처리에서 병목 현상이 발생할 수 있으므로, 처리 구조에 따른 평균 FPS 변화를 비교함으로써 제안 구조의 성능 개선 효과를 검증하고자 하였다. 실험 환경 사양은 다음 표 2와 같다. FPS 측정은 언리얼 엔진에서 제공하는 성능 측정 기능을 활용하여 수행하였으며, 동일한 시뮬레이션 조건에서 중앙 집중형 처리 구조와 분산 서버 기반 병렬 처리 구조의 평균 FPS 값을 10회 테스트하여 비교·분석하였다.

표 2. 실험환경 사양

Table 2. Experiment environment specifications

Item	Specification
CPU	Intel Core Ultra 7 265K (3.90 GHz)
GPU	NVIDIA GeForce RTX 5070 Ti (16 GB)
Network	Gigabit Ethernet (1 Gbps)
Number of Distributed Nodes	10
Object Counts	300
Operating System	Windows 11

4-4 실험 결과 분석

그림 2는 중앙 집중형 처리 구조와 제안된 분산 서버 기반 병렬 처리 구조의 FPS 변화를 비교한 실험 결과를 나타낸다. 중앙 집중형 처리 구조에서는 플레이어 객체 수가 증가함에 따라 FPS가 감소하는 현상을 확인했다. 특히 다수의 객체에서 신체 파트 단위 상태 갱신이 동시에 발생하는 구간에서 프레임 저하 현상이 두드러지게 나타났다. 실험 결과 중앙 집중형 구조의 평균 FPS는 최소 18 FPS까지 감소하였으며, 객체 상태 반영 지연은 평균 55ms로 측정되었다. 반면, 제안된 분산 서버 기반 병렬 처리 구조에서는 객체 수가 증가하더라도 FPS 감소 폭이 상대적으로 완만하게 나타났으며, 일정 수준 이상의 프레임률을 안정적으로 유지하였다. 실험 결과 평균 FPS는 57 FPS를 유지하였고, 객체 상태 반영 지연은 평균 17ms로 감소하였다. 또한 프레임 안정성(Frame Stability)은 기존 구조의 58%에서 91%로 향상되어 객체 움직임의 연속성이 더 안정적으로 유지되는 것을 확인하였다. 이는 객체 상태 연산이 분산 서버로 분담되면서 중앙 서버의 연산 부하가 완화되었기 때문으로 분석된다. 성능 비교표는 표 3과 같다.

표 3. 중앙 집중형 구조와 제안 구조의 성능 비교

Table 3. Performance comparison between conventional centralized architecture and proposed architecture

Performance Metric	Conventional Centralized Architecture	Proposed Distributed Parallel Architecture	Improvement
Average FPS	18 FPS	57 FPS	+216.7%
Object State Reflection Latency	55ms	17ms	-69.1%
Frame Stability	58%	91%	+33%p
Processing Structure	Centralized Processing	Distributed Parallel Processing	-
Visual Continuity	Frequent Motion Discontinuity	Stable Motion Visualization	Improved

V. 결 론

3차원 GIS 시뮬레이션 환경에서 다수의 동적 객체 상태 데이터를 효율적으로 처리하기 위한 분산 서버 기반 병렬 처리 구조를 제안하였다. 기존 시스템은 객체 상태 데이터의 수집, 연산 및 갱신 과정이 중앙 서버에 집중되는 단일 처리 구조로 운영되어 왔으며, 객체 수 증가 시 연산 부하 집중으로 인한 병목 현상이 발생하는 구조적 한계를 가지고 있었다. 이를 개선하기 위해 본 연구에서는 객체 상태 데이터 처리 과정을 기능 단위로 분리하고, 객체 움직임 연산을 분산 서버에서 병렬 처리하도록 구조를 설계하였다. 특히 플레이어 객체를 신체

파트 단위로 분할하여 위치 좌표 및 회전 값 연산을 분산 처리함으로써 연산 부하를 효과적으로 분담하도록 구성하였다. 성능 검증을 위해 언리얼 기반의 3차원 시뮬레이션 환경에서 다수의 플레이어 객체가 동시에 동작하는 시험 시나리오를 구성하였으며, 프레임률(FPS)을 성능 평가 지표로 활용하였다. 실험 결과, 제안 구조는 객체 수가 증가함에 따라 중앙 집중형 처리 구조 대비 프레임률 저하를 완화하고 객체 움직임의 시각적 연속성을 향상시키는 것으로 나타났다. 이를 통해 분산 병렬 처리 구조가 대규모 동적 객체 환경에서 효과적인 처리 방식임을 확인하였다. 제안된 분산 서버 기반 병렬 처리 구조는 객체 상태 데이터 처리 부하를 분산함으로써 중앙 서버의 연산 부담을 완화하고 시스템의 전반적인 처리 안정성을 향상시킬 수 있고, 객체 움직임이 끊어지는 현상을 감소시켜 시각적 표현의 자연성을 향상시킬 수 있다. 또한, 다수의 동적 객체가 동시에 존재하는 대규모 3차원 GIS 시뮬레이션 환경에서도 안정적인 성능을 유지할 수 있어 도시 관리, 재난 대응, 디지털 트윈 기반 가상훈련 시스템을 포함한 다양한 응용 분야에 활용할 수 있다. 향후 연구에서는 제안 구조의 확장성과 실용성을 높이기 위한 다양한 발전 방향을 고려할 수 있다. 먼저, 본 연구에서 분산 처리 대상은 플레이어 객체의 신체 파트 단위로 한정되어 있으므로, 차량, 드론, 시설물과 같이 파트 단위로 분할이 어려운 다양한 유형의 동적 객체에 대한 적용 가능성 검증이 필요하다. 이를 통해 제안 구조의 범용성 및 실제 활용 가능성을 더욱 향상될 수 있을 것으로 판단된다. 또한, 객체 상태 데이터의 중요도와 갱신 우선순위를 고려한 적응형 데이터 처리 기법을 적용한다면, 제한된 자원 환경에서도 처리 효율성과 시각적 연속성을 같이 향상시킬 수 있을 것으로 기대된다. 마지막으로 실제 대규모 사용자 환경에서 장시간 운용 실험을 수행하여 시스템의 안정성, 확장성 및 실효성을 종합적으로 검증하는 추가 연구가 필요하다.

감사의 글

본 연구는 대한민국 정부(산업통상자원부 및 방위사업청) 재원으로 민군협력진흥원에서 수행하는 민군기술협력사업의 연구비 지원으로 수행되었습니다.

협약지원(25SNCV03)

참고문헌

- [1] H.-D. Lee, Victim Tree Detection System of Drone Images Based on 3D GIS and Deep Learning Method, Ph.D. Dissertation, Kyungil University, Gyeongsangbukdo, 2022.
- [2] H. Lee, J. Ahn, and Y. S. Kim, "Analysis of BIM/GIS Convergence Research Trends Using LDA Topic

- Modeling," *Journal of Korean Society for Geospatial Information Science*, Vol. 33, No. 2, pp. 33-48, 2025. <https://doi.org/10.7319/kogsis.2025.33.2.033>
- [3] S. Gang, D.-H. Ryu, T. Kim, H.-C. Park, J. M. Kim, and Y. J. Choung, "Development of Core Module and Web System for a Visualization Platform for the 3D GIS Service of Disaster Information using Unity," *Journal of Korea Multimedia Society*, Vol. 20, No. 3, pp. 520-532, 2017.
- [4] Y. C. Shim, A Study on Real-Time Distributed and Parallel Processing for Real-Time Map Generation in Smart Cities, Master's Thesis, University of Seoul, Seoul, 2018.
- [5] C.-R. Yoon, H.-C. Kim, K.-O. Kim, and C.-J. Hwang, "Realtime Video Visualization Based on 3D GIS," *Journal of the Korea Spatial Information Systems Society*, Vol. 11, No. 1, pp. 196-202, 2009.
- [6] L. Gong and M. Ding, "Urban Traffic Digital Twin System Development in Unity," *Scientific Reports*, Vol. 15, No. 1, 40085, 2025. <https://doi.org/10.1038/s41598-025-23943-7>
- [7] M. Kim, B. H. Park, and S. Lee, "Creating a Cooperating VR-AR Space with Synchronizing Objects and Interactions," *Journal of Digital Contents Society*, Vol. 21, No. 11, pp. 1913-1920, 2020. <https://doi.org/10.9728/dcs.2020.21.11.1913>
- [8] S. H. Moon, C. W. Jung, and Y. H. Ahn, "A Study on the Development of a Discrete-Event Simulation Model for the Modular Steel Frame Production Process," in *Proceedings of the Architectural Institute of Korea Academic Conference*, Seoul, pp. 746-747, 2025.
- [9] G.-M. Chung, I.-K. Jeong, and J. Kim, "Implementation of a Massive Metaverse Platform Using Niagara Particle Simulation in Unreal Engine," *Journal of Digital Contents Society*, Vol. 25, No. 5, pp. 1387-1397, 2024. <https://doi.org/10.9728/dcs.2024.25.5.1387>



조장원(Jang-Won Jo)

2022년~현 재: ㈜유토비즈 대리

2025년~현 재: 대전대학교 IT소프트웨어공학과

※ 관심분야 : 가상현실, 증강현실 등



배종환(Jong-Hwan Bae)

2013년 : 한남대학교 국방전략대학원
(공학석사)

2018년 : 공주대학교 대학원
군사과학정보학과 (공학박사)

1998년~2002년: 군인공제회 C&C 선임연구원

2003년~2017년: M&D정보기술/ARES 개발부 이사

2017년~현 재: ㈜유토비즈 대표이사

※ 관심분야 : 국방M&S, 데이터연동, 워-게임, 가상현실,
증강현실 등



유현배(Hyun-Bae You)

2000년 : 일본국립쓰쿠바(筑波)대학교
대학원(공학박사-지능기능공
학전공)

2002년~현 재: 나사렛대학교 오웬스교양대학 교수

※ 관심분야 : 증강현실, 유니버설디자인, AI 공학,
현대선교신학



박선희(Seon-Hui Bak)

2007년 : 공주대학교 영상예술대학원
(공학석사)

2016년 : 부산외국어대학교 대학원
(ICT창의융합 공학박사)

2012년~2017년: 아이에이치테크 이사

2017년~현 재: ㈜유토비즈 기업부설연구소 소장

※ 관심분야 : HCI, 빅데이터(Bigdata), ICT융합, 가상현실,
증강현실, 인터랙티브 등