

<https://doi.org/10.7236/JIIBC.2018.18.4.123>

JIIBC 2018-4-18

대규모 멀티플레이어 온라인 게임을 위한 영역 기반 하이브리드 구조의 대역폭 요건

Bandwidth Requirement of Region-based Hybrid Architectures for Massively Multiplayer Online Games

김진환*

Jin-Hwan Kim*

요약 많은 대규모 멀티플레이어 온라인 게임(MMOG)들은 서버 측의 막대한 대역폭 요건이 필요한 클라이언트-서버 구조를 사용한다. 본 논문에서 클라이언트-서버 구조와 peer-to-peer 구조의 장점을 결합하며 게임 상태를 변경시키는 중요한 이벤트들만 서버에서 처리하는 영역 기반의 하이브리드 게임 구조를 제시한다. 이 하이브리드 구조에서 중앙 서버는 게임을 영역들로 분할하며 중요 갱신 작업을 각 영역 내에서 분산시키는 플레이어용 영역 서버로 설정한다. 즉 중앙 서버는 영역 서버를 통하여 플레이어들에게 상태 갱신 작업을 전송한다. 동일 영역에 있는 모든 플레이어들은 게임 상태에 영향을 주지 않는 갱신 작업만 직접 상호 교환한다. 분담된 작업은 중앙 서버의 대역폭을 매우 감소시키며 중앙 서버가 더 많은 플레이어들을 동시에 실행시킬 수 있도록 한다. 실험 결과 3 단계 계층의 영역 기반 하이브리드 구조는 클라이언트-서버 구조보다 규모조정 능력이 우수하며 영역 서버로 작용하는 플레이어들의 충분한 대역폭을 요구하는 반면 중앙 서버의 상당한 대역폭을 감소시킬 수 있는 것으로 나타났다.

Abstract Many massively multiplayer online games(MMOG) use client-server architectures that have enormous server-side bandwidth requirements. In this paper, we propose a region-based hybrid game architecture that combines the advantages of client-server and peer-to-peer architectures, so that only critical events changing game state are processed by the server. In this hybrid architecture, the central server divides the game into regions and assigns a player as a regional server to distribute important updates for that region. Thus, the central server sends state updates to the players through the regional servers. All players in the same region directly exchange updates without affecting game state. This division of labor greatly reduces server bandwidth significantly and enables it to serve larger number of concurrent players. Our experiments show that the region-based hybrid architecture with three-level hierarchy scales better than the client-server architecture, saving considerable bandwidth at the central server while requiring enough bandwidth of players acting as regional servers.

Key Words : massively multiplayer, hybrid architecture, region-based, bandwidth, game state

*정희원, 한성대학교 멀티미디어공학과
접수일자 2018년 5월 30일, 수정완료 2018년 6월 30일
게재확정일자 2018년 8월 10일

Received: 30 May, 2018 / Revised: 30 June, 2018 /
Accepted: 10 August, 2018

*Corresponding Author: kimjh@hansung.ac.kr

Dept. of Multimedia Engineering, Hansung University, Korea

I. 서 론

대규모 멀티플레이어 온라인 게임(MMOG; Massively Multiplayer Online Game)을 수행하는 단체의 주요 관심사는 대역폭 요건이다^[1, 2]. 중앙 서버가 게임을 주도할 경우 모든 플레이어는 갱신 작업을 서버에게 전송하며 서버는 새로운 게임 상태를 만들어서 관련된 모든 플레이어들에게 전송한다^[3]. 최악의 경우 서버는 플레이어들의 수의 제곱에 비례하는 통신 대역폭 요건이 필요하게 된다^[4].

서버의 통신 대역폭 문제를 해결하고자 플레이어들로 하여금 게임 갱신 작업을 직접 분산시킴으로써 클라이언트의 대역폭 요건이 강화되는 peer-to-peer(이하 P2P로 기술함) 게임 구조^[5, 6]가 개발되었다. 그러나 P2P 구조에서 플레이어들은 상태 일관성, 사건의 순서화, 부정 행위 등을 직접 해결해야 하며 특히 부정 행위 방지는 매우 어려운 일이다. 클라이언트-서버 구조에서 서버의 대역폭이 병목 현상이긴 하나 서버는 게임 상태를 제어하기 때문에 상태의 일관성을 유지하기 쉽고 플레이어들 간의 부정행위를 방지하기에 용이한 장점이 있다^[7].

본 논문에서는 MMOG를 위하여 클라이언트-서버 구조와 P2P 구조의 장점들을 결합한 하이브리드 구조를 제시한다. 수천 명 이상의 플레이어들이 하나의 게임 세계를 공유하는 MMOG는 대부분 RPG(Role Playing Game), RTS(Real-time Strategy), FPS(First Person Shooting) 게임 등이며 각 플레이어는 가상 세계에서 캐릭터의 역할을 가정한다^[8]. 일반적으로 MMOG에서 플레이어는 세 가지 종류의 액션을 수행하게 된다. 플레이어가 게임 상태에 영향을 주지 않으면서 이동할 때 발생하는 위치 변경 액션과 플레이어가 특정 아이템을 획득하는 플레이어-객체 상호 작용 그리고 플레이어들 간의 공격 행위 즉 플레이어-플레이어 상호 작용 액션 등이 존재한다^[8]. 특히 플레이어-객체 상호 작용과 플레이어-플레이어 상호 작용은 게임 상태에 영향을 주는 중요한 액션이다. 본 논문의 하이브리드 구조에서 서버는 게임을 다수의 영역으로 분할하며 각 영역에서 특정 플레이어를 영역 서버로 설정하게 된다.

게임 상태에 중대한 영향을 주지 않는 위치 변경 액션은 영역 내 플레이어들이 상호 간에 직접 교환하며 게임 상태에 영향을 주는 중요 액션들은 플레이어가 중앙 서버에 직접 전송하게 된다. 중앙 서버는 게임 상태의 일관

성을 유지하며 플레이어들이 유발하는 부정 행위도 예방할 수 있고 또한 게임 상태를 직접 플레이어들에게 전송하는 대신 영역 서버들에게 전송함으로써 상당량의 대역폭을 절감할 수 있다. 반면 영역 서버는 영역 내 플레이어들에게 서버로부터 수신한 게임 상태를 다시 전송해야 하므로 일반 플레이어보다 더 증가된 통신 대역폭이 필요하게 된다.

본 논문에서 제시된 영역 기반 하이브리드 구조에서 중앙 서버는 영역마다 특정 플레이어를 영역 서버로 설정함으로써 서버의 대역폭을 절감할 수 있으며 동시에 실행되는 플레이어들의 수가 증가되더라도 유연하게 수용할 수 있으므로 기존의 클라이언트-서버 구조보다 규모조정 능력이 우수하다. P2P 구조도 서버의 통신량과 작업 부하를 플레이어들에게 분산시킬 수 있기 때문에 규모조정성과 비용 절감 측면에서 MMOG에 적합한 구조로 연구되고 있으나 아직 보안이나 게임 관리에 관한 약점이 문제시되고 있다^[9]. 게임 상태 관리 이외에 사용자 계정이나 패스워드 등 민감한 게임 데이터를 유지 관리하기 위해서도 아직 중앙 서버가 필요하다. 본 논문에서는 상태의 일관성과 보안이 장점인 클라이언트-서버 구조와 규모조정성과 신속한 응답이 장점인 P2P 구조를 결합한 3 단계 계층의 하이브리드 구조를 제시한다.

제시된 하이브리드 게임 구조에서 중앙 서버와 영역 서버 그리고 플레이어들은 역할을 분담함으로써 영역 서버와 플레이어들의 통신 대역폭을 충분히 활용하고 중앙 서버의 통신 대역폭을 대폭 감소시킬 수 있다. MMOG에서 영역의 수와 영역 내 플레이어들의 수에 따른 중앙 서버의 대역폭과 영역 서버의 대역폭에 대한 실험 결과가 비교 분석되었다.

본 논문에서는 MMOG를 위한 하이브리드 구조가 2장에서 제시되며 3장에서 데이터의 복제본이 유지되는 MMOG의 게임 액션이 기술된다. 4장에서는 하이브리드 게임 구조에서 서버, 영역 서버, 플레이어들의 통신 대역폭이 분석되었으며 5장에서 결론이 기술된다.

II. MMOG를 위한 하이브리드 구조

1. MMOG

MMOG는 수천명 이상의 플레이어들이 동일한 게임 세계를 공유함으로써 기존의 온라인 게임과는 차별화된

다^[10]. 대부분의 MMOG는 역할 분담을 하는 RPG 또는 실시간 전략 게임인 RTS이며 RTS와 RPG가 결합된 형태도 존재한다^[5]. 전형적인 MMOG로 EverQuest, Ultima Online, Star Wars Galaxies, The Sims Online and Simcountry 등이 있다. 1 인칭 슈팅 게임 FPS인 Quake와 Doom 그리고 네트워크화된 RTS 게임 Warcraft III도 MMOG로 분류될 수 있다. MMOG의 기본적인 전제는 각 플레이어가 가상 세계에서 특정 캐릭터의 역할을 수행하는 것이다. 각 플레이어는 게임에서 자신의 캐릭터를 대표하는 아바타를 통해서 게임 세계를 보고 듣고 여러 행위를 수행하며 경험하는 것이다.

단독 혹은 집단의 일부로서 임무나 모험을 수행하는 게임 캐릭터가 게임 세계의 여러 지역을 탐험하면서 경험을 축적하여 점수를 획득하거나 다른 플레이어들과 상호 작용을 하는 행위, 특정 아이템을 찾거나 수익을 추구하는 행위 등이 전형적인 게임에 포함된다. 따라서 MMOG마다 미세한 차이는 있지만 기본 게임 메카니즘, 자료 구조, 통신 패턴 등은 매우 유사하다^[5].

2. 영역 기반 하이브리드 게임 구조

클라이언트-서버 구조는 게임 상태의 일관성을 유지하기는 용이하나 플레이어들의 수가 증가할 때 중앙 서버의 대역폭이 많이 필요하다는 문제가 있으며 P2P 구조는 서버를 사용하지 않는 대신 플레이어들이 야기하는 부정행위에 취약하다는 보안 문제가 있다. 본 논문에서는 두 구조의 장점을 결합함으로써 게임 상태의 일관성을 유지하고 서버의 대역폭 요건을 완화하여 플레이어들의 수가 급증하는 경우에도 이를 수용할 수 있는 하이브리드 구조를 제시한다(그림 1 참조).

중앙 서버는 게임 세계를 여러 영역으로 분할하고 각 영역의 특정 플레이어들을 영역 서버로 설정한다. 그림 1에서 일반 플레이어는 단일 선으로 작성된 원 안에 P로 표시되며 영역 서버는 이중 선으로 작성된 원 안에 P가 표시되어 있다. 각 영역 서버는 중앙 서버로부터 수신한 게임 상태를 영역 내의 모든 플레이어들에게 전송하는 역할을 수행한다. 영역 내의 플레이어들은 P2P 구조로써 게임 상태를 변경시키지 않는 위치 변경 액션을 직접 상호 교환하며 그림 1에서 실선으로 표시되었다. 반면 게임 상태를 변경 시키는 액션은 플레이어가 중앙 서버에 직접 전송하며 그림 1에서 점선으로 표시되었다.

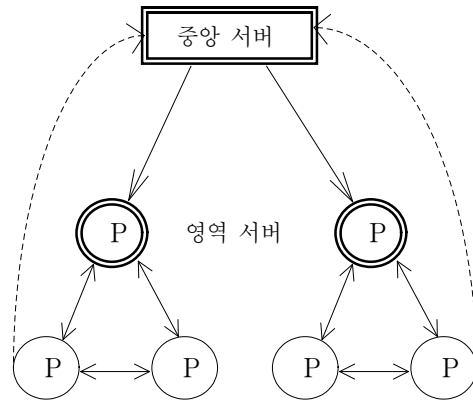


그림 1. 중앙 서버와 영역 서버가 있는 하이브리드 구조
 Fig. 1. A hybrid architecture with central server and regional servers

영역 내의 플레이어들의 수가 증가할 때 작업 부하를 분산하는 것은 이 하이브리드 구조의 중요한 문제이다. MMOG의 플레이어들은 주어진 시간 간격 동안 게임 세계에서 근거리를 이동하기 때문에 제한된 움직임과 시야를 비롯한 제한된 감지 능력을 가지고 있다. 즉 플레이어들은 시야 내에 있는 객체들 그리고 다른 플레이어들과 상호 작용을 할 수 있는 것이다^[11]. 각 플레이어는 게임 세계에서 자신의 위치와 시야에 기반하여 관련된 게임 상태만 수신하게 하는 관심 관리(interest management) 기법은 플레이어가 접근할 수 있는 게임 상태의 규모를 수치적으로 제한한다^[8].

관심 관리에 가장 많이 사용되는 기법은 게임 세계를 소규모의 영역으로 분할하는 지역화(zoning) 기법이다^[12]. 본 논문에서도 플레이어들이 제한된 이동 속도와 감지 능력을 가지고 있다는 사실을 기반으로 게임 세계가 영역으로 분할되는 것을 가정한다. 이 영역은 관심 구역(AOI: Area of Interest)으로도 기술되며 게임 지도의 위치상 동일한 영역에 있는 플레이어들은 해당 구역에 대한 관심 그룹을 형성하며 구역 내 상태 갱신 행위만 그룹 내에서 교환한다. 임의의 플레이어는 자신이 속한 영역에서 다른 영역으로 이동할 수도 있으며 게임 기법에 따라 다른 이웃 영역의 갱신 과정도 파악할 수 있다.

각 플레이어는 자신의 관심 구역에 있는 객체 그리고 플레이어들과 상호 작용을 하게 되므로 게임 세계의 모든 객체들 대신 관심 구역에 있는 객체들의 복제본만 필요한 것이다. 따라서 관심 관리 기법은 복제된 객체 갱신

작업시 통신량을 감소시키는 중요한 역할을 수행한다. 게임 세계에 존재하는 모든 객체들의 복제본을 유지하는 것과 비교하면 필요한 컴퓨팅 자원과 통신 대역폭이 상당히 감소될 수 있는 것이다. 관심 구역이 지역화되는 방법에 따라 영역의 모양과 규모가 달라질 수 있으나 본 논문에서는 이에 대한 구체적 내용은 기술하지 않기로 한다. 제시된 하이브리드 게임 구조에서 중앙 서버는 게임 세계를 플레이어들의 수에 따라 다수의 영역으로 분할하고 각 영역마다 컴퓨팅 자원과 통신 대역폭의 여유가 있는 특정 플레이어를 영역의 서버로 설정하는 것을 가정한다.

III. 복제된 객체와 게임 액션

1. 객체 유형과 복제

MMOG에서 전형적인 게임 세계는 네 가지 객체 유형으로 구성된다^[5]. 첫째, 풍경이나 지형 정보 등 게임 종료 전까지 거의 변화가 없는 불변의 객체들이며 게임 시작 시 초기화되고 클라이언트 측에 설치된다. 둘째, 입력 장치로 플레이어에 의해 제어되는 게임 세계의 캐릭터 또는 아바타 객체이며 게임 세계에서 위치가 정해진다. 셋째, 음식, 무기, 도구 등 변경될 수 있는 객체들이며 플레이어들은 이 객체들과 상호 작용이 가능하다. 넷째, bots로 불리는 NPC(Non-Player Character)는 대부분 플레이어가 아닌 인공 지능에 의해 제어된다. 본 논문에서 언급되는 게임 객체는 불변의 객체를 제외한 아바타, 변경 가능한 객체, NPC 등을 의미한다.

대부분의 게임 엔진들은 기본 사본 복제(primary-copy replication) 기법을^[5] 사용하며 각 객체의 갱신을 책임지는 사본을 기본 사본(primary copy 또는 master copy)으로 나머지 사본들을 2차 사본(secondary copy) 또는 복제본(replica)이라고 정의한다. 각 플레이어는 자신이 속한 영역 내의 플레이어들이 보유하는 객체의 기본 사본에 대응하는 복제본을 가지고 있다. 객체에 대한 갱신 작업은 반드시 기본 사본에서 먼저 수행되어야만 한다. 만일 어떤 플레이어가 자신이 가지고 있는 복제본 객체에 대하여 갱신 작업을 수행할 경우 이 갱신 과정은 기본 사본을 가진 플레이어에게 먼저 전송된다. 기본 사본을 보유한 해당 플레이어가 갱신 과정을 승인할 경우 갱신된 객체는 복제본을 가진 영역내의 모든 플레이어들에게 전송됨으로써 객체에 대한 변경 작업이 진행되는 것이다.

2. 플레이어 상태와 게임 액션

플레이어의 상태는 게임 세계에서 자신의 위치 그리고 자신의 능력, 건강 정도, 소유물 등과 같은 게임 아바타의 상태를 포함한다. 아바타 상태는 지속적이며 로그인 세션이 변경되어도 유지될 수 있다. NPC와 게임 객체들의 상태는 게임 내에서 역할에 따라 지속적이거나 일시적일 수 있다.

일반적으로 플레이어에게 위치 변경, 플레이어-객체 상호작용, 플레이어-플레이어 상호작용 등 세 종류의 액션이 허용된다. 플레이어들은 게임 규칙에 따라 아바타들의 상태는 물론 NPC를 포함한 객체들 또는 다른 플레이어들의 상태도 변경시킬 수 있다. 예를 들면 병에 있는 액체를 마실 경우 병이라는 객체의 상태는 가득찬 상태에서 비어있는 상태로 변경되며 플레이어 객체의 갈증을 나타내는 매개변수의 수치도 변경되는 것이다. 또한 플레이어가 다른 플레이어와 전투를 할 경우 두 플레이어 객체들의 강건함을 나타내는 매개변수들도 변경된다. 게임 상태를 변경하지 않는 플레이어의 위치 변경 액션은 본 논문에서 상태 불변 액션으로 기술하며 게임 상태가 변경되는 플레이어-객체 상호작용과 플레이어-플레이어 상호작용은 상태 변경 액션으로 기술한다. 본 논문에서 상태 불변 액션은 영역 내의 모든 플레이어들이 직접 상호 교환하며 상태 변경 액션은 플레이어가 중앙 서버에게 직접 전송한다. 중앙 서버는 수신한 플레이어의 상태 변경 액션의 승인 여부를 결정한 후 영역 서버들에게 객체가 변경된 결과를 전달하고 영역 서버는 이 결과를 다시 영역 내 모든 플레이어들에게 전달한다(그림 1 참조).

지속적인 플레이어의 상태와 지형을 제외하고 대부분의 다른 게임 상태들은 주기적으로 다시 만들어진다. 이는 게임이 진행되면서 모든 NPC 상대자들이 결국 소멸되며 모든 음식이 소진되고 모든 탐구 행위도 해결되기 때문이다. 게임 상태의 재건 과정은 게임 전체가 리셋되거나 모든 NPC들과 객체들이 주기적으로 각각 다시 생성되는 것으로 구현된다. 이런 결과로 초래된 게임 세계는 규모가 방대하며 전형적으로 플레이어들이 상호 간에 연결된 정적인 영역으로 분할된다. 특정 서버에 속한 플레이어들을 다른 서버로 이동시킬 수 있는 연결 과정은 ‘터널’이라고 하는 게임 기법을 사용하여 구현될 수 있다^[5]. 각 영역은 각 클라이언트가 자신의 메모리 용량에 적합한 크기의 데이터를 보관하도록 다시 분할될 수도 있다.

지형과 플레이어들을 위한 그래픽 요소들은 전형적으로 게임 클라이언트 소프트웨어의 부분으로 설치되며 정규 소프트웨어 갱신 기법을 사용하여 갱신될 수 있다. 영역의 지형이 그래픽 기법과 무관한 게임 지도 자체는 게임이 진행되는 동안 변경되지 않는 반면 지도에 있는 동적 구성요소들은 객체로 처리된다.

IV. 성능 분석

1. 실험 환경과 변수

실제 MMOG에서 발생하는 플레이어의 움직임과 데이터 객체의 갱신 등의 결과를 추적하기 어렵기 때문에 본 논문에서는 가상의 MMOG가 하이브리드 구조에 적용되는 게임 모델을 가정한다. 이 모델에서 플레이어 객체는 다른 MMOG와 유사하게 자신의 위치 그리고 다른 게임 상태를 관리하며 각 플레이어는 이동, 음식 섭취, 전투 등 세 가지 액션을 수행할 수 있다. 음식 섭취는 전형적인 플레이어-객체 상호작용이며 전투 행위는 전형적인 플레이어-플레이어 상호작용이다. 이 게임 모델에서 각 플레이어는 자신과 NPC를 포함한 객체들을 매초 10회의 위치 변경을 수행하며 영역 내 다른 플레이어들에게 전송한다. 실제 FPS 게임인 Quake II에서는 플레이어가 초당 20회 정도 위치를 변경하며 이 결과를 다른 플레이어들에게 전송한다^[13].

각 플레이어가 보유한 객체들의 평균 수를 N_{object} 로 영역 내 플레이어들의 수를 N_{player} 로 기술하며 객체의 평균 크기 S_{object} 는 200 바이트로 가정한다. delta 코딩 기법을 사용하면 객체 갱신 메시지를 24바이트로 감소시킬 수 있다^[13]. 대부분의 게임들이 UDP 통신 기법을 사용하므로 메시지 손실이 발생할 경우 게임 상태가 부정확할 수 있어 본 논문에서도 객체의 속성 대신 객체 전체를 전송하는 것으로 가정한다. 각 플레이어가 자신이 보유한 객체들의 위치 변경 즉 상태 불변 액션을 자신을 제외한 다른 플레이어들에게 전송할 때 필요한 통신 대역폭 B_{No_Change} (Bytes/초)은 다음과 같다.

$$B_{No_Change} = N_{object} * S_{object} * (N_{player} - 1) * 10 \quad (1)$$

각 플레이어는 상태 변경 액션인 음식 섭취와 전투 행위를 1분에 평균 각각 3회씩 수행한다고 가정한다. 음식 섭취 횟수와 전투 행위 횟수는 각각 N_{eating} 과 $N_{fighting}$ 으로 표기된다. 음식 섭취와 전투 행위에서 각각 1개의

객체만 상태가 변경되는 것을 가정할 때 플레이어가 중앙 서버에게 전송시 필요한 대역폭 B_{Change} 은 다음과 같다.

$$B_{Change} = S_{object} * (N_{eating} + N_{fighting}) / 60 \quad (2)$$

B_{Change} 는 B_{No_change} 에 비해서 크기가 매우 작다. 플레이어들로부터 수신한 B_{Change} 를 중앙 서버가 컴퓨팅한 후 갱신된 객체들을 영역 서버들에게 전송한다. 전체 플레이어들의 수를 N_{total} 로 영역 서버들의 수를 N_{region} 으로 기술한다. 중앙 서버가 영역 서버들에게 전송할 때 필요한 대역폭 B_{Server} 는 다음과 같다.

$$B_{Server} = N_{total} * B_{Change} * N_{region} \quad (3)$$

각 영역 서버는 중앙 서버로부터 수신한 $N_{total} * B_{Change}$ 바이트를 영역 내 플레이어들에게 다시 전송하게 되고 이 대역폭 B_{Region} 은 다음과 같다.

$$B_{Region} = N_{total} * B_{Change} * (N_{player} - 1) \quad (4)$$

2. 통신 대역폭 분석

클라이언트-서버 구조에서 중앙 서버는 모든 플레이어들로부터 수신한 모든 액션 요청을 처리한 후 다시 모든 플레이어들에게 전송해야 하므로 플레이어 수 제공에 비례하는 통신 대역폭이 필요하다^[4]. 본 논문의 하이브리드 구조에서 중앙 서버는 플레이어들의 상태 변경 액션만 수신해서 처리 결과를 영역 서버에게 전송한다. 실험에서 플레이어들의 수 N_{total} 은 6000 명으로 영역 내 플레이어들의 수 N_{player} 는 25 명부터 100명까지 25명씩 증가시키는 것으로 설정하였다. 영역 내 플레이어들의 수가 25 명씩 증가할수록 영역의 수 N_{region} 은 240, 120, 80, 60 순서로 감소된다. 각 플레이어는 16 개 이상 48 개 이하의 객체들을 유지하며 평균 객체 수 N_{object} 는 32로 설정한다.

실험에서 N_{player} 를 25부터 100까지 증가시킬 경우 N_{region} 은 240에서 60으로 감소하므로 중앙 서버의 대역폭 B_{Server} 도 급격히 감소하는 결과가 나타났다(그림 2 참조). 반면 영역 서버는 플레이어의 역할까지 수행해야 하므로 실제 통신 대역폭 B_{Region_actual} 은 B_{Region} , B_{No_change} , B_{Change} 의 합이 된다.

$$B_{Region_actual} = B_{Region} + B_{No_change} + B_{Change} \quad (5)$$

각 플레이어의 실제 통신 대역폭 B_{Player} 도 B_{No_change} 와 B_{Change} 의 합으로 구성된다.

$$B_{Player} = B_{No_change} + B_{Change} \quad (6)$$

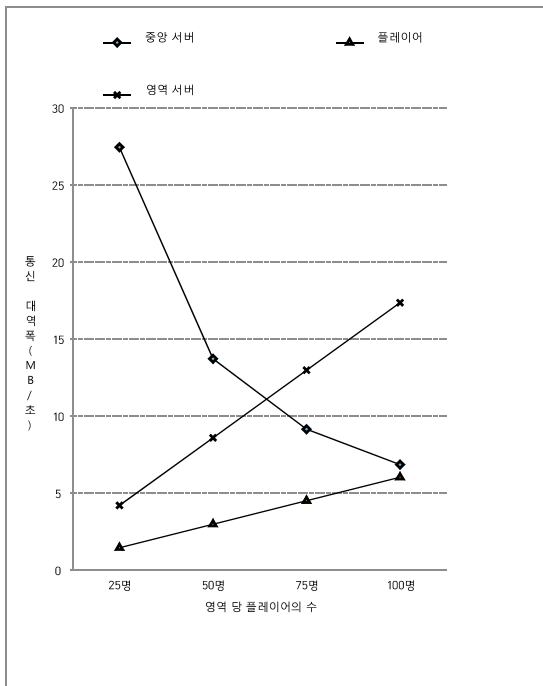


그림 2. 영역 당 플레이어의 수 N_{player} 에 따른 대역폭
Fig. 2. Bandwidth for N_{player} , number of players per region

B_{change} 는 실제 20 바이트로 N_{player} 의 최소값인 25일 때 B_{No_change} 인 1.46MB에 비해서 매우 작다. 플레이어의 대역폭은 영역 당 플레이어들의 수에 비례하여 완만하게 증가되고 있는 반면 영역 서버의 대역폭은 영역 내 플레이어들의 수가 증가할수록 플레이어 대역폭의 2 배보다 약간 작은 대역폭을 더 필요로 하고 있다. 따라서 영역 서버의 대역폭은 영역 내 플레이어 대역폭의 3 배에 근접한 대역폭이 나타나고 있다. 영역 내 플레이어들의 수가 증가할수록 영역 서버의 대역폭이 급증함에 따라 영역 서버 역할을 수행하는 플레이어의 처리 능력과 용량 등을 고려할 필요가 있다. 실험 결과 N_{player} 가 50일 때 중앙 서버의 대역폭이 영역 서버보다 크며 75일 때 중앙 서버보다 영역 서버의 대역폭이 더 증가하므로 중앙 서버보다 영역 서버의 대역폭이 작아지도록 영역 당 플레이어의 수를 작게 유지하는 것이 바람직하다.

영역을 구분하지 않고 중앙 서버가 6000명의 플레이어들에게 직접 전송할 경우 약 686.64MB의 대역폭이 필요하고 이는 N_{region} 이 240($N_{player}=25$)일 때의 대역폭 27.46MB에 비하면 약 25 배 정도가 된다.

플레이어의 상태 변경 액션인 N_{eating} 과 $N_{fighting}$ 이 1 분 당 3회에서 6회로 각각 증가할 경우 중앙 서버의 대역폭과 영역 서버의 대역폭은 그림 3과 같이 나타난다.

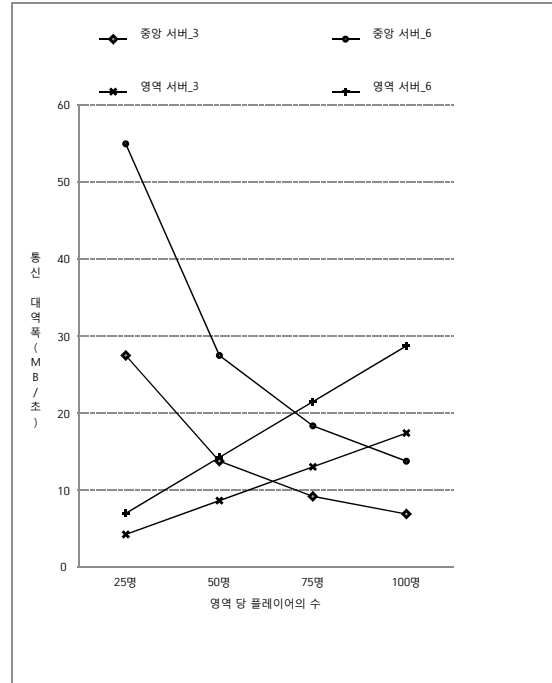


그림 3. N_{eating} 과 $N_{fighting}$ 이 증가된 중앙 서버와 영역 서버의 대역폭
Fig. 3. Bandwidth of central server and regional server for increasing N_{eating} and $N_{fighting}$

이 경우 상태 변경 액션은 플레이어가 중앙 서버에 직접 요청하므로 플레이어의 대역폭은 그림 2와 거의 동일하다. N_{eating} 과 $N_{fighting}$ 이 분당 6회인 중앙 서버와 영역 서버의 대역폭은 그림 3에서 중앙 서버_6과 영역 서버_6으로 표시되었다. N_{player} 가 50일 때 중앙 서버_6의 대역폭이 영역 서버_6보다 크며 N_{player} 가 75일 때 중앙 서버_6의 대역폭이 영역 서버_6보다 작게 나타나는 결과는 N_{eating} 과 $N_{fighting}$ 이 분당 3 회인 경우와 동일하게 나타났다. 그러나 N_{eating} 과 $N_{fighting}$ 이 각각 6인 경우 중앙 서버의 대역폭이 영역 서버보다 작아지기 시작하는 영역 당 플레이어의 수는 N_{eating} 과 $N_{fighting}$ 이 각각 3인 경우보다 다소 증가한 것으로 나타났다. 즉 그림 3에서 중앙 서버_6과 영역 서버_6이 만나는 점은 중앙 서버_3과 영역 서버_3이 만나는 점보다 x 축 상 우측에 있으므로 영역 당 플레이어의 수가 더 많음을 의미하는 것이다. 이와 같이 상태 변경

액션 요청이 증가할수록 중앙 서버의 대역폭 이하이면서 영역 서버가 수용할 수 있는 플레이어들의 수가 일부 증가될 수 있음을 의미한다.

V. 결 론

본 논문에서 수천 명 이상의 플레이어들이 참여하는 MMOG를 위하여 클라이언트-서버 구조와 P2P 구조^[14]의 장점을 모두 활용하는 영역 기반의 하이브리드 구조를 제시한다. 이 하이브리드 구조에서 서버는 게임 세계를 다수의 영역으로 분할하며 각 영역에서 처리 능력이 확보된 플레이어를 영역의 서버로 설정한다. 각 영역 내 플레이어들은 상태 불변 액션을 상호 간에 직접 교환하며 상태 변경 액션은 플레이어가 직접 중앙 서버로 전송한다. 중앙 서버는 상태 변경 액션 결과를 영역 서버를 통하여 플레이어들에게 전송함으로써 통신 대역폭을 대폭 절감할 수 있다. 반면 이 하이브리드 구조에서 컴퓨팅 능력이 향상된 스마트폰과 PC를 사용하는 플레이어들의 대역폭과 영역 서버의 대역폭은 증가하게 된다^[15].

중앙 서버는 게임 세계의 주요 데이터를 관리하며 보안성을 강화할 수 있고 플레이어들의 대역폭을 활용함으로써 급증하는 플레이어들을 수용할 수 있다. 하이브리드 구조에서 플레이어들의 수와 게임 환경을 고려한 영역 분할과 구성에 대한 구체적 연구와 실험 결과가 더욱 필요할 것으로 전망된다.

References

- [1] J. Y. Wang, K. Zhanga and H. Jacobsen, "Demo: combat state-aware interest management for online games," 18th ACM/IFIP/USENIX Middleware Conference, pp. 17-18, 2017.
- [2] J. Jardine and D. Zappala, "A hybrid architecture for massively multiplayer online games," International ACM SIGCOM Workshop on Network and System Support for Games, pp. 60-65, 2008.
- [3] Suznjevic, M., Stupar, I. and Matijasevic, M., "Traffic modeling of player action categories in a MMORPG," International ICST Conf. on Simulation Tools and Techniques, pp. 280 - 287, 2012.

- [4] J. D. Pellegrino and C. Dovrolis. "Bandwidth requirement and state consistency in three multiplayer game architectures," 2nd Workshop on Network and System Support for Games, pp. 52 - 59, 2003.
- [5] B. Knutsson, H. Lu, W. Xu, and B. Hopkins. "Peer-to-Peer support for massively multiplayer games," 23rd Annual Joint Conference of the IEEE Computer and Communications Societies, 2004.
- [6] H.-H. Lee and C.-H. Sun, "Load-balancing for peer-to-peer networked virtual environment," 5th ACM SIGCOMM workshop on Network and System Support for Games, pp. 14, 2006.
- [7] I. Barri, C. Roig and F. Gine, "Distributing game instances in a hybrid client-server/P2P system to support MMORPG playability," Multimedia Toos and Applications, v. 75, Issue 4, pp. 2005-2029, Feb. 2016.
- [8] A. Yahyavi and B. Kenne, "Peer-to-peer architectures for massively multiplayer online games: a survey," ACM Computing Surveys, v. 46, no. 1, Oct. 2013.
- [9] J. Kim, "Reduction method of network bandwidth requirement for the scalability of multiplayer game server systems," The Journal of the Institute of Internet, Broadcasting and Communication, v. 13, no. 4, pp. 29-38, 2013.
- [10] L. Fan, P. Trinder and H. Taylor, "Design issues for peer-to-peer massively multiplayer online games," Int'l Journal of Advanced Media and Communication, v. 4, no. 2, pp. 108-125, 2010.
- [11] Y. Makbily, C. Gotsman and R. Bar-Yehuda, "Geometric algorithms for message filtering in decentralized virtual environments," Symposium on Interactive 3D Graphics, pp. 39-46, 1999.
- [12] MChen, J., Wu, B., Delap, M., Knutson, B., Lu, H., and Amza, C., "Locality aware dynamic load management for massively multiplayer games," ACM SIGPLAN Symp. on Principles and Practice of Parallel Programming (PPoPP). ACM, pp. 289 - 300, 2005.
- [13] A. Bharambe, A. J. Pang and S. Seshan.

"Colyseus: A Distributed Architecture for Online Multiplayer games," International Conference on Networked Systems Design & Implementation, pp. 3-6, 2006.

- [14] Y. Jung, S. Cho, J. Lee, K. Jeong, "A Design of P2P Cloud System Using The Super P2P," International Journal of Internet, Broadcasting and Communication, v. 7, no. 1, pp. 42-48, 2015.
- [15] H. Ko, S. Jung, I. Lee, J. Cho, J. Cha, "A Study on Compatibility between LTE and WLAN System for Mobile Satellite Wireless Package System," International Journal of Internet, Broadcasting and Communication, v. 7, no. 2, pp. 130-136, 2015.

저자 소개

김진환(정회원)



- 1986년 : 서울대학교 컴퓨터공학과 졸업(학사)
- 1988년 : 서울대학교 컴퓨터공학과 졸업(석사)
- 1994년 : 서울대학교 컴퓨터공학과 졸업(박사)
- 1994년 ~ 1995년 : 서울대학교 컴퓨

터신기술공동연구소 특별연구원

- 1995년 ~ 현재 : 한성대학교 컴퓨터공학부 교수

<주관심분야 : 멀티미디어 시스템, 실시간 게임 시스템>

※ This research was financially supported by Hansung University