

<http://dx.doi.org/10.7236/IIBC.2014.14.4.91>

IIBC 2014-4-14

멀티플레이어 온라인 게임을 위한 P2P 구조의 객체 복제와 일관성 제어 기법

Object Replication and Consistency Control Techniques of P2P Structures for Multiplayer Online Games

김진환*

Jinhwan Kim*

요 약 멀티플레이어 온라인 게임을 위한 주요 구조는 전형적으로 클라이언트-서버, 멀티 서버, P2P(peer-to-peer) 구조 등이 있다. P2P 구조는 본질적인 분산과 협동 특성으로 인하여 구축 비용이 저렴하며 플레이어들 간에 직접 통신을 수행함으로써 빠른 응답시간과 높은 규모조정성을 가질 수 있다. 그러나 P2P 구조는 여러 가지 어려움이 존재한다. 플레이어들 간에 게임을 분산시키므로 제어 유지가 어렵고 특정 플레이어의 고의적인 부정행위에 취약해지는 경향도 있다. 또한 갱신과정의 충돌 현상이 여러 사이트에서 발생될 수 있기 때문에 P2P 시스템에서 일관성 제어를 제공하는 것도 더욱 어렵다. 비일관성을 회피 또는 정정하기 위하여 대부분의 멀티플레이어 게임은 객체에 대한 갱신 주 사본에 먼저 수행되는 주 사본 기법을 사용한다. 본 논문은 멀티플레이어 온라인 게임을 위한 P2P 구조에서 각 객체에 대한 일관성을 제공하며 갱신 결과의 전송 메카니즘이 존재하는 주 사본 모델과 이에 대한 성능 분석 결과를 기술한다.

Abstract The main game architectures for multiplayer online games are the traditional client-server architectures, multi-server architectures and P2P(peer-to-peer) architectures. P2P architectures, due to their distributed and collaborative nature, have low infrastructure costs and can achieve high scalability as well as fast response time by creating direct connections between players. However, P2P architectures face many challenges. Distributing a game among peers makes maintaining control over the game more complex. These architectures also tend to be vulnerable to churn and cheating. Providing consistency control in P2P systems is also more difficult since conflicting updates might be executed at different sites resulting in inconsistency. In order to avoid or correct inconsistencies, most multiplayer games use a primary-copy replication approach where any update to the object has to be first performed on the primary copy. This paper presents the primary-copy model with the update dissemination mechanism that provides consistency control over an object in P2P architectures for multiplayer online games. The performance for this model is evaluated through simulation experiments and analysis.

Key Words : Multiplayer, P2P, Consistency, Replication, Primary-copy

*정회원, 한성대학교 멀티미디어공학과
접수일자 2014년 5월 14일, 수정완료 2014년 7월 5일
게재확정일자 2014년 8월 8일

Received: 14 May, 2014 / Revised: 5 July, 2014

Accepted: 8 August, 2014

*Corresponding Author: kimjh@hansung.ac.kr

Dept. of Multimedia Engineering, Hansung University, Korea

I. 서 론

멀티플레이어 온라인 게임(MOG; Multiplayer Online Game)은 막대한 이익이 창출 가능한 인기있는 온라인 사업 중 하나이며 새롭고 도전적인 요소를 가지고 있다. 많은 플레이어들이 게임에 참여할수록 게임 환경도 상호작용이 증가하고 더욱 복잡해지며 플레이어들의 관심도 증가하게 된다. 현재 World of Warcraft^[1]는 거의 천이백만명의 플레이어들이 참여하고 있는 성공적인 멀티플레이어 온라인 게임으로 인식되고 있다.

전자 오락인 비디오 게임은 제어기로 실행되며 시각적인 feedback을 발생시킴으로써 사용자에게 상호작용 기능을 제공한다. 이에 비하여 여러 플레이어들에 의해 실행되는 MOG에서 플레이어들은 단순히 독자적인 적의 입장이거나 아니면 팀으로 참여하여 게임을 수행할 수 있다. 플레이어들은 서로를 적으로 여기며 참여할 수도 있지만 가상의 적들이 인공 지능으로 제어되는 온라인 게임에도 참여할 수도 있다. 대규모 MOG는 수십만명의 플레이어들을 지원할 수 있으며 인터넷을 기반으로 대부분의 게임이 수행될 수 있다. World of Warcraft, Eve Online^[2], Final Fantasy XI^[3] 등은 현재 사업 규모가 계속 증가하는 MOG들이다.

참여하는 플레이어들의 수에 따라 MOG는 엄청난 네트워크 트래픽과 처리 부하를 유발할 수 있다^[4]. 따라서 MOG는 동시에 수천명 이상의 플레이어들을 지원할 수 있는 규모조정성(scalability)과 일관성(consistency), 보안, 신속한 응답시간 등의 도전적 요인을 해결할 수 있어야 한다. 그렇지 않으면 플레이어인 고객의 만족도가 급격히 감소될 수 있다. 게임 실행과 게임 상태의 전송이 서버에 의해 전적으로 제어되는 클라이언트-서버 시스템은 현재 가장 많이 사용되는 게임 구조이다^[5]. 그러나 P2P(Peer-to-peer) 구조는 여러 면에서 게임의 기본 구조로 활용될 수 있는 장점이 있다. 클라이언트 노드들이 직접 통신을 수행하거나 게임 상태 처리 작업의 일부를 수행한다면 컴퓨터 처리 능력과 네트워크 통신 대역폭 면에서 서버의 요건을 상당히 감소시킬 수 있기 때문이다.

컴퓨팅 작업과 통신량을 플레이어들에게 분산시킬 수 있는 P2P 구조는 최근 활발히 연구되고 있으며 잠재적으로 높은 규모조정성, 저렴한 비용, 좋은 성능 등을 달성할 수 있다. 본 논문에서는 MOG를 P2P 구조에 적용하기 위

한 객체 복제 기법과 일관성 제어 기법을 제시한다.

본 논문에서는 MOG를 위한 게임 객체와 구조를 2장에서 분석하며 3장에서 P2P 구조를 위한 객체 복제 모델과 멀티캐스트 통신 기법이 기술된다. 4장에서는 복제된 데이터를 사용하는 P2P 게임 모델의 일관성 제어 기법과 통신 대역폭 분석 결과가 기술되며 5장에서 결론이 기술된다.

II. MOG 설계 원칙과 구조

1. 설계 원칙

가. 객체 유형

최근 비디오 게임의 게임 세계는 네 가지 유형의 구성 요소로 구성된다^[6]. 첫째는 풍경, 지형 정보 등 생성된 이후 게임 수행동안 절대 변하지 않는 불변(immutable) 객체이고 둘째는 입력 장치를 이용하여 플레이어가 직접 제어하는 캐릭터 또는 아바타(avatar)이다. 아바타는 게임 세계에서 위치가 설정된 이후 위치가 변경된다. 셋째는 음식, 무기, 변형될 수 있는 도구 등 변경가능한(mutable) 객체이다. 플레이어들은 다른 플레이어들과의 상호작용시 이러한 객체들을 사용할 수 있다. 넷째는 플레이어가 아닌 인공 지능으로 제어될 수 있는 NPC(Non-Player Character)이다. 본 논문에서도 게임 객체는 아바타, 변경가능한 객체, NPC 들을 언급한다.

나. 플레이어 상호작용

플레이어 상호작용은 플레이어의 갱신, 플레이어와 객체 간의 상호작용, 플레이어들 간의 상호작용 등 세가지 영역으로 구분된다^[6]. 플레이어의 갱신은 플레이어 자신에게만 영향을 주는 게임 세계와의 상호 작용이며 위치 변경이나 플레이어의 아바타에 대한 그래픽 변경 등이 이에 해당된다. 최적화되지 않은 단순한 게임에서 대부분의 플레이어 상호작용은 위치 변경 과정에 해당된다. 플레이어와 객체 간의 상호작용은 플레이어와 변경가능한 객체 간의 상호작용을 의미하며 도구를 집어서 창고에 넣거나 음식을 섭취하는 행위들이 이에 해당될 수 있다. 플레이어들 간의 상호작용은 한 플레이어와 다른 플레이어들 간의 상호작용을 의미한다. 예를 들면 다른 플레이어를 공격함으로써 공격받은 플레이어는 부상당하고 공격한 플레이어는 일정 점수가 증가하는 등의 행위

가 이에 해당된다. NPC와 플레이어의 상호작용은 게임 설계에 따라 플레이어와 객체 간의 상호작용 또는 플레이어들 간의 상호작용으로 간주될 수 있다. 동일한 객체에 대한 갱신 과정이 동시에 발생하여 충돌할 경우 일관성 유지를 위해서 상호작용의 유형 파악이 중요하다.

다. 객체 복제

플레이어가 게임에 참여할 경우 여러 유형의 게임 객체로 구성된 게임 세계의 예시(instance)를 수신하게 된다. 대부분의 게임 엔진들은 주 사본(primary-copy replication) 기법을 사용하고 있다. 절대적인 사본이 존재하는 객체와 캐릭터를 주 사본(primary copy)이라고 하며 나머지 사본들을 부 사본(secondary copy 또는 replica)이라고 정의한다. 각 플레이어는 다른 플레이어들과 연관된 게임 객체의 사본을 가지게 되며 임의의 객체에 대한 갱신은 주 사본에서 먼저 수행되어야 한다. 주 사본과 부 사본의 분산 방법은 게임 구조에 따라 달라질 수 있다. 만일 플레이어가 부 사본인 객체에 대하여 갱신을 수행하고자 할 경우 주 사본에 대하여 먼저 갱신 요청을 해야 한다. 주 사본을 보유한 플레이어가 갱신 요청의 수락 여부를 결정하며 객체를 갱신했을 경우 해당 객체의 부 사본을 가진 모든 플레이어들에게 갱신 결과를 전송함으로써 객체에 대한 갱신 과정이 종료되는 것이다.

라. 일관성 제어

멀티플레이어 게임같은 분산 구조에서 동시에 실행되는 갱신과정의 충돌은 여러 노드에서 발생할 수 있으며 비일관적 상태를 유발할 수 있다. 이러한 비일관성을 회피 또는 교정하기 위한 일관성 메커니즘이 반드시 필요하다. 예를 들어 두 플레이어가 제 3의 플레이어를 거의 같은 시점에서 공격할 경우 모든 플레이어들은 동일한 순서로 실행된 갱신과정을 파악할 수 있어야 하며 먼저 공격한 결과가 다른 모든 사본에 성공적으로 적용되어야 한다. 일반적으로 모든 사본들에 대하여 동일한 순서로 갱신과정이 실행될 경우 모든 노드들은 동일한 상태를 가지게 된다. 그러나 메시지들의 전송 과정에서 순서가 변경되거나 메시지가 손실될 경우 비일관성이 발생할 수 있다. 대부분의 게임들은 속도가 빠르지만 메시지 손실이 발생할 수 있는 비신뢰적인 UDP 메시지 프로토콜을 사용한다. 특별히 여러 개의 객체 상태를 변경하고 원자성(atomicity)이 요구될 경우에는 신뢰성이 있는 TCP 또

는 임계 영역을 위한 2 단계 commit 프로토콜을 사용하여 갱신 과정을 전송할 수도 있다.

모든 상호작용을 고립성(isolation)과 원자성을 보장하는 트랜잭션으로 처리할 경우 가장 강력한 일관성 형태를 가질 수 있다. 그러나 트랜잭션으로 처리할 경우 비용이 증가하고 모든 게임의 행위에 다 적용시키기는 어렵다. 또한 원자성을 보장하기 위하여 2 단계 commit 프로토콜을 수행할 경우 지연시간이 증가하여 게임의 상호작용성을 현저하게 감소시킬 수도 있다. 따라서 현실적인 대안으로 궁극적 일관성이 필요하다. 궁극적 일관성은 일관성이 다소 약화된 형태이며 임의의 객체 사본이 일시적으로는 비일관적일 수 있으나 갱신 행위가 오랜 시간 동안 중지될 경우 모든 객체 사본들이 궁극적으로 동일한 상태를 유지할 수 있음을 의미한다.

상호작용성은 대부분의 게임을 성공시킬 수 있는 요인이므로 게임에서 일관성을 종종 희생시키기도 한다. 게임은 대개 비일관성을 예방하는 대신 비일관성을 해결하는 기법을 제공하며 결과적으로 궁극적 일관성을 유지하는데 주력한다.

2. 게임 구조

가. 클라이언트-서버 구조

MOG를 위한 클라이언트-서버 구조에서 서버는 변경 가능하거나 움직이는 객체와 아바타의 주 사본(primary copy)을 보유하며 게임 세계의 전체 상태를 유지한다. 클라이언트들은 게임 세계에 관한 필요한 정보를 수신하기 위하여 서버와 연결을 시도한다. 모든 플레이어의 갱신 작업과 상호 작용 내용은 게임 진행과 충돌 과정의 해결을 위하여 서버에게 전송되며 서버는 갱신 작업을 실행한 이후 관련된 모든 플레이어들에게 갱신된 객체들을 전송해야 하는 책임이 있다.

서버를 사용하는 집중화된 구조는 게임 세계에 대하여 높은 제어 수준을 제공할 수 있다. 게임 상태는 게임 개발업체의 가장 중요한 자산이 될 수 있으며 게임 업체는 게임 상태를 쉽게 변경하거나 수정할 수 있고 소프트웨어 패치, 캐릭터 변경, 새로운 임무 추가 등과 같은 필요한 갱신 작업을 소프트웨어 상에서 제어할 수 있다. 즉 게임 업체들이 이 구조를 사용하는 주된 이유는 관리와 제어가 용이하다는 것이다. 일관성 관리가 용이한 것도 집중화된 구조의 장점으로 부각된다. 서버는 모든 갱신 작업을 실행하며 충돌과정을 해결하기 때문에 일관성 관

리가 분산성이 높은 P2P 구조에 비하여 용이하다.

그러나 클라이언트-서버 구조의 큰 단점은 규모조정성이 작다는 것이다. 고성능의 서버라 하더라도 수천명이 넘는 플레이어들을 동시에 수용하는 것이 어렵고 각 플레이어에 대한 지연시간 요건을 충족시키기 위해서는 결국 플레이어들의 수를 감소시켜야 하는 상황이 발생한다. 실제로 인기있는 FPS(First Person Shooting) MOG인 Quake II도 클라이언트-서버 구조를 사용하며 동시에 수백명 정도만 수용하는 규모조정성을 유지하고 있다^[7].

클라이언트-서버 시스템에서는 비용과 결합 허용성의 문제도 발생한다. 대규모의 작업 부하를 처리할 수 있는 한 개의 서버는 매우 고가이고 결합 발생시 게임 수행이 중단되며 게임 상태가 손상될 수도 있다. 이러한 문제는 결합 허용을 위한 별도의 backup 서버를 추가함으로써 해결될 수도 있으나 backup 서버의 유지 자체가 시스템을 복잡하게 만들며 비용 증가 요인이 된다.

나. P2P 구조

P2P 구조에서는 모든 노드가 서버와 클라이언트 역할을 동시에 수행할 수 있다. 게임 시스템에서 플레이어가 존재하는 각 노드는 일부 게임 객체의 주 사본을 유지할 책임이 있으며 이들에 대한 갱신 결과를 다른 노드들에 전송하게 된다. 주 사본을 가지고 있는 플레이어는 부 사본을 가진 플레이어로부터 갱신 요청을 받게 되며 갱신 여부를 결정할 수 있다. 주 사본을 갱신할 경우 해당 플레이어는 관련된 다른 모든 플레이어들에게 갱신된 내용을 전송하게 된다. 반면 클라이언트-서버 구조에서는 모든 객체의 주 사본이 서버에 존재하게 되며 클라이언트들은 부 사본만을 유지하게 된다.

P2P 구조에서 작업 부하는 모든 노드들에 분산되며 새로운 노드가 추가될 때 시스템에 새로운 자원도 추가되는 것이므로 규모조정성을 손쉽게 증가시킬 수 있다. 따라서 P2P 구조는 규모조정성에 대한 잠재력이 매우 크다. 게임 제공업체 입장에서 추가 자원에 대한 비용 부담이 거의 없는 것이 이 구조의 특별한 장점이기도 하다. 부하가 모든 peer 즉 플레이어가 있는 노드에 분산되기 때문에 고가의 서버를 유지할 필요가 없으며 게임 관리에 대한 책임도 각 노드가 소량의 부하를 담당하므로 전체 노드들에 분산된다. 임의의 플레이어 노드에서 결합이 발생하더라도 다른 많은 플레이어들에게 부정적인 영향을 미치지 않게 된다. 또한 플레이어들 간에 직접 통신

이 적절하게 수행되며 서버에게 갱신 상태를 전송한 후 서버가 다시 플레이어들에게 전송하는 추가 과정이 필요 없기 때문에 통신 지연시간을 클라이언트-서버 구조보다 감소시킬 수 있다.

그러나 P2P 구조는 여러 단점을 가지고 있으며 가장 취약한 부분은 보안이다. 부정행위 자체는 클라이언트-서버 구조보다 P2P 구조에서 더 쉽게 수행될 수 있다^[8]. 또한 게임의 전체 상태를 유지하는 중앙 서버가 없기 때문에 게임 관리와 제어가 더욱 어려워지는 문제도 있다. 즉 게임 상태가 플레이어들에게 분산되어 있기 때문에 게임 업체가 게임을 완벽하게 제어하기 어렵다. 갱신 과정의 충돌도 여러 노드들에서 발생할 수 있기 때문에 P2P 구조에서 일관성 제어 기법의 적용이 훨씬 더 어려워지며 결과적으로 비일관성이 쉽게 발생할 수 있다.

III. P2P 구조의 주 사본 기법

1. 멀티캐스트 통신

갱신 작업의 전송을 위한 주 사본 기법은 전형적으로 멀티캐스트 메카니즘을 기반으로 구현되고 있다. 플레이어가 갱신 결과를 전송해야 할 플레이어들을 한 그룹으로 고려할 경우 그룹 통신은 유니캐스트나 브로드캐스트보다 멀티캐스트 기법이 유리하다. 따라서 멀티캐스트 기법의 성능이 게임 환경의 비용과 성능에 많은 영향을 미칠 수 있다. 주 사본 기법에서 모든 객체의 갱신 과정은 객체에 연관된 노드 즉 부 사본이 존재하는 노드들에게 전송된다. 이 기법의 장점은 갱신과정 전송시 지연시간이 적게 걸린다는 것이다.

그러나 주 사본 기법에서 각 플레이어마다 높은 대역폭이 요구되는 단점이 있다. 갱신 요청을 수신해야 하는 플레이어들의 수, 갱신 빈도수, 객체의 크기와 수 등에 따라 플레이어의 통신 대역폭이 증가하게 된다^[8]. 이러한 변수들은 게임에 따라 달라지며 실제 Quake II의 경우 평균 객체 크기가 200바이트, 초당 10번의 갱신 횟수, 8-64개의 객체들을 8-64 플레이어들에게 전송할 경우 1-66 Mbps의 대역폭이 필요하게 된다^[7]. 갱신 빈도수도 게임의 형태에 따라 달라진다. 실제로 FPS 게임은 RTS(Real-Time Strategy)나 RPG(Role Playing Game) 게임보다 갱신 빈도수가 더 많다.

2. 데이터 사본 구조

대부분의 멀티플레이어 게임에서 주 사본을 유지하는 방법과 갱신 결과를 다른 부 사본에 전송하는 방법은 게임의 구조에 따라 상이할 수 있다. 게임 세계를 여러 지역으로 분할하는 P2P 시스템에서는 임의의 플레이어가 조정자 역할을 수행할 수도 있다. 조정자는 자신의 노드에 모든 게임 객체들의 주 사본을 유지하고 자신이 유지하는 객체에 대한 모든 갱신 요청을 수신하며 갱신 결과를 멀티캐스트 기법으로 다른 플레이어들에게 전송하게 된다. 이러한 시스템에서 작업부하의 균형을 유지하기 위해 과부하가 걸린 노드들의 경우 해당 영역의 크기를 감소시킬 수 있다. 즉 조정자가 유지해야 하는 객체들의 수를 감소시킬 수 있는 것이다. 그러나 NPC는 인공지능 기법을 이용하여 다음 행위를 결정해야하기 때문에 조정자에 많은 계산 부하가 걸릴 수 있는 단점이 존재한다.

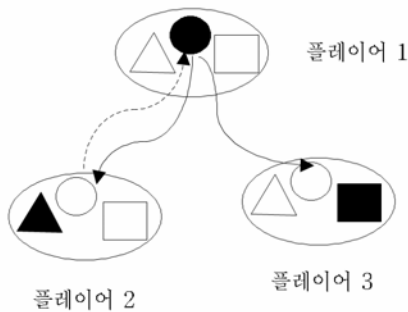


그림 1. 객체 원에 대한 플레이어 2의 갱신 요청
 Fig. 1. Update request of player 2 for object circle

반면 조정자 없이 모든 플레이어들이 상호 간에 직접 통보하는 all-to-all 시스템에서는 각 플레이어가 자신의 아바타를 비롯한 객체의 주 사본을 유지하며 갱신 작업도 자신의 노드에서 수행할 수 있게 된다. 객체를 갱신한 결과는 직접 다른 플레이어들에게 전송하게 된다. 그림 1에서 원, 네모, 세모는 세 개의 객체를 의미하며 각 객체의 주 사본은 검은 색으로 표시되고 객체의 부 사본은 흰색으로 표시된다. 플레이어 1, 2, 3은 각각 원, 세모, 네모 객체의 주 사본을 가지고 있고 나머지 객체들의 부 사본을 가지고 있는 것을 의미한다. 주 사본 모델에서 객체의 모든 부 사본은 읽기 전용이며 플레이어가 임의로 변경할 수 없다. 예를 들어 플레이어 2가 객체 원을 갱신하고자 주 사본을 가진 플레이어 1에게 갱신 요청을 하면(그

림 1의 점선) 플레이어 1이 갱신 허용 여부를 결정하게 된다. 둘 이상의 플레이어들이 동시에 갱신 요청을 할 경우 플레이어 1은 갱신 요청 작업들을 순서화하여 하나의 갱신 요청 작업만 허용하게 되는 것이다. 플레이어 1이 플레이어 2의 갱신 요청을 허용하여 객체 원의 주 사본이 갱신되고 갱신된 객체 원은 플레이어 2와 플레이어 3에게 멀티캐스트 기법을 이용하여 동시에 전송된다(그림 1의 실선).

특별한 조정자가 없는 all-to-all 시스템은 객체와 NPC에 대한 상태 관리가 취약해 질 수 있기 때문에 이를 해결하기 위한 여러 기법들이 제시된 바 있다^[9-11].

노드들이 서로 연결된 상태에 따라 P2P 시스템은 구조화와 비구조화 방식으로 분류된다. 구조화된 P2P 시스템에서는 임의의 노드가 다른 노드로 메시지를 통신망에서 전송할 수 있도록 구체적인 그래프 구조를 형성하기 위하여 결정적인(deterministic) 프로토콜이 사용된다. 반면 비구조화된 P2P 시스템에서는 플레이어들 간 통신망 연결의 조직과 최적화를 위한 결정적인 알고리즘이 없다. 통신망 연결은 일반적으로 임의로 연결되거나 적당한 형태로 구성되도록 확률적 메커니즘을 사용하게 된다. 이러한 통신망에서 탐색 작업은 확률적 방법으로 수행되며 탐색 속도를 향상시키기 위하여 중복 메커니즘인 flooding 기법이나 데이터 복제 기법이 사용된다^[12,13].

P2P 기반 게임의 기본 개념은 게임 상태를 플레이어들에 분산시켜서 처리, 통신, 저장 등의 과정이 수행되도록 하는 것이다. 객체의 주 사본을 사용하는 기법도 모든 객체의 주 사본을 플레이어들에게 분산시키고 있으며 P2P 시스템을 사용하여 갱신 작업을 전송하게 된다. 대부분의 비구조화된 P2P 시스템은 상호 통보를 기반으로 하고 있다^[14]. 이 시스템의 대표적인 게임은 PSense^[15]이다. PSense에서 임의의 플레이어가 이동할 경우 관성 지역내의 모든 플레이어들에게 이동한 플레이어의 갱신된 위치가 즉시 전송된다.

IV. 객체 사본과 일관성 제어

1. 비일관성의 발생

다양한 메커니즘에 의해서 구현되는 일관성의 여러 수준은 상이한 객체와 동일한 중요도로 구성되지 않는 상호작용의 여러 형태에 따라 고려될 수 있다^[16]. 예를 들

면 많은 가상 게임 객체들은 자산으로 고려되며 실제 금액으로 거래될 수 있어서 일관성 제어를 매우 중요한 요건으로 설정할 필요가 있다. 반면 플레이어의 위치 갱신은 낮은 일관성 요건을 갖게 되며 궁극적인 일관성으로도 충분히 유지할 수 있다.

모든 갱신 과정이 주 사본에서 먼저 수행되는 주 사본 모델에서는 부 사본들이 존재하는 노드에서 갱신 과정간의 충돌 현상이 발생하지 않기 때문에 일관성 관리가 단순화된다. 대신 모든 갱신 과정은 주 사본에 대하여 순서적으로 처리되어야 한다. 그러나 주 사본에서 갱신 과정이 종료된 이후 부 사본들은 갱신 결과를 수신하기 때문에 이 시간동안 부 사본들은 오래된 데이터를 보유하고 결과가 발생한다. 플레이어들은 이러한 상태를 관찰하고 주 사본에 대한 의미없는 갱신 요청을 시도할 수도 있다. 즉 플레이어들은 자신이 보유한 오래된 데이터 사본에 기반한 행위가 예상하지 못한 결과를 초래할 수도 있게 된다.

2. 일관성 유지 기법

갱신 전송 시간 때문에 발생하는 데이터의 진부성과 비일관성을 은닉하기 위한 다양한 기법들이 P2P 구조의 게임에 적용되고 있다. 이 기법들은 PCM(Predictive Contract Mechanism)^[17]과 multi-resolution simulation^[18] 등 두가지로 분류된다. dead-reckoning 기법^[19]은 PCM의 가장 보편적인 형태이며 해양이나 항공에서 비행기의 현재 위치를 이전 위치와 움직임 벡터에 기반하여 계산하도록 설계되었다. 게임에서도 플레이어의 위치를 이전 프레임에서의 위치, 속도, 움직임 방향 등에 기반하여 계산해서 다음 프레임에 적용하고 있다. dead reckoning 기법은 향후 발생할 수 있는 충돌을 탐지하기 위해서도 사용되며 메시지들이 지연시간이나 분실로 인하여 정시에 도착하지 못할 때도 사용된다. 모든 사본들이 dead reckoning 기법을 수행할 수 있으며 이 경우 dead reckoning으로 계산된 상태와 실제 상태와의 차이가 허용치보다 클 경우 주 사본은 상태 갱신 결과를 모든 부 사본들에게 전송하게 된다. dead reckoning은 일반적으로 수백 millisecond 정도의 지연시간에 허용될 수 있으며 플레이어의 움직임에 국한되어 사용되고 있다. 실제 게임에서 갱신 과정을 자주 전송하지 않는 플레이어들에 대하여 dead reckoning 기법이 사용될 수 있다.

3. 여러 수준의 일관성 제어

한 행위가 두 개 이상의 객체들을 갱신할 경우 상이한 주 사본들 사이에서 조정자의 역할도 필요하며 원자성과 고립성이 보장되는 트랜잭션으로 실행될 필요가 있다. 그러나 locking이나 다수결 기법처럼 기존의 프로토콜들은 비용이 많이 발생하기 때문에 대부분의 게임들은 약한 일관성 수준을 채택하게 된다. 여러 유형으로 분석되는 게임 행위는 일관성 요건에 따라 여러 영역으로 분류되었다^[20]. 상호작용마다 상이한 수준의 민감성을 갖는 특성에 따라 영역화된 것이다. 각 영역을 위한 일관성 알고리즘이 제공된다. 한 개의 객체만 대상으로 하는 행위에 대해서는 두 가지의 낮은 수준이 유지된다. 첫째, 일관성이 전혀 없는 수준이며 사본들이 best-effort 방법으로 갱신된다. 즉 객체의 색깔 변경 등 그래픽 효과는 이 정도 수준으로 충분하며 해당 노드의 자원에 여유가 있을 때 갱신 과정이 수행되는 것이다. 둘째, 낮은 일관성 수준에서는 데이터의 진부성에 대한 허용치를 제공한다. 플레이어의 점진적인 움직임이 이 영역으로 포함되기에 적절하다. 즉 dead reckoning 기법이 낮은 일관성 메커니즘의 예로 고려될 수 있다.

여러 객체에 대한 행위를 위해서는 세 가지 일관성 수준이 있다. 트랜잭션 시스템에서 제공되는 고립성은 한 트랜잭션의 실행이, 동시에 실행되는 다른 트랜잭션들에 의해서 간섭 받지 않는다는 것을 의미한다. 즉 트랜잭션은 다른 트랜잭션들과 고립되어서 수행되는 것처럼 보여야 한다. 이 고립성과는 다소 차이가 있으나 중간 일관성 수준에서는 고립성 자체가 제공되지 않고 오래된 데이터에 대한 판독이 허용되므로 플레이어는 예상하지 못한 행위의 결과를 받을 수도 있다. 예를 들면 플레이어가 액체가 가득찬 병을 생각했었으나 실제로 빈 병을 획득할 수도 있는 것이다. 높은 일관성 수준은 객체의 가격처럼 분명히 명시된 속성에 관해서만 고립성이 제공된다. 그리고 정확한 일관성 수준은 오래된 데이터의 판독을 허용하지 않는 완벽한 고립성을 제공하고 있다. 일관성 수준이 높을수록 이를 유지하기 위한 메커니즘의 비용이 증가하게 된다.

4. 대역폭 분석

본 논문에서는 주 사본 모델에서 일관성을 낮은 수준과 중간 수준으로 유지할 때 필요한 플레이어의 upload 통신 대역폭을 비교 분석하였다. 각 플레이어의 upload

대역폭 B_{upload} 는 갱신할 객체의 평균 크기(바이트) M , 갱신할 객체의 평균 수 N , 초당 갱신되는 횟수 U , 갱신된 객체를 전송해야 하는 플레이어의 수 P 의 곱으로 결정된다.

$$B_{upload} = M \times N \times U \times P \quad (1)$$

본 논문에서 M 은 200 바이트, N 은 32개, U 는 10회, P 는 16 명에서 64 명까지 16 명씩 증가시키며 플레이어의 움직임(movement) 행위에 대한 실험을 수행하였다. 플레이어의 움직임 행위는 해당 캐릭터의 위치와 목적지 속성을 사용하여 구현된다. 정기적인 시간 간격을 두고 게임은 현재의 위치 속성을 판독하며 약간씩 위치를 갱신하여 목적지에 접근하게 된다. 본 실험에서는 각 플레이어의 노드에서 플레이어의 아바타 캐릭터 1 개와 31개의 NPC들을 포함하여 매초 32 개의 객체들이 갱신되는 것으로 가정하였다.

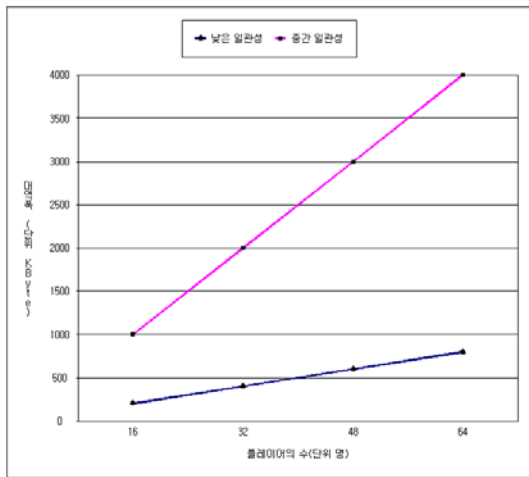


그림 2. 부 사본을 가진 플레이어의 수에 따른 B_{upload}
 Fig. 2. B_{upload} for the number of players with replica

일관성이 낮은 수준에서는 매초 10 회씩 갱신되는 객체들의 위치 속성을 관련 플레이어들에 모두 전송하는 대신 매초 1회 또는 2회 정도만 전송하고 나머지는 각 플레이어의 노드에서 dead reckoning 기법이 적용되어 위치가 갱신되는 것으로 가정한다. 본 실험에서는 낮은 수준의 일관성이 유지될 경우 매초 2 회만 주 사본 객체를 가진 노드가 나머지 부 사본을 가진 노드들에게 전송하는 것으로 가정하여 실험을 수행하였다. 중간 수준의 일

관성에서는 매초 10 회씩 갱신된 각 객체의 위치 속성이 주 사본 노드에서 부 사본 노드로 모두 전송된다. 그림 2에서 낮은 수준의 일관성과 중간 수준의 일관성에서의 대역폭 B_{upload} 가 비교되고 있다.

낮은 수준의 일관성이 유지되는 모델에서의 대역폭은 중간 수준의 일관성이 유지되는 모델에서의 대역폭보다 약 1/5 정도의 크기가 필요하다. 일관성 수준을 낮게 유지하면 필요한 통신 대역폭의 크기는 감소하지만 각 노드에서 dead reckoning 기법을 유지하기 위한 비용이 수반된다.

수식 1과 동일한 주 사본 모델에서 각 플레이어의 download 대역폭 $B_{download}$ 는 매초 각 플레이어로부터 수식 1의 $M \times N \times U$ 를 수신하게 되며 플레이어의 수가 P 일 경우 B_{upload} 와 크기가 같다. 따라서 각 플레이어마다 필요한 통신 대역폭은 B_{upload} 와 $B_{download}$ 를 합한 대역폭이며 결과적으로 B_{upload} 의 두 배가 된다.

5. 낙관적 일관성 제어

모든 게임에서 주 사본 모델을 사용하는 것은 아니며 일부 게임에서는 플레이어에게 신속한 응답을 제공하기 위하여 임의의 사본에 대한 갱신 요청이 허용되기도 한다. 그러나 이 경우 비일관성을 처리하기 위한 프로토콜이 반드시 필요하다. 많은 게임에서 각 노드에 있는 사본의 갱신 과정이 지역적으로 요청되었을 때 이를 낙관적으로 허용하고 있다. 이는 갱신 요청을 주 사본으로 전송함으로써 발생하는 지연시간을 은닉하기 위한 것이다.

임의의 행위가 부 사본과 주 사본에서 상이한 갱신 결과를 유발할 경우 비일관성 해결 프로토콜이 필요하게 된다. Colyseus^[7] 게임에서는 동시에 여러 갱신 과정이 발생할 경우 특별한 충돌 해결 기법없이 가장 최근에 수신한 갱신 과정만 인정하고 있다. 충돌 해결의 대안책은 일관적인 이전의 상태로 복구(roll back)하는 것이다. Timewarp^[21]는 모든 사본들이 낙관적으로 갱신 과정을 실행하는 분산 시뮬레이션과 데이터베이스를 위해 제시된 복구 기반 메카니즘이다. 이 기법에서는 노드들이 순서가 잘못된 갱신 과정을 수신하여 비일관성이 발생하면 이전 상태들을 추적하여 복구하게 된다. 그러나 복구 방법은 플레이어들을 완전히 만족시키기 어렵기 때문에 게임 환경에서 성능을 개선하기 위한 Timewarp 최적화 기법들도 제시된 바 있다^[22, 23].

V. 결론

대규모 멀티플레이어 온라인 게임들은 현재 급성장하는 사업 영역이며 아직 해결하지 못한 어려운 문제점들을 가지고 있다. 대부분의 MOG들은 클라이언트-서버 구조를 기반으로 하고 있으나 스트리밍 시스템^[24]에 적용되는 P2P 구조에 대한 관심도 지속적으로 유지하고 있다. P2P 구조는 고가의 서버를 유지하기 위한 비용을 감소시킬 수 있고 게임의 규모조정성도 향상시킬 수 있다. P2P 구조는 높은 규모조정성과 저가의 구성 비용을 제공하는 반면 제한된 대역폭, 부정행위에 대한 취약점, 데이터의 일관성 보장 등 해결되어야 하는 여러 문제들을 가지고 있다. 본 논문에서는 게임 객체의 주 사본과 부 사본이 구성된 P2P 구조에서 게임 시스템의 일관성을 적절하게 유지할 수 있는 기법과 통신 대역폭이 분석되었다.

향후 대규모 멀티플레이어 온라인 게임에서 게임의 행위와 객체의 특성에 따라 여러 수준의 일관성이 유지될 수 있는 기법은 물론 일관성을 객관적으로 측정할 수 있는 프레임워크에 대한 구체적 연구와 실험적 결과가 필요할 것으로 전망된다.

References

- [1] Blizzard Entertainment, <http://www.worldofwarcraft.com/pvp/battlegrounds>, 2011.
- [2] EVE Online, <http://www.eveonline.com>, 2011.
- [3] Final Fantasy XI, <http://www.playonline.com/ffllus>, 2011.
- [4] 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.
- [5] 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.
- [6] N. Knutsson, H. Lu, W. Xu and B. Hopkins, "Peer-to-peer support for massively multiplayer games," IEEE International Conference on Computer Communications, 2004.
- [7] 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.
- [8] J. Goodman and C. Verbrugge, "A Peer Auditing Scheme for Cheat Elimination in MMOGs," International ACM SIGCOMM Workshop on Network & System Support for Games, pp. 9-14, 2008.
- [9] C. GauthierDickey, V. Lo, and D. Zappla, "Using n-trees for scalable event ordering in peer-to-peer games," International Workshop on Network & Operating Systems Support for Digital audio & Video, ACM, pp. 87-92, 2005.
- [10] S. Y. Hu, S. C. Chang and J. R. Jiang, "Voronoi state management for peer-to-peer massively multiplayer online games," International IEEE Conference on Consumer Communications & Networking, pp. 1134 - 1138), 2008.
- [11] M. Varvello, S. Ferrari, S. E. Biersack and C. Diot, "Distributed avatar management for Second Life," International ACM SIGCOMM Workshop on Network & System Support for Games, pp. 1-6, 2009.
- [12] M. Sozio, T. Neumann and G. Weikum, "Near-optimal dynamic replication in unstructured peer-to-peer networks," ACM International Symposium on Principles of Database Systems, pp. 281-290, 2008.
- [13] W. Nagel, W. Walter, W. Lehner, E. Leontiadis, V. Dimakopoulos and E. Pitoura, "Creating and maintaining replicas in unstructured peer-to-peer systems," International Euro. Conference on Parallel Processing, 4128. Springer, pp. 1015-1025, 2006.
- [14] Y. Chu, S. G. Rao, S. Seshan and H. Zhang, "A case for end system multicast," IEEE Journal on Selected Areas in Communications, v. 20, no. 8, pp. 1456-1471, 2002.
- [15] A. Schmieg, M. Stieler, S. Jeckel, P. Kabus, B. Kemme and A. Buchmann, "pSense-maintaining a dynamic localized peer-to-peer structure for position based multicast in games," International Conference on Peer-to-Peer Computing, IEEE. pp. 247-256, 2008.

- [16] A. Chandler and J. Finney, "On the effects of loose causal consistency in mobile multiplayer games," In International ACM SIGCOMM Workshop on Network & System Support for Games, ACM, pp. 1 - 11. 2005.
- [17] IEEE, IEEE standard for distributed interactive simulation application protocols, IEEE standard 1278.1-1995.
- [18] J. A. Hamilton., D. A. Nash and U. W. Pooch, Distributed simulation, CRC, 1997.
- [19] L. Pantel and L. Wolf, "On the suitability of dead reckoning schemes for games," International ACM SIGCOMM Workshop on Network & System Support for Games, ACM, pp. 79-84, 2002.
- [20] K. Zhang and B. Kemme, "Transaction models for massively multiplayer online games," International Symposium on Reliable Distributed Systems, IEEE, pp. 31 - 40, 2011.
- [21] D. R. Jefferson, "Virtual time," ACM Trans. on Programming Language Systems, v. 7, no. 3, pp. 404 - 425, 1985.
- [22] S. Ferretti, "A synchronization protocol for support in peer-to-peer multiplayer online games in overlay networks," International Workshop on Distributed Event-Based Systems, ACM, pp. 83 - 94, 2008.
- [23] L. Chan, J. Yong, J. Bai, B. Leong and R. Tan, "Hydra: a massively-multiplayer peer-to-peer architecture for the game developer," International ACM SIGCOMM Workshop on Network & System Support for Games, ACM, pp. 37-42, 2007.
- [24] T. Kim and E. Kim, "An Integrated Prefetching/Caching Scheme for P2P Live Streaming," The Journal of the Institute of Internet, Broadcasting and Communication, v. 14, no. 1, pp. 69-76, 2014.

저자 소개

김진환(정회원)



- 1986년 : 서울대학교 컴퓨터공학과 졸업(학사)
- 1988년 : 서울대학교 컴퓨터공학과 졸업(석사)
- 1994년 : 서울대학교 컴퓨터공학과 졸업(박사)
- 1994년 ~ 1995년 : 서울대학교 컴퓨터신기술공동연구소 특별연구원
- 1995년 ~ 현재 : 한성대학교 멀티미디어공학과 교수
<주관심분야 : 멀티미디어 시스템, 실시간 게임 시스템>

※ This research was financially supported by Hansung University.