

게임에서의 물리기반 시뮬레이션 기술의 현황과 전망

최민규(광운대학교)

1. 서론

21세기에 접어들면서 콘솔 및 컴퓨터 게임은 더 이상 어린이들이나 소수의 사람들만이 향유하는 특별한 무엇이 아닌 대부분의 사람들이 즐겨보았거나 또는 즐기고 있는 너무나도 보편적인 여가선용 수단이 되었다. 오늘날의 3D 게임기술은 현실을 그대로 모방하거나 또는 현실에서 실제로 볼 수 없는 초현실적인 것까지도 그럴듯하게 보여줌으로써 게임 이용자들에게 많은 즐거움을 선사하고 있다. 또한, 이러한 유희적 측면뿐만 아니라 교육 및 산업적 측면에서도 큰 기여를 하고 있다. 이와 같은 콘솔 및 컴퓨터게임의 성공요인으로는 여러가지를 들 수 있지만, 그들 중에서도 현실에서와 거의 유사한 물리적 사실성 및 특수 효과가 대표적인 요인이라고 해도 과언이 아니다. 게임의 물리적 사실성 및 화려한 시각적 효과의 이면에서는 “자연계의 물리 현상을 컴퓨터를 이용하여 사실적으로 재현하는 물리기반 시뮬레이션(physically-based simulation)” 기법이 그 핵심적인 역할을 수행하고 있다.

게임에 사용되고 있는 물리기반 시뮬레이션 기술은 단지 게임만을 위해 개발된 게임 전용 기술은 아니다. 보다 사실적인 물리 현상 묘사에 대한 게임 이용자들의 요구와 게임 시장에서 보다 큰 이윤을 남기기 위한 게임 개발사들과 그래픽카드 개발사들의 지속적인 연구와 투자에 의해 컴퓨터그래픽스를 포함한 여러 분야의 물리기반 시뮬레이션 기술들이 게임에 효과적으로 도입된 것이다. 특히, 기존에는 실시간에 이루어지기 힘들었던 컴퓨터 그래픽스 분야의 고급 물리기반 시뮬레이션 기법들이 컴퓨터그래픽스 및 그래픽 하드웨어의 상호보완적인 발전에 힘입어 게임에 사용될 수 있을 정도로 충분히 빨라지게 되었으며, 특수한 환경에서만 사용가능했던 기술들이 게임에도 적용할 수 있게 안정성이 향상되었기 때문이다.

최근에 들어서는 게임에서의 물리기반 시뮬레이션을 그 주목적으로 하는 많은 연구들이 학계에서 활발히 진행되고 있으며, 게임 개발자들을 위한 학회가 창립되기도 하였다. 이 글은 게임에서의 물리기반 시뮬레이션의 기본

개념 및 현재의 기술수준, 그리고 앞으로의 발전 방향을 고찰해 보고자 한다.

II. 게임에서의 물리 기반 시뮬레이션

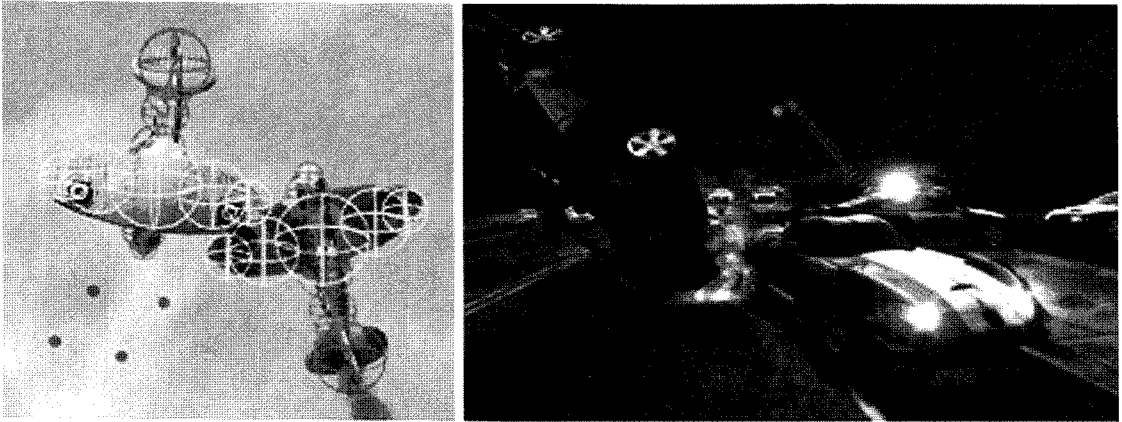
현실세계에는 중력, 관성력, 마찰력, 충돌력, 반발력 등에 따른 여러 물리현상이 존재한다. 최근의 게임들은 현실세계와 유사한 가상현실을 만들기 위하여 이러한 물리현상을 사실적으로 묘사하기 위한 노력을 기울이고 있다. 물리현상을 효과적으로 묘사하기 위한 물리 엔진을 게임 제작사에서 직접 개발하는 경우도 있지만 구현이 어렵고, 오랜 개발기간이 소요되기 마련이어서 최근에 제작된 대부분의 게임들에서는 상용물리엔진이나 무료물리엔진이 사용되고 있다. 제작중인 게임에서 특별한 물리현상을 묘사할 필요가 있다면 그 부분만을 부가적으로 구현하여 기존의 물리 엔진과 함께 사용하기도 한다.

게임에 필요한 대표적인 물리 구성 요소로서는 강체 동역학(rigid body dynamics), 충돌 검사 및 반응(collision detection & response), 기구학 및 역기구학(kinematics & inverse kinematics), 모션 캡처 데이터와 물리 기반 시뮬레이션(motion capture data & physically-based simulation), 파티클 시스템(particle system), 질량-용수철 모델(mass-spring model), 의복 시뮬레이션(cloth simulation), 헤어 시뮬레이션(hair simulation), 변형체 시뮬레이션(deformable body simulation), 유체 시뮬레이션(fluid simulation) 등을 들 수 있다. 이외에도 기계공학, 유체역학 등에서 연구되고 있는 많은 물리 현상들에 대한 시뮬레이션 기법들이 지속적으로 게임에 도입되는 추세이다.

강체 동역학은 최근 게임에서 가장 널리 사용되고 있는 게임물리기술이라 할 수 있다. 강체 동역학은 FPS(일인칭 슈팅) 게임 등에서 총알의 움직임, 기술하거나 충격에 의해 날아가는 물체들의 위치 및 방향을 기술하기 위



〈그림 1〉 강체 시뮬레이션 : 크라이텍에서 제작한 “크라이시스(Crysis)”



〈그림 2〉 충돌 검사 및 반응 : XNA, “니드 포 스피드(Need for Speed)”.

해서 사용되고 있다. <그림 1>은 크라이텍(Crytek)에서 제작한 게임 “크라이시스(Crysis)”에서의 강체 시뮬레이션을 보여준다.

물체가 허공에서 혼자 움직이는 경우가 아니라면 다른 물체와 충돌할 수 있다. 이때 충돌 현상을 제대로 처리하지 않는다면 물체끼리의 침투, 심지어는 물체간의 관통이 발생할 수 있다. 이는 레이싱게임 등에서 심각한 문제가 된다. 현실에서는 물체들이 충돌할 경우 서로를 튕겨 내거나, 미끄러지거나 또는 정지된 상태가 된다. 이러한 충돌 현상을 정확히 시뮬레이션 하기 위해서는 우선 충돌 및 접촉을

빠르게 감지하고 그에 따른 충격량(impulse) 또는 충격력(collision force)을 계산하여 물체의 선운동량(linear momentum) 및 각운동량(angular momentum)을 갱신해야 한다. <그림 2>의 왼쪽은 마이크로소프트의 차세대 게임 개발 플랫폼 XNA에서의 충돌 검사 과정을 도식적으로 보여 주며, 오른쪽은 레이싱 게임의 대표작이라 할 수 있는 “니드 포 스피드(Need for Speed, EA)”에서의 충돌처리 및 반응을 보여준다.

대부분의 3D 게임 캐릭터들은 다관절체(articulated body)로 표현된다. 다관절체의 자

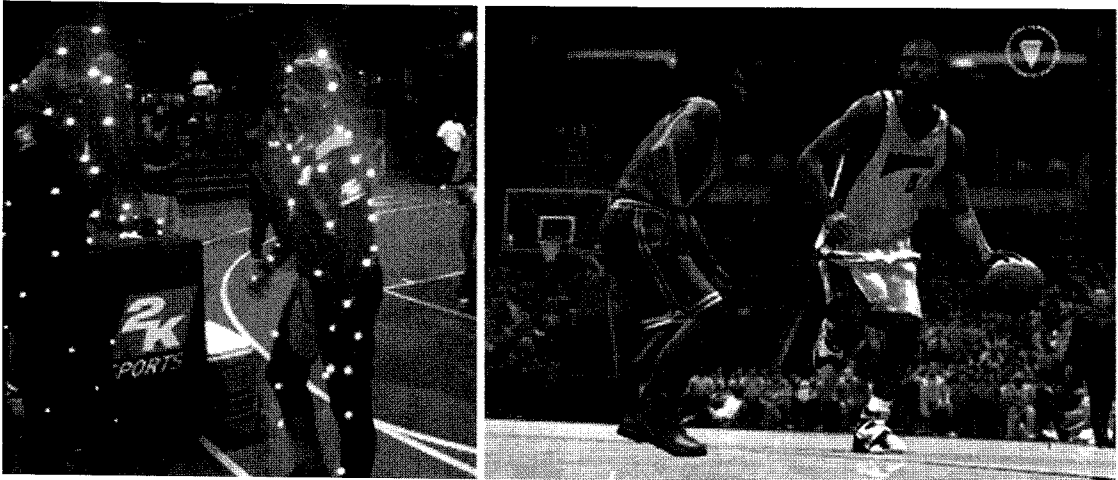


원본

역기구학 적용

역기구학 및 수작업

〈그림 3〉 역기구학을 사용한 캐릭터 자세의 결정 : SEC에서 제작한 “원다와 거상”



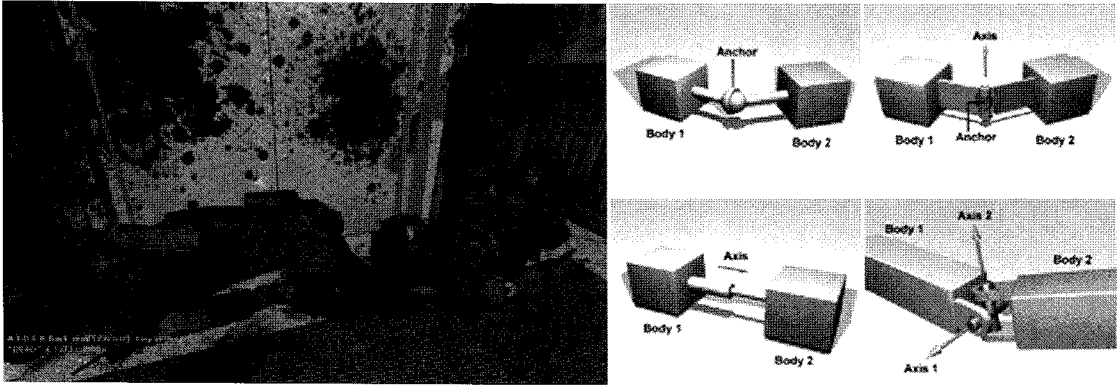
〈그림 4〉 모션 캡처를 이용한 게임 제작 : 2K SPORTS에서 제작한 NBA2K8

세 및 동작을 효과적으로 기술하기 위해서는 정기구학 및 역기구학(kinematics and inverse kinematics)을 사용할 수 있다. 정기구학이란 다관절체의 관절각(joint angle)과 링크의 길이(link length)로 부터 다관절체의 말단(end-effector)의 위치 및 방향을 계산하는 것이다. 역기구학은 반대로 말단의 위치 및 방향으로 부터 관절각을 계산하는 것이다. 역기구학은 <그림 3>에서와 같이 캐릭터의 자세를 주변 환경에 맞추어 수정하는 경우에 효과적으로 사용될 수 있다. 대부분의 3D 게임에서는 미리 저장되어 있는 이동동작을 게임 이용자의 제어에 따라 단순히 재생한다. 이와 같은 경우 캐릭터가 지면 위를 미끄러지듯이 달리는 것처럼 보이는 부작용이 있을 수 있다. 소니 컴퓨터 엔터테인먼트에서 제작한 “원대와 거상(Shadow of the Colossus)”에서는 <그림 3>에서와 같이 역기구학을 이용하여 지형의 기복에 맞도록 보행 애니메이션을 보정함으로써 이러한 부작용을 제거하였다.

최근 여러 스포츠 게임 및 대전 격투게임 등

에서는 사실적이면서 역동적인 동작을 손쉽게 얻기 위해 모션 캡처(motion capture)를 사용하고 있다. <그림 4>는 광학식 마커(optical marker)를 실제 연기자에 부착한 후, 비디오 카메라를 이용하여 마커의 움직임을 포착하고, 이로부터 연기자의 관절각들을 계산하여 게임 캐릭터의 동작을 만들어낸 2K SPORTS의 “NBA2K8”의 스크린 샷을 보여준다. 아직까지는 모션 캡처로 얻은 데이터를 그대로 재사용하거나 역기구학을 이용하여 간단하게 수정하는 정도에 그치고 있지만, 조만간 모션 캡처 데이터로부터 물리기반 제어모형을 학습하고, 이를 이용하여 사용자의 반응에 역학적으로 그럴듯하게 반응하는 캐릭터를 만들 수 있을 것으로 기대된다.

캐릭터가 폭발에 의해 날아가는 동작이나 죽는 동작 등을 만들어내기 위해서는 <그림 5>의 왼쪽에서와 같은 봉제인형 시뮬레이션(rag doll simulation)이 사용되고 있다. 봉제인형 시뮬레이션에서는 다관절체로 이루어진 캐릭터의 관절을 <그림 5>의 오른쪽에서와 같이



〈그림 5〉 봉제인형 시뮬레이션 : “카운터스트라이크(Counter Strike)”

해석하여 그에 따른 다관절체 동역학 (articulated rigid body dynamics)을 시뮬레이션 한다. <그림 5>의 왼쪽은 발브(Valve)에서 제작한 “카운터스트라이크(Counter Strike)”에서의 봉제인형 시뮬레이션에 대한 스크린 샷이다.

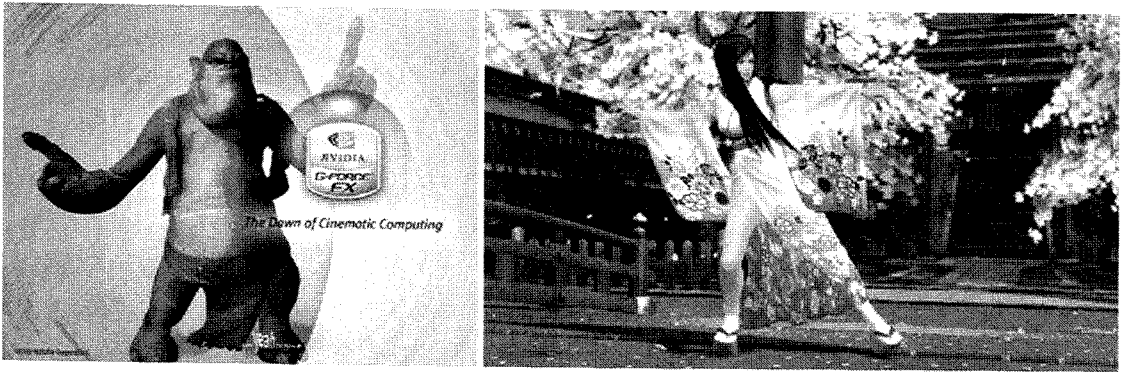
최근의 컴퓨터게임들은 파티클 시스템 (particle system)을 통한 화려한 특수 효과들을 선보이고 있다. 파티클의 사전적 의미는 물리에서 말하는 입자 혹은 미립자이며, 역학적 관점에서 바라볼 때는 질량, 위치, 속도 등의 속성을 가지고 있지만 길이나 넓이 혹은 부피와 같은 공간적인 크기에 대한 속성을 가지고 있

지 않은 질점이다. 파티클 시스템은 중력, 열의 대류, 용수철에 의한 힘 등과 같이 물리적 힘을 받는 파티클들의 집합체를 말한다. 파티클들의 움직임은 상미분방정식으로 표현되며 이를 적분하여 파티클들의 애니메이션을 얻을 수 있다. <그림 6>은 크라이텍에서 제작한 “크라이시스(Crysis)”에서의 파티클 시스템을 이용한 폭발 효과를 보여준다.

캐릭터의 사실성을 높이는데 있어서 의복의 역할은 새삼스럽게 강조할 필요가 없을 정도로 중요하다. 기존의 게임에서는 대부분 미리 만들어진 의복 애니메이션을 상황에 따라 재



〈그림 6〉 파티클을 이용한 불꽃 시뮬레이션 : 크라이텍에서 제작한 “크라이시스(Crysis)”



<그림 7> 의복과 헤어 시뮬레이션 : NVIDIA Ogre 데모, DOA(Dead or Alive)

사용하는 간단한 방식으로 캐릭터의 옷을 움직였다. 하지만 다양한 상황에서 다양한 형태의 역동적인 동작을 필요로 하는 격투게임 등에서 자연스러운 옷의 움직임은 이와 같은 방식으로 보여주기에는 너무 많은 저장 공간이 필요한 단점이 있다. 또한 주변 환경과의 상호작용을 반영할 수 없다는 단점이 있다. 이러한 단점을 극복하기 위하여 최근에는 게임캐릭터의 의복을 실시간에 시뮬레이션하기 위한 시도가 활발히 진행되고 있다.

파티클 사이의 거리에 따른 힘이 주요한 물리 법칙인 파티클 시스템을 질량-용수철 시스템(mass-spring system)이라 한다. 최근에 제작된 게임들에서는 <그림 7>에서와 같은 의복 시뮬레이션(cloth simulation)을 위해 질량-용수철 시스템을 채택하고 있으며, 그래픽 카드 개발사들은 실시간 의복 시뮬레이션을 위하여 GPU(Graphics Processing Unit)를 사용하는 예제들을 시연하고 있다. <그림 7>의 왼쪽은 NVIDIA의 Ogre 데모의 스크린 샷이며, 오른쪽은 테크모(Tecmo)에서 개발한 “데드 오어 얼라이브(Dead or Alive)”에서의 실시간 의복 시뮬레이션이다. 아직까지는 소수의 캐릭터

가 등장하는 경우에만 실시간 성능을 낼 수 있으며, 또한 옷의 움직임에 큰 댐핑(damping)을 가함으로써 옷의 찰랑거림을 다소 미사실적으로 만드는 면이 있다.

사실적인 게임 캐릭터 생성을 위해서는 의복 시뮬레이션뿐만 아니라 헤어 시뮬레이션(hair simulation)도 필수적이다. 헤어도 과거에는 의복과 마찬가지로 애니메이터가 수작업으로 미리 만들어둔 결과에 따라 애니메이션 되었지만, 근래에는 물리기반 시뮬레이션을 통해서 애니메이션 되고 있다. 캐릭터의 옷과 달리 머리카락은 수만개로 이루어져 있다. 따라서 모든 머리카락을 실시간에 시뮬레이션 하는 것은 최고성능의 컴퓨터를 사용하더라도 불가능한 일이다. 게임에서는 실시간 성능을 위해 대표적인 수백가닥의 머리카락을 시뮬레이션하고 그 주변의 머리카락은 대표 머리카락의 움직임을 보간하여 생성한다. <그림 7>의 오른쪽은 테크모(Tecmo)에서 제작한 “데드 오어 얼라이브(Dead or Alive)”에서 헤어 시뮬레이션을 통해 만든 자연스러운 헤어의 움직임을 보여주고 있다.

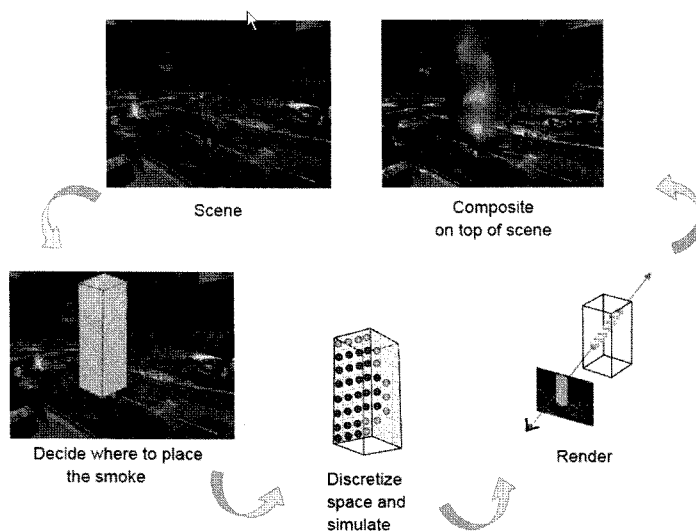
옷이나 머리카락 등의 매우 유연한 물체와



〈그림 8〉 파손 시뮬레이션 : “루카스아츠”에서 제작한 “스타워즈 : 더 포스 언리쉬드”

달리 현실세계의 많은 물체들은 어느 정도는 강체와 유사한 양상을 보이다가 갑자기 파손(fracture) 되는 성질을 가지고 있다. 파손이 일어나는 물체에 대한 시뮬레이션은 레이싱 게임, 대전 격투 게임, FPS 게임 등에서 자동차, 캐릭터, 총알 등에 의해 주변 물체가 부서지는 특수 효과에 필수적이다. 파손 현상을 물리적으로 엄밀히 시뮬레이션 하기 위해서는 연속체 역학(continuum mechanics)을 이용해야 하

지만 게임에서 요구하는 실시간 성능을 내기 어렵다. 따라서 물체가 부서지는 형태를 미리 정의한 후에 충돌을 감지하고 그 충돌에 따른 힘이 임계값(threshold) 이상일 때 물체들이 부서지게 하는 비교적 간단한 방식이 널리 사용되고 있다. <그림 8>은 루카스아츠(LucasArts)에서 제작한 “스타워즈 : 더 포스 언리쉬드”(Star Wars : The Force Unleashed)”에서의 파손 시뮬레이션을 보여준다. 이 게임에서는



〈그림 9〉 “헬게이트 런던(Hellgate London)”에서의 연기 시뮬레이션

DMM 물리 엔진을 이용하여 파손 현상을 실시간으로 시뮬레이션 하였다.

게임에서의 시각적인 즐거움을 높이는데 있어서 물, 연기, 불과 같은 유체(fluid)의 사실적인 묘사가 차지하는 비중이 점차 높아져 가고 있다. 유체를 기술하기 위해서는 나비에-스토크스 방정식(Navier-Stokes equation)을 풀어 시간에 따른 유체의 속도장(velocity field) 및 압력장(pressure field)을 구해야 한다. <그림 9>는 플래그십 스튜디오(Flagship Studio)에서 제작한 “헬게이트 런던(Hellgate London)”에서 사용된 유체 시뮬레이션 과정을 보여준다. 이 게임에서는 빗물이 튀기는 생동감 넘치는 특수 효과를 볼 수 있다. 또한 다양한 연기 시뮬레이션 효과, 즉 연기의 농도 및 흐름 등이 현실의 연기와 흡사한 수준의 그래픽 효과를 보여주기도 한다.

III. 게임 물리 엔진 기술 현황 및 동향

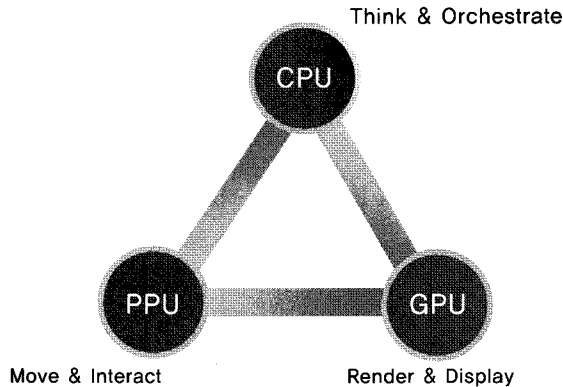
게임물리엔진은 게임 내에서의 물체의 움직임, 반응 등에 필요한 다양한 물리 요소를 관장한다. 기존의 게임들에서는 물체의 움직임이 몇가지로 미리 정해져 있어서 게임 이용자가 자신이 원하는 대로 물체들을 움직이지 못하는 경우가 많았다. 하지만, 컴퓨터 사양이 높아지고 게임이용자들의 시각적 품질에 대한 요구가 높아짐에 따라 보다 사실적인 움직임을 제공하기 위한 많은 물리 엔진들이 개발되었다. 이들은 지금까지 언급한 다양한 물리현상을 모두 다 지원하거나 또는 그 일부분만을 효과적으로 지원하고 있다. 현재 널리 사용되고 있는 물리 엔진으로

는 Havok 5, PhysX, DMM 등의 상용 물리 엔진과 ODE, Tokamak, Chipmunk, Bullet 등의 오픈소스 물리엔진이 있다.

Havok 엔진은 상용물리엔진으로써 충돌 감지, 큰 충돌 메쉬(mesh)를 간결하게 표현할 수 있는 MOPP 기술, 제약조건(constraint)을 가지는 동역학 시뮬레이션, 차량 시뮬레이션(vehicle dynamics), 데이터 직렬화, 비주얼 디버깅 등을 제공한다. Havok 엔진이 사용된 게임으로는 “카운터 스트라이크 : 하프 라이프 2(Counter-Strike : Half-Life 2, 발브, 2004)”, “에이지 오브 엠파이어 3(Age of Empires 3, 앙상블 스튜디오, 2005)”, “엘더스 크롤4 : 오블리비언(The Elder Scroll IV : OBLIVION, 베테스다 소프트웨어, 2004)”, “오토어썰트(Auto Assault, 엔씨소프트, 2005)” 등이 있다.

Novodex에서 제작한 PhysX물리엔진은 Xbox 360에서 게임을 개발하는 용도로 사용할 때에는 라이선스 비용이 있지만 PC게임을 제작할 때에는 무료로 사용할 수 있다. PhysX엔진은 강체 시뮬레이션, 파티클 시스템, 유체 생성 및 시뮬레이션, 의복 시뮬레이션, 캐릭터 제어, 차량 시뮬레이션, 변형체 시뮬레이션, 3D 외력장(force field) 시뮬레이션 등을 지원한다. PhysX엔진을 사용한 게임으로는 에픽 게임스(Epic Games)의 “기어스 오브 워(Gears Of War, 2006)”와 “언리얼 토너먼트 3(Unreal Tournament 3, 2007)”, 그리고 “시티 오브 빌런스(City of Villains, 엔씨소프트, 2005)”, “레드 스틸(Red Steel, 유비소프트, 2006)” 등이 있다.

DMM은 상용물리엔진으로 대부분의 오픈소스 라이브러리에서 제공하지 않는 변형이



〈그림 10〉 게임 성능 트라이앵글

나 파손 같은 기능도 지원하며, 완성도 높은 상용 게임들에서 사용되고 있다. DMM이 지원하는 물리요소로는 강체 시뮬레이션, 파손, 의복 시뮬레이션, 로프(rope) 시뮬레이션, 배(sailboat) 시뮬레이션 등이 있다. DMM을 사용한 게임으로는 루카스아츠(LucasArts)의 “스타워즈 : 더 포스 언리쉬드(Star Wars : The Force Unleashed, 2007)”, “인디아나 존스 (Indiana Jones, 2007)” 등이 있다.

ODE(Open Dynamics Engine)는 오픈 소스 물리 엔진으로는 가장 널리 사용되고 있으며, 강체 역학을 시뮬레이션 할 때 높은 성능을 자랑한다. C/C++ API로 되어 있어서 플랫폼에 독립적이고 안정적이다. 또한 진보된 관절 종류들과 마찰을 고려한 충돌 감지를 제공하고 차량 시뮬레이션이나 가상 환경에서의 물체와 가상물체를 표현하는데 유용하다. ODE를 사용한 게임으로는 “스토커 : 쉐도우 오브 체르노빌(Stalker : Shadow of Chernobyl, GSC 게임월드, 2007)”와 “퀘이크 4(Quake 4, Id 소프트웨어, 2005)” 등이 있다. 스토커에서는 한 장면에 수백개의 물체가 나오기도 하며, 물체의 파손, 총알과 수류탄의

궤도, 폭풍파, 다관절체 애니메이션, 몬스터의 공중부양, 물체나 캐릭터를 관통 하는 총알 등이 효과적으로 표현되고 있다.

이외에도 Tokamak, Chipmunk, Bullet 등의 오픈소스 물리 엔진들이 있다. Tokamak 물리 엔진은 BSD 라이선스 규칙을 따르는 오픈소스 물리엔진으로서 강체 시뮬레이션, 정기구학 및 역기구학, 파티클 시스템과 파손 시뮬레이션 등을 지원한다. Chipmunk는 GDC 2005에서 발표된 논문을 기반으로 작성된 오픈소스 물리엔진으로써 이차원 강체 시뮬레이션을 지원한다. Bullet는 Zlib라이선스의 오픈소스 라이브러리며 강체 시뮬레이션 기능을 지원한다.

최신 3D 게임에서는 3D 그래픽 구현에 관한 전반적인 데이터 처리를 GPU가 담당하고, CPU는 많은 계산이 필요한 게임 AI와 논리 처리를 담당하고 있다. 따라서 화려한 시각적 효과를 가진 고사양의 게임을 즐길수록 CPU와 GPU에 부하가 걸리기 마련이다. 게임의 자유도나 인공지능이 높으면 높을수록 이러한 현상은 더욱 가속화된다. AGEIA가 개발한 PPU(Physics Processing Unit)는 이

러한 부하를 덜어 게임을 더욱 사실적으로 살아 숨 쉬게 할 수 있는 애드온 제품으로서 3D 폴리곤의 이동 및 주변 환경과의 상호 연산을 PPU가 전담하여 처리하도록 한다. 따라서 GPU는 PPU가 보조해 주는 만큼 3D 그래픽 데이터 처리에 대한 부하를 덜어낼 수 있는 장점이 있다. AGEIA의 PPU는 PhysX 물리 엔진을 가속화할 수 있으며, <그림 10>에서와 같이 “게임 성능 트라이앵글(gaming power triangle)”의 세번째 요소로서의 역할을 수행한다.

IV. 게임 물리 기술의 향후 전망

영화에 사용되는 고급 물리 기반 기술과 게임에 사용되는 실시간 물리 기반 기술이 다루고자 하는 근본적인 현상은 동일하다. 단지 사실성, 안정성, 속도를 종합적으로 고려할 때 영화에서는 사실성이 시뮬레이션 속도보다 더 중요하고 게임에서는 사실성 보다는 안정성 및 속도가 더 중요할 뿐이다. 하드웨어의 급격한 발전에 힘입어 오프라인과 온라인을 대표하던 영화와 게임의 시각적 품질 차이 또한 급속도로 줄어들어가고 있으며 이러한 추세가 더욱 가속화될 것으로 전망된다는 점을 감안할 때, 가까운 미래에는 영화와 게임에 사용되는 물리 기반 기술들 간의 경계마저도 모호해질 것으로 예상된다. 이는 “ACM SIGGRAPH 2007” 학회에서의 주목할 만한 두개의 코스(course) “LucasArts and ILM : A Case Study in Film and Game Convergence(Visual Effect)”와 “Crossing the Line : Moving from Film to

Games (and Possibly Back)”에서도 확인할 수 있다.

최근에 들어 게임개발자를 위한 학회 GDC(Game Developers Conference)가 창립되었고, ACM SIGGRAPH에서 게임개발자들을 위한 코스들의 숫자가 매년 증가하고 있다는 것을 감안한다면 게임물리분야에 대한 학계의 노력이 한층 커져가고 있다는 것을 알 수 있다. 학계의 이러한 노력이 게임업계의 노력과 맞물려서 미래의 게임물리기술은 지금까지 설명한 물리현상들뿐만 아니라 현재 여러 분야에서 활발히, 또한 새롭게 연구되고 있는 물리 기반 시뮬레이션을 포괄하여, 종래에는 영화에서와 같은 수준의 사실적인 캐릭터와 화려한 특수효과를 게임에서도 맞출 수 있게 할 것으로 기대된다.

저자소개



최민규

1996년 2월 한국과학기술원 전산학과 학사
 1998년 2월 한국과학기술원 전산학과 석사
 2003년 2월 한국과학기술원 전산학과 박사
 2003년 3월-2005년 2월 서울대학교 전기컴퓨터
 공학부 박사후연구원
 2005년 3월-2007년 2월 광운대학교 전임강사
 2007년 3월-현재 광운대학교 조교수

주관심 분야 : 변형체 애니메이션, 동작 계획, 동작 편집, 영상처리