

GPU를 활용한 고성능 연체 객체 시뮬레이션을 위한 조화진동 모델과 야코비 반복법 기반 수치 적분 기술

강영민

동명대학교 게임공학과

ymkang@tu.ac.kr

Numerical Integration based on Harmonic Oscillation and Jacobi Iteration
for Efficient Simulation of Soft Objects with GPU

Young-Min Kang

Dept. of Game Engineering, Tongmyong University

요 약

실시간 그래픽스 응용에서 연체의 움직임을 효율적으로 생성하기 위해 다양한 방법이 제안되었다. 연체 구성 요소들의 위상을 유지하기 위해서는 서로를 묶는 힘이 존재할 수밖에 없으며, 이는 강직도(stiffness)로서 수치적분의 시간간격의 크기를 제한하고 효율성을 떨어트린다. 이를 해결하기 위해 시간간격을 늘릴 수 있는 암시적 적분이 제안되었지만, 대규모 행렬이 포함된 선형시스템을 풀어야 해서 계산복잡도가 크게 높아진다. 이 문제를 개선한 근사 기법들은 댐핑 효과의 증가와 정확성의 손실을 초래할 수밖에 없다. 본 논문에서는 선형시스템을 풀지 않고도 안정성은 크게 높이기 위해 조화진동에 근거하여 스프링 힘을 적분하고, 이를 근사 암시적 기법과 결합하여 안정성을 극대화 하는 방법을 제안한다. 이 기법은 GPU를 통한 병렬화가 용이하여 거대한 규모를 가진 연체 객체의 움직임을 실시간에 생성할 수 있다.

ABSTRACT

Various methods have been proposed to efficiently animate the motion of soft objects in realtime. In order to maintain the topology between the elements of the objects, it is required to employ constraint forces, which limit the size of the time steps for the numerical integration and reduce the efficiency. To tackle this, an implicit method with larger steps was proposed. However, the method is, in essence, a linear system with a large matrix, of which solution requires heavy computations. Several approximate methods have been proposed, but the approximation is obtained with an increased damping and the loss of accuracy. In this paper, new integration method based on harmonic oscillation with better stability was proposed, and it was further stabilized with the hybridization with approximate implicit method. GPU parallelism can be easily implemented for the method, and large-scale soft objects can be simulated in realtime.

Keywords : Soft Objects, Implicit Integration, Harmonic Oscillation

Received: Sep. 10. 2018 Revised: Oct. 17. 2018

Accepted: Oct. 17. 2018

Corresponding Author: Young-Min Kang(Tongmyong University)

E-mail: ymkang@tu.ac.kr

© The Korea Game Society. All rights reserved. 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.

ISSN: 1598-4540 / eISSN: 2287-8211

1. 서 론

옷감과 같은 연체(soft object)의 동작을 생성하는 것은 사실적인 애니메이션을 위해 매우 중요한 기술이며, 다양한 시도가 제안되었다.

그래픽스 분야에서 옷감과 같은 연체의 움직임이 변형가능한 표면 문제로 모델링 되었고[1], 이후 이 문제를 해결하기 위한 다양한 방법이 제안되었다[2,3,4]. 그러나 이러한 기법들은 옷감 시뮬레이션을 위한 운동방정식이 갖고 있는 높은 수준의 경직도 때문에 수치적분을 위해 짧은 시간 간격을 사용해야 하는 문제를 극복하지 못 해 효율성 측면에서 한계를 가진다.

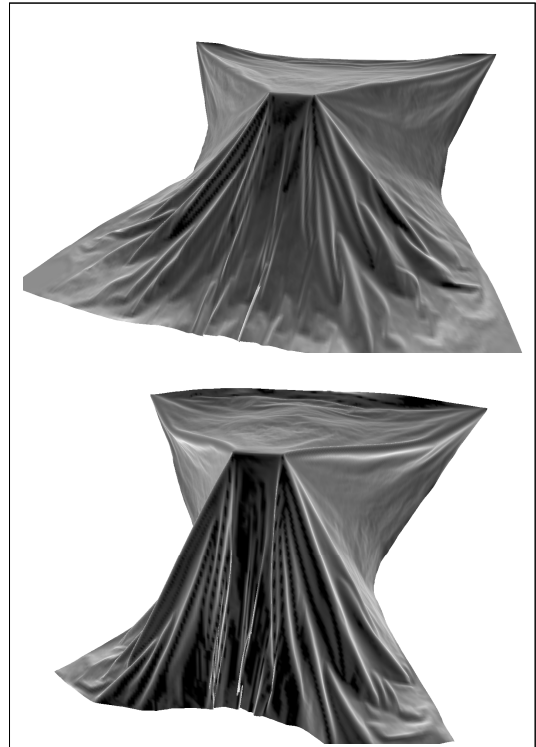
이러한 문제를 극복하기 위해 암시적 적분법을 이용하는 방법이 제안되었고[5], 이 방법의 효율성을 개선하기 위한 근사기법들도 제안되었다[6,7]. 암시적 적분법은 높은 수준의 안정성을 보장하지만, 연체 객체를 구성하는 기하 요소가 복잡해지면 계산량이 급격히 증가하며, 이를 계산한 근사 기법들은 정확성과 안정성이 떨어지는 문제가 있기 때문에 매우 복잡한 연체, 특히 경직도가 매우 높은 직물과 같은 객체의 실시간 애니메이션은 여전히 어려운 문제이다. 본 논문은 [Fig. 1]과 같이 높은 기하적 복잡도와 큰 경직도를 가진 연체 객체를 상호작용적 환경에서 효율적으로 애니메이션 할 수 있는 기법을 제안한다.

2. 안정적 연체 시뮬레이션

2.1 경직도에 의한 불안정성 문제

연체의 동작은 변형된 상태를 원래의 상태로 되돌리기 위한 복원력을 시뮬레이션 하는 것으로 이 복원력의 크기가 크면 클수록 시스템의 경직도가 높다고 한다. 높은 경직도는 시스템을 불안정하게 하는 것으로 잘 알려져 있다[8].

경직도가 높은 미분방정식을 안정적으로 적분하기 위해서는 매우 짧은 크기의 시간 간격을 할 수 밖에 없어 실시간 시뮬레이션을 어렵게 만든다.



[Fig. 1] Interactive Animation Result with Hybridization of Approximate Implicit Method and Force Integration with Harmonic Oscillation Model.

2.1.1 암시적 적분법에 의한 안정성 확보

수치해석에서 시간에 의존적인 미분 방정식의 수치적 근사해를 얻기 위해 사용하는 방법으로 명시적 기법(explicit method)와 암시적 기법(implicit method)이 있다. Y^t 가 현재 시간 t 에서의 상태라고 하고, 우리가 구하려는 미래의 상태를 Y^{t+h} 라고 하자. 이때, h 는 작은 시간 간격이다. 명시적 기법은 현재의 상태를 어떤 함수 Φ 에 단순히 입력해 해를 얻지만, 암시적 기법은 다음과 같은 방정식을 풀어 미래의 상태 Y^{t+h} 를 계산한다.

$$\Psi(Y^t, Y^{t+h}) = 0 \quad (\text{Eq. 1})$$

암시적 기법을 통한 수치적분은 불안정성 문제

를 일으키지 않는다. 스프링의 힘을 적분하여 속도를 갱신하고, 이 속도를 이용하여 위치를 갱신하기 위해 암시적 적분법의 일종인 역 오일러 방법(backward Euler method)은 다음과 같다.

$$\Delta \mathbf{v}^{t+h} = h \mathbf{M}^{-1} \mathbf{f}^{t+h} \quad (\text{Eq. 2})$$

이때, \mathbf{v} 는 속도, \mathbf{f} 는 속도 벡터이며, \mathbf{M} 은 질량 행렬이다. 그런데 Eq. 3에 나타난 힘 \mathbf{f}^{t+h} 는 시물레이션을 통해 얻은 미래의 상태로 결정되는 힘을 의미하기 때문에 아직 우리가 알지 못 하는 값이다. 따라서 이 값은 근사치를 구해서 사용할 수밖에 없다. 이를 위해 우리는 다음과 같이 테일러 급수를 이용하여 미래 힘의 근사치를 구한다.

$$\mathbf{f}^{t+h} = \mathbf{f}^t + \frac{\partial \mathbf{f}^t}{\partial \mathbf{x}} \Delta \mathbf{x}^{t+h} \quad (\text{Eq. 3})$$

이때, n 개의 입자에 작용하는 힘 벡터 \mathbf{f} 을 위치 벡터로 미분한 $\partial \mathbf{f}^t / \partial \mathbf{x}$ 는 3×3 크기의 행렬 \mathbf{J}_{ij} 가 $n \times n$ 개 있는 야코비(Jacobi) 행렬 \mathbf{J} 이다. 행렬의 i 행 j 열 요소는 i 질점에 작용하는 힘 \mathbf{f}_i 을 질점 j 의 위치 \mathbf{x}_j 로 미분한 값으로 i 에 연결된 모든 스프링이 만들어내는 힘의 총합이지만, \mathbf{x}_j 에 대한 편미분에서 고려될 힘은 스프링 (i, j) 에 의한 힘 \mathbf{f}_{ij} 뿐이다. 따라서 \mathbf{J}_{ij} 는 $\partial \mathbf{f}_{ij}^t / \partial \mathbf{x}_j$ 이다. \mathbf{J}_{ij} 는 \mathbb{R}^3 의 원소인 벡터를 \mathbb{R}^3 의 원소인 벡터로 미분한 것이고, 이것은 $\mathbb{R}^{3 \times 3}$ 의 원소인 행렬이 된다. \mathbf{J} 행렬은 이 \mathbf{J}_{ij} 를 $n \times n$ 개 가지고 있기 때문에 $\mathbf{J} \in \mathbb{R}^{3n \times 3n}$ 이 된다. 특히 \mathbf{J}_{ii} 는 이 행렬의 대각성분이 되는 $\mathbb{R}^{3 \times 3}$ 행렬이며, $\partial \mathbf{f}_i^t / \partial \mathbf{x}_i$ 이므로 다음과 같다. 아래 식에서 S 는 스프링의 집합을 의미한다.

$$\mathbf{J}_{ii} = \frac{\partial \mathbf{f}_i}{\partial \mathbf{x}_i} = \sum_{(i,j) \in S} \frac{\partial \mathbf{f}_{ij}}{\partial \mathbf{x}_i} = - \sum_{(i,j) \in S} \frac{\partial \mathbf{f}_{ji}}{\partial \mathbf{x}_i} = - \sum_{(i,j) \in S} \mathbf{J}_{ij} \quad (\text{Eq. 4})$$

그러면 속도의 변화는 다음과 같다.

$$\Delta \mathbf{v}^{t+h} = h \mathbf{M}^{-1} (\mathbf{f}^t + \mathbf{J} \Delta \mathbf{x}^{t+h}) \quad (\text{Eq. 5})$$

미래 상태에서 얻게 되는 입자들의 위치 $\Delta \mathbf{x}^{t+h}$

는 $h(\mathbf{v}^t + \Delta \mathbf{v}^{t+h})$ 로 표현할 수 있으므로, 이를 대입하여 정리하면 다음과 같은 선형 시스템의 해를 푸는 문제가 된다.

$$\Delta \mathbf{v}^{t+h} = (\mathbf{M} - h^2 \mathbf{J})^{-1} (h \mathbf{f}^t + h^2 \mathbf{J} \mathbf{v}^t) \quad (\text{Eq. 6})$$

이렇게 해서 얻는 속도 변화를 이용하여 입자들의 속도를 갱신하고, 이에 따라 위치를 갱신하면 불안정성 문제가 발생하지 않는다.

2.1.2 암시적 적분법의 한계

암시적 적분법을 통해 시스템의 불안정성을 해결할 수 있고, 큰 시간 간격을 적용하여 옷감과 같은 연체의 시물레이션 효율성을 개선할 수 있다. 하지만 Eq. 6에 나타난 선형 시스템은 $\mathbf{M} - h^2 \mathbf{J}$ 의 역행렬을 구하는 문제인데, 이 행렬의 크기가 $\mathbb{R}^{3n \times 3n}$ 이며, 질량 입자의 수 n 이 늘어나면 급격히 규모가 커져 계산이 어려워진다. 이러한 문제를 해결하고 선형 시스템의 해를 효율적으로 근사하기 위해 경직도만을 고려한 상수 행렬을 도입하고, 이 행렬의 역행렬을 필터(filter)로서 힘에 적용하는 방법이 제안되었다[9]. 하지만, 이 방법 역시 매우 큰 행렬 자체를 사용하지 않을 수는 없기 때문에 많은 수의 질점을 가진 연체에는 적용하기 어렵다.

3. 시물레이션 효율성 개선

암시적 적분법이 비록 안정성을 보장하기는 하지만, 그 계산의 복잡도 때문에 실시간 애니메이션에 적용하는 것이 어렵다. 이런 문제를 극복하기 위한 사실성 증대기법이 제안되기도 했다[10,11]. 그러나 적분 방법 자체를 개선하는 것이 중요하다.

3.1 근사 암시적 기법

암시적 기법은 Eq. 6에 나타난 선형시스템의 해를 푸는 것이다. 이때 $\mathbf{A} = (\mathbf{M} - h^2 \mathbf{J})$ 라 하고, $\mathbf{b} = h \mathbf{f}^t + h^2 \mathbf{J} \mathbf{v}^t$ 라고 하면 문제를 $\Delta \mathbf{v}^{t+h} = \mathbf{A}^{-1} \mathbf{b}$ 로

표현할 수 있다. 그런데 단순한 역행렬 계산법으로는 이 문제를 효율적으로 해결할 수 없다.

많은 경우 질량-스프링을 구성하는 입자의 수가 수 천 개에 이르기까지 하기 때문에 행렬의 크기가 매우 커지고, 이 행렬의 역행렬을 구하는 것이 실시간이나 효율적 시간 내에 이루어지지 않는다. 그런데 힘의 야코비 행렬인 \mathbf{J} 의 경우에는 3×3 크기의 부행렬(submatrix)을 하나의 성분으로 간주할 때 i 행 j 열 성분이 0행렬이 아니라면 질점 i 와 j 가 연결되어 있을 때 뿐이므로 \mathbf{A} 행렬은 희소행렬(sparse matrix)이다. 따라서 모두 n 개의 질점이 존재하고, 스프링의 개수가 e 라고 하면, \mathbf{J} 행렬에 0 행렬이 아닌 3×3 부행렬은 모두 $n+2e$ 개 존재한다. 이것은 일반적인 질량-스프링 모델을 고려할 때, \mathbf{J} 의 성분 개수인 $n \times n$ 보다 훨씬 적은 수이다. 또한 \mathbf{A} 의 i 번째 대각 성분은 질점 i 의 질량행렬인 $m_i \mathbf{I}_{33}$ 의 값에 $h^2 \mathbf{J}_{ii}$ 를 뺀 $m_i \mathbf{I}_{33} - h^2 \mathbf{J}_{ii}$ 의 값인데, \mathbf{J}_{ii} 의 값은 $-\sum_{(i,j) \in E} \mathbf{J}_{ij}$ 의 값이므로, 대각 성분은 $m_i \mathbf{I}_{33} + h^2 \sum_{(i,j) \in E} \mathbf{J}_{ij}$ 로 한 행에서 대각성분의 값이 다른 비대각성분의 값을 모두 더한 값에 질량이 더해진 형태이므로 대각성분이 선형시스템의 특징을 좌우한다고 할 수 있다.

만약 \mathbf{A} 와 대각행렬이라고 가정하면 \mathbf{A}^{-1} 을 구하기 위해 어렵게 역행렬 구하기 알고리즘을 사용할 필요가 없다. 대각행렬의 역행렬은 각 대각성분의 역수만 구하면 되기 때문이다. 다시 말해 속도의 변화 $\Delta \mathbf{v}$ 는 선형 시스템 풀이가 아니라 다음과 같이 각각의 질점 별로 개별적으로 계산이 가능하다.

$$\Delta \mathbf{v}_i = \mathbf{A}_{ii}^{-1} \mathbf{b}_i \quad (\text{Eq. 7})$$

문제는 \mathbf{A} 가 대각성분이 중요한 역할을 수행하는 행렬이기는 하지만, 완전한 대각행렬은 아니라는 점이다. 이럴 경우에 위의 방식은 실제 해에 근접한 근사해가 된다. 그러나 \mathbf{A} 가 대각행렬 \mathbf{D} 와 비대각성분으로 구성된 나머지 행렬 \mathbf{R} 로 구분했을 때, $\mathbf{A} = (\mathbf{D} + \mathbf{R})$ 라고 표현할 수 있다. 따라서 선형 시스템은 $(\mathbf{D} + \mathbf{R})\Delta \mathbf{v} = \mathbf{b}$ 로 표현할 수 있고, 다음과 같은 관계식을 얻을 수 있다.

$$\begin{aligned} \mathbf{D}\Delta \mathbf{v} + \mathbf{R}\Delta \mathbf{v} &= \mathbf{b} \\ \mathbf{D}\Delta \mathbf{v} &= \mathbf{b} - \mathbf{R}\Delta \mathbf{v} \\ \Delta \mathbf{v} &= \mathbf{D}^{-1}\mathbf{b} - \mathbf{D}^{-1}\mathbf{R}\Delta \mathbf{v} \end{aligned} \quad (\text{Eq. 8})$$

그러면 $\Delta \mathbf{v}_i = \mathbf{A}_{ii}^{-1} \mathbf{b}_i$ 는 $\mathbf{D}_{ii}^{-1} \mathbf{b}_i$ 와 같은 값이므로 이렇게 얻은 $\Delta \mathbf{v}$ 는 오차 $\epsilon = \mathbf{D}^{-1}\mathbf{R}\Delta \mathbf{v}$ 를 가진다. 이 오차는 \mathbf{R} 이 0 행렬에 가까워지면 0에 근접하게 된다. 그런데 이 \mathbf{R} 이 0행렬은 아니지만 \mathbf{D} 보다는 중요하지 않은 경우, 즉 대각성분이 특성을 좌우하는 대각지배행렬일 경우 야코비 반복법을 통해 오차를 줄여나갈 수 있다. 야코비 반복법은 우리가 구하려고 하는 벡터를 좌우변에 동시에 나타나게 한 다음 식을 이용한다.

$$\Delta \mathbf{v}^{(k+1)} = \mathbf{D}^{-1}(\mathbf{b} - \mathbf{R}\Delta \mathbf{v}^{(k)}) \quad (\text{Eq. 9})$$

즉 k 번째 반복을 통해 얻은 미지수 벡터가 $\Delta \mathbf{v}^{(k)}$ 라면 반복을 한 번 더 수행하여 $\Delta \mathbf{v}^{(k+1)}$ 을 얻을 수 있다. 이 방법에서 반복의 횟수가 k 라고 하면 $O(k(n+e))$ 의 시간 복잡도를 갖는다.

3.2 조화진동 기반 적분 안정성 개선

명시적 기법은 $\mathbf{v}_i^{t+h} = \mathbf{v}_i^t + h \mathbf{f}_i^t / m_i$ 와 같이 현재 우리가 알고 있는 미분 추정치를 사용하여 수치 적분하는데, 이 미분 추정치가 시간 간격 h 동안 일정하게 유지되지 않고, 변화하는 함수이기 때문에 부정확성을 초래한다. 그리고, 이 부정확성은 시스템을 불안정하게 만드는 원인이 된다.

암시적 적분법은 이를 해결하기 위해 테일러 전개를 통해 미래의 미분치를 근사하여 추정하고, 이를 적분에 이용하기 때문에 안정성을 확보할 수 있었다. 그러나, 암시적 기법은 선형시스템을 풀어야 하는 새로운 문제를 일으키는 것을 앞서 살펴 보았다.

만약 상태의 미분이 어떻게 변화하는지를 잘 알고, 시간간격 내에서 상태의 이 미분치를 더 정확하게 적분하는 방법이 있다면, 명시적 적분을 사용하더라도 안정성을 크게 높일 수 있을 것이다. 즉

다음과 같은 적분이 가능하다면 선형시스템 풀이 없이 안정적 시뮬레이션이 가능하다.

$$\mathbf{v}^{t+h} = \mathbf{v}^t + \frac{1}{m} \int_t^{t+h} \mathbf{f}(t) dt \quad (\text{Eq. 10})$$

이것은 결국 함수 $\mathbf{f}(t)$ 의 해석적으로 다룰 수 있는 닫힌 형태(closed form) 표현을 안다면 이를 해석적으로 적분하여 새로운 상태를 얻을 수 있다는 것이다.

훅(Hooke)의 법칙에 의해 계산되는 힘은 현재의 스프링 상태에서 얼마의 힘이 발생하는지는 알려주지만, 그것이 어떻게 변화할 것인지는 알려주지 않는다. 따라서 정확한 적분을 위해 훅의 법칙에 의해 결정되는 힘이 아니라, 스프링의 운동에 대한 해석에 기반한 힘 모델이 필요하다. 정적인 객체에 스프링으로 연결된 질량의 운동은 조화진동으로 잘 모델링되어 있다. 이러한 조화진동 기반 적분은 이미 제안된 바가 있지만, 댐핑을 고려하지 않아 안정성 개선에 한계가 있다[12].

3.2.1 댐핑이 없는 조화진동 모델의 적분

조화 진동의 단순한 형태는 정적인 객체에 질량 m 을 가진 질점이 스프링 강도 κ 의 스프링으로 연결되어 있을 때에, 이 질점이 1차원 공간에서 진동하는 것이다. 질점의 위치는 $x = A \sin \omega t$ 와 같은 식으로 표현할 수 있다. 이때 A 는 진동의 폭이며, ω 는 $\sqrt{\kappa/m}$ 로 결정된다.

우리가 다루는 문제는 하나의 질점이 정적 객체에 붙어 있는 것이 아니라, 운동하는 두 질점이 스프링의 양쪽에 붙어 있는 모델이다. 스프링 (i,j) 에 의해 두 질점 i 와 j 가 연결되어 있고, 각각의 위치를 \mathbf{x}_i 와 \mathbf{x}_j 로 표현했다. 속도는 \mathbf{v}_i 와 \mathbf{v}_j 이며, 각각의 질량은 m_i 와 m_j 로 표현된다.

벡터 $\mathbf{x}_j - \mathbf{x}_i$ 는 간략히 \mathbf{x}_{ij} 로 표현하고, 비슷하게 상대속도 역시 $\mathbf{v}_j - \mathbf{v}_i$ 를 \mathbf{v}_{ji} 로 표현하자. 질점의 위치는 시간에 대한 함수이므로 시간 t 에서의 질점 i 위치를 \mathbf{x}_i^t 와 같이 표현한다. 시간에 의존적인 다른

물리적 상태 역시 비슷한 방식으로 표현하자. (i,j) 스프링의 휴지상태 길이와 현재 시간 t 에서의 길이는 앞서 사용한 방식과 같이 l_{ij}^0 와 l_{ij}^t 로 표현된다. 이는 $|\mathbf{x}_{ij}|$ 와 같은 값이다.

어떤 스프링 (i,j) 가 일으키는 힘의 크기를 결정하는 핵심적 요소는 스프링의 변형 정도 δ_{ij}^t 이다. 이 값은 $l_{ij}^t - l_{ij}^0$ 의 값이다.

두 질점이 연결되어 있을 때에, 이 변형 정도는 조화진동 모델에 따라 다음과 같은 진동을 할 것이다.

$$\delta_{ij}^t = A_{ij} \sin \left(\sqrt{\frac{\kappa(m_i + m_j)}{m_i m_j}} t \right) \quad (\text{Eq. 11})$$

이 식에서 m_i 가 무한에 접근하면 ω_{ij} 는 $\sqrt{\kappa/m_j}$ 에 접근하고, 정적인 객체에 연결된 j 의 단순 조화진동으로 접근하게 된다. m_j 가 무한에 접근할 경우에는 질점 i 의 단순 조화진동이 된다. 따라서 $m_i m_j / (m_i + m_j)$ 를 스프링으로 연결된 두 질점 i 와 j 의 조화진동에서 질량으로 작용하는 M_{ij} 라고 볼 수 있다. 그러면 위의 진동은 다음과 같이 표현할 수 있다.

$$\delta_{ij}^t = A_{ij} \sin \sqrt{\frac{\kappa}{M_{ij}}} t \quad (\text{Eq. 12})$$

이 모델에서 결정되지 않은 것은 스프링 (i,j) 의 진동의 폭인 A_{ij} 이다. 이를 결정하기 위해서는 에너지 보존 법칙이 적용된다. 스프링의 전체 에너지는 운동 에너지와 위치 에너지로 구성된다. 진폭을 알면 스프링의 총 에너지를 알 수 있는데, 스프링 에너지 총합 E_{tot} 은 $\kappa A_{ij}^2 / 2$ 이다. 운동 에너지 E_k 는 변형속도 $\dot{\delta}_{ij}^t$ 에 의해 다음과 같이 계산된다.

$$E_k = \frac{1}{2} M_{ij} \dot{\delta}_{ij}^{t^2} = \frac{1}{2} \left(\frac{m_i m_j}{m_i + m_j} \right) \dot{\delta}_{ij}^{t^2} \quad (\text{Eq. 13})$$

위치 에너지 E_p 는 스프링 변형에 의해 결정되며

$E_p = \kappa \delta_{ij}^{t^2} / 2$ 이다. 따라서 에너지 보존 법칙에 따라 스프링의 총 에너지 E_{tot} 은 $E_k + E_p$ 와 일치하여야 하므로 다음을 얻는다.

$$\kappa A_{ij}^2 = \kappa \delta_{ij}^{t^2} + \left(\frac{m_i m_j}{m_i + m_j} \right) \dot{\delta}_{ij}^{t^2} \quad (\text{Eq. 14})$$

따라서 스프링 (i, j) 의 진폭은 다음과 같다.

$$A_{ij} = \sqrt{\delta_{ij}^{t^2} + \left(\frac{m_i m_j}{\kappa(m_i + m_j)} \right) \dot{\delta}_{ij}^{t^2}} \quad (\text{Eq. 15})$$

위의 식을 풀기 위해서는 스프링 변형의 미분치인 $\dot{\delta}_{ij}^t$ 를 알아야 한다. 이 값은 다음과 같다.

$$\frac{d}{dt} \delta_{ij}^t = \frac{d}{dt} (l_{ij}^t - l_{ij}^0) = \frac{d}{dt} l_{ij}^t \quad (\text{Eq. 16})$$

이 값은 다음과 같이 구할 수 있다.

$$\begin{aligned} \frac{d}{dt} l_{ij}^t &= \frac{d}{dt} \sqrt{\mathbf{x}_{ji}^t \mathbf{x}_{ji}^t} \\ &= \frac{1}{2} \mathbf{x}_{ji}^t \mathbf{x}_{ji}^t^{-\frac{1}{2}} \frac{d}{dt} \mathbf{x}_{ji}^t \mathbf{x}_{ji}^t \\ &= \frac{1}{2} \mathbf{x}_{ji}^t \mathbf{x}_{ji}^t^{-\frac{1}{2}} \cdot 2 \mathbf{x}_{ji}^t \mathbf{v}_{ji}^t \\ &= \hat{\mathbf{x}}_{ji}^t \mathbf{v}_{ji}^t \end{aligned} \quad (\text{Eq. 17})$$

이때 $\hat{\mathbf{x}}$ 는 \mathbf{x} 를 정규화한 벡터로 $\mathbf{x}/|\mathbf{x}|$ 를 의미한다. 이렇게 $\dot{\delta}_{ij}^t$ 를 계산하고 나면, 진폭을 쉽게 계산할 수 있다.

조화 진동은 두 개의 시간 변수 t 와 T 로 표현할 수 있다. 이 두 시간 변수를 이용하여 진동을 표현하면 다음과 같다.

$$\delta_{ij}^t = A_{ij} \sin \omega_{ij} T \quad (\text{Eq. 18})$$

이때 시간 변수 t 는 일반적인 시간이며, T 는 진동 주기 내에서 정의되는 시간으로서 0에서 $2\pi/\omega$ 의 범위 내에 있는 시간이다. 조화진동에 기반한 모델을 정확히 시뮬레이션하기 위해서는 T 변수를 이용하여 적분하여야 하며, 이 값은 다음 같다.

$$T = \frac{1}{\omega} \sin^{-1} \left(\frac{\delta_{ij}^t}{A_{ij}} \right) \quad (\text{Eq. 19})$$

이제 스프링 (i, j) 가 조화진동을 일으키기 위해 이 스프링 방향으로 생성되는 힘의 크기 f_{ij}^t 를 시간 t_0 에서 $t_0 + h$ 구간에서 다음과 같이 해석적으로 적분할 수 있다. 이때 f_{ij}^t 는 스프링 방향으로의 힘의 크기이므로 스칼라 값이다.

$$\begin{aligned} \int_{t_0}^{t_0+h} f_{ij}^t dt &= \int_{t_0}^{t_0+h} \kappa \delta_{ij}^t dt \\ &= \kappa A_{ij} \int_{T_0}^{T_0+h} \sin \omega_{ij} t dt \\ &= - \frac{\kappa A_{ij}}{\omega_{ij}} \cos \omega t \Big|_{T_0}^{T_0+h} \end{aligned} \quad (\text{Eq. 20})$$

이 적분을 이용하여 질량-스프링 모델을 더욱 정확히 시뮬레이션할 수 있다. 하나의 스프링 (i, j) 가 발생시키는 힘을 적분한 이 양을 ϕ_{ij} 라고 정의하면 이 값은 다음과 같이 계산할 수 있다.

$$\begin{aligned} \phi_{ij} &= \int_{T_0}^{T_0+h} f_{ij}^t dt \\ &= \frac{\kappa A_{ij}}{\omega} [\cos \omega (T_0 + h) - \cos \omega T_0] \end{aligned} \quad (\text{Eq. 21})$$

이 스프링이 발생시킨 힘의 적분은 각각의 질점 i, j 에 나뉘어 저야 하고, 각각의 질점 i, j 에 가해진 힘의 적분총량을 ϕ_i 와 ϕ_j 는 다음과 같이 ϕ_{ij} 로 얻어진 값을 가벼운 쪽이 더 많이 영향을 받도록 나눈 값을 두 질점을 묶고 있는 스프링 방향으로 서로 반대가 되도록 적용하면 된다. 따라서 특정 질점 i 에 가해지는 힘의 총량은 다음과 같이 계산할 수 있다. 이때 \mathbf{f}_i^{ext} 는 질점 i 에 대해 외부에서 가해지는 외력이다.

$$\phi_i = \sum_{(i,j) \in S} \frac{m_j}{m_i + m_j} \phi_{ij} \hat{\mathbf{x}}_{ij} + \mathbf{f}_i^{ext} h \quad (\text{Eq. 22})$$

각 질점의 속도 변화는 그 질점에 연결된 모든 스프링에 의해 생성되는 속도변화를 모은 것이기 때문에 다음과 같이 질점 i 의 속도 변화를 구할

수 있다. 이때, \mathbf{f}^{ext} 는 시스템 외부에서 가해지는 외력이다.

$$d\mathbf{v}_i = \frac{\boldsymbol{\phi}_i}{m_i} \quad (\text{Eq. 23})$$

이때 S 는 질량-스프링 모델의 스프링 집합이다. 이러한 방법으로 속도를 갱신할 경우 명시적 오일러 기법과 같이 현재 알려진 미분치를 그대로 수치적분하는 것에 비해 더욱 안정적인 결과를 얻을 수 있다. 그리고, 이 방법은 각각의 스프링 단위로 힘을 계산하듯이 속도의 변화를 계산하기 때문에 전체적인 시간 복잡도가 암시적 적분법에 비해 매우 낮은 장점이 있다. 이 방식은 명시적 기법이지만 안정적인 방법이라 할 수 있다.

3.2.1 댐핑 효과를 고려한 조화진동 모델

댐핑은 안정성 뿐만 아니라 에너지를 적절히 소산시키며 더욱 사실적인 동작을 생성하는 데에 필요하다. 댐핑이 포함될 경우 스프링 진동이 다음과 같다는 것이 잘 알려져 있다.

$$\delta_{ij}^t = A_{ij} e^{-\alpha_{ij} t} \sin \omega_{ij}^d t \quad (\text{Eq. 24})$$

이때, α 는 댐핑 힘에 비례하는 계수이며, ω_{ij}^d 는 댐핑으로 지연된 진동 주파수라고 할 수 있다.

우리는 우선 이 식에서 유도되는 속도의 변화 ϕ_{ij} 를 계산한 뒤에, α 와 ω_{ij}^d 의 값을 계산하는 법을 살펴볼 것이다. 이를 위해 우선 힘을 다음과 같이 적분하여야 한다.

$$\int_{t_0}^{t_0+h} f_{ij}^t dt = \kappa A_{ij} \int_{T_0}^{T_0+h} e^{-\alpha_{ij} t} \sin \omega_{ij}^d t dt \quad (\text{Eq. 25})$$

잘 알려진 미적분 기술을 이용하여 우리는 그 결과가 다음과 같음을 알 수 있다.

$$\left. \frac{\kappa A_{ij} e^{-\alpha_{ij} t}}{\alpha_{ij}^2 + \omega_{ij}^{d2}} (-\alpha_{ij} \sin \omega_{ij}^d t - \omega_{ij}^d \cos \omega_{ij}^d t) \right|_{T_0}^{T_0+h} \quad (\text{Eq. 26})$$

따라서, 스프링 (i,j) 가 발생시킨 힘의 크기를

적분한 결과는 다음과 같다.

$$\begin{aligned} & \kappa A_{ij} \frac{e^{-\alpha_{ij} T_0} e^{-\alpha_{ij} h}}{\alpha_{ij}^2 + \omega_{ij}^{d2}} [-\alpha_{ij} \sin(\omega_{ij}^d T_0 + \omega_{ij}^d h) \\ & \quad - \omega_{ij}^d \cos(\omega_{ij}^d T_0 + \omega_{ij}^d h)] \\ & - \kappa A_{ij} \frac{e^{\alpha_{ij} T_0}}{\alpha_{ij}^2 + \omega_{ij}^{d2}} [-\alpha_{ij} \sin \omega_{ij}^d T_0 - \omega_{ij}^d \cos \omega_{ij}^d h] \end{aligned} \quad (\text{Eq. 27})$$

간략한 표현을 위해 다음과 같이 C^t , C^h , s^t , s^h , c^t , c^h 를 정의하자.

$$\begin{aligned} C^t &= e^{-\alpha_{ij} T_0} / (\alpha_{ij}^2 + \omega_{ij}^{d2}) \\ C^h &= e^{-\alpha_{ij} h} / (\alpha_{ij}^2 + \omega_{ij}^{d2}) \\ s^t &= \sin \omega_{ij}^d T_0, \quad s^h = \sin \omega_{ij}^d h \\ c^t &= \cos \omega_{ij}^d T_0, \quad c^h = \cos \omega_{ij}^d h \end{aligned} \quad (\text{Eq. 28})$$

스프링 (i,j) 의 변형 속도 변화 ϕ_{ij} 는 이 정의를 이용하여 다음과 같이 표현할 수 있다.

$$\begin{aligned} \phi_{ij} &= -\kappa A C^t \\ & \quad (s^t [C^h \alpha_{ij} c^h - C^h \omega_{ij}^d s^h - \alpha_{ij}] \\ & \quad + c^t [C^h \alpha_{ij} s^h + C^h \omega_{ij}^d c^h - \omega_{ij}^d]) \end{aligned} \quad (\text{Eq. 29})$$

이제 이 값으로 질량-스프링 모델의 움직임을 시뮬레이션 할 수 있다. 그런데 이 식에서 아직 정해지지 않은 값은 α_{ij} 와 ω_{ij}^d 이다. 이를 알아야 스프링의 변형 속도를 계산할 수 있다. 댐핑이 포함된 단순 조화진동은 $\ddot{\mathbf{x}} + (c_d/m)\dot{\mathbf{x}} + (\kappa/m)\mathbf{x} = 0$ 이다.

이때 c_d 는 댐핑 계수이다. 이 미분방정식을 푸는 것은 잘 알려져 있다. 댐핑이 없을 경우에 진동의 각속도 ω 가 $\sqrt{\kappa/m}$ 으로 알려져 있다. 댐핑 적용 각속도 ω^d 를 구하기 위해 ξ 를 $c_d/2\sqrt{\kappa m}$ 으로 정의하면, 위의 미분방정식은 $\ddot{\mathbf{x}} + 2\xi\omega\dot{\mathbf{x}} + \omega^2\mathbf{x} = 0$ 와 같이 표현할 수 있다. 이 방정식의 해는 다음과 같다.

$$\mathbf{x}^t = A e^{-\xi\omega t} \cos \omega \sqrt{1-\xi^2} t \quad (\text{Eq. 30})$$

여기서 $\xi\omega$ 가 α 이며, $\omega\sqrt{1-\xi^2}$ 이 댐핑에 의해 지연된 각속도 ω^d 가 된다. 앞서 살펴본 바와 같이 두 개의 질량이 연결된 스프링에서의 진동 각속도 ω 가 $\sqrt{\kappa/M_{ij}}$ 이고, M 은 $m_i m_j / (m_i + m_j)$ 이다.

ξ 값은 두 질량이 스프링에 미치는 질량 효과인 M 을 $c_d/2\sqrt{\kappa m}$ 의 m 에 적용하여 구할 수 있다. 따라서 특정 스프링 (i, j) 의 α 와 ω^d 의 값은 아래와 같이 구할 수 있다.

$$\xi_{ij} = \frac{c_d}{2\sqrt{\kappa M_{ij}}} \quad (\text{Eq. 31})$$

$$\alpha_{ij} = \xi_{ij}\omega_{ij} = \frac{c_d}{2\sqrt{\kappa M_{ij}}} \sqrt{\frac{k}{M_{ij}}} = \frac{c_d}{2M_{ij}}$$

$$\omega_{ij}^d = \sqrt{\kappa/M_{ij}} \sqrt{1 - \frac{c_d^2}{4\kappa M_{ij}}}$$

3.4 조화진동과 자코비 반복법의 혼합

조화진동 모델은 오일러 기법과 같은 전통적인 명시적 기법에 힘의 적분을 더 나은 방식으로 수행하기 때문에 안정성이 다소 높아진다. 그러나, 이렇게 해서 얻어지는 안정성은 암시적 적분법에 비해 크게 떨어진다. 반면 암시적 기법은 그 안정성에도 불구하고 메시의 구조가 복잡해질 경우 시스템에 포함된 행렬의 크기가 너무 커지고, 이 행렬의 역행렬을 구하는 것이 쉽지 않게 된다. 이러한 한계를 개선하기 위해 야코비 반복법에 기반한 근사 암시적 기법을 Eq. 9과 같이 적용할 수 있다. 이때, Eq. 9에 나타나는 \mathbf{b} 는 현재 시간 t 에서의 힘 \mathbf{f}' 와 점성력 $h\mathbf{J}\mathbf{v}$ 를 단순히 시간간격 h 로 오일러 적분한 것이다. 이 값을 더 좋은 값으로 개선하면 더 좋은 적분 결과를 얻을 수 있을 것이다. 이를 위해 기존의 \mathbf{b} 와 조화진동 기반 모델에서 구한 ϕ 의 가중합을 구한다. 간단한 방법은 각각의 가중치를 0.5로 두는 것이다. 따라서, 각 질점의 속도 변화는 다음과 같이 구할 수 있다.

$$\Delta \mathbf{v}_i = \mathbf{D}_{ii}^{-1} \tilde{\mathbf{b}} \quad (\text{Eq. 32})$$

$$= (\mathbf{M}_{ii} - h^2 \mathbf{J}_{ii}^* - h \mathbf{J}_{ii}^*)^{-1} [$$

$$\frac{1}{2} (\mathbf{f}'_i + h^2 \mathbf{J}_{ii}^* \mathbf{v}_i + h^2 \sum_{(i,j) \in S} \mathbf{J}_{ij}^* \mathbf{v}_j) +$$

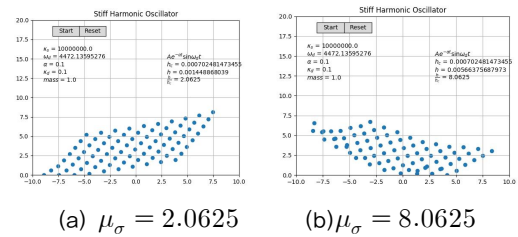
$$\frac{1}{2} (\phi_i + \mathbf{f}_i^{ext})$$

$$]$$

이 기법의 장점은 각 입자 상태의 갱신을 독립적으로 수행할 수 있어 명시적 기법처럼 병렬화가 용이하기 때문에 GPU를 활용한 병렬처리를 쉽게 구현할 수 있다는 것이다.

4. 실험결과

스프링의 진동 각속도가 ω 일 경우, 단순한 수치 적분은 π/ω 보다 작은 시간간격이 필요하다. 이 값을 임계시간 h_c 로 정의하였다. 시스템의 안정성은 사용된 시간간격 h 가 이 임계시간보다 얼마나 큰지를 평가하는 $\mu_\sigma = h/h_c$ 로 정의하였다.



[Fig. 2] Stability of Force Integration based on Harmonic Oscillation Model

안정성 실험의 결과가 [Fig. 2]에 나타나 있으며, 임계시간보다 훨씬 큰 시간간격인 $u_\sigma = 2.0625$ 와 $\mu_\sigma = 8.0625$ 로 크게 설정된 상태에서도 안정적인 시뮬레이션이 가능했다.

[Table 1]은 최대경직도 κ_{\max} 를 측정된 것이다. c_1 은 2,500 개의 질점을 가진 $2m^2$ 면적의 천이 40 kg의 질량, 0.5의 댐핑 상수와 0.1msec의 시간간격으로 시뮬레이션한 것이고, c_2 는 질점의 수와 댐핑상수를 각각 10,000 개와 3.5로 높였다. c_3 는 질점의 수가 40,000 개이며 댐핑 상수는 1.5이다. 명시적 오일러는 E, 조화진동 모델은 HO, 야코비 반복 3회를 적용한 근사 암시적 기법은 JI, 암시적 적분법은 IM, 조화진동과 야코비 반복법의 혼합 기법은 HY로 표현했다. 해당 기법이 시뮬레이션을 수행할 수 있는 최대경직도 κ_{\max} 가 측정되었다. 암시적 기법은 안정성이 보장되므로 제외하였다.

[Table 1] κ_{\max} Comparison Results

C	E	HO	JI	HY
c_1	9.2×10^5	1.1×10^6	9.8×10^5	2.7×10^6
c_2	1.1×10^5	3.1×10^5	3.6×10^5	1.1×10^6
c_3	5.9×10^3	1.0×10^5	1.1×10^5	5.0×10^5

[Table 2] Update Time Comparison Result

입자	HO	JI	IM	HY
20×20	2.02	4.06	19.48	4.46
30×30	2.43	4.16	86.90	4.56
40×40	2.46	4.15	252.55	4.55

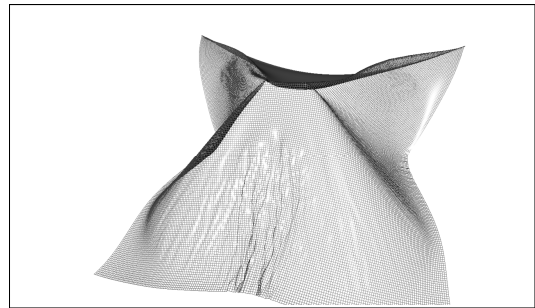
[Table 3] Tolerance for Large Steps (h)

입자	stiffness			
	512	1024	2048	4096
E	2.5×10^{-3}	1.8×10^{-3}	1.3×10^{-3}	1.0×10^{-3}
HY	1.3×10^{-2}	1.0×10^{-2}	7.6×10^{-3}	6.5×10^{-3}

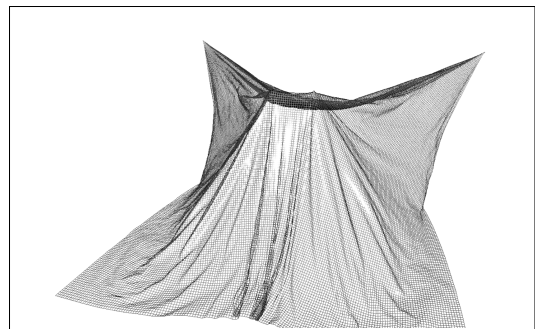
[Table 1]은 각각의 기법이 주어진 입자수의 상태를 갱신하는 데에 소요되는 시간을 비교한 것이다. 하이브리드 기법은 명시적 기법의 상태 갱신과 같은 수준의 계산복잡도를 가져서 표에 나타난 바와 같은 효율적 성능을 보인다. 이 표의 숫자는 같은 조건에 대해 오일러 적분이 사용하는 시간을 1로 보았을 때의 상대적 소요시간이다.

[Table 2]는 단순 오일러 기법(E)과 야코비 반복에 조화진동 모델을 혼합한 기법(HY)에 대해 경직도를 변화시켰을 때 허용되는 시간간격을 비교한 것이다. 혼합기법이 더욱 높은 안정성을 보인다.

[Fig. 3]은 제안된 기법이 성능상의 잇점 뿐만 아니라 생성된 애니메이션이 표현하는 연체의 사실성도 크게 개선할 수 있음을 보여준다. 그림의 (a)는 명시적 오일러 적분법을 사용하여 40,000 개의 절점을 가진 연체를 경직도 $\kappa=5,000$ 으로 시뮬레이션 한 것이고, (b)는 조화진동 모델과 야코비 반복법에 기반하여 동일한 연체에 $\kappa=500,000$ 을 적용한 결과이다. 제안된 기법은 더 높은 경직도를 시뮬레이션 할 수 있을 뿐만 아니라 더욱 자연스러운 주름을 생성할 수 있다.



(a) Result of Explicit Euler Method



(b) Result of Harmonic Oscillation Method

[Fig. 3] Comparison of Wrinkles of Explicit Euler and Harmonic Oscillation Model

5. 결론

본 연구를 통해 기존의 명시적 오일러 적분보다 안정성이 크게 개선할 수 있으며, 암시적 적분법이 가진 계산 복잡도의 한계를 극복할 수 있는 새로운 수치적분 모델을 제안하였다. 제안된 기법은 매우 복잡한 연체가 큰 경직도를 가지고 있는 상황에서도 빠르게 상태를 갱신할 수 있으며, GPU 등을 활용한 병렬화가 용이한 명시적 상태 갱신 방식으로 구현된다. 야코비 반복법은 각 입자에 대해 스프링을 통해 인접한 입자만을 고려하여 상태를 갱신할 수 있으며, 조화진동 기법 역시 하나의 입자에 연결된 스프링만 국지적으로 고려하면 상태를 갱신할 수 있다. 따라서 이 두 기법은 GPU와 같은 데이터 병렬처리 장치에서 간단히 입자별 시뮬레이션을 구현할 수 있다. 이 두 기법의 장점을 혼

합한 하이브리드(hybrid) 기법 역시 같은 방식으로 병렬화할 수 있어 실시간 응용에 적용할 수 있다.

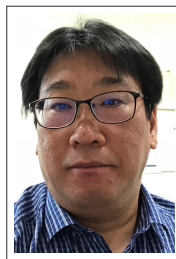
실험을 통해 확인할 수 있는 바와 같이 이 기법은 단순히 안정성이 높을 뿐만 아니라 생성된 애니메이션 결과가 자연스러운 옷감 주름을 생성할 수 있어 다양한 주름을 통해 옷감의 질점을 더욱 자연스럽게 생성할 수 있다. 따라서 이 기법은 게임이나 가상현실과 같은 실시간 환경에서 매우 유용한 연체 시물레이션 기법을 제공할 수 있다.

ACKNOWLEDGMENTS

이 논문은 2016학년도 동명대학교 교내학술연구비 지원에 의하여 연구되었음(2016A010-1).

REFERENCES

- [1] D. Terzopoulos, J. Platt, A. Barr, Fleischer K. "Elastically deformable models". Computer Graphics (Proc. of SIGGRAPH 87) 1987; 21(4): 205 - 214.
- [2] P. Volino, N. Magnenat-Thalmann, S. Jianhua, and D. Thalmann. "An evolving system for simulating clothes on virtual actors". IEEE Computer Graphics & Applications, 16(5):42 - 51, September 1996.
- [3] M. Meyer, G. DeBunne, M. Desbrun, and A. H. Bar. "Interactive animation of cloth-like objects in virtual reality". The Journal of Visualization and Computer Animation, 12:1 - 12, 2001.
- [4] X. Provat. "Deformation constraints in a massspring model to describe rigid cloth behavior". Graphics Interface '95, pages 147 - 154, May 1995.
- [5] D. Baraff and A. Witkin. "Large steps in cloth simulation". Proc. of SIGGRAPH 98, pages 43-54, July 1998.
- [6] M. Desbrun and M.-P. Gascuel. "Animating soft substances with implicit surfaces". Proc. of SIGGRAPH 95, pages 287 - 290, August 1995.
- [7] Y.-M. Kang and H.-G. Cho. "Real-time animation of complex virtual cloth with physical plausibility and numerical stability". Presence - Teleoperators and Virtual Environments, 13(6):668 - 680, MIT Press. 2004.
- [8] Y.-M. Kang. "Real-time Cloth Animation with Physical Plausibility and Numerical Stability", Ph.D Thesis, Pusan National University. 2003.2.
- [9] M. Desbrun, P. Schröder, A. Barr. "Interactive animation of structured deformable objects", In Graphics Interface '99, Los Angeles, CA, 1999; 1 - 8.
- [10] Y.-M. Kang. "Realistic Rendering of Woven Surface using Procedural Bump Mapping", Journal of Korea Computer Game Society, 10(3):103-120, Korea Computer Game Society, 2010.6.
- [11] Y.-M. Kang. "Adaptive Mesh Structure for Realtime Paper Crumple Simulation", Journal of Korea Computer Game Society, 9(4):97-105, Korea Computer Game Society, 2009.8.
- [12] Y.-M. Kang and C.-S. Cho. Photorealistic Cloth in Realtime Applications, Computer Animation and Virtual Worlds, DOI:10.1002/cav.1456, 2012.



강영민 (Kang, Young-Min)

약 력 : 1996년 부산대학교 전산학과 이학사
 1999년 부산대학교 전산학과 이학석사
 2002년 제네바대학교 미라랩 임시연구원
 2003년 부산대학교 전산학과 이학박사
 2003년-2005년 한국전자통신연구원 연구원
 2005년-현재 동명대학교 게임공학과 교수

관심분야 : 컴퓨터 그래픽스, GPU 병렬 처리