
네트워크 기반 햅틱 협업 시스템의 네트워크 적응적 물체 강도 조절 기법

Network-adaptive Control of Object Stiffness for Networked Haptic Collaboration System

손석호, Seokho Son, 이석희, Seokhee Lee, 김종원, JongWon Kim
광주과학기술원 정보기전공학부, 네트워크미디어 연구실

요약 햅틱 협업을 위한 네트워크는 기본적으로 지연, 지터, 손실의 제약을 가진다. 햅틱은 정보의 속도에 민감하므로 네트워크 환경에서 협업을 이루어내기에는 많은 제약이 있다. 특히 협업의 품질을 감소시키는 네트워크 지연을 보상하기 위한 연구가 필요하다. 본 논문에서는 물체의 좌표 전송을 기반으로 한 햅틱 협업이 높은 수준의 지연시간을 지닌 네트워크에서 발생할 수 있는 문제를 정의하고 그 원인을 파악하여 안정적인 협업을 유지하기 위하여 네트워크 지연 문제를 보상할 수 있는 기법을 제시한다. 네트워크 지연에 의해서 물체를 밀 때 더 많은 힘을 사용되는 현상과 클라이언트들이 물체를 들어올릴 때 물체가 진동하는 현상이 발생된다. 이 문제를 해결하기 위해 물체의 강도를 변경하는 방법을 제안한다. 지연 시간의 수준이 증가하여도 지연 문제가 발생하지 않게 함과 동시에 최대한 원래 물체 강도를 유지하기 위해서 클라이언트가 물체에 발생시키고자 하는 힘과 서버에서 물체에 발생시키는 힘을 같도록 만드는 수식을 유도한다. 이 수식을 이용하여 지연의 크기에 관계없이 클라이언트가 물체의 위치를 제어할 수 있다. 지연 보상 기법을 통해 햅틱 미디어의 품질을 유지하면서 지연에 의한 문제를 해결하는 방안을 제시하며, 실제 실험을 통하여 결과를 확인한다.

핵심어: *Haptic collaboration, Haptic delay, stiffness modification, shared object*

1. 서론

네트워크를 통해 햅틱 협업을 이루어 내기 위해 많은 연구들이 진행되고 있지만 현실적인 느낌대로 서로 촉감을 공유하고 상호작용하는 것은 아직까지 많은 제약이 있다. 햅틱은 지터, 손실, 지연에 민감하여 햅틱 장비의 진동, 비정상적인 힘의 반환 등 많은 문제가 발생할 수 있다. 이러한 햅틱의 특성 때문에 지연, 지터, 손실 등이 존재하는 현재의 네트워크에서는 햅틱의 민감한 특성을 만족시켜주기 힘들다.

햅틱의 특성을 만족시키기 위해서는 언급된 네트워크의 세 가지 문제점을 해결하여야 한다. 햅틱에서 네트워크의 문제를 해결하기 위한 연구로써 지터 문제 해결을 위한 버퍼 사용연구 [1], 패킷 손실률을 감소시키기 위해서 FEC (forward error collection) 기법을 적용하는 [2] 등 다양한 측면에서의 연구가 진행되고 있다. 나머지 요소인 네트워크 지연에 대한 문제를 해결하기 위한 연구의 예로써 M. Fujimoto는 단일 클라이언트를 위한 네트워크 기반 햅틱의 지연문제를 해결하기 위해 동적 물체 강도를 변경하는 연구를 예로 들 수 있다 [3]. M. Fujimoto는 일정한 지연이 발

생하면 임의로 측정된 상수를 지연으로 나눈 값을 물체의 강도로 설정한다. 하지만 지연이 발생한 상황과 발생하지 않았을 때의 역감을 서로 같게 만들기 위해서는 더 정확한 물체 강도변경이 필요하다.

본 연구의 목적은 햅틱의 요구사항 중에서 지연 문제를 해결하는 데에 초점을 둔다. 지연 문제를 두 가지로 정의하고 문제의 발생 원인을 밝히며, 이 두 가지 문제를 보상하기 위해 지연 크기에 따라 동적으로 물체의 강도를 바꾸는 방법에 대한 연구를 진행하였다. 기존에 제안된 단순 상수 값을 이용한 물체 강도 변경 방법은 햅틱 미디어의 품질 저하 문제가 있음을 주시하고 햅틱 협업 품질을 유지함과 동시에 햅틱 미디어의 품질 또한 같이 유지 시킬 수 있게 지연에 의해서 발생하는 서버와 클라이언트 사이의 물체위치 차이를 최소화 시킬 수 있는 물체 강도를 찾는다. 본 연구에서는 서버와 클라이언트 사이의 힘을 같게 만드는 물체 강도 변경 기법을 사용하여 클라이언트에서 발생하였던 지연문제를 해결함과 동시에 햅틱 미디어의 품질을 최대한 유지하는 기법을 제안한다.

본 논문은 다음과 같이 구성된다. 2절에서는 네트워크 지연에 따른 문제점을 정의하고 그 원인을 밝히며, 3절에서는 알맞은 물체 강도변경을 위해 서버와 클라이언트의 물체 힘의 비교를 기반으로 동적물체 강도변경 기법을 제안한다. 4절은 제안된 기법의 유용성을 실험을 통해서 입증하고 마지막 5절에서는 결론 및 향후 연구에 대해서 언급한다.

2. 네트워크 지연에 따른 문제점

네트워크 지연은 단순히 물체 위치 차이뿐 아니라 협업의 품질을 떨어뜨리는 현상을 일으킨다. 이 현상 중 주요한 두 가지 문제를 정의하고 그 원인을 밝힌다.

2.1 서버와 클라이언트 구조의 시스템에서 지연 문제

햅틱 미디어 전송에는 미디어의 종류에 따라 크게 두 가지 구조로 나뉜다. 한 가지 구조는 물체의 위치 전송이고 나머지 하나는 힘 정보 자체를 전송하는 것이다. 물체의 위치 전송에는 주로 공유된 물체를 통한 협업 시스템 등이 있으며[5], 힘 정보 자체를 전송하는 경우는 기계 원격 제어, 원격 수술 등이 있다[6]. 본 논문은 물체의 위치가 전송되는 구조의 시스템을 따른다. 물체의 위치를 공유하며 협업하기 위해서는 서버 클라이언트 모델과 피어 투 피어 모델을 생각할 수 있다. 본 논문에서는 서버 클라이언트 모델을 이용할 때 생기는 문제 해결에 주안을 두고 있다. 그림 1은 서버 클라이언트 모델 햅틱 협업 시스템을 구조화 한 것이다. 클라이언트들은 각 사용자가 제어하는 자신의 HIP (Haptic Interaction Point)를 서버에게 전송한다. 서버는 클라이언트들의 HIP 정보를 이용하여 물체의 위치를 물리적 모델을 기반으로 계산하고 계산된 물체의 위치를 다시 클라이언트들에게 전송한다. 클라이언트들은 별도의 물체위치 계산 없이 서버에서 전송되어온 물체 정보를 통해 자신의 물체위치를 갱신시킨다.

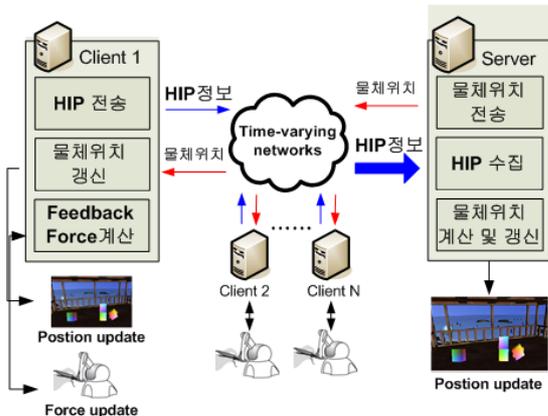


그림 1. 서버, 클라이언트 구조의 햅틱 협업 시스템.

물체의 위치를 통해 협업하는 서버 클라이언트 구조의 협업 시스템은 네트워크의 시간지연에 따라 햅틱 협업 품질을 심각히 저하 시키는 두 가지 현상이 발생한다.

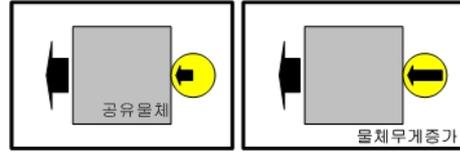


그림 2. 지연에 의한 물체무게증가.

지연에 의한 첫 번째 문제는 클라이언트에서 느끼는 비현실적 무게감이다. 이 문제는 그림 2와 같이 클라이언트의 사용자가 HIP를 통해 물체를 움직이고자 할 때 일어난다. 클라이언트가 처음 물체를 밀기 시작했을 때, 이미 서버에서는 클라이언트의 HIP와 서버의 물체 좌표를 통해 물체의 이동을 계산한다. 하지만 계산된 정보는 지연에 의하여 클라이언트에 늦게 반영된다. 즉, 클라이언트가 물체를 움직였지만 사용자가 움직일 때 들인 힘만큼 물체의 위치가 바로 반영되지 않으므로 클라이언트에서 물체는 지연시간이 끝나는 시점까지 정지해 있게 된다. 협업 주체는 물체를 밀기 위해 지속적으로 힘을 가하고 있다. 물체를 밀면 그 힘만큼 움직여야 하지만 지연에 의하여 물체가 움직이지 않으므로 물체가 더 무거워진 듯한 비정상적인 힘을 인지하게 된다. 지연이 끝나는 시점에서 서버에서 전송되어오는 정보와 같이 클라이언트도 물체의 위치를 갱신하게 되고 클라이언트의 사용자는 물체에 힘을 가하고 있는 상태이므로 서버에서는 다시 물체의 위치를 갱신하며, 이 정보는 클라이언트에게 지연시간이 끝난 이후에 전달되어 처음의 상황으로 돌아가 문제가 반복된다. 결국 물체는 가다 서다를 반복하게 된다. 이 현상을 연속적으로 관찰하면 결과적으로 사용자가 물체를 밀는 동안 전체 요구되는 힘이 증가하게 된다고 할 수 있다.

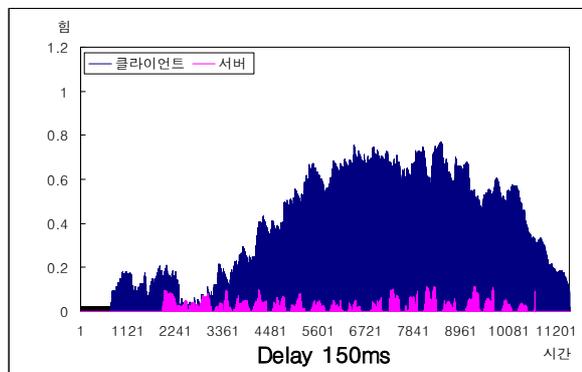


그림 3. 지연에 의한 서버와 클라이언트의 힘 계산 차이.

그림 3은 단 방향 지연이 150ms 일 때, 클라이언트가

물체를 미는 실험의 결과이다. 각 서버와 클라이언트가 물체를 움직일 때 사용한 힘을 측정하였다. 이 실험에서 클라이언트는 물체를 특정 위치로 밀기위해 많은 힘을 주어야한다. 클라이언트는 서버에서 계산한 물체 위치정보를 통해 물체 위치를 갱신하므로 서버와 클라이언트의 물체 위치정보는 지연시간 이후부터 같아진다. 그러므로 물체를 움직인 거리는 클라이언트와 서버가 같지만 서버에서 계산된 힘은 클라이언트가 준 힘의 크기와는 다름을 직관적으로 확인할 수 있다. 클라이언트는 자신이 원하는 만큼 물체를 밀기 위해서 훨씬 더 큰 힘을 주어야 한다. 서버에서 물체에 계산되는 힘을 관찰해 보면 힘의 발생이 불연속적임을 알 수 있다. 이 현상은 앞에서 설명한 비현실적 무게감의 원인이 서버와 클라이언트의 물체 위치 차이라는 것을 보여준다. 서버에서는 아직 클라이언트의 HIP가 서버의 물체에 닿지 않았기 때문에 물체의 움직임을 가다 서다를 반복함을 확인할 수 있다. 150ms에서의 실험으로 단적인 예를 보였으나 이 문제는 지연이 커질수록 더 큰 문제를 일으킴을 짐작할 수 있다.

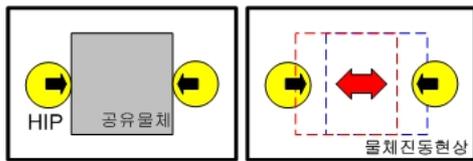


그림 4 지연에 의한 물체진동현상.

지연에 의한 두 번째 문제는 협업 상황에서 생기는 문제로서 그림 4와 같이 여러 협업 주체들이 함께 물체를 들어올릴 때 발생된다. 여러 주체가 물체를 들어올리기 위해서 주체들은 물체의 중심을 향해 서로 다른 방향에서 물체를 밀게 된다. 또한 물체를 들어올리는 동안 물체의 고도가 유지되도록 하기 위해서 클라이언트는 계속해서 HIP를 물체에 밀착 시키고 있어야한다. 하지만 이 상황에서 지연에 의해 물체의 위치 차이가 발생하게 되면 클라이언트에서는 HIP를 물체에 밀착 시키고 있다고 해도 서버에서는 물체와 클라이언트의 HIP 사이에 틈이 발생한다. 이 틈에 의해서 물체는 한쪽에서만 밀고 있는 듯한 불안정한 상태를 보이게 된다. 지연시간이 끝나고 물체의 위치가 클라이언트에게 전송될 때는 클라이언트가 물체와 밀착상태를 유지하려 하므로 HIP를 물체의 안쪽으로 더 밀게 되는데 이 경우 앞의 불안정 원인과 반대로 서버에서는 너무 큰 힘을 물체에 적용하게 된다. 결국 밀고 있는 행동은 동일하지만 물체에 주어지는 힘의 크기는 작아졌다가 커졌다가를 반복하게 되고, 이 효과는 물체의 진동으로 나타나게 된다. 정의된 두 가지 문제 모두 지연에 의한 서버와 클라이언트사이의 물체 위치 차이에 의해 발생하는 것이다. 이 두 가지 문제는 네트워크 지연이 커질수록 악화된다.

본 논문은 물체의 위치를 전송 시 네트워크 지연에 의해 발생하는 문제를 해결하는 데에 초점을 둔다. 지연문제를 해결하기 위한 방법으로 직접 네트워크를 제어하는 연구보다 햅틱의 렌더링의 성질을 이용하여 지연을 보상하는 위치에 있다. 또한 클라이언트들이 물체를 들어올리는 상황을 고려하여 햅틱 협업 시에 생길 수 있는 문제도 문제영역에 포함된다.

위에서 밝힌 네트워크 지연에 의한 두 가지 주요한 현상을 해결하기 위해서 본 논문은 물체의 강도를 변경하는 방법을 이용한다. 물체에 계산되는 힘은 물체의 위치를 수정하기 위해서 사용되는데 이 힘은 물체의 강도를 통해 계산된다. 그러므로 물체의 강도를 조절하면 물체에 적용되는 힘을 조절할 수 있으므로 지연에 따른 강도 변경을 통해 지연에 따라 힘을 수정할 수 있다. 2.2절에서 관련연구에 대해 논한 다음 3절에서는 지연과 물체 속성에 적절한 물체 강도 변경 기법에 대하여 설명한다.

2.2 관련연구

서론에서 언급한 바와 같이 M. Fujimoto는 한 클라이언트에서 물체의 밑면을 밀어 올릴 때 네트워크 지연에 의한 물체 진동 현상을 문제로 인식하고, 이 문제를 해결하기 위해서 물체의 강도를 변경하는 방법을 제안하였다 [3]. 협업 상황을 고려하지 않은 것을 제외하면 문제의 영역은 거의 동일하며, 한 클라이언트가 물체를 위로 들어올리는 행동은 앞서 밝힌 바와 같이 두 클라이언트가 협업을 통해 물체를 들어올릴 때 발생하는 문제와 같은 물체 진동 문제를 일으킨다. 하지만 한 클라이언트가 물체를 위로 밀어 올리는 일은 드문 일이며 물체 진동 문제뿐만 아니라 물체를 밀 때 발생하는 비이상적 힘도 큰 영향을 끼치는 문제중 하나이므로 앞서 밝힌 바와 같이 문제 정의를 명확히 할 필요성이 있다.

M. Fujimoto는 물체의 강도를 변경하는 방법으로 특정 상수를 네트워크 지연으로 나누어서 물체의 강도로 이용하는 기법을 이용한다. 이 방법은 네트워크 지연에 의한 물체 진동 효과를 완화 시킬 수 있다. 하지만 단순히 지연의 크기에 따라 물체 계수를 수정한다면 서버에서 클라이언트에 전달될 원하는 대부분의 물체의 속성이 무시되고 단순히 지연에 의해 힘이 결정된다. 이는 결국 햅틱 협업의 품질을 떨어뜨리게 된다. 또한 적절한 상수 값이 없으므로 물체의 크기나 무게와 같은 속성이 변할 때마다 다시 상수 값을 측정해줘야 할 필요성이 있다. 본 논문에서는 지연 값에 의한 단순 물체 강도 변경 방법에 문제가 있음을 주목하고, 본래 물체 강도와 물체 속성들을 최대한 유지하면서 지연문제를 해결하는 방법을 제안한다.

3. 제안된 동적 물체 강도변경 기법

3.1 피드백 힘 계산 모델

동적 물체 강도변경 기법을 소개하기 전에 기본적으로 햅틱 장비에 힘을 전달해줄 피드백 힘(feedback force)을 계산하는 방법을 알아야 할 필요성이 있다. 일반적으로 피드백 힘 계산을 위해서 수식 (1)과 같은 Spring-Damper 모델을 사용한다[4].

$$F = K_S \times X + K_D \times V. \quad (1)$$

K_S 는 탄성계수, X 는 HIP와 프록시 거리, K_D 는 제동계수, V 는 HIP의 속도이다. 그림 5에서 볼 수 있듯이 물체에 대한 힘은 네 가지 요소를 통해 결정된다. 탄성계수는 물체의 속성 중 하나인 물체의 강도와 같다. X 는 사용자가 물체를 밀기 위한 깊이로써 물체의 접점인 프록시와 HIP사이의 거리이다. K_S 와 X 의 곱으로 힘이 계산되므로, K_S 와 X 는 햅틱 장비에 전달하기 위한 힘 계산의 주된 요소가 된다. 본 논문에서는 K_S 와 X 를 변경하는 방법에 중점을 두고 있다.

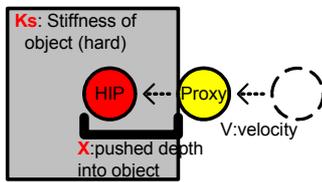


그림 5. 피드백 힘 계산 모델 모형.

3.2 힘의 비교에 기반을 둔 동적 물체 강도 변경 기법

서버는 단순히 클라이언트의 HIP 정보만으로 물체의 움직임 여부를 판단한다. 클라이언트에서 물체를 일정 힘을 통해 움직이고 싶을 때 움직인 물체의 위치는 결국 Spring-Damper 모델에 따라 프록시와 HIP 사이의 거리에 의해 결정된다. 클라이언트에서 생각하는 깊이 정보와 서버에서의 깊이 정보 값이 같다면 물체는 같은 힘을 받게 되며 물체의 변경된 위치 또한 같게 된다. 하지만 지연에 의해 클라이언트는 이 깊이 정보를 정확히 전달할 수 없다. 그 이유는 앞서 문제정의에서 밝힌 바와 같이 서버와 클라이언트의 물체가 서로 다른 위치에 있기 때문이다. 이 차이는 지연에 의해 생긴 것이므로 지연에 의한 서버와 클라이언트의 물체 위치 차이에 클라이언트에서 원하는 힘을 발생시키기 위한 프록시와 HIP의 거리를 더해주면 결과적으로 서버는 클라이언트가 원래 발생시키고자 한 힘으로 물체의 위치를 갱신시킬 수 있다. 이처럼 클라이언트의 HIP가 서버의 물체에 영향을 줄 수 있게 물체 강도를 변경하여야 한다. 다음 수식들은 앞서 말한 이론을 토대로 유도한 수식이다. 지연에 의한 물체의 위치차를 보상하기 위해서 Spring-Damper 모델 [2]은 수식 (2)와 같이 재정의 된다.

엔트가 원래 발생시키고자 한 힘으로 물체의 위치를 갱신시킬 수 있다. 이처럼 클라이언트의 HIP가 서버의 물체에 영향을 줄 수 있게 물체 강도를 변경하여야 한다. 다음 수식들은 앞서 말한 이론을 토대로 유도한 수식이다. 지연에 의한 물체의 위치차를 보상하기 위해서 Spring-Damper 모델 [2]은 수식 (2)와 같이 재정의 된다.

$$F = K_S \times X \quad (\text{simple spring model})$$

$$F_{server} = F_{client}$$

$$K_S \times X_s = K_S \times b \times (X_c + X_{delay})$$

$$b = X_c / (X_c + X_{delay}) \quad (X_s = X_c)$$

$$F = K_S \times X \times X_c / (X_c + X_{delay})$$

$$F = K_S \times X \times X_c / (X_c + X_{delay} \times delay/c). \quad (2)$$

$$K_S' = K_S \times b. \quad (3)$$

$$F = K_S' \times X$$

$F_{server} = F_{client}$ 를 이용하여 같은 힘이 계산되게 만든다. X_s 는 서버에서 물체를 누른 깊이이고 X_c 는 클라이언트가 물체를 누른 깊이이다. X_{delay} 는 지연에 의한 서버와 클라이언트의 물체의 위치 차이이다.

수식 (2)를 통해 클라이언트의 HIP가 지연에 의한 물체의 위치 차이와 물체를 밀고 싶은 정도를 더한 것과 같은 위치에 놓인다는 것을 알 수 있다. HIP의 위치가 서버와 클라이언트에 대해 같다는 것은 물체에 같은 힘이 주어진다라는 것과 의미가 상통한다. HIP의 위치가 조정 되지만 물체를 동적으로 변경하였기 때문에 클라이언트는 원래 자신이 주고자 했던 크기의 힘만 줘도 서버의 물체를 움직일 수 있게 된다. 수식 (2)에 $delay/c$ 요소를 추가한 것을 지연의 크기에 조금 더 민감하게 반응하게 만들기 위해서이다. 결론적으로 수식 (3)을 이용해 물체의 강도를 수정하면 클라이언트가 물체를 움직이려고 발생시킨 힘이 서버에서도 같게 계산되도록 만든다. 변경된 물체 강도는 사용자에게 전달되는 햅틱 피드백 힘은 같지만 물체에 가하는 힘은 더 크게 만드는 역할을 하고 서버에서는 사용자가 원하는 크기의 힘을 발생시키게 만든다.

물체 강도 변경 기법의 세부 알고리즘은 다음과 같다. 서버는 단지 클라이언트들에게서 HIP정보를 수집하여 물체의 위치를 갱신하고 갱신된 물체 위치를 다시 재분배하는 역할만 수행한다. 서버와 클라이언트들은 모두 동기화가 이루어져 있다고 가정한다.

클라이언트는 밀어진 물체의 깊이와 지연에 의한 물체의 위치 차이를 구하고 물체의 강도를 변경하는 과정을 반복한

다.

- 1) 서버에서 전송된 물체 정보를 일정시간동안 히스토리에 저장한다. 이 때 물체의 위치는 현재 클라이언트의 시간정보와 함께 입력한다.
- 2) 서버에서 전송된 패킷의 시간 정보와 가장 가까운 히스토리의 시간 정보를 찾고 히스토리에서 해당 물체 위치 정보를 읽어온다.
- 3) 현재 전송된 물체위치와 히스토리에서 검색된 물체위치의 거리를 측정한다. 이 거리는 네트워크 지연에 의해 생긴 물체위치 격차이다.
- 4) 프록시와 HTP의 거리를 측정한다. 이 값은 사용자가 물체를 움직이기 위해 민 깊이이다.
- 5) 물체 위치 차이와 떨어진 깊이를 이용하여 수식 (3)에 적용한다.

4. 실험 및 결과

4.1 실험 환경

실험은 그림 1 에서와 같이 햅틱 협업 시스템에서 이루어진다. 서버와 클라이언트 사이에는 동기화 기법이 적용되어 같은 시간을 사용한다. 협업은 서버가 함께 참여하여 하나의 클라이언트 역할을 함께 수행한다. 2개의 클라이언트만 존재하므로 네트워크 부하에 대한 영향에 대한 고려는 제외하였다. 일정한 지연을 생성시키기 위해 네트워크 에뮬레이터를 이용하였다[5].

실험은 단 방향 지연이 50ms, 100ms, 150ms, 그리고 200ms의 상황에 대하여 수행되었다. 지연에 의한 비정상적인 힘을 제어하는 경우, 성능 측정을 위해 서버와 클라이언트에서 각 물체에 적용되는 힘을 비교하여 성능 측정 요소로 이용하였다. 그리고 지연에 의한 물체 진동에 대한 성능을 측정하기 위해서 물체의 X좌표 값을 통해 물체의 진동을 확인한다. 물체 진동 실험의 경우 클라이언트와 서버가 같은 좌표 값을 가지므로 클라이언트에 대한 수치만 이용하였다.

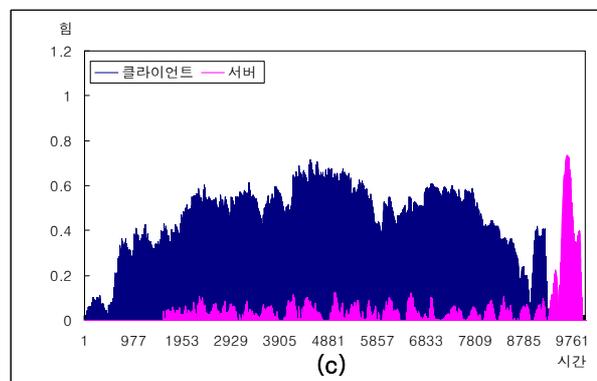
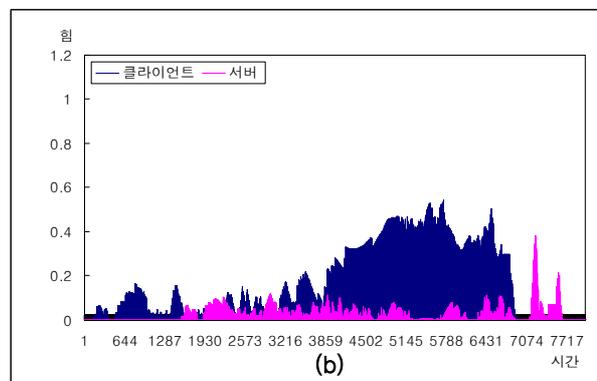
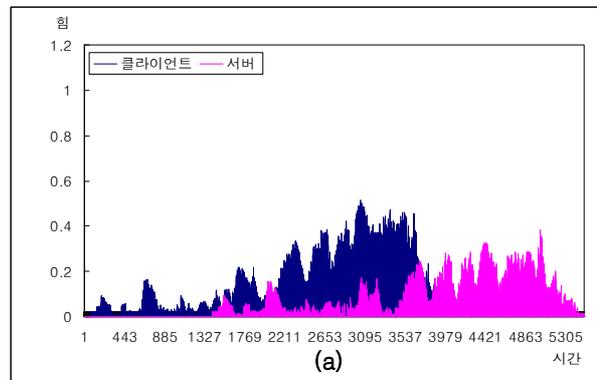
물체의 크기는 가로, 세로, 높이가 각각 0.7이고 물체의 질량은 0.1이며, c의 값으로 20을 속성으로 지정하였다.

4.2 실험 결과

그림 6에서는 물체를 밀 때 지연에 의해서 발생하는 문제를 보여준다. 결과에서 관찰할 수 있듯이 지연이 없을 때는 클라이언트가 계산하는 피드백 힘과 서버가 계산하는 피드백 힘이 거의 동일함을 알 수 있다. 하지만 지연이 증가할수록 서버와 클라이언트가 계산하는 피드백 힘의 차이가 더 커짐을 알 수 있다. 지연이 커질수록 클라이언트는 물체

를 움직이기 위해서 더 큰 힘을 필요로 하게 되고 서버는 비연속적으로 위치를 계산하는 현상을 실험 결과를 통해 확인할 수 있다.

그림 7은 제안된 보상 기법을 적용하여 물체를 미는 실험의 결과이다. 50ms, 100ms 에서는 서버와 클라이언트가 거의 동일한 힘을 계산할 수 있었다. 150ms, 200ms 에서는 서버에서 약간 비연속적으로 힘을 계산 하는 것을 발견 가능하지만 기법이 적용되지 않은 경우에 비해서는 월등히 나은 결과를 보여준다, 물체를 움직이기 위해 클라이언트가 요구하는 힘이 증가하지 않으므로 지연에 의한 첫 번째 문제인 비이상적 힘이 발생하는 현상이 해결될 수 있음을 결과를 통해 확인하였다.



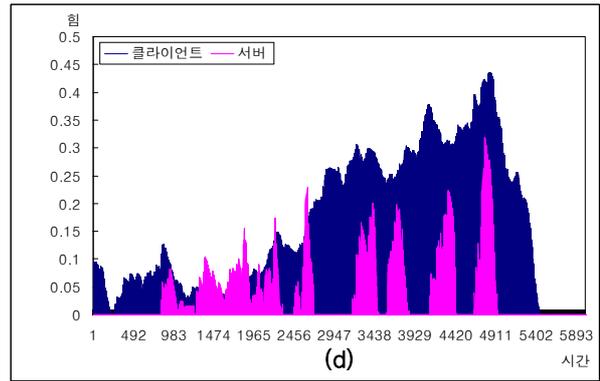
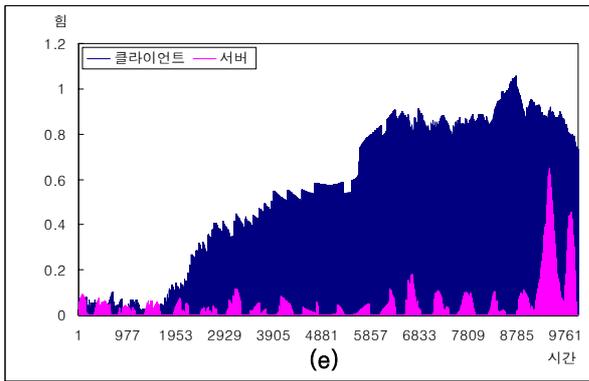
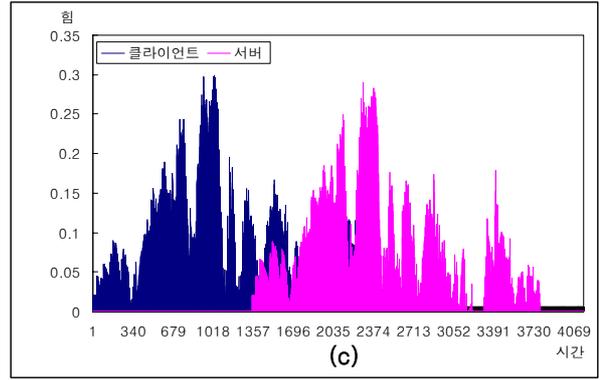
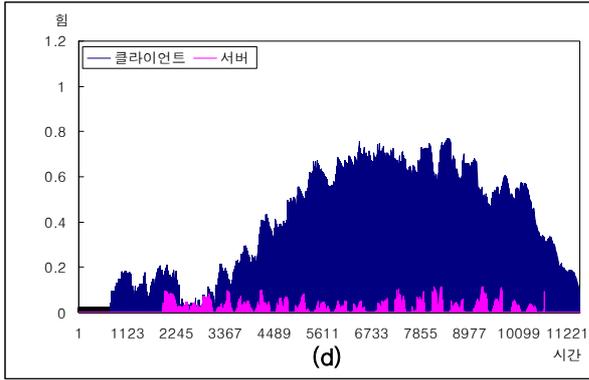


그림 6. 지연에 따른 서버와 클라이언트 피드백 힘 측정 보상기법 미적용; (a)지연 없음, (b)50ms, (c)100ms, (d)150ms 그리고 (e)200ms 지연.

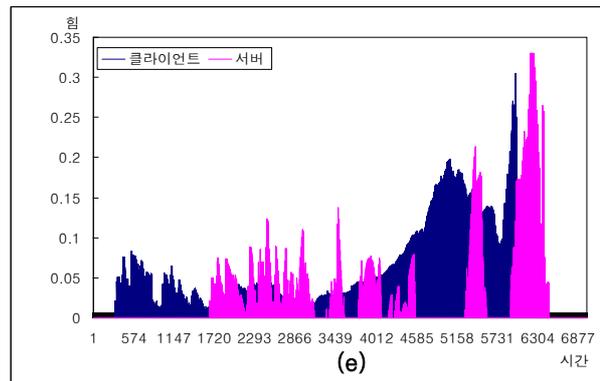
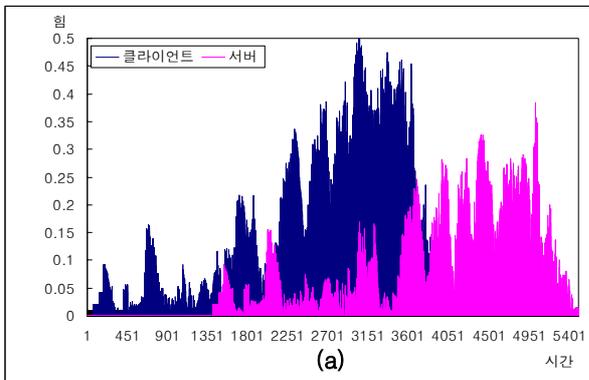


그림 7. 지연에 따른 서버와 클라이언트 피드백 힘 측정 보상기법 적용; (a)지연 없음, (b)50ms, (c)100ms, (d)150ms, 그리고 (e)200ms 지연.

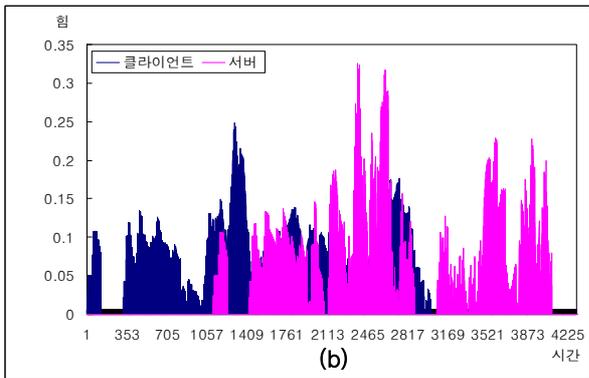


그림 8은 지연 보상이 이루어지지 않은 경우 물체를 들어 올릴 때 지연에 의해 발생하는 물체 진동을 물체 좌표 값을 통해 보여준다. 지연이 커짐에 따라 물체의 좌표가 진동함을 그래프를 통해 확인할 수 있다. 200ms에서 측정된 결과는 물체를 들어올린 상태로 유지하기 힘들 정도의 물체진동으로 인하여 햅틱 협업의 품질이 매우 저하되었음을 보이고 있다.

반면, 제안된 기법을 통해 지연 문제를 보상할 수 있음을 그림 9를 통해 확인할 수 있다. 제안된 기법을 사용할 경우 지연의 크기와 무관하게 저 수준의 진동만이 발생함을 보인

다. 이와 같이 지연에 의한 두 번째 문제에서도 제안된 보상 기법은 효과가 있음을 확인하였다.

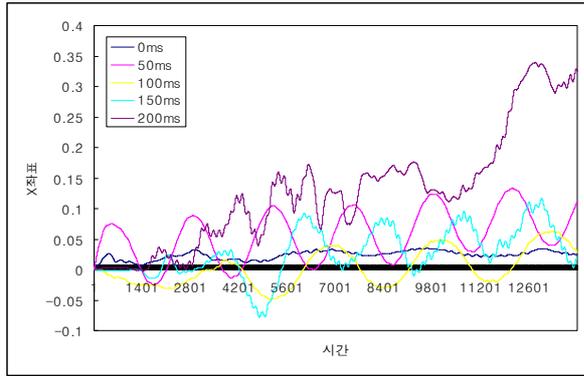


그림 8. 물체를 들어올릴 때 지연에 따른 결과
(보상 기법 미적용).

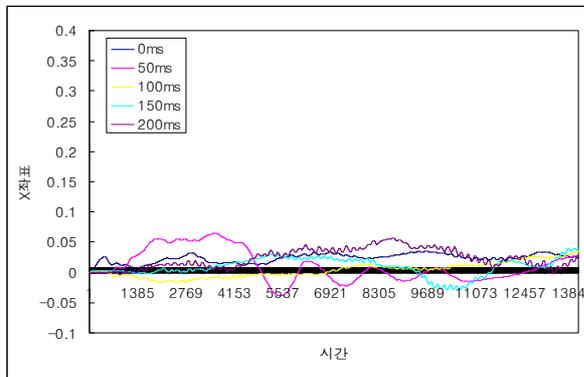


그림 9. 물체를 들어올릴 때 지연에 따른 결과
(보상 기법 적용).

5. 결론 및 향후 연구

본 논문에서는 네트워크 기반 햅틱 협업에서 발생하는 지연에 의한 두 가지 문제를 비정상적인 무게감, 물체를 들어올릴 때 발생하는 물체의 진동으로 정의하고 그 원인을 밝혔다. 지연 문제의 원인이 힘 전달의 비연속성에 있음을 인지하여 이 두 가지 문제를 보상하기 위해 지연에 의한 물체의 위치 차이에 따라 동적으로 물체의 강도를 바꾸는 방법에 대한 연구를 진행하였다.

단순히 측정된 상수와 지연에 의한 수식으로는 지연문제를 해결할 수는 있으나 물체의 속성에 대한 고려가 없으므로 햅틱 미디어의 품질을 떨어뜨리는 효과를 일으킨다. 본 논문에서는 물체 강도 변경 기법으로, 지연이 있는 상황에서도 현실감 있는 힘 생성을 위해 클라이언트가 물체에 발생시키고 싶은 힘과 지연에 의한 서버와 클라이언트의 물체 위치 차이를 극복할 수 있는 크기의 물체 강도를 설정한다. 이 지연 보상 기법은 물체의 강도와 클라이언트가 발생시키

고자 하는 힘, 그리고 지연의 크기 등이 물체 강도 변경의 요소로 사용된다. 그러므로 제안된 지연 보상 기법을 통해서 지연에 의한 문제를 최소화하여 햅틱 협업의 질을 올림과 동시에 최대한 물체의 속성과 힘을 유지시켜 햅틱 미디어의 품질을 함께 높이는 방안을 제시하였다.

향후 본 연구에서는 더 정확한 물체 강도 변경 값 측정을 위한 연구가 필요하다. 특히 지연에 의한 물체 위치 차이를 구하기 위해 예전의 지연정보를 이용하므로, 정확한 물체의 위치 차이라 보기 어렵다. 이를 보완하기 위해 정확한 위치 차이를 예측하는 기법을 연구할 것이다. 또한 고품질의 햅틱 협업을 지원하기 위해 햅틱 협업 시스템을 위한 네트워크 적응적 전송 설계 [7] 등과의 통합을 연구할 필요성이 있다.

감사의 글

본 연구는 산업자원부 지방기술 혁신 사업
(RTI 05-03-01) 지원으로 수행되었음.

참고문헌

- [1] S. Lee, and J. Kim, "Intra-media synchronization scheme for haptic interactions in distributed virtual environments," in Proc. SPIE ITCOM, vol. 6015, 2005.
- [2] Z. Cen, M. W. Mutka, D. Zhu, and N. Xi. "Supermedia transport for teleoperations over overlay networks," in Proc. NETWORKING, pp. 1409-1412, Oct. 2005.
- [3] M. Fujimoto and Y. Ishibashi, "A compensation scheme for network delay jitter of haptic media in networked virtual environments," in Proc. 8th World Multi-Conference on Systemics, Cybernetics and Informations, vol. 3, Jul. 2004.
- [4] G. Burdea, "Virtual reality systems and applications," in Proc. IEEE Electro'93 International Conference, Apr. 1993.
- [5] 이석희, Le Hai Dao, 김종원, "촉각지원 네트워크 협업 테스트베드의 구현," in Proc. HCI, pp. 244-250, 2006.
- [6] B. Tay, N. Stylopoulos, S. De, D. Rattner. W, M. Srinivisan. A. "Measurement of in-vivo force response of intra-abdominal soft tissues for surgical simulation." in Proc. MMVR Conference pp. 514-519, 2002.
- [7] S. Lee, S. Moon, J. Kim, "A network-adaptive transport scheme for haptic-based collaborative virtual environments." in Proc. ACM NetGames, vol 1, 2006.