

# 디바이스 서버를 이용한 모션 베이스 특성에 따른 시각적 동기화

\*박형근, \*김종국, \*정의정, \*송재복, \*\*고희동

\*고려대학교 기계공학과, \*\*KIST IMRC

## Device Server Based Synchronization of a Simulator Reflecting Motion Base Characteristics

\*Hyung-Geun Park, \*Jong-Kuk Kim, \*Ui-Jung Jung,

\*Jae-Bok Song, \*\*Hee-Dong Ko

\*Mechanical Engineering, Korea Univ., \*\*KIST IMRC

### 요 약

가상공간에서 탑승자에게 운동감을 제공하기 위해서 수많은 형태의 모션 시뮬레이터가 개발되어 왔다. 하지만 이러한 형태의 시뮬레이터들은 그 고유한 형태에 의해 정해진 동역학과 위시아웃 알고리즘을 가지지만 정밀한 시뮬레이터가 될 수 있다. 본 논문은 기존의 좁은 시야공간에서의 시뮬레이터가 아닌 넓은 시야를 제시하고, 모션 베이스와 독립적으로 운영되는 영상 제시기관을 이용하여 탑승자에게 운동감을 제공하는 분리형 시뮬레이터를 대상으로 하여 영상과 모션을 동기화시켜서 제시하는 방법에 대해 논의하고자 한다.

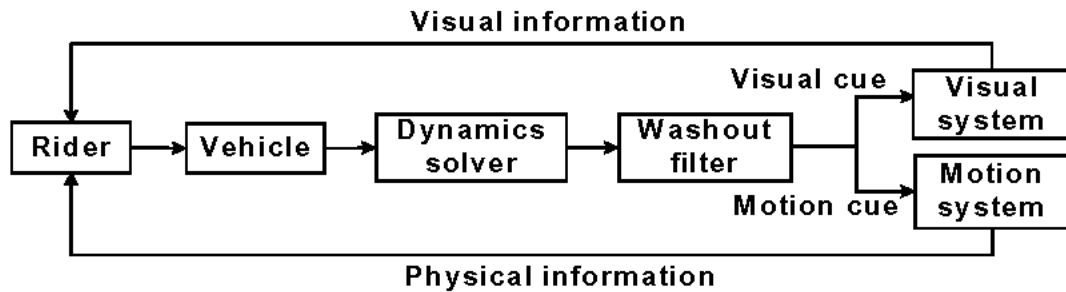
*Keywords: Motion Simulator, Virtual Reality, Vehicle Dynamics, Washout algorithm*

### 1. 서론

가상환경에서 움직이는 운동체에 대하여 탑승자가 그에 부합하는 운동감을 느끼도록 하기 위해 여러 형태의 모션 시뮬레이터가 개발되어 운용되고 있다. [1, 2] 그림 1은 전형적인 모션 시뮬레이터에서의 신호 흐름을 보여준다. 우선 탑승자가 주행체(자전거, 자동차, 비행기 등)를 조작하면(자동차의 경우 가감속 및 조향), 이들 신호를 주행 동역학의 입력으로 취급하여 동역학의 해, 즉 주행체의 운동을 구하게 된다. 그러나 실제 주행체의 운동 범위에 비해서 모션 베이스의 운동 범위가 매우 제한되어 있으므로,

위시아웃 필터를 사용하여 동역학에서 계산된 실제 운동을 모션 베이스의 운동 범위 내에서 탑승자가 실제 운동과 유사한 느낌을 가질 수 있는 운동으로 변환하게 된다. 이렇게 변환된 운동은 영상 제시기 및 모션 베이스의 제어부로 전달되어 영상이나 모션의 변화를 야기하게 된다. 탑승자는 이러한 시각 및 운동 정보를 통해서 실제 주행체에 탑승하고 있다는 경험을 하게 된다.

이러한 시뮬레이터의 경우, 주행체의 동역학과 위시아웃 필터 등에 대한 연구는 오래 전부터 활발히 진행되고 있으나, 영상과 모션 간의 동기화에 대한 연구는 별로 없다. [3, 4] 이는



[그림 1] Flow chart of motion simulator

현재의 가상현실 시스템이 주로 영상을 중심으로 기술되어 있으며, 모션 베이스가 사용된다라도 주로 직관에 의한 튜닝 방식에 의해서 모션을 제어하기 때문이다. 그러나 이러한 방식으로는 각 시스템 특성에 의한 차이를 고려하지 못하므로, 효과적으로 영상과 모션을 동기화시키는 것은 어렵다. 따라서 본 연구에서는 영상 제시기와 모션 베이스의 특성의 차이를 반영하는 동기화 방법을 논하고자 한다.

시뮬레이터는 그 형태에 따라 자동차 시뮬레이터, 자전거 시뮬레이터, 비행기 시뮬레이터 등으로 구분할 수 있다. 한편, 모션 베이스와 영상 제시기의 상대적인 위치에 따라서 분류할 수도 있는데, 영상 제시기가 모션 베이스 내부에 있는 일체형 시뮬레이터와 영상 제시기가 모션 베이스 외부에 있는 분리형 시뮬레이터로 구분하기로 한다. 본 논문에서는 주로 탑승자에게 좀더 넓은 시야를 제공해 줄 수 있는 분리형 시뮬레이터에 대해 논하고자 한다.

분리형 시뮬레이터와 일체형 시뮬레이터의 구분이 중요한 이유는 모션 베이스의 움직임이 영상 제시기의 위치와 형태에 따라서 다르게 적용되어야 하기 때문이다. 일체형 시뮬레이터의 경우에는 모션 베이스에 의한 운동이 영상과 탑승자의 상대 위치에 변화를 주지 않으므로 계산된 동역학을 바로 영상과 모션에 반영하는 것이 가능하다. 그러나 분리형 시뮬레이터의 경우에는 영상의 위치가 정적인데 비해 모션 베이스의 움직임에 따라 탑승자의 시점이 변화하므로 이를 동기화시키는 작업을 수행하여야 한다.

본 논문은 상대적으로 영상과 모션의 동기화

가 어려운 분리형 시뮬레이터를 대상으로 하며, 다음과 같이 구성되어 있다.

## 2. 분리형 시뮬레이터의 동기화

분리형 시뮬레이터의 경우에는 운동감 제시를 위한 모션 베이스의 운동과 워시아웃 필터에 의해 생기는 변이좌표(tilt coordination)의 차이를 반영해야 하므로 동기화가 어렵다. 일체형 시뮬레이터의 경우에는 영상 제시기와 모션 베이스가 같이 움직이기 때문에 이러한 고려할 필요가 없다. 그러나 분리형 시뮬레이터일 경우에는 모션 베이스의 움직임에 의해 탑승자의 시점이 변화되므로, 이를 고려하지 않으면 탑승자에게는 제시된 모션이 오히려 역효과를 유발시킬 수 있다. 즉, 영상에서의 동역학은 탑승자가 정지된 시점에서 영상을 바라본다는 가정 하에 수행되므로, 모션 베이스의 운동에 의해 탑승자는 고정된 시점에서 벗어나게 되며, 이로 인해서 원근의 왜곡과 가속도의 변화가 생기게 때문에 오히려 불완전한 운동감을 탑승자에게 제시하게 된다.

기본적으로 모든 시뮬레이터는 탑승자의 시점의 방향, 모션 베이스의 수평면과 영상 제시기의 수평선이 정확히 평행을 이루어야 탑승자는 가상공간상에서 시선에 평행하게 자신이 위치하고 있음을 인식하게 된다. 그러므로 모션 베이스의 수평면과 영상 제시기의 수평선을 동일하게 맞추어야 한다. 또한, 정확한 동기화를 위해서는 영상 제시기가 제시하는 시계(field of view)에 맞추어야 탑승자에게 정확한 원근감을 제시할 수 있어야 한다.[5]

## 2.1 회전 및 위치 변화에 따른 시점의 변화

분리형 시뮬레이터의 경우에 모션 베이스가 탑승자에게 운동감을 제공하기 위해 위치 및 회전 운동을 수행하면, 이에 맞추어서 영상 제시기에서 탑승자의 변화된 시점을 동기화시켜야 할 필요성이 있다.

일례로, 그림 2와 같이 자동차 시뮬레이터에서 영상 제시기를 통해서  $7\text{m/s}^2$ 이라는 가속도를 탑승자에게 제공하고자 하는 경우를 고려하여 보자. 영상 제시기를 통해  $7\text{m/s}^2$ 이라는 가속도를 탑승자에게 제시하고, 모션 베이스 역시 순간적으로  $7\text{m/s}^2$ 이라는 가속도로 움직일 경우에, 탑승자는 시각적으로 총  $14\text{m/s}^2$ 이라는 가속도를 경험하게 될 것이다. 즉, 탑승자가 경험하는 가속도는 다음과 같이 구할 수 있다.

$$a_r = a_g + a_m \quad (1)$$

여기서,  $a_r$ 는 탑승자(rider) 시점에서의 가속도,  $a_g$ 는 그래픽 렌더링 서버에서 제시하는 영상의 가속도,  $a_m$ 은 모션 베이스의 가속도이다. 그림 3은 이러한 상황을 수정하여 사용자에게 적용한 경우이며, 이를 그래픽 렌더링 서버에서 구현하기 위한 가속도는 다음과 같다.

$$a_g = a_d - a_m \quad (2)$$

여기서,  $a_d$ 는 주행 동역학 계산에 의해서 구한 값으로, 실제 탑승자가 경험하여야 하는 가속도이다. 이러한 값을 대입하여 그래픽 렌더링 서버에 값을 대입할 경우, 그림 3와 같이 탑승자의 입장에서 정확한 가속도의 값이 반영되는 것을 볼 수 있다.

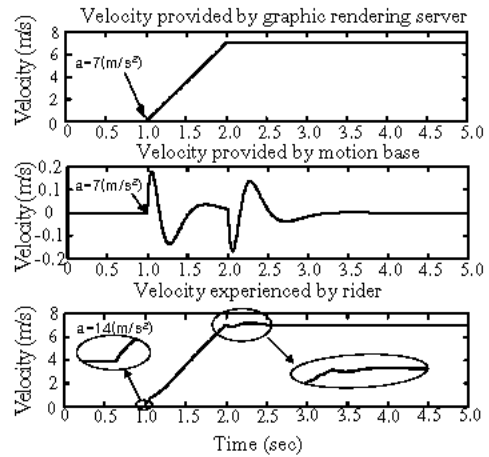


그림 2 분리형 시뮬레이터에서 사용자 시점에 따른 동기화가 적용되지 않은 경우

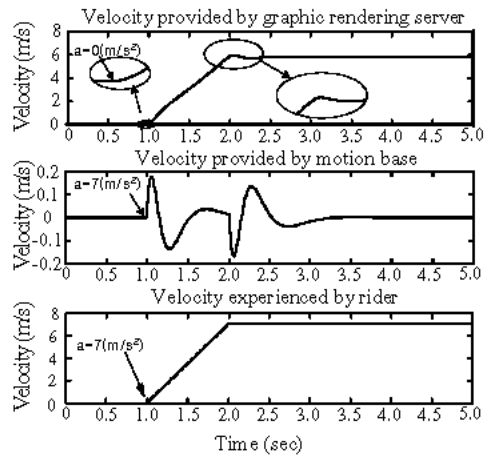


그림 3 분리형 시뮬레이터에서 사용자 시점에 따른 동기화가 적용된 경우

한편, 조향 동작이 수행되는 경우에 모션 베이스의 회전에 의해서 탑승자의 시선 방향(eye direction)이 변경되므로, 제공되는 영상 역시 사용자의 시선 방향으로 바뀌어야 한다.

$$\gamma_{NW}_g = \gamma_{NW}_m \quad (3)$$

이와 같은 동기화 가정을 거치지 않은 경우에는 직선 가속의 경우에는 긴물이 갑자기 뚫출하는 것처럼 느끼거나, 왜곡된 위근을 느끼게 된다.

## 2.2 워시아웃 필터에 의한 시점의 변화

다음으로 고려할 사항은 워시아웃 필터에 의해서 생기는 모션 베이스의 변이좌표(tilt)

coordination)을 지리하는 문제이다. 위시아웃 필더는 실제 주행체에서 큰 범위의 운동을 운동 범위가 좁은 모션 베이스가 그대로 구현할 수 없으므로, 주행 영역에서 커진된 주행체의 운동을 모션 베이스의 운동 범위 내의 운동으로 직렬화 변환시켜 주는 알고리즘이다. 이를 위해서 직선 가속도의 경우 사용자에 중력 가속도 방향을 실제 중력방향과 어긋나게 제공하고, 사용자에게 계속적으로 가속도를 느끼도록 인이 좌표를 생성하게 된다. 모션 베이스의 디바이스 시계가 위시아웃 필더의 연산결과로 생성하는 변이좌표의 값을 영상 제시기가 반영하도록 하여야 한다.

이들 움직임이 반영되지 않을 경우 분리형 시뮬레이터 내의 탑승자는 영상 제시기로부터 부적절한 값을 제공받게 된다. 일례로, 위시아웃 필더는 탑승자에게 직선가속도를 제공하기 위해 탑승자를 회전 방향으로 회전시키게 되는 데, 이를 영상 제시기가 반영하지 못한 경우, 고정된 가상환경의 수평선에 의해 탑승자는 운동체가 가속된다고 느끼는 것이 아니라, 운동체가 기울어져서 가속된다는 느낌을 받게 된다. 그러므로 이러한 변이좌표의 경우, 영상 제시기에서 다음 처리 같이 반영되 주어야 할 필요가 있다.  $roll_{in}$ ,  $pitch_{in}$ 을 각각  $x$ 축,  $y$ 축으로의 위시아웃에 대한 변이좌표라고 할 때, 그래픽 렌더링 시터에서 표현해야 할 값은 다음과 같다.

$$\begin{aligned} pitch_s &= pitch_{in} \\ roll_s &= roll_{in} \end{aligned} \quad (4)$$

$y$ 축 회전방향의 경우, 탑승자는 직선가속도에 대한 가속을 느끼기 위한 부분이며,  $x$ 축 회전방향의 경우는 원진리에 대한 경우를 영상 제시기에서 구현해 주기 위한 식이다.

### 3. 모션과 영상의 동기화

모션이 수반되지 않는 영상 제시의 경우에는 동기화적인 모든 정보는 영상을 통해서만 제시된다. 즉, 중력에 의한 정보, 주행체와 접촉하는 사형 정보 등을 모두 영상을 통해 구현하게 된다. 이 때, 영상 제시기와 탑승자 사이는 공간

적으로 고정되어 있다. 이를 해결하기 위해 일반적으로 일체형 혹은 분리형 시뮬레이터를 사용하고 있다.

분리형 시뮬레이터는 일체형 시뮬레이터보다 규모가 큰 영상 제시기를 사용할 수 있으며, 이로 인해 탑승자에게 진짜 시야를 제공하는 것이 가능하다. 따라서 모션 베이스와 동기화를 통하여 탑승자에게 모션을 제공할 때, 일체형 시뮬레이터보다 높은 몰입감을 제공하는 것이 가능하다.

분리형 시뮬레이터에서는 영상 제시기가 고정되어 있으며, 영상 제시기는 주로 정적인 지리를 수행한다. 일반적으로 모션 베이스의 운동 영역 내의 운동을 모션 베이스가 지리하고, 전체적인 운동 기술은 영상 제시기가 맡아서 처리하게 된다. 예를 들어, 수직모드 시뮬레이터의 경우에, 전체적인 정적인 배경은 통하여 제시하고, 모션 베이스의 운동으로는 수평에서의 물방이나 비행 등의 모션을 나타내는 등의 역할 분담을 한다면 좀더 몰입감을 높일 수 있다.

이와 같이 분리형 시뮬레이터에서 동역학 모델에서 계산한 운동이 모션 베이스의 운동 범위 내에 있는 경우에는 모션 베이스의 운동만으로 충분히 모션을 표현해 줄 수 있다. 그러나 동역학 모델에서 계산한 운동이 모션 베이스의 운동 범위를 벗어난 경우에는 모션과 영상의 적절한 역할 분담을 통하여 그림자와 같은 의사 운동(pseudo motion)의 형태로 지리하여야 한다. 예를 들어, 비행기가 후락하는 시뮬레이터의 경우, 모션 베이스 자체가 시뮬과 수직하게 동원 수 없으므로 영상과 함께 이를 반영할 필요가 있다.



[그림 3] 의사 운동

$pitch_s$ 는 모션 베이스가 제공할 수 있는 최대의  $x$ 축 회전각,  $pitch_{max}$ 는 모션 베이스의 운동 범

위의 한계점을 기준으로 할 때 더 움직여 주어야 할 회전각,  $pitch_d$  영역학 계산에 의해 나온 x축 회전각,  $pitch_g$ 를 그래픽 렌더링 서버에 적용해주는  $pitch_d$ ,  $pitch_m$ 을 모션 베이스가 실제로 움직여야 하는 회전각이라 하자. 의사 운동을 표현해야 할 경우인  $pitch_m > pitch_d$ 인 경우에는

$$pitch_{over} = pitch_d - pitch_m \quad (5)$$

이 되지만, 이와  $pitch_m < pitch_d$ 인 경우에는

$$pitch_{over} = 0$$

이 된다. 이를 그래픽 렌더링 서버에 적용하면 다음과 같은 식이 된다.

$$pitch_g = pitch_{tilt} + pitch_{over} \quad (6)$$

위와 같은 식에 의해, 롤링 및 요잉 운동에 대해서도 다음과 같다.

$$roll_g = roll_{tilt} + roll_{over} \quad (7)$$

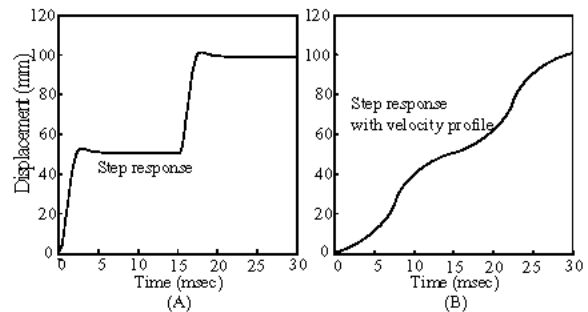
$$yaw_g = yaw_{tilt} + yaw_{over} \quad (8)$$

#### 4. 분리형모델에서의 통신 시간지연

시뮬레이터의 경우, 동기화의 또 하나의 문제가 통신에 의한 시간지연을 들 수 있다. 본 시뮬레이터의 경우, 그래픽 렌더링 서버와 디바이스 서버로 분리하여 분산서버로 가상현실을 구현하고 있다.[6]

UDP에 의한 그래픽 렌더링 서버와 디바이스 서버와의 통신 속도는 2msec 미만이며, 모션 베이스와 디바이스 서버 사이의 통신은 병렬통신으로 속도는 1msec 미만이다. 그러나 주요 병목지역은 그래픽 렌더링 서버에서의 정보의 갱신과 디바이스 서버의 갱신주기이다. 그래픽 렌더링 서버는 갱신에 지연되는 시간은 약 30msec정도이며, 디바이스 서버는 15msec의 갱신주기를 가진다. 이러한 갱신주기의 시간 지연으로 인해 모션 베이스의 운동이 연속적이지 못하고 정지되는 현상이 발생하게 되는데, 모션 베이스를 제어하는 DSP(digital signal

processor) 프로세서에서 부드러운 동작을 유도하기 위해 1msec 단위로 궤적을 생성해 줄 수 있는 속도 프로파일을 적용시켜 이러한 문제를 해결하였다. 즉, 그림 5와 같이 15msec 동안 DSP 내부에서 모션 베이스로 다음 궤적까지 새로운 궤적을 만들어서 부드럽게 작동하도록 하였으며, 각 축이 동기화되어 움직일 수 있도록 하기 위해 정해진 시간에 지정좌표로 도달하도록 하여, 어느 축이 먼저 수행되는 일이 없도록 하였다.



[그림 5] (A) Step response without velocity profile  
(B) Step response with velocity profile

#### 5. 결론

모션 베이스와 영상 제시기가 분리되어 있는 분리형 시뮬레이터에서는 모션 베이스에 있는 탑승자의 시점이 모션에 의해서 변할 수 있다. 따라서 주행 동역학의 해를 영상과 모션에 적절히 분배하여 영상과 모션을 동기화하여 구현하여야 한다. 이 연구에서는 이러한 상황에 대하여 모션 베이스의 운동을 어떻게 생성하여야 하는가에 대하여 논의하였다. 이렇게 영상과 동기화된 모션을 구현함으로써, 탑승자에게 보다 사실적인 시뮬레이션이 가능하였다.

#### 후기

본 연구는 감성공학 기반기술 개발 사업의 일부로 한국과학기술연구원원위 지원하에 수행되었다.

#### 참고 문헌

- [1] Barbagli F., Ferrazzin D., Avizzano C. A., Bergamasco M., "Washout Filter Design

- for a Motorcycle Simulator," Proc. of IEEE Conference on Virtual Reality, pp.225-232, 2001.
- [2] Schmidt S.F., Conrad B., "Motion drive simulator signals for piloted flight simulators," NASA CR-1601, 1970.
- [3] Bee S.T., Burns P.C., Dixon P.J., "Driver-vehicle performance measurement in a simulator," Monitoring of Driver and Vehicle Performance (Digest No: 1997/122), IEE Colloquium on, pp. 101-103, 1997.
- [4] Correia M.J., Guedry F.E., "The vestibular system: Basic biophysical and physiological mechanisms," In R.B. Masterson (Ed.), Handbook of Behavioral Neurobiology, Plenum Publishing Corp. 1978.
- [5] 정의정, "퍼지 로직에 기초한 위시아웃 알고리즘을 적용한 주행 시뮬레이터의 개발" 고려대학교 기계공학과 석사학위논문 2001.
- [6] 차양환, "네트워크 트래픽 예측 및 최적화 설계," 고려대학교 컴퓨터과학 기술대학원 정보산업학과 석사학위 논문, 1998.