

컨테이너 크레인의 실시간 제어 시뮬레이션

홍 금 식, 박 배 정, 김 성 훈, 정 재 룡

부산대학교 기계공학부 및 지능기계공학과

1. 서론

본 논고는 네트워크(network)를 통하여 플랜트를 제어하고자 하는 연구의 일환으로, PC(personal computer)가 제어기이고 또한 플랜트의 동역학을 또 다른 PC에서 구현할 때 발생하는 문제점과 해결방법에 대하여 기술하고자 한다. 즉, 실제의 플랜트 대신에 하나의 PC가 그 역할을 담당하게 되고, 제어기가 실시간으로 LAN을 통하여 플랜트를 제어하는 상황을 시뮬레이션 하고자 한다. 본 연구에서 제어기 개발을 위한 프로그래밍 환경으로는 서울대학교 제어계측신기술 연구센터에서 개발된 CEMTool을 사용한다[1].

통상적으로 컴퓨터를 이용한 제어시스템의 시뮬레이션에서는 플랜트의 운동방정식과 제어기가 동일한 컴퓨터에서 프로그래밍되어 시뮬레이션이 수행된다. 이러한 경우 플랜트 동역학과 제어알고리즘의 계산을 수행하는 데 걸리는 전체시간에 고려치 않고 시간증분(time step)을 충분히 작은 값을 설정하면 계산을 수행하는 데 걸리는 전체 시간에 관계없이 만족할만한 결과를 얻을 수 있다. 이때 설정된 시간증분은 실시간의 개념이 아니므로 실제 시간에 근거한 시간증분 이내에 계산이 끝나지 않아도 무관하며, 전체 계산속도는 오직 PC의 연산속도에만 의존하게 된다. 따라서 이러한 방법으로 설계된 제어기가 우수한 성능을 보인다 하더라도 실험으로 곧바로 구현 가능한지에 대한 여부는 결정할 수 없다.

그러나 플랜트와 제어기를 독립된 두 대의 PC에 분리하여 프로그램을 하는 경우 좀더 실제적인 상황을 고려할 수 있다. 두 대의 PC를 이용한 제어시스템의 설계의 필요성은 플랜트와 제어기가 원거리에 위치하고 있을 때 더욱 뚜렷해진다. 이것은 바로 플랜트와 제어기 사이의 통신문제를 해결하여야 하기 때문이다. 정도의 차이는 있을지언정 모든 제어시스템에 있어서 플랜트와 제어기 사이의 통신문제는 발생할 수 있다. 즉, 플랜트가 거동을 하게 되면 각종 센서로부터 그 측정값을 제어기가 수집 및 처리하고 제어입력

을 결정한 후 다시 플랜트로 입력을 가하게 될 때 통신으로 인한 시간지연은 중요한 이슈가 될 수 있다. 이러한 과정이 두 대의 PC를 이용하게 되면 고려될 수가 있다. 그리고 기존의 제어시스템 설계방법에는 적시성의 요소인 실시간 처리에 대한 개념이 빠져있다. 왜냐하면 프로그램이 돌아가는 시간 증분은 신중히 고려되지 않아도 시스템 운동방정식의 해가 구해지기 때문이다. 이러한 제어기 설계방법은 설계된 제어기의 성능을 확인해보는 수단으로는 충분히 평가될 수 있겠으나, 실제 실험을 위한 대비책으로는 부족하기 때문에 실제 실험을 할 때에 예측하지 못한 많은 어려움을 겪게 될 수 있다. 또한 시뮬레이션에서 확보된 결과가 실제 실험결과와 차이를 보일 경우에 그 원인을 밝혀내는데 소홀히 할지 모른다. 이것은 올바른 시뮬레이션 및 실험에 대한 분석이라고 볼 수 없다.

그러나 본 논고에서 제시하는 두 대의 PC를 이용하여 플랜트와 제어기를 독립적으로 프로그래밍하고 네트워크를 통한 시뮬레이션을 수행할 경우 다음의 사항이 추가되므로 좀더 실제적인 제어문제를 다룰 수 있다.

첫째, 플랜트의 운동방정식으로부터 해가 구해지는 시간, 제어기가 플랜트에 가할 제어입력을 결정하는 시간, 플랜트와 제어기 사이에 데이터 교환이 이루어지는 시간 등을 모두 포함할 수 있는 시간증분을 결정하는 것부터 신중히 고려하게 된다. 왜냐하면, 플랜트가 설정된 시간증분 이내에 운동방정식의 해를 계산하지 못하면 제어기는 그 출력을 받지 못하게 되고 이러한 경우 제어입력이 결정될 수가 없으므로 원하는 제어시스템의 성능 및 안정도를 보장할 수 없게 된다.

둘째, 플랜트와 제어기가 사용하는 시간은 똑같이 흘러가기 때문에 두 대의 PC간의 시간적 동기 문제가 중요한 요인이 된다. 이러한 방법으로 실시간 처리를 하지 못한다면 고성능의 제어기로서의 전환문제, 시간증분의 재설정문제, 입출력 데이터의 조정 등 부가적인 문제도 제어설계자가 고려하게 된다.

이것이 두 대의 PC를 이용하여 제어시스템을 설계하는 주된 장점이라고 할 수 있겠다.

2. 제어시스템의 구성

표 1은 기존의 제어시스템 설계방법과 두 대의 PC를 이용한 방법의 알고리즘 차이를 개념적으로 보여주고 있다.

표 1. PC를이용한제어시스템설계방법

기존의 설계방법	두 대의 PC를 이용한 설계방법	
	Plant	Controller
<pre>initialize(); for(; ;){ controller(); plant(); }</pre>	<pre>initialize(); for(; ;){ in=receivelan(); plant(); sendlan(); }</pre>	<pre>initialize(); for(; ;){ sendlan(); controller(); out=receivelan(); }</pre>

기존의 제어시스템 설계방법은 제어기와 플랜트가 하나의 루프 안에서 같이 계산되도록 구성되지만, 두 대의 PC를 이용한 설계방법은 제어기가 보낸 제어입력을 플랜트가 받아서 플랜트의 운동방정식 해가 구해지고, 다시 그 결과를 제어기로 보내는 과정이 포함되어 있다. 미루어 짐작하기에 이 두 대의 PC 사이에는 어떤 약속에 의해서 동작되어야 함을 알 수가 있다.

2.1 플랜트의 구성

플랜트를 구성하는 것은 제어하고자 하는 시스템을 표현하는 단계이다. 우선 제어대상인 플랜트를 모델링하여야 한다. 모델링하는 과정에서 실제 시스템을 가장 잘 표현할 수 있도록 고려하여야 하겠다. 그리고 플랜트의 운동방정식을 얻게되면 운동방정식으로부터 수치해를 구하기 위하여 Euler, Runge-Kutta 등의 방법을 사용하여 미분방정식의 해를 구할 수 있도록 구성한다. 이제 구해진 플랜트의 수치적인 해로부터 시스템 상태나 출력을 제어기로 전달할 수 있도록 입출력관계를 구성한다. 또한, 수치해로부터 플랜트의 움직임을 시각적으로 표현할 수 있는 동영상 윈도우를 구성한다. 이것은 시스템 운동방정식의 수치해가 매 시간중분마다 시스템의 거동을 표현하기 때문에 시각적으로 플랜트의 거동을 이해하는데 도움이 된다. 이제 수식과 동영상을 모두 갖춘 플랜트가 갖추어 지게 되면, 이 플랜트의 제어기를 구성하도록 한다. 단, 플랜트에 여러 가지 다른 제어기법이 적용될 수 있도록 시스템 운동방정식 이외에 적절한 외부요소를 추가하여 유연성있게 구성한다. 예를 들면 시스템의 비

선형성, 시간지연, 모델의 불확실성, 외란 등의 요소 등이 적절히 포함될 수 있도록 한다. 제어설계자가 필요하다고 생각되는 중요한 출력요소는 그래프로 표현하여 시뮬레이션 결과를 평가한다.

2.2 제어기의 구성

이제 시스템 모델링과 해석과정에서 플랜트가 구성되면 다음 단계로 플랜트가 목표로 하는 제어능의 구현과 안정성을 보상에 줄 수 있는 제어기를 설계한다. 이때 설계되는 제어기 또한 다른 한 대의 PC에서 독립적으로 프로그램이 되며 통신기능 및 데이터 입출력 기능을 갖추도록 구성되어야 한다. 제어기를 설계할 때 플랜트에 추가되었던 여러 가지 외부요소들에 의한 영향을 최소로 하여 가장 좋은 성능을 얻을 수 있도록 구성하되 서로 다른 제어기법들이 다양하게 적용될 수 있도록 확장성을 좋게 한다. 다음으로 가장 기본이 되는 고전적 제어기법인 PID 제어기가 적용되었을 때의 결과와 비교하여 설계된 제어기가 우수한 결과를 나타냄을 입증한다. 실제 실험장치에서 수행하기 어려운 극한 상황(emergency stop)에 대한 실험에 대해서도 소프트웨어적으로 구현가능 하므로, 이러한 상황에 대비하여 극한 상황이 발생하였을 경우에도 시스템을 가장 빨리 안정화시킬 수 있는 제어방법도 고려한다. 또한, 실제 하드웨어로 구현된 시스템에 적용될 수 있는 가장 간단하면서도 견실한 제어기를 선정한다.

2.3 LAN을 이용한 플랜트/제어기의 통신

플랜트와 제어기가 구성되면 이들 사이의 통신문제를 고려하여야 한다. 여기서 두 대의 PC는 LAN을 통하여 통신이 이루어지며, 한 대는 플랜트가 되고, 다른 한 대는 제어기의 역할을 수행하도록 구성된다. 이것의 개념도는 그림 1과 같다.

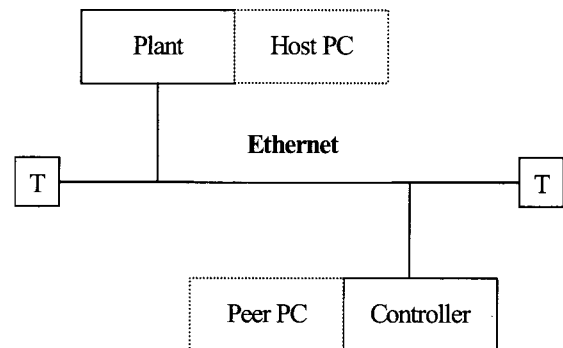


그림 1. LAN을 통한 플랜트/제어기 통신.

CEMTool/SIMTool은 기본적으로 LAN을 통한 통신기능을 지원해 주고 있다. 이러한 기능을 이용하

기 위해서는 플랜트와 제어기로 동작하는 PC가 네트워크 상에서 할당받은 컴퓨터 이름을 지정해 주어야 한다.

구체적으로, 호스트(host)는 플랜트로 지정된 컴퓨터의 컴퓨터 이름을 설정하고, 피어(peer)는 제어기가 있는 컴퓨터의 이름을 설정한다. 그리고 플랜트의 모델은 C 혹은 C++로 프로그램을 하고 제어기는 CEMTool/SIMTool로 구성하도록 한다. 단, 플랜트 또는 제어기 어느 한쪽에 기준하여 두 대의 PC가 시간적으로 동기화 되도록 프로그램 한다.

3. 플랜트/제어기를 구성할 때 고려되어야 할 사항: 실시간 처리의 개념

3.1 실시간의 정의

본 논고에서 사용하는 실시간 처리에 대한 정의는 다음과 같이 한다. 플랜트 및 제어기의 알고리즘 계산이 설정된 시간증분 안에서 계산이 완전히 수행될 수 있어서 시간지연이 발생하지 않는 경우에 대해 실시간 처리라고 정의한다. 이것은 아주 제한적인 개념의 실시간 정의이기는 하지만, PC에서의 실시간에 대한 정의는 실제로 어렵기 때문에 충분히 명백한 정의로 받아들여질 수 있을 것이다. 왜냐하면 PC에서 사용되는 DOS 혹은 WINDOWS 같은 운영체제는 엄격히 실시간 운영체제가 아니기 때문이다. 실시간 제어의 정확한 개념은 실시간 운영체제(RTOS: real time operating system)의 사용과 실시간 스케줄링(scheduling)으로 실시간으로 모든 산출이 이루어지는 적시성과 실시간 이내에 어떤 계산이 수행되지 못할 경우 그 값을 대체할 수 있는 신뢰성(reliability)을 갖는 제어개념을 말한다[3]. 그러나 PC에서 LAN을 통한 통신은 핸드 셰이킹(hand shaking)의 방법에 의해서 이루어지므로 실제 설정된 시간증분 안에서 계산이 끝나지 않더라도 문제가 되지 않는다. 데이터를 주고받는 과정이 이루어지지 않으면, 계산이 느린 컴퓨터가 지연요소가 되어 다음 단계로 진행이 되지 못하므로 엄격히 실시간 정의에서 말하는 적시성의 요소를 갖추지 못할 수 있다. 즉, 실시간으로 연산이 이루어지지 않더라도 PC로부터 통제가 이루어지지 않는다. 따라서 PC로써 실시간 제어를 하기 위해서는 핸드 셰이킹에 의해서 시간지연이 발생하지 않도록 시간증분을 신중하게 선택하여야 한다. 그리고 설정된 시간증분 내에서 i) 플랜트의 운동방정식의 해가 구하여 지고, ii) 제어기가 플랜트에 가할 제어입력을 계산하고, iii) 플랜트/제어기가 데이터를 모두 주고 받을 수 있어야 한다. 이것은 제어시스템 설계자가 사전에 엄밀히 고려하여야 한다.

3.2 시간증분의 선정

일반적으로 PC에서 사용되는 시간증분은 CPU의 연산속도에 의존한다. 따라서 얼마나 작은 시간증분을 사용할 수 있는가 하는 문제도 CPU의 연산속도에 의존하여 결정될 수 있다. 그러나 CPU의 계산에 영향을 미치는 외부 요소들(컴퓨터에서 사용하는 시스템 프로그램의 수, 기본적인 입출력 장치들, 디스플레이 속도 등)로 인해서 CPU의 연산속도를 기준으로 시간증분을 설정하기는 매우 어렵다. 따라서 CPU의 연산속도 대신에 PC의 내부 타이머가 1초에 18.2회의 인터럽트(interrupt)를 호출하는 것을 이용하여, 이것을 기준으로 시간증분을 설정하는 것이 바람직하다. 즉, PC의 내부 타이머 클럭(clock)을 기준으로 한 0.054945초를 시간증분으로 사용하여 이 시간 이내에 플랜트 운동방정식 해의 계산, 제어알고리즘의 계산, 플랜트와 제어기의 통신이 모두 가능하면 앞에서 정의한 실시간 처리의 개념이 된다. 그러나 플랜트의 시스템 차수가 높고, 계산량이 많은 경우나 제어알고리즘이 복잡하여 이 시간증분 이내에 계산을 마칠 수 없는 경우에는 시간 스케일링(time scaling) 기능을 이용한다. 이것은 플랜트의 동적 거동을 임의적으로 몇 배 천천히 계산하여 제어기를 동작시킨 다음 모든 계산이 이루어 졌을 때 다시 시간을 그 배수만큼 나누어서 결과를 보는 방법이다. 이러한 기능을 이용하면 플랜트와 제어기가 아무리 복잡하다 하더라도 앞에서 정의한 실시간 처리의 개념을 적용할 수가 있는 것이다. CEMTool에서는 이러한 시간 스케일링 기능을 지원한다. 이제 이러한 내용들을 바탕으로 실제 두 대의 PC를 사용한 자동제어실험을 예제를 통해서 알아보자.

4. 컨테이너 크레인 제어에의 응용

4.1 모델설명

4.1.1 제어개요

컨테이너 크레인은 항만에서 화물용 컨테이너들을 선박으로부터 컨테이너 트럭으로 하역/선적하는 장비이다. 컨테이너 크레인의 최대 관건은 주어진 시간에 얼마나 많은 수의 컨테이너를 운반할 수 있는가이다. 항만에서 화물의 운반과정을 살펴보면, 화물을 들어올려 일정한 높이가 되는 순간부터 트롤리가 가속하기 시작하고 최대속력에 이른 후 그 속력으로 일정지점까지 등속으로 움직이다가 목표 지점 전방에서 감속하기 시작한다. 그리고 트롤리가 목표지점에 도착하면 화물을 내리는 일련의 동작으로 이루어진다. 이때, 가속 및 감속과정에서 운

반물의 흔들림은 자연적으로 발생하게 되며, 목표 지점에 도달한 후에도 이러한 흔들림이 계속해서 남아있을 경우에는 적하작업이 어려워 작업능률이 저하된다. 또한, 운반 도중에 발생하는 돌발적인 사태와 바람등과 같은 외란에 의한 흔들림도 신속하게 제거하여 줄 필요가 있다. 따라서, 크레인 제어의 목적은 최단시간에 트롤리를 목표위치로 이동시키고, 트롤리의 가감속으로 인하여 발생하는 화물의 흔들림과 운반과정에서 로우프의 길이변화, 바람 등으로 인하여 발생하는 예측하지 못한 흔들림까지도 트롤리가 목표위치에 도착하였을 때 0이 되게끔 하는 데 있다. 또한, 트롤리가 주행도중 갑작스럽게 정지하는 경우에도 빠른 시간 내에 흔들림을 감쇠시켜 줄 수 있는 제어성능도 필요하다. 화물의 신속한 이송과 목표지점에서의 0의 흔들림을 얻기 위하여 전체 제어루프를 두 단계로 나누어 고려한다.

4.1.2 모델링

그림 2는 컨테이너 크레인의 개략적인 도식도이다. 트롤리의 수평방향으로의 이동은 트롤리에 연결된 로우프를 트롤리 드럼에 원하는 방향으로 감음으로서 좌우로 이동하게 된다. 컨테이너의 상하방향으로의 이동은 호이스트 모터의 회전으로 인해 로우프가 호이스트 드럼에 감김으로서 상승하게 되며, 화물 및 스프레더의 자중으로 인해 하강하게 된다. 화물을 감아올리고 내리는 것도 트롤리의 경우와 마찬가지로 호이스트 모터를 양방향으로 구동함으로써 올리고 내릴 수 있으나, 화물의 자중에 의해 하강하는 모델을 대상으로 한다.

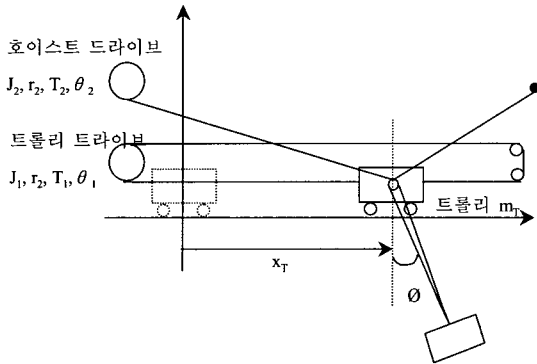


그림 2. 컨테이너 크레인의 도식도.

x 를 트롤리의 변위, l 을 로우프의 길이, 그리고 ϕ 를 컨테이너의 흔들림각이라 할 때 이들에 대한 운동방정식은 다음과 같다[4].

$$(m_1 + m_T + m_L)\ddot{x} + m_L l \ddot{\phi} - m_L l \dot{\phi}^2 \sin \phi + 2m_L l \cos \phi \dot{\phi} + m_L l \cos \phi \ddot{\phi} = F_1 - b_1 \dot{x} \quad (1)$$

$$\frac{1}{2} m_L \dot{x} \sin \phi + 2(m_2 + \frac{1}{4} m_L) \dot{l} - \frac{1}{2} m_L l \dot{\phi}^2 - \frac{1}{2} m_L g \cos \phi = F_2 - b_2 \dot{l} \quad (2)$$

$$l\ddot{\phi} + 2\dot{l}\dot{\phi} + g \sin \phi + \ddot{x} \cos \phi = 0 \quad (3)$$

여기서, m_T = 트롤리의 질량, m_L = 컨테이너의 질량, m_1 = 트롤리 회전부의 등가질량, F_1 = 트롤리 모터의 견인력, F_2 = 호이스트 모터의 견인력, 그리고 b_1 및 b_2 는 트롤리와 호이스트의 움직임에 저항하여 나타나는 비보존 마찰계수이다. 컨테이너의 흔들림각도 ϕ 는 작고 로우프의 길이가 일정하다고 가정할 때 최종적으로 얻어지는 단순화된 시스템은 다음과 같다.

$$(m_1 + m_T)\dot{x} + b\dot{x} = F_1, \quad x(0) = x_0, \quad \dot{x}(0) = 0 \quad (4)$$

$$(2m_2 + \frac{1}{2} m_L)\dot{l} - \frac{1}{2} m_L g = F_2, \quad l(0) = l_0, \quad \dot{l}(0) = 0 \quad (5)$$

$$\ddot{\phi}(t) + \frac{g}{l} \phi(t) = -\frac{\ddot{x}(t)}{l}, \quad \phi(0) = 0, \quad \dot{\phi}(0) = 0 \quad (6)$$

4.1.3 경로계획

그림 3은 컨테이너가 출발점에서부터 목표지점까지 운반되는 경로를 나타낸다. 각 구간에서의 컨테이너의 운반과정을 살펴보면, AB 구간에서는 주변에 장애물이 없는 높이까지 화물을 상승시키게 된다. B 지점에서 화물을 상승시키는 최대 속도가 얻어진다. BC 구간은 트롤리가 가속되는 구간으로서 트롤리와 호이스트는 서로 독립적으로 제어가 된다. 이때 호이스트는 트롤리가 주행도중 만날 수 있는 모든 장애물을 피할 수 있는 높이까지 상승하게 되고, C 지점에서 트롤리의 최대속도가 얻어진다. BC 구간을 통한 트롤리의 가속패턴은 가속이 끝난 시점에서 흔들림이 없어야 한다는 경계조건으로부터 유도된다. CD 구간을 통해서는 트롤리가 최대속도로 주행하는 최대정속구간이다. 트롤리의 최대속도는 기계적인 요소에 의해서 결정된다. DE 구간에서는 트롤리는 감속하고 호이스트는 하강하게 되는데 E 지점에서 화물을 하강시키는 최대속도가 된다. 최종적으로 EF 구간을 통하여 화물이 목표높이로

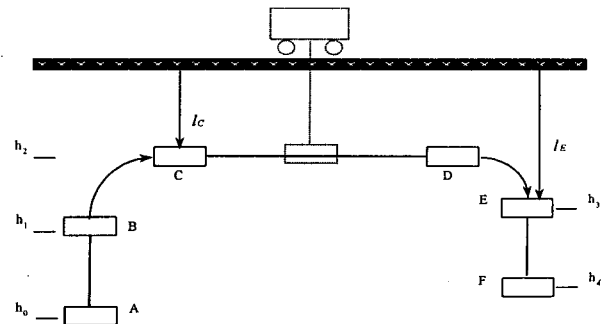


그림 3. 컨테이너의 운반경로.

내려지게 된다.

4.1.4 제어계획

경로계획을 통한 화물의 운반과정을 바탕으로 크게 2 부분으로 구별되는 제어전략을 제시한다.

첫번째는 BE 구간으로 트롤리가 시간최적제어를 통해서 유도된 속도선도를 따라서 주행하는 부분이며, 이 구간에서는 트롤리와 호이스트가 서로 독립적으로 제어가 된다. BE 구간에서는 단순화된 모델(6)이 플랜트로 사용된다.

두번째는 EF 구간으로 시스템의 비선형성이 모두 포함된 운동방정식 (1)~(3)으로부터 비선형 제어가 설계된다. 잔류흔들림을 신속하게 제어할 목적으로 부분 피드백 선형화와 가변구조제어의 기법을 병행한 비선형 제어기법을 적용하게 된다. 그림 4는 트롤리 주행거리와 로우프 길이에 따른 제어계획을 나타내고 있다. x_d 는 트롤리의 목표지점을 나타내고, x_{min} 은 시간최적주행을 하기 위한 최소한의 거리를 나타낸다. 로우프의 길이가 일정하고, 바람도 없고, 또한 초기 흔들림도 아주 작은 경우 ($\phi_0 < \phi_{min}$)에는 개루프의 트롤리 주행이 매우 효과적이다(#1). 여기서 ϕ_0 는 초기 흔들림각도이고, ϕ_{min} 은 흔들림정도를 판단하기 위한 기준값이다. 그러나, 로우프의 길이가 일정한 경우라도, 초기에 심한 흔들림 있는 경우에도 가속시 Error Feedback을 적용하여 최고속도에 도달시점에서 초기흔들림에 의한 영향을 최소화 할 필요가 있다(#2). 로우프의 길이가 일정한 경우는 그림 3의 경로계획에서 B지점의 높이가 h_2 가 되는 경우로 호이스팅을 완전히 마친후 트롤리 주행을 시작하겠다는 개념이다. 시간적으로는 중간높이 h_1 지점에서 트롤리 주행을 시작하는 것보다 손해이지만 일정한 로우프의 길이로 주행한다는 측면에서 간단한 제어법칙도 매우 효과적이다. 로우프의 길이가 변하는 경우에 가속시에는 C 지점의 로우프 길이를 기준으로 하고, 감

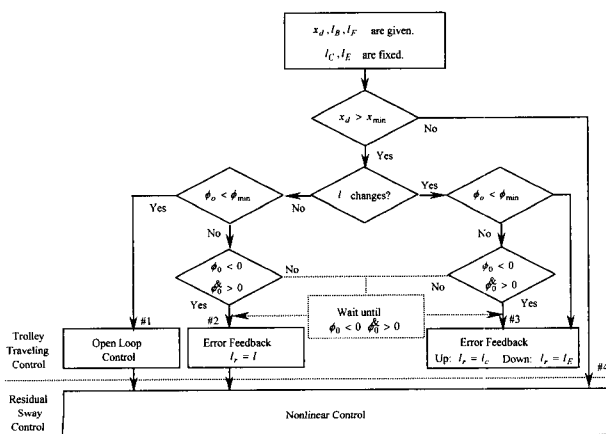


그림 4. 트롤리 주행제어 흐름도.

속시에는 E 지점의 로우프 길이를 기준으로 하는 것이 바람직하다(#3). 그리고, 목표위치(x_d)가 bang-bang 제어기법으로 트롤리의 최대속도까지 가속후 다시 정지하기 까지 필요한 최소거리(x_{min}) 보다 작을 경우에는 처음부터 비선형 제어기법을 사용하는 것이 바람직하다(#4).

4.2 모델윈도우의 구성

다음의 그림 5는 컨테이너 크레인의 제어실험을 위한 모델윈도우를 보이고 있다. 전체 모델윈도우의 구성은 anti-sway system을 운영하는데 필요한 사항들을 기초로 하여 실제 anti-sway system의 감시 프로그램과 유사하게 구성한다. 또한 크레인 시스템의 동역학 계산에 따른 움직임을 동영상으로 표현하여 실제 시스템의 거동을 이해하는데 도움이 되도록 한다. 모델윈도우의 구성을 살펴보면 크레인 시스템 실험을 위한 조작부분, anti-sway system 감시를 위한 출력부분, ethernet 통신을 위해 address를 지정할 수 있는 부분, 제어실험을 위한 시험모델을 선정하는 부분, 크레인 시스템 제어를 통해 나타나는 결과를 그래프로 나타내는 부분으로 나뉜다.

4.2.1 제어실험을 위한 시험모델

■선형시불변 시스템(LTI system): 흔들림 각도가 작고, 로우프 길이가 변하지 않는다는 가정하에 단순화된 시스템으로서 (6)과 같이 표현된다. 시간최적제어와 입력성형(Input Shaping) 기법 등이 적용될 수 있다.

■선형시변 시스템(LTV system): (6)과 같이 표현되는 선형 시스템에서 로우프 길이 변화를 고려한 것이다. 시뮬레이션은 로우프 길이변화 및 초기흔들림의 영향을 고려한다.

이때 시간최적제어에 의한 기준 속도선도로부터

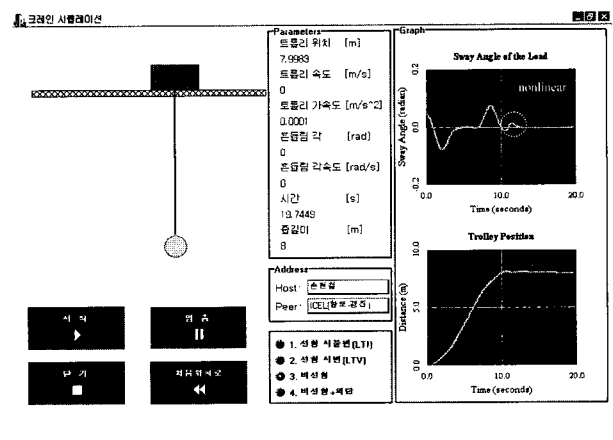


그림 5. 컨테이너 크레인의 모델윈도우. (시간최적주행+가변구조제어의 시뮬레이션)

유도된 흔들림과 실제 시스템으로부터의 흔들림과의 차이를 피드백하여 속도선도에 오차입력을 가하는 오차 피드백 제어가 적용된다[4].

■비선형 시스템: (1)-(3)으로 표현되는 시스템을 고려한다. 이 모델에는 로우프의 길이변화와 흔들림 각도가 결합된 동역학이 포함된다. 여기서는 2단계 제어전략이 적용되는데 트롤리의 주행제어는 기준 속도선도에 의한 오차 피드백제어, 목표위치에서의 잔류 흔들림제어는 피드백 선형화를 통한 가변구조 제어가 적용된다.

■비선형 시스템 + 외란: (1)-(3)의 비선형 시스템에 바람 등에 의한 외란의 영향을 추가한다. 단, 외란의 값의 범위는 정해져 있는 것으로 가정하며 가변구조제어의 외란에 대한 견실한 성능을 평가할 수 있다.

4.2.2 ANTI-SWAY SYSTEM 출력부분

■트롤리 위치: 트롤리가 정확한 목표위치에 도달하는지를 알 수 있다. 실제 항만에서 사용되는 크레인 시스템의 경우에 트롤리 위치오차의 범위는 $\pm 10\text{mm}$ 이다.

■트롤리 속도: 시스템에 가해지는 속도를 알 수 있다. 시스템의 기계적 특성을 고려하여 최대속도를 결정할 수 있다.

■트롤리 가속도: 실제 제어기로부터 계산되어 플랜트에 가해지는 입력이다.

■흔들림 각도: 제어입력에 의해서 나타나는 흔들림 각도를 나타낸다. 실제 항만에서 사용되는 크레인 시스템의 경우에 흔들림의 변위(흔들림 각도가 0일때를 기준으로부터 벗어난 컨테이너의 수평거리)는 $\pm 30\text{mm}$ 이다.

■흔들림 각속도: 흔들림 각도의 변화량을 알 수 있다.

■시간: 설정된 목표위치에 도달하는 시간을 알 수 있다. 빠른 시간 내에 화물을 운반하는 것이 중요하므로 트롤리가 도달하였을 때의 시간이 출력된다.

■로우프 길이: 로우프의 길이변화를 알 수 있다. 축소모델 뿐만 아니라 실제 항만에서 사용되는 값을 이용할 수 있다.

4.2.3 Ethernet 통신을 위한 Address 지정

CEMTool은 기본적으로 LAN을 통한 통신뿐만 아니라 직렬(serial)통신, 병렬(parallel)통신, 포트(port)를 이용한 통신기능을 갖추고 있다. LAN을 이용한 통신은 네트워크에 연결되어 있는 두 대의 PC에 부여된 이름을 지정하면 된다.

■Host Name: 크레인 시스템의 플랜트가 구성된

컴퓨터의 이름을 지정한다.

■Peer Name: 크레인 시스템의 제어기가 구성된 컴퓨터의 이름을 지정한다.

4.2.4 그래프 출력 부분

크레인 시스템 제어의 중요한 상태를 그래프로 표현하여 제어기의 성능을 관찰한다.

■Sway Angle of the Load: 목표위치에서 흔들림이 0이 되는지 확인한다.

■Trolley Position: 트롤리가 정확하게 목표위치에 도달하는지 확인한다.

4.2.5 동영상 윈도우

동역학 계산에 의한 결과를 토대로 크레인 시스템의 거동을 동영상으로 구현하여 실제 크레인 시스템의 거동을 알 수 있다. 트롤리의 움직임, 컨테이너의 흔들림각도, 로우프 길이변화 등이 동영상으로 표현된다.

4.3 시뮬레이션

■선형시불변 시스템: 시간최적제어

시뮬레이션에 적용된 시간최적제어의 입력은 계단형 속도선도를 사용한다. 이유는 노치형이 시간적으로 가장 유리하지만 실제 적용시에 구조물 자체에 진동을 가할 수 있으므로 실제의 상황을 고려하기 위해서이다[5]. 로우프 길이가 일정하고 초기흔들림이 없는 경우에 대해서는 시간최적제어에 의한 개루프 운전으로도 트롤리를 정확하게 목표위치에 위치시킬 뿐만 아니라 목표위치에서의 컨테이너의 흔들림이 0이 됨을 알 수 있다.

■선형시불변 시스템: 입력성형기법

시간최적제어가 적용된 경우와 마찬가지로 로우프 길이가 일정하고, 초기 흔들림이 없는 경우에 대해서는 정확하게 목표위치에 도달하고 흔들림이 0이 된다. 그림 6은 SIMTool을 이용한 제어기이다.

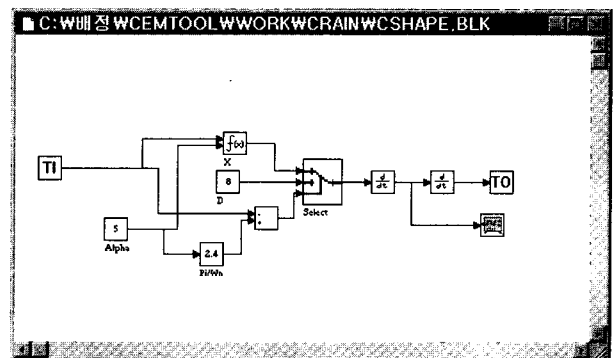


그림 6. 입력성형기법.

■선형시변 시스템: 오차피드백 제어

로우프 길이가 변하고, 초기흔들림이 존재하는 경우에 대해서도 트롤리를 목표위치에 도달시키고, 목표위치에서의 흔들림을 최소로 유지할 수가 있음을 알 수 있다. 단, 오차 피드백의 경우 시뮬레이션 및 실험을 통하여 피드백 이득값을 적절하게 조절할 필요가 있다.

■비선형 시스템: 가변구조제어

그림 5는 비선형 제어를 이용한 시뮬레이션의 결과 그래프를 나타내고 있다. 로우프 길이가 변하고 초기 흔들림이 있는 상황에서도 트롤리를 정확하게 목표위치에 위치시키고, 목표위치에서의 컨테이너의 흔들림이 0이 되게 함을 알 수 있다. 특히 바람 등과 같은 외란으로 남을 수 있는 목표지점에서의 잔류흔들림이 비선형 구간에서 충분히 제어가 됨을 알 수 있다.

■비선형 시스템+외란: 가변구조제어

비선형 플랜트에 바람이나 마찰 등과 같은 외란의 영향에 의한 시스템 동역학을 고려하기 위하여 외란을 추가한 시스템에 가변구조제어를 적용한다. 외란의 존재에도 불구하고 트롤리를 정확하게 목표위치에 위치시키고, 목표위치에서 컨테이너의 흔들림을 0으로 함을 알 수 있다. 따라서 비선형성 및 외란의 존재에도 가변구조제어기는 견실한 성능을 나타냄을 시뮬레이션을 통해서 알 수 있다.

4.4 요약

본 시뮬레이션은 컨테이너 크레인의 운전시 흔들리게 되는 화물의 흔들림제어에 관한 것이다. 플랜트 파라미터의 시변특성, 모델의 불확실성 및 외란 등을 고려하여 트롤리의 주행제어와 잔류흔들림 제어의 2단계에 걸친 제어를 제시하였다. 트롤리의 주행제어는 신속한 화물의 운송을 위하여 시간최적제어에 바탕을 둔 속도선도와 오차 피드백 제어를 기본으로 하고, 잔류흔들림 제어는 화물의 하강시 로우프의 길이가 변화하는 것을 고려한 피드백 선형화와 가변구조 제어를 병합한 비선형제어를 채택하였다. 실험을 통하여 제안된 알고리즘을 확인하였으며, 본 연구의 결과는 실제의 항만 크레인에 직접적으로 사용될 수 있다고 판단된다.

5. 결론

본 논고는 두 대의 PC를 LAN으로 연결하여 사용한 실시간 시뮬레이션의 개념을 소개하였다.

이러한 방법은 기존의 한 대의 PC에서 행해지던 시뮬레이션보다 실험적 고찰 및 실시간 처리

에 대한 개념이 포함되므로 보다 바람직한 방법이라 사료되며, 복잡한 시스템이라 할지라도 프로그램에 의한 가상 플랜트를 구현하기만 한다면 여러 가지 제어기를 적용할 수 있으므로 교육적 목적으로서도 충분한 가치가 있다고 판단된다. 또한 국내에서 개발된 소프트웨어인 CEMTool의 활성화에도 기대를 해본다.

참고문헌

- [1] 황우현, 권육현, CEMTool을 이용한 제어시스템의 설계, 대광서림, 1998.
- [2] K. J. Astrom and B. Wittenmark, *Computer Controlled Systems Theory and Design*, Prentice-Hall, 1984.
- [3] 김학배, "시스템 개발을 위한 특성 분석", 제어·자동화·시스템공학회지, vol. 4, no. 1, pp. 22-27, 1998.
- [4] Keum-Shik Hong, Sung-Chull Sohn, and Man-Hyung Lee, "Sway control of a container crane (Part I): Modeling, control strategy, error feedback control via reference velocity profiles, *J. of Control, Automation, and Systems Engineering*, vol. 3, no. 1, pp. 23-31, 1997.
- [5] Keum-Shik Hong, Sung-Chull Sohn, and Man-Hyung Lee, "Sway control of a container crane (Part II): Regulation of the pendulum sway through patternizing trolley moving velocity", *J. of Control, Automation, and Systems Engineering*, vol. 3, no. 2, pp. 132-138, 1997.
- [6] Keum-Shik Hong, Jae-Hoon Kim, and Kyo-Il Lee, "Control of a container crane: fast traversing and residual sway control from the perspective of controlling an underactuated system", *Proc. of the American Control Conference*, vol. 2, pp. 1294-1298, 1998.
- [7] A. Marttinen, J. Virkkunen, and R. Salminen, "Control study with a pilot crane", *IEEE Trans. on Education*, vol. 33, no. 3, pp. 298-305, 1990.
- [8] J-J. E. Slotine and W. Li, *Applied nonlinear control*, Prentice Hall, 1991.

저자소개

홍 금 식 (洪 金 植)

1957년 8월 25일생

1979년 서울대학교 기계설계학과 (공학사)

1987년 Columbia University 기계공학과 (석사)

1991년 University of Illinois at Urbana, Applied Mathematics (이학석사)

1991년 University of Illinois at Urbana, 기계산

업공학과 (공학박사)

1982년 - 1985년 대우중공업 기술연구소 (인천)

1991년 - 1992년 University of Illinois, Postdoctoral
Research Associate

1993년 - 현재 부산대학교 기계공학부 조교수

<관심분야>

- 적응제어, 비선형제어
- 분포계수시스템의 제어
- 시스템 이론 및 산업응용.

<연락처>

Tel. (051) 510-2454. Fax. (051) 514-0685

Email: kshong@hyowon.pusan.ac.kr

박 배 정 (朴 倍 正)

1974년 8월 5일생

1998년 부산대학교 제어기계공학과 (공학사)

1998년 - 현재 부산대학교 지능기계공학과
석사과정

<관심분야>

- 컨테이너 크레인 제어
- 자동차 현가장치 제어
- 실시간 제어, 로봇제어

<연락처>

Email: bjpark@hyowon.pusan.ac.kr

김 성 훈 (金 成 勳)

1975년 1월 10일생

1997년 부산대학교 제어기계공학과 (공학사)

1997년 - 현재 부산대학교 지능기계공학과
석사과정

<관심분야>

- 예측제어, 시스템규명, 적응제어
- 철강분야의 공정제어
- 실시간 제어

<연락처>

Email: shkim20@hyowon.pusan.ac.kr

정 재 룡 (鄭 載 龍)

1976년 7월 6일생

1999년 부산대학교 기계공학부 (공학사)

1999년 - 현재 부산대학교 지능기계공학과
석사과정

<관심분야>

- 컨테이너 크레인 제어, 로봇제어

<연락처>

Email: jrjung@hyowon.pusan.ac.kr