

〈논 문〉 SAE NO. 943709

## 실시간 시뮬레이션을 위한 병렬적분

Parallel Integration for Real-Time Simulation

이운성,\* J. Samson\*\*  
W. S. Lee,

### ABSTRACT

A parallel integration approach is proposed for real-time simulation of controlled mechanical systems. The proposed approach, which employs the dual-rate integration method in a parallel computing environment, is developed to deal with stiffness and high frequency characteristics of the controlled mechanical systems effectively. Numerical experiments are performed to demonstrate the effectiveness of the approach in shared memory multiprocessors, Alliant FX/8 and Alliant FX/80.

주요기술용어 : Parallel Integration(병렬적분), Real-Time Simulation(실시간시뮬레이션), Parallel Processing(병렬처리), Dual-Rate Integration(이중비율적분)

### 1. 서 론

컴퓨터 기술의 눈부신 발달에 힘입어 복잡한 기계시스템의 실시간 시뮬레이션이 가능해지고 있으며, 작게는 제어용 하드웨어와 알고리즘을 간편하고 경제적으로 검증할 수 있는 하드웨어 포함루프(hardware-in-the-loop) 시뮬레이션<sup>1)</sup>으로부터 크게는 자동차의 개발 및 검증, 운전자의 인자연구(human factors research) 등에 효과적으로 이용되는 운전시뮬레이터<sup>2,3)</sup>까지 그 적용범위를 넓혀 가고 있다. 실시간 시뮬레이션에서는 시뮬레이션의 정확도와 속도라는 서로 상반되는 요구를 동시에 만족시켜야하기 때문에, 비실시간(non-real-time) 시뮬레이션과는 다른 사항을 고

려하여야 한다. 컴퓨터 프로세서(processor)의 처리속도가 빨라짐에 따라 더욱 복잡하고 실제 시스템에 가까운 모델-예를 들면, 유압 및 전기회로의 동특성을 포함하는 다물체 차량동력학 모델-이 고려되고 있으나, 이는 다시 시뮬레이션 시간을 증가시켜 실시간의 구속조건을 만족시키기 어렵게 하며, 따라서 운동방정식을 수치적으로 푸는 실시간 시뮬레이션 전용 수치적분기법의 개발이 필요하게 된다.

시뮬레이션 결과가 실시간에서 얻어져, 예를 들면 운전 시뮬레이터에서 컴퓨터화상과 플래트폼운동의 생성 또는 하드웨어포함루프 시뮬레이션에서 제어입력의 결정 등의 다른 목적으로 쓰이는 실시간 시뮬레이션에서 적분기법의 선정에

\* 정희원, 국민대학교 자동차공학과

\*\* Department of Mechanical Engineering, University of Missouri-Kansas City, U.S.A

는 안정성, 정확성, 편리성 등 여러가지가 고려되어야 한다. 따라서 스텝마다 운동방정식을 한번 푸는 다스텝명시기법(explicit multi-step methods)에 상수스텝을 사용하는 방법이 널리 이용되었다.<sup>24)</sup> Howe<sup>5)</sup>는 Adams-Basforth의 예측기법보다 정확도와 안정성이 좋은 새로운 Adams-Moulton 예측-정정기법을 제안했다. 그러나 스텝마다 운동방정식을 두번 풀어야 하는 이 기법은 복잡한 기구학적 구조로 인해 운동방정식의 생성 및 풀이에 시간이 많이 소요되는 기계시스템의 실시간 시뮬레이션에는 적합하지 않다. 독일 Mercedes-Benz 자동차회사의 운전시뮬레이터에서는 암시적(implicit) Euler기법을 변형시켜 0.01초의 큰 스텝으로 차량 실시간시뮬레이션을 수행하였으나, 이에 쓰인 차량운동방정식은 “주된” (“dominant”) 좌표만을 이용하여 단순화되었고 유압이나 전기회로의 동특성도 무시되었다.<sup>6)</sup>

본 연구에서는 제어되는 복잡한 기계시스템의 실시간 시뮬레이션에 효과적으로 이용될 수 있는 병렬수치적분기법을 개발하고자 한다. 여러 서브시스템이 결합되어 하나의 시스템을 형성하는 경우 각각의 동특성 및 응답속도가 다름으로 해서 생기는 적분상의 어려움을 효과적으로 해결하는 이중비율(dual-rate) 적분기법에 기초를 두고 다물체동력학의 효율적인 해석을 가능케 하는 병렬처리방법(parallel processing methodology)을 적용하는 기법을 개발하고자 한다. 기계시스템의 기구학적 구조의 차이로 인해 모든 시스템의 실시간 시뮬레이션에 적용될 수 있는 범용기법을 개발하기는 어려우나, 본 연구에서 개발한 기법은 그 가능성 및 효과를 보여 준다.

## 2. 제어되는 다물체 기계시스템의 운동방정식

제어되는 다물체 기계시스템의 운동방정식은 각 서브시스템의 운동방정식을 세워 결합함으로써 생성되는 데, 그 과정을 이 절에서 간단히 기술한다. 운동방정식 수립의 자세한 과정은 참고문 7에 기술되어 있다.

다물체 기계서브시스템의 Newton-Euler 운동

방정식은 다음과 같다.<sup>8)</sup>

$$\delta q^T(M\ddot{q} - Q) = 0 \quad (1)$$

여기서  $q$ 는 장체들의 운동을 표시하는 상대좌표벡터,  $M$ 은 일반화된 질량행렬,  $Q$ 는 기계적 또는 제어거동에 의한 일반화된 힘을 나타낸다. 가상변위  $\delta q$ 는 이 시스템안의 밀폐된 기구학적 루프로부터 생성되는 아래의 구속방정식과 모순이 되지 않아야 한다.

$$\Phi(q) = 0 \quad (2)$$

따라서 상대좌표  $q$ 는 전부가 독립적이지는 않다. 이를 종속좌표와 독립좌표, 즉  $q = [u^T, v^T]^T$ 로 나누면, 위의 방정식 (1), (2)를 다음과 같이 다시 쓸 수 있다.

$$[\delta u^T, \delta v^T] \left\{ \begin{bmatrix} M_{uu} & M_{uv} \\ M_{vu} & M_{vv} \end{bmatrix} \begin{bmatrix} \ddot{u} \\ \ddot{v} \end{bmatrix} - \begin{bmatrix} Q_u \\ Q_v \end{bmatrix} \right\} = 0 \quad (3)$$

$$\Phi(u, v) = 0 \quad (4)$$

여기서 방정식 (3)의  $\delta u$ 와  $\delta v$ 는 구속방정식 (4)과 모순되지 않아야 한다. Lagrange 승수정리<sup>9)</sup>에 의하면, 아무런 구속조건이 없는  $\delta u$ 와  $\delta v$ 에 대한 다음과 같은 방정식을 세울 수 있는 Lagrange 승수벡터  $\lambda$ 가 존재한다.

$$[\delta u^T, \delta v^T] \left\{ \begin{bmatrix} M_{uu} & M_{uv} \\ M_{vu} & M_{vv} \end{bmatrix} \begin{bmatrix} \ddot{u} \\ \ddot{v} \end{bmatrix} + \begin{bmatrix} \Phi_u^T \\ \Phi_v^T \end{bmatrix} \lambda - \begin{bmatrix} Q_u \\ Q_v \end{bmatrix} \right\} = 0 \quad (5)$$

방정식 (4)를 1차변이(first-order variation)시켜  $\delta u$ 에 대해 풀면

$$\delta_u = -\Phi_u^{-1} \Phi_v \delta_v \equiv R_1 \delta_v \quad (6)$$

같은 방법으로, (4)를 시간에 대해 두번 미분하고,  $\ddot{u}$ 에 대해 풀면

$$\ddot{u} = -\Phi_u^{-1} \Phi_v \ddot{v} - \Phi_u^{-1} (\Phi_u \dot{u} + \Phi_v \dot{v}) \equiv R_1 \ddot{v} + R_2 \quad (7)$$

방정식 (6)과 (7)을 (5)에 대입하고,  $\delta u$ 의 계수항을 영으로 놓으면 아래와 같이 기계서브시스템의 운동방정식을 독립좌표만의 미분방정식 형태로 구할 수 있다.

$$\bar{M}\ddot{v} = \bar{Q}$$

그리고

$$\bar{M} = R_1^T M_{uu} R_1 + R_1^T M_{uv} + M_{vu} R_1 + M_{vv} \quad (9)$$

$$\bar{Q} = R_1^T Q_u + Q_v - R_1^T M_{uu} R_2 - M_{vu} R_2 \quad (10)$$

전기나 유압회로로 많이 구성되는 제어서브시스템의 동특성은 아래와 같은 상태방정식의 형태로 표시된다.

$$\dot{x} = f(x, v, i, w) \quad (11)$$

$$y = h(x, v, i, w) \quad (12)$$

여기서  $x$ 는 전류, 유압력 등의 상태변수벡터를 표시하고,  $w$ 는 제어알고리즘을 거쳐 가해지는 제어입력을 나타낸다.  $y$ 는 기계서브시스템에 힘 또는 모멘트로써 가해지는 출력벡터를 표시한다. 이 상태방정식은 전기회로의 경우 Kirchhoff 법칙<sup>9)</sup>을 적용하거나, 유압회로의 경우 연속방정식과 액체의 압축성<sup>10)</sup>을 이용하여 만들어진다.

따라서 제어되는 기계시스템의 운동은 방정식 (8)~(12)에 의해 표시된다.

### 3. 병렬적분

#### 3.1 이중비율적분

제어되는 기계시스템을 구성하는 각 서브시스템의 서로 다른 동특성 및 응답속도로 인해 전체시스템은 “경직”(“stiff”)되는데, 이는 수치적분 효율성에도 영향을 미친다. 실시간 시뮬레이션에 널리 쓰이는 상수스텝의 명시적(explicit) 적분기

법에서는 제어서브시스템의 경직도 또는 높은 주파수 특성으로 인해 아주 작은 스텝을 써야만 안정성을 유지할 수 있게 된다. 반면에 암시적(implicit) 적분기법에서는 명시적 적분기법보다는 좋은 안정성으로 해서 큰 스텝을 쓸 수 있으나 (Benz 운전시뮬레이터의 0.01초), 자코비안(jacobian)을 이용해서 비선형방정식의 반복해를 구해야하는 어려움이 있다. 게다가 복잡한 기계서브시스템의 자코비안을 보통 유한차이법을 이용해서 구하기 때문에 매 스텝당 계산량이 많아진다.

이런 서로 다른 시간스케일과 주파수특성을 다루는 가장 좋은 방법은 서브시스템을 분리해 각각에 가장 알맞는 적분기법과 스텝크기를 적용하는 것이다.<sup>11)</sup> 기계서브시스템은 제어서브시스템에 비해 느리고 평坦한 응답특성을 갖기 때문에 Fig.1과 같이 매 스텝당 한번씩 계산하는 명시적 기법에 큰 스텝을 효과적으로 적용할 수 있다. 반면에 빠른 제어서브시스템에는 안정성이 좋은 낮은 차수의 명시적 기법에 작은 스텝을 적용하든가, 또는 절대안정성(absolute stability)을 가지며 효율적인 반 암시적(semi-implicit) 기법을 효과적으로 이용할 수 있다. 반 암시적 기법은 선형방정식의 해만을 필요로 하며<sup>11)</sup>, 해를 구하는 데 쓰이는 제어서브시스템의 자코비안은 많은 경우에 간단하게 구할 수 있기 때문에 암시적 기법보다 훨씬 더 효율적이다. 이중비율적분에 의한 효율성증가는 방정식 계산량, 적분알

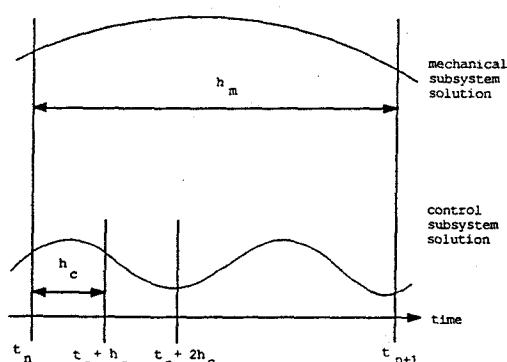


Fig.1 Integration step sizes for mechanical and control subsystems

고리즘, 보간(interpolation) 또는 보외(extrapolation) 알고리즘 등에 좌우되나, 기계시스템의 경우에는 방정식 계산양이 가장 큰 요소로 작용한다. 제어서브시스템의 방정식 계산양은 기계서브시스템에 비해 훨씬 작기 때문에, 기계서브시스템에 큰 스텝의 적분기법을 적용하면 효율성이 많이 증가되리라고 기대된다.

이중비율적분의 적용시 대상시스템의 분리는 적분기법의 효율성 및 안정성에 영향을 미치는 중요한 과제이다. 대상시스템의 물리적 특성이 밝혀지지 않은 경우 고유치(eigenvalues)를 관찰하여 빠른(fast) 부분과 느린(slow) 부분으로 분리하기도 하나 비선형시스템에의 적용 문제 및 적분 중 시스템 동 특성의 변화 가능성 등의 이유로 해서 신뢰성을 갖지 못한다. 따라서 대상시스템의 물리적 특성에 대한 사전 지식의 중요성이 강조되고 있다.<sup>12)</sup>

제어서브시스템의 출력은 일반화된 힘으로써 기계서브시스템에 작용하고, 기계서브시스템의 운동 및 피드백은 제어서브시스템의 상태를 결정하므로, 두 서브시스템은 서로 연결되어 있다. 이 연결성은 이중비율적분 안에서 두 서브시스템이 스텝크기가 다르게 적분될 경우 정보교환에 따른 인터페이스 문제를 일으킨다. 즉 제어서브시스템이 더 작은 스텝으로 적분될 경우 중간점에서 필요한 기계서브시스템의 정보는 얻어지지 않는다.(Fig.1) 이 정보를 얻기 위해서는 어느 서브시스템을 먼저 적분하느냐에 따라 보외 또는 보간이 필요하게 되는데, 이는 이중비율적분의 실제적용에 대한 두 가지 기법으로 이어진다.<sup>12)</sup> 또한 범용목적으로 개발된 이중비율적분기법에서는 적분의 정확도를 높히기 위해 다음 스텝까지 예측해서 적분하고, 이의 상대오차를 계산하고, 오차가 클 경우 스텝크기를 줄여 다시 적분하고, 정확도를 확인하는 예측, 평가, 정정, 평가 과정을 반복하게 된다.

적분기법의 정확도가 요구되는 수준을 어느 정도 만족한다고 가정하면, 실시간시뮬레이션 환경에서는 적분기법의 안정성이 가장 중요한 과제가 되고, 또 실시간의 구속조건을 만족하기 위해 오차계산 및 적분스텝 재조정과정은 생략

해야 한다. 이런 점에 비추어 볼때 기계서브시스템의 보외정보를 이용하여 작은 스텝으로 제어서브시스템을 적분하고 큰 스텝으로 기계서브시스템을 적분하는 방법이 실시간시뮬레이션에 가장 적합하게 된다. 이중비율적분의 안정성을 해치지 않는다면 보외에서 오는 부정확성을 피하기 위해 같은 크기의 스텝을 두 서브시스템에 적용할 수도 있다.

### 3.2 병렬처리

앞에서 기술한 이중비율적분은 두 서브시스템의 방정식 계산 및 적분을 각각 독립적으로 시행하는데 그 기본원리를 둔다. 이는 병렬컴퓨터의 여러 프로세서를 이용해 각 서브시스템의 계산을 독립적으로 할수 있기 때문에 병렬처리환경에서 더욱 효과적이 된다. 두 서브시스템의 분리로 인해 Fig.2에서처럼 높은 페벨의 병렬도(parallelism)가 우선 생성되며, 또한 독립된 연쇄(kinematic chain)를 따라 각 서브시스템의 운동방정식을 계산하기 때문에 기구학적 구조에 따른 낮은

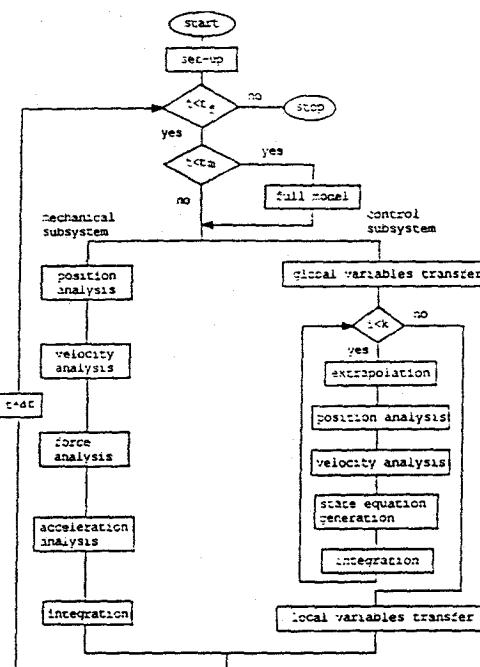


Fig.2 Computational flow diagram for parallel integration

레벨의 병렬도도 생성할 수 있다. 각 체인의 독립적 계산에 기초를 둔 다물체 동력학의 병렬처리기법은 최근 활발히 연구되고 있다.<sup>4,13)</sup>

제어되는 기계시스템 동특성 해석의 병렬처리에는 해석과정 중 여러가지 데이터와 정보를 각 프로세서끼리 자주 교환해야하기 때문에 메모리 공유 다중프로세서(shared memory multiprocessors)를 이용하는 것이 가장 적합하다. 그러나 이런 병렬컴퓨터의 컴파일러는 현재 병렬처리의 국지적인 최적화기능만을 갖고있기 때문에 병렬처리 효율성을 극대화하기 위해서는 시스템의 기구학적 구조, 각 병렬처리과제의 계산량, 프로세서의 수 및 성능 등 여러 인자를 고려한 균형있는 병렬처리계획(multitasking plan)에 기초를 두고 전체적으로 최적화된 병렬처리 프로그램을 생성해야 한다.<sup>13)</sup>

### 3.3 알고리즘

실시간시뮬레이션을 위한 병렬적분 알고리즘을 다음과 같이 제안한다.(Fig.2) :

- 1) 시뮬레이션 시작 때 대상 시스템의 실계수와 초기조건 값을 읽는다.
- 2) 시작  $t=t_n$  일 때 시뮬레이션 최종시각에 도달했는지를 확인한다.
- 3) 병렬적분 수행중 제어서브시스템 방정식을 중간단계에서 계산하기 위해서는 기계서브시스템의 보의정보가 필요하다. 따라서 처음 몇 스텝동안은 전체시스템을 동시에 적분해서 기계서브시스템에 대한 정보를 저장한다. 병렬적분이 일단 시작되면 각 서브시스템의 해석이 독자적으로 진행된다.
- 4) 기계서브시스템에서는 독립된 연쇄를 따라 병렬처리기법을 적용하여 운동방정식을 세우고 가속도를 계산한 후, 이를 다음 스텝, 즉  $t_n+h_m$ 까지 적분한다. 적분효율성을 높히기 위해 Table 1에 제시된 2차 또는 3차의 Adams-Bashforth 적분기법을 적용한다. 적분후 주 루프로 돌아간다.
- 5) 제어서브시스템에서는 병렬처리로 인해 변수 값이 변하지 않도록 우선 국지변수에 값을 저장한다. 미리 저장된 기계서브시스

Table 1 Integration formulas for  $y'=f(t, y)$  in parallel integration

#### Mechanical Subsystem

2nd order Adams-Bashforth

$$y_{n+1} = y_n + h/2(3f(t_n, y_n) - f(t_{n-1}, y_{n-1}))$$

3rd order Adams-Bashforth

$$y_{n+1} = y_n + h/12(23f(t_n, y_n) - 16f(t_{n-1}, y_{n-1}) + 5f(t_{n-2}, y_{n-2}))$$

#### Control Subsystem

Euler

$$y_{n+1} = y_n + h/2f(t_n, y_n)$$

Semi-implicit Euler

$$\Delta_n = [I - hf_y(t_n, y_n)]^{-1}hf(t_n, y_n)$$

$$y_{n+1} = y_n + \Delta_n$$

Semi-implicit midpoint

$$\Delta_n = \Delta_{n-1} + 2[I - hf_y(t_n, y_n)]^{-1}(hf(t_n, y_n) - \Delta_{n-1})$$

$$y_{n+1} = y_n + \Delta_n$$

템 정보를 보의하여 서브시스템 상태방정식을 계산한다. 여기서도 서브시스템의 기구학적 구조, 액추에이터의 수 등을 고려해 병렬처리기법을 적용한다. 다음 제어서브시스템 상태변수만을  $t_n+h_m$ 까지 적분한다. 이 과정은  $t=t_n+h_m$ 이 될 때 까지 계속된다. 적분안정성을 높히기 위해 Euler 기법에 작은 스텝을 적용하든가, 또는 절대안정성을 가진 반 암시적 Euler 또는 중간점기법을 적용한다.(Table 1) 적분 후 국지변수 값을 옮기고 주 루프로 돌아간다.

- 6) 각 서브시스템의 독자적 계산이 합쳐진 후, 다시 단계 2로 되돌아간다.

### 4. 수치실험 및 고찰

앞 절에서 제안한 병렬적분 알고리즘의 효율성을 검증하기 위해 미니수퍼급 병렬컴퓨터 Alliant FX/8(프로세서 8개)과 Alliant FX/80(프로세서 6개)을 이용하여 아래와 같이 수치실험을 수행하고 그 결과를 고찰하였다.

#### 4.1 유압액추에이터 구동 로봇팔

Fig.3은 유압서보시스템에 의해 구동되는 로봇팔의 구조를 보여준다. 그림에서 각 강체는 원내의 번호로 지정되고, 기계서브시스템의 운동은 레블루트 조인트 각을 지정하는 상대좌표  $q_1 \cdots q_4$  와 원통조인트의 복진 및 회전운동을 지정하는  $q_5$ 와  $q_6$ 로 표시된다. 강체 3-4-5-6으로 이루어지는 기구학적 밀폐루프로 인해 자유도는 3이 되며, 독립좌표는  $q_1 \cdots q_3$ 로 선정된다. 유압서보시스템은 스팔밸브와 복동액추에이터로 구성되며, 이의 운동은 각 유압실린더의 압력을 상태변수  $x_1 \cdots x_6$ 로 정의하여 표시된다. 시스템 계수 값은 Table 2에 주어진다.

병렬적분 알고리즘을 적용하기 위한 병렬과제 그래프(task flow graph)는 각 과제의 자세한 설

명은 생략하고 Fig.4에 주어진다. 병렬적분에서 생성되는 여러 레벨의 병렬도를 효과적으로 다룰 수 있도록 Kuhl<sup>[4]</sup>이 개발한 병렬파제프리미티브(tasking primitives)를 이용하여 병렬프로그램을 만들었다. Alliant FX/8의 8개의 프로세서를 나누

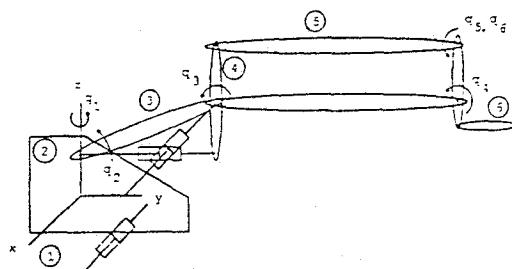


Fig.3 Mechanical manipulator actuated by hydraulic servomechanism

Table 2 Physical parameters of mechanical manipulator

#### Hydraulic Subsystem

##### Mechanical Subsystem

Body no.	Center of gravity (m)			Mass (kg)	Moment of inertia (kg·m <sup>2</sup> )		
	x	y	z		I <sub>xx</sub>	I <sub>yy</sub>	I <sub>zz</sub>
1	0	-3.0	0	-	-	-	-
2	0	0	0	219	1356	1356	1356
3	0	3.2	2.4	73	678	67.8	678
4	0	3.2	2.4	15	67.8	67.8	5.4
5	0	5.0	3.4	30	203.4	13.6	203.4
6	0	6.8	2.2	15	67.8	40.7	27.1

#### Hydraulic Subsystem

Parameter	Actuator 1	Actuator 2	Actuator 3
Supply pressure(kN/m <sup>2</sup> )	$1.38 \times 10^4$	$1.38 \times 10^4$	$1.38 \times 10^4$
Bulk modulus(MN/m <sup>2</sup> )	$1.03 \times 10^3$	$1.03 \times 10^3$	$1.03 \times 10^3$
Area of actuator(m <sup>2</sup> )	$1.94 \times 10^{-2}$	$1.55 \times 10^{-2}$	$1.55 \times 10^{-2}$
Initial volume(left) (m <sup>3</sup> )	$1.23 \times 10^{-2}$	$3.28 \times 10^{-3}$	$3.28 \times 10^{-3}$
Initial volume(right) (m <sup>3</sup> )	$1.23 \times 10^{-2}$	$3.28 \times 10^{-3}$	$3.28 \times 10^{-3}$
Initial pressure(left) (kN/m <sup>2</sup> )	$8.14 \times 10^3$	$6.9 \times 10^3$	$6.9 \times 10^3$
Initial pressure(right) (kN/m <sup>2</sup> )	$5.66 \times 10^3$	$6.9 \times 10^3$	$6.9 \times 10^3$
Flow coefficient	$1.49 \times 10^{-3}$	$7.45 \times 10^{-4}$	$7.45 \times 10^{-4}$

어 5개는 기계부분, 나머지 3개는 유압부분의 계산을 맡도록 하였다. 유압서보시스템에는  $2 \times 10^{-4}$ 초의 스텝으로 Euler 기법을 적용하고, 기계서보시스템에는  $1 \times 10^{-3}$ 초의 스텝으로 3차 Adams-Bashforth 기법을 적용하였다. 유압서보시스템의 각 스플밸브에 1.2, -0.9, 1.2mm의 계단 입력을 가하고 3초동안 시뮬레이션을 수행하여, 그 결과를 전체시스템에  $2 \times 10^{-4}$ 초의 스텝으로 3차 Adams-Bashforth 기법을 적용하여 얻은 결과와 비교하였다. 그 결과로써 Fig.5는 강체 6의 수직방향 위치, 속도, 가속도를 보이는 데, 두 적분기법 사이의 차이점은 거의 보이지 않는다.

Fig.6은 적분스텝 당의 계산량과 사용된 프로

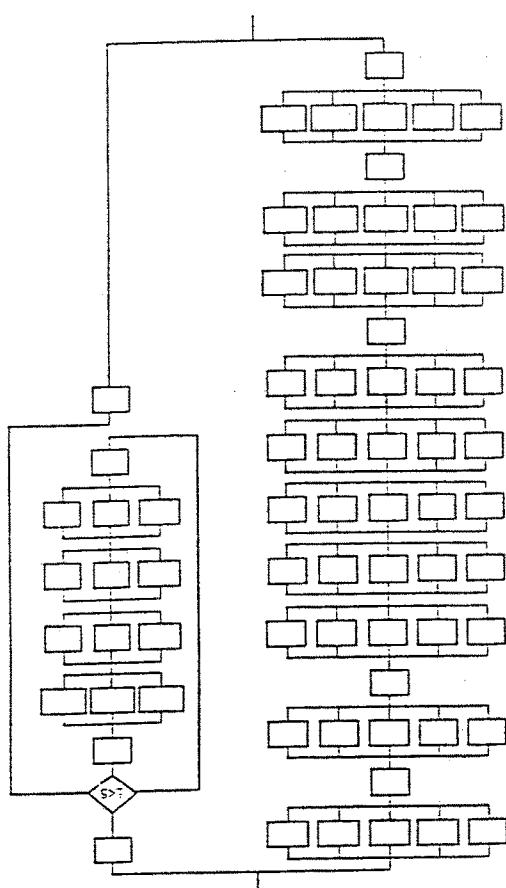


Fig.4 Task flow graph for parallel implementation of manipulator dynamics

세서 수와의 관계를 보인다. 1개의 프로세서를 이용했을 때와 비교할 때 최대속도증가비는 4.04를 기록한다. 프로세서 사용비율은 4개의 프로세서의 경우 66.2%인 반면 8개의 경우 50.4%로 떨어지는 데 이는 더 많은 프로세서가 쓰일 경우 직렬계산부분의 비중이 커지기 때문이다.

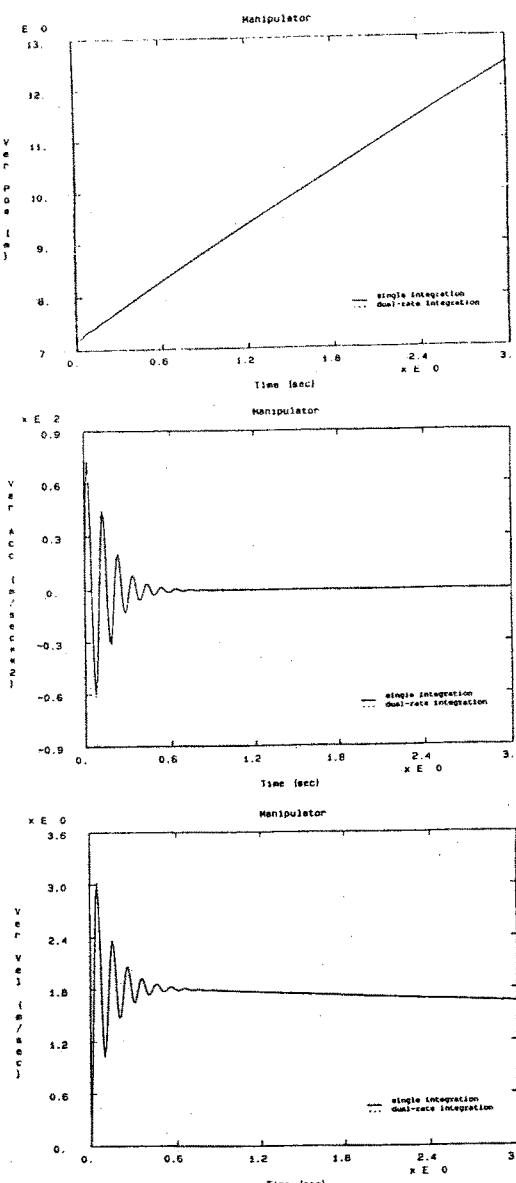


Fig.5 Vertical position, velocity, and acceleration of body 6

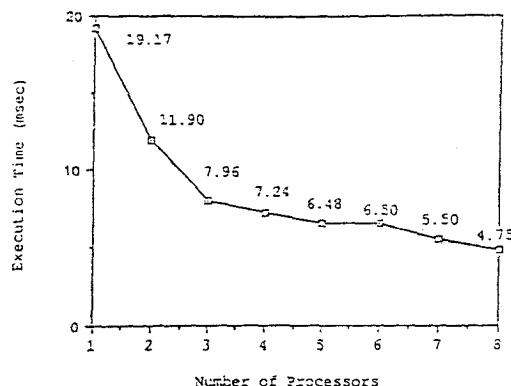


Fig.6 Performance of parallel implementation of manipulator dynamics

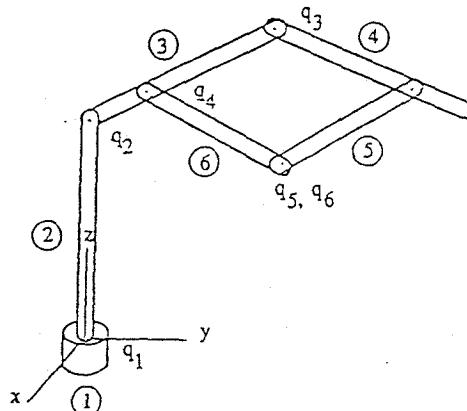


Fig.7 Robot arm driven by DC motors

#### 4.2 직류모터 구동 로봇팔

직류모터로 구동되는 로봇팔의 구조는 Fig.7에 보인다. 기계서브시스템의 구조는 앞 예와 같고  $q_1 \sim q_6$ 은 상대좌표로 지정된다. 제어서브시스템의 상태변수는 강체 1과 2, 2와 3, 3과 4 사이의 레볼루트 조인트에 장착된 직류모터의 전류  $x_1 \sim x_3$ 로 정의된다. 시스템의 실계수는 Table 3에 주어진다. 이 시스템은 외란에도 불구하고  $q_1=0$ ,  $q_2=45$ ,  $q_3=300$ 도의 값을 유지하도록 하는 밀폐

루프 시스템으로, 직류모터의 입력  $w$ 는 본래의 비선형 운동방정식을 선형화한 후 풀위치지정 제어기법을 이용하여 다음과 같이 결정된다.

$$w = -K(z - z^*) \quad (13)$$

여기서  $z$ 와  $z^*$ 는 이 시스템의 현재와 기준상태벡터를 나타내고,  $K$ 는 피드백개인 행렬로써 MATLAB<sup>15)</sup>를 이용해서 구했다.

Table 3 Physical parameters of robot arm

#### Mechanical Subsystem

Body no.	Center of Gravity (m)			Mass (kg)	Moment of Inertia (kg-m <sup>2</sup> )		
	x	y	z		I <sub>xx</sub>	I <sub>yy</sub>	I <sub>zz</sub>
1	0	0	0	—	—	—	—
2	0	0	1	2	0.7	0.7	0.04
3	0	0.866	2.503	2	0.7	0.04	0.7
4	0	2.598	2.512	2	0.7	0.04	0.7
5	0	2.385	1.888	1.5	0.4	0.04	0.4
6	0	1.084	1.881	1.5	0.4	0.04	0.4

#### Motor Subsystem

Parameter	Motor 1	Motor 2	Motor 3
Resistance( $\Omega$ )	5	5	5
Inductance(h)	0.02	0.02	0.02
Back emf Gain(V/(rad/sec))	0.5	0.5	0.5
Gain(n-m/A)	20	15	10

$$K = \begin{bmatrix} 337.4 & 0.18 & 0.072 & 6834 & 0.018 & 0.005 & 0.2 & 0.0 & 0.0 \\ 0.079 & 13203 & 505.1 & -0.002 & 139.2 & 46.97 & 0.0 & 0.36 & -0.059 \\ -0.046 & 534.4 & 403.5 & 0.002 & 100.8 & 62.03 & 0.0 & 0.053 & 0.32 \end{bmatrix} \quad (14)$$

이 예의 병렬과제 그래프는 앞 예와 비슷해 생략하나, 차이점은 제어서브시스템의 모터운동 방정식은 앞의 유압서보시스템 방정식보다 간단하기 때문에 Alliant FX/80을 이용하여 시뮬레이션을 수행할 때 1개의 프로세서 만을 써 직렬 처리를 한다는 것이다. 이 시스템의 경직도가 앞 예보다 적기 때문에 3차 Adams-Basforth와 Euler 기법에 기초를 둔 병렬적분기법을 적용할 경우 적분스텝을 각각  $2 \times 10^{-2}$ 초와  $5 \times 10^{-3}$ 초까지 증가시킬 수 있다. 참고로 전체시스템에 3차 Adams-Basforth 기법만을 적용할 경우  $2 \times 10^{-3}$ 초보다 큰 스텝을 쓰면 안정성을 잃게된다. Euler 기법 대신에 절대안정성을 가진 반 암시적 중간점기법을 제어서브시스템에 적용하면 두 서브시스템에 같은  $3 \times 10^{-2}$ 초의 큰 스텝을 쓸 수 있게 된다. 그러나 제어서브시스템 계산량은 기계서브시스템에 비해 훨씬 적기 때문에 병렬적분 내에서 각 서브시스템을 독립적으로 다룰 경우 CPU 시간을 증가시키지 않으면서 제어서브시스템의 적분스텝을 1/4까지 줄일 수 있었다. Fig.8은 각각의 적분기법을 적용하여 계산된 레볼루트 조인트 각을 비교하는 데 차이는 아주 작다.

## 5. 결 론

본 연구에서는 제어되는 기계시스템을 구성하는 각 서브시스템의 서로 다른 시간스케일 및 주파수 특성으로 인한 적분상의 어려움을 해결하고 실시간시뮬레이션에 효과적으로 적용할 수 있는 병렬적분기법을 개발하였고, 그의 가능성 및 효율성을 미니수퍼급 병렬컴퓨터를 이용하여 검증하였다. 제안된 기법의 효율성을 극대화하기 위해서는 서브시스템 동특성, 보외방법, 실시간 조건, 프로세서의 수 및 성능 등의 여러 요소를 고려하여 적분기법, 차수 및 스텝크기를 결정해야하며, 잘 조정된 병렬처리계획이 필수적이다.

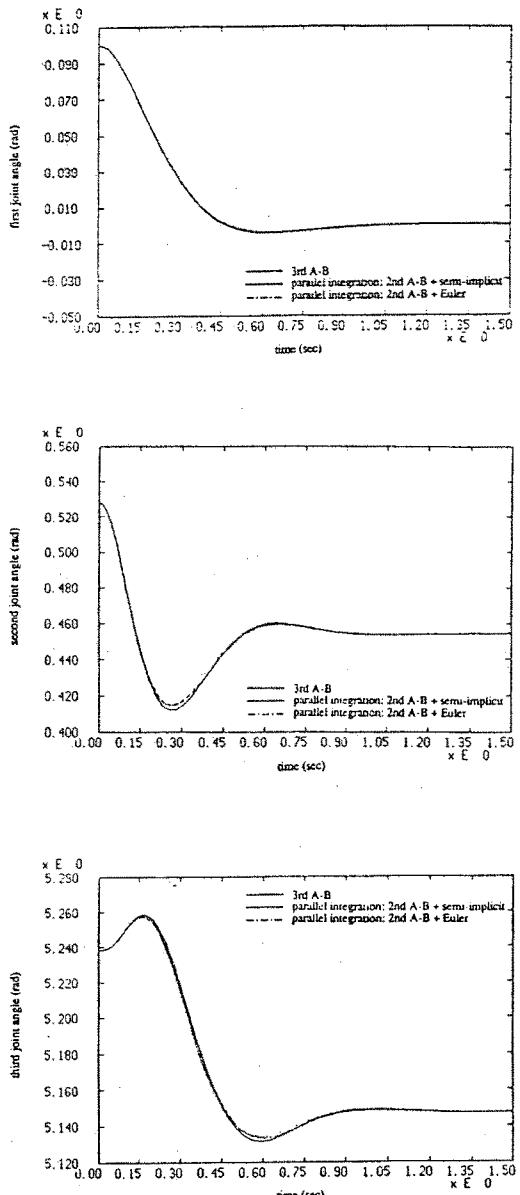


Fig.8 Revolute joint angles

병렬처리 효율성을 일관성있게 분석하기 위한 방법은 병렬처리 컴퓨터의 하드웨어와 알고리즘을 동시에 고려하여야 가능하며, 좀 더 체계적인 연구가 필요하다. 경제적인 실시간시뮬레이션을 위해 병렬로 연결된 DSP 보드를 고려할 수도 있다.

## 참 고 문 헌

1. Fennel, H., Mahr, S., and Schleysing, R., "Transputer-Based Real-Time Simulator-A High Performance Tool for ABS and TCS Development", SAE Paper 920643.
2. Haug, E. J., Andersen, D., Berbaum, K., Evans, D., Hinrichs, J., Kim, S-S., Kuhl, J., Lee, W-s., Lenel, J., Stoner, J., Tsai, F-F., Bernard, J., Malwitz, D., Turpin, D., Sinacori, J., Allen, Wl., and McAreavy, J., Feasibility Study and Conceptual Design of a National Advanced Driving Simulator, DOT HS 807 596, National Highway Traffic Safety Administration, 1990.
3. Drosdol, J., and Panik, F., "The Daimler-Benz Driving Simulator : A tool for Vehicle Development", SAE paper 850334, 1985.
4. Tsai, F. F., and Haug, E. J., Automated Methods for High Speed Simulation of Multibody Dynamic Systems, Technical Report R-47, Center for Simulation and Design Optimization, University of Iowa, 1989.
5. Howe, R. M., "The Choice of Integration Methods for Real-Time Simulation Based on Dynamic Accuracy and Stability", Modeling and Simulation, Vol.19, 1988, pp.457-466.
6. Rill, G., "Vehicle Dynamics in Real-Time Simulation", 10th IAVSD Symposium, Czechoslovakia, 1987, pp.337-347.
7. Lee, W-S., and Tak, T. O., "Dual-Rate Integration of Controlled Multibody Mechanical Systems", Proc. of IEEE Conference on Decision and Control, pp.674-679, 1989.
8. Haug, E. J., Intermediate Dynamics, Allyn & Bacon, Boston, Massachusetts, 1989.
9. Franklin, G. F., Powell, J. D., and Emami-Naeini, A., Feedback Control of Dynamic Systems, 2nd Edition, Addison-Wesley, 1991.
10. McCloy, D. and Martin, H. H., Control of Fluid Power : Analysis and Design, 2nd Edition, Ellis, 1980.
11. Bader, G., and Deuflhard, P., "A Semi-Implicit Mid-Point Rule for Stiff Systems of Ordinary Differential Equations", Numerische Mathematik, Vol.41, 1983, pp.373-398.
12. Wells, D. R., Multirate Linear Multistep Methods for the Solution of Systems of Ordinary Differential Equations, Technical Report No. UIUCDCS-R-82-1093, Department of Computer Science, University of Illinois, 1982.
13. Lee, W-S., and Tak, T. O., "Parallel Processing of Controlled Multibody Mechanical System Dynamics", Engineering with Computers, Vol.7, No.3, 1991, pp.161-172.
14. Kuhl, J., and Tayyab, A., private communication, 1989.
15. MATLAB User's Manual, The Math Works, 1990.