

실시간 제어 성능 향상을 위한 지터 제어 기법의 비교 연구

(Comparative Study on Jitter Control Methods for Improving Real-Time Control Performance)

박 문 주 [†] 임 양 미 ^{**}
(Moonju Park) (Yang-Mi Lim)

요 약 본 논문에서는 실시간 제어 시스템에서의 지터를 줄이기 위한 스케줄링 기법들을 비교 분석한다. 기존의 실시간 지터에 대한 연구들은 동적우선순위 스케줄링 기법에 초점을 맞추는 반면, 본 논문의 연구는 실제 상황에서 더 널리 사용되는 고정우선순위 스케줄링 기법들에 중점을 둔다. 기존의 연구들에서 사용한 지터의 측정법은 상대적 값을 측정하기 때문에 유용하지 않은 경우가 있음을 지적하고, 새로운 측도를 제시하여 시뮬레이션을 통한 실험을 수행하였다. 실험 결과 본 논문에서 제시된 지터 한계를 이용한 고정우선순위 할당 기법이 지터 감소 및 제어 성능 향상에서 우수한 성능을 보임을 알 수 있다.

키워드 : 실시간, 제어, 지터

Abstract This paper compares and studies scheduling methods to reduce jitter in real-time control systems. While previous research has focused on dynamic-priority scheduling schemes, this paper focuses on fixed-priority scheduling which is more widely used. It is pointed out that previously defined jitter measures might not be useful in enhancing the control performance of a real-time task because the measures are relative values. We present a new jitter measure and a new scheduling scheme for fixed-priority tasks. The experimental results through simulation show that the new scheduling scheme reduces jitter and enhances control performance.

Key words : Real-time, Control, Jitter

1. 서 론

다양한 분야에 IT기술이 이용되면서, 컴퓨터를 이용한 제어 시스템의 중요도가 높아지고 있다. 이러한 컴퓨터 제어 시스템에서는 소프트웨어를 이용한 디지털 제어 장치가 점점 과거의 아날로그(analog) 제어 장치를

대치하고 있다. Time-to-market의 압력이 높아지고 비용을 절감하여 원가를 절감해야 하는 필요가 대두되면서, 하나의 프로세서 상에서 다수의 실시간 제어 응용을 수행시켜 시스템의 처리기 수를 줄이고 효율을 높이고자 하는 시도가 계속되고 있다.

실시간으로 제어가 필요한 시스템에서는 수행 결과가 목적인 값에 얼마나 근접한 지를 나타내는 실시간 제어 성능(real-time control performance)이 시스템의 효율성을 결정한다. 실시간 제어 시스템에 멀티태스킹이 도입되면서, 태스크 간의 수행 간섭으로 인한 지터(jitter) 문제가 발생하여 이에 대한 해결책이 요구되고 있다. 지터란 특정 이벤트의 발생으로부터 수행을 완료할 때까지의 시간의 변동 정도를 나타낸다. 이러한 지터는 단순한 지연(delay)과 달리 변화하는 양이기 때문에, 제어 시스템의 성능을 저하시키는 문제점을 가지고 있다[1].

최근 10년간 실시간 제어 시스템에서의 지터 문제를 해결하기 위한 연구가 활발히 진행되어 왔다. Marti 등은 [2]에서 제어 시스템 설계 시 지터를 보정하도록 하

· 이 논문은 인천대학교 2009년 자체연구비 지원에 의하여 연구되었음

[†] 정 회 원 : 인천대학교 컴퓨터공학과 교수
mpark@incheon.ac.kr
(Corresponding author임)

^{**} 정 회 원 : 서울산업대학교 매체공학과 교수
yosimi@chol.com

논문접수 : 2009년 9월 28일

심사완료 : 2009년 10월 20일

Copyright©2010 한국정보과학회 : 개인 목적이나 교육 목적인 경우, 이 작품의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

정보과학회논문지: 컴퓨터의 실제 및 래터 제16권 제1호(2010.1)

는 연구를 발표하였다. Cervin 등은 [3]에서 EDF(Earliest Deadline First)[4] 스케줄링에서 입출력 지터의 상한을 구하고 지터 한계(jitter margin)의 개념을 이용해 지터를 제어할 수 있음을 보였다. 주기보다 빠른 마감시한(deadline)을 할당함으로써 지터를 줄이는 방법은 이미 알려져 있었으나, 실제로 제어 시스템에 대한 적용을 연구한 것은 Buttazzo 등의 연구[5]에서 비롯되었다. 이후 이 연구를 확장하여 지터 감소의 문제를 마감시한 할당과 제어 성능에 대한 최적화 문제로 전환하여 해결하려는 연구가 진행되었다[6]. 그러나 이러한 연구들은 모두 EDF와 같은 동적우선순위 기반 스케줄링 기법에 국한되고 있다.

본 논문에서는 고정우선순위 스케줄링 기법을 포함하여 지터 감소를 위한 기법들을 비교 분석하도록 한다. 기존의 연구[5]에서 비교 대상으로 제시한 상대적 지터 값이 제어 성능과의 연관성을 나타내기에 부족함이 있음을 지적하고, “간섭 지터”를 정의하여 이를 성능 비교 대상으로 하였다. 또한 EDF에 적용되었던 주기보다 짧은 마감시한 할당 기법이 지터 한계 개념을 응용하여 고정우선순위 스케줄링 기법에서 쉽게 구현될 수 있음을 보이고, 실험을 통하여 이 기법이 선점형 또는 비선점형 DM(Deadline Monotonic) 스케줄링보다 높은 성능을 달성함을 보인다.

본 논문은 총 5절로 구성되어 있다. 2절에서는 분석과 모델링에 사용된 시스템 모델을 설명하고, 3절에서는 지터를 분석하고 이를 감소시키기 위한 스케줄링 기법을 제시한다. 4절에서는 시뮬레이션을 통한 실험 결과를 제시하고 분석하며, 5절에서 결론을 맺는다.

2. 시스템 모델

본 논문에서 고려하는 시스템 모델은 전형적인 실시간 태스크 모델에 기반한다. N개의 주기 태스크 $\{\tau_1, \tau_2, \dots, \tau_n\}$ 는 하나의 프로세서 상에서 수행된다. 각 태스크 τ_i 는 무한개의 태스크 인스턴스(task instance), 즉 작업(job)으로 이루어지며, 각 작업의 최악 경우 수행시간(WCET: Worst Case Execution Time)은 동일하다. 태스크 τ_i 는 주기, 상대적 마감시한, WCET의 세 가지 시간 제약 조건을 가지며, 이들은 각각 T_i, D_i, C_i 로 표시된다. 태스크 $\tau_i = (T_i, D_i, C_i)$ 로 간략히 표시할 수 있다. WCET는 상대적 마감시한보다 작아야만 하지만, 상대적 마감시한은 주기와 같거나, 혹은 작거나 클 수도 있다.

각 태스크는 독립적이며, 따라서 배타적 공유 자원은 사용하지 않는 것으로 가정한다. 이는 제어 시스템에서 일반적으로 적용할 수 있는 가정으로, 제어 시스템의 안

정성을 위하여 공유 자원은 고려하지 않는 경우가 많다.

본 논문에서는, 분석을 위하여 제어 시스템의 샘플링 지연(sampling delay)은 무시할 수 있는 것으로 가정한다. 실제 시스템에서는 이러한 지연이 지터(jitter)를 발생시킬 수 있으나, A/D 전환기(A/D converter)에서 주기적으로 샘플링하도록 프로그래밍하여 이러한 지터를 없애도록 구현할 수 있다.

3. 지터 제어 기법

3.1 지터와 제어 시스템 안정성

지터가 제어 대상 시스템의 안정성에 주는 영향은 지터 한계(jitter margin: J_m)의 형태로 표현될 수 있다 [3]. 만일 제어 알고리즘의 출력 시간이 지터 한계보다 작다면, 제어 시스템의 안정성이 보장된다.

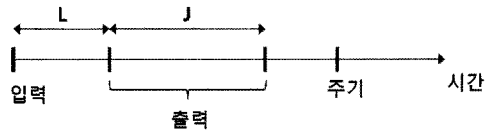


그림 1 지연(delay)의 분석: 고정 지연과 지터

그림 1의 L은 입력지연과 같은 고정지연을 의미하며, J는 출력을 위한 계산 시간과 스케줄링에 의한 지연시간을 포함하는 가변적인 시간을 나타낸다. L은 언제나 고정된 값이므로, 스케줄링에서 관심 있는 변수는 J(지터)이다. 최대 지연값은 L+J로 주어지며, 이 값이 지연 한계(delay margin: D_m)보다 작으면 시스템은 안정하다. L은 상수이므로, J의 값이 (지연 한계 - L) 보다 작으면 역시 시스템은 안정하게 되고, (지연 한계 - L) 을 지터 한계라고 하며 L의 함수가 된다($J_m(L)$). 지터 한계 값은 제어 시스템의 특성으로부터 계산될 수 있다[3].

[5]에서는 지터의 종류를 세분하여 세 가지 형태의 지터에 대한 스케줄링 정책들의 영향을 실험적으로 연구하였다. [5]에서 정의한 지터는 다음과 같다.

RTJ: Response Time Jitter
- 최대 응답 시간과 최소 응답시간의 차
INJ: Input Jitter
- 최대 수행시작 시간과 최소 수행시작 시간의 차
IOJ: Input-Output Jitter
- 최대 작업완료 시간과 최소 작업완료 시간의 차

그러나 [5]에서 평가된 위 세 가지 지터 값들은 상대적인 값이기 때문에 제어 시스템의 안정을 보장하는 지터 한계와 다른 의미를 가지게 된다. 즉, 지터 한계는 샘플링 주기로부터의 상대적 지연 정도를 나타내는 데 반해, [5]에서 정의되고 측정된 지터는 지터간의 상대적

크기이기 때문에 서로 다른 의미를 가지며, RTJ가 작다고 해서 시스템의 안정도가 높다고 하기 어렵다. 이러한 결과는 [5]의 실험 결과에서도 나타나며, 오히려 INJ와 IOJ의 합에 영향을 받는 것 같다는 의견을 제시하고 있다. 예로서, 다음과 같은 세 개의 실시간 태스크를 고려하자: $\tau_1 = (4, 4, 1)$, $\tau_2 = (5, 5, 2)$, $\tau_3 = (10, 10, 3)$.

이 태스크들의 프로세서 이용률은 95%로, 선점형 EDF에서 스케줄 가능하다. 이 태스크들이 선점형 EDF에 의해 스케줄 되는 경우의 태스크 3의 최악의 경우 스케줄은 다음 그림과 같다.

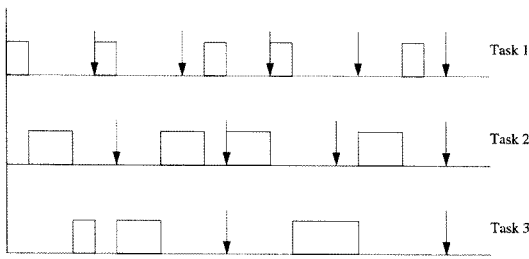


그림 2 EDF 스케줄링

태스크 3의 최대 응답시간은 7이며, 최소 응답시간은 6이므로 RTJ는 1이다. 각 태스크의 RTJ, INJ, IOJ의 값은 다음 표와 같이 구해진다.

표 1 EDF에서의 지터 값

	태스크 1	태스크 2	태스크 3
RTJ	2	2	1
INJ	2	2	3
IOJ	0	1	2

실제 제어 태스크의 경우, 태스크 3의 성능이 제일 나쁠 것으로 기대되지만, 위 표에서 볼 수 있는 바와 같이 RTJ 값은 태스크 3이 가장 작다. 이는 태스크 3의 응답시간이 전체적으로 늦기 때문에, 상대적 지터는 작아지기 때문이다. 따라서 [5]에서 제시한 지터 값들은 제어 성능의 지표로 사용할 경우 태스크의 특성에 따라 적절하지 않을 수 있다.

본 논문에서는 [3]에서 제시된 지터 한계의 의미에 충실한 성능 지표를 얻기 위하여, 측정에 사용할 값을 “최대 응답 시간과 수행시간의 차”로 정의하고 이를 간섭 지터(interference jitter)라고 부른다. 간섭 지터의 의미는, 이러한 지터가 멀티태스킹 환경에서 발생하는 것으로 다른 태스크의 간섭에 의해 발생함을 나타내는 것이다. 즉, 만일 한 프로세서에 하나의 제어 태스크만 존재한다면 A/D, D/A 전환 및 입력 신호에 의한 지연만이 존재할 것이며, 이러한 지연은 제어 알고리즘에서 충분

히 고려될 수 있다.

3.2 지터 제어 스케줄링 방법

[2]에서 제시된 지터 감소 방법은 태스크 분할(TS: Task Splitting), 주기보다 빠른 마감시한 할당(AD: Advanced Deadline), 그리고 비선점형(NP: Non Preemptive) 스케줄링이다. 이중 태스크 분할 방법은 강인 제어(robust control) 시스템에서는 사용할 수 없으며 그 외의 경우에도 일반적인 태스크 모델에 비해 나은 성능을 보이지 않기 때문에[2], 본 논문의 고려 대상에서는 제외하도록 한다.

고정우선순위 스케줄링에서 선점형 스케줄링에 AD 방법을 채용할 경우에는 Deadline Monotonic(DM) 스케줄링이 최적(optimal)임이 알려져 있다[7]. AD 방법을 채용하기 위해서, [3]에서는 최소 응답시간과 최대 응답시간을 구하여 두 값의 차가 지터 한계보다 작도록 마감시한을 할당하는 방법을 제시하였다. 그러나 응답시간의 계산은 의사-다항시간을 요구하며, 또한 다른 태스크의 마감시한이 변경되면 다시 계산을 해야 하므로 [3]의 방법은 마감시한 할당에 많은 시간이 소요된다.

고정우선순위 스케줄링에서는, DM 스케줄링을 채용할 경우 단순히 빠른 상대적 마감시한이 높은 우선순위를 의미하기 때문에, 마감시한의 계산 없이 우선순위 변경만으로 AD방법을 구현할 수 있다. 하나의 프로세서에 하나의 태스크가 수행될 때, 태스크의 지터는 0이 되며, 이때의 고정지연은 태스크의 수행시간과 같다. 따라서 고정지연 L을 태스크의 수행시간으로 설정하고, 지터 한계 Jm을 이로부터 구할 수 있다. 구해진 지터 한계를 마감시한 할당에 사용하는 대신, 지연 한계 $Jm(C_i) + C_i$ 를 구하여 이 값이 작은 태스크에 높은 우선순위를 부여하면 AD 방법이 구현된다.

고정우선순위에 대한 AD 방법 적용의 예를 보기 위하여 아래와 같은 제어 태스크를 고려하자. 하나의 프로세서 상에서 3개의 제어 태스크가 동작하며, 각 태스크는 서로 다른 DC servo 모터를 제어한다. 제어 대상(plant)의 특성은 다음과 같다.

$$P_1(s) = \frac{1000}{s(s+1)}, P_2(s) = \frac{800}{s(s+1)}, P_3(s) = \frac{1000}{s(s+0.5)}$$

제어 알고리즘으로는 PID가 사용되며, 각 서보 모터를 제어하는 태스크들은 다음과 같다.

$$\tau_1 = (10, 10, 2), \quad \tau_2 = (5, 5, 2), \quad \tau_3 = (4, 4, 0.4)$$

태스크 특성의 시간 단위는 밀리초(ms)이다. 이 태스크들의 고정지연을 수행시간과 같이 놓았을 때 지터 한계는 각각 τ_1 에 대하여 $Jm(2)=1.7$, τ_2 에 대하여 $Jm(2)=3.2$, τ_3 에 대하여 $Jm(0.4)=1.4$ 이다. 따라서 각 태스크의 지연 한계는 τ_1 이 3.7, τ_2 는 5.2, τ_3 는 1.8이 되며, 이에 따라

우선순위는 τ_3, τ_1, τ_2 의 순서로 주어진다.

NP 방법에 의한 지터 감소는 [5]에 기술된 바와 같이 구현이 간단하고 임출력 지터(IOJ)가 0이 되는 장점이 있다. 그러나 태스크들의 수행시간이 크게 차이나는 경우, 비선점 구간으로 인한 블로킹(blocking) 시간의 증가로 인해 제어 성능이 떨어질 수 있다.

4. 실험 결과 및 분석

본 실험에서는 3.2절에 제시된 DC 서보 모터의 PID 제어태스크를 하나의 프로세서 상에서 동작시켜, 일반적인 선점형 DM 스케줄링과, NP 방법, 그리고 AD 방법의 세 가지 스케줄링 기법을 사용했을 때의 간섭 지터를 측정하였다. 지터의 측정은 시뮬레이션을 통하여 이루어졌으며, 시뮬레이션 도구로는 MATLAB을 이용한 TrueTime 패키지[8]가 이용되었다. 태스크의 수행시간은 random함수를 사용하여 최대 수행시간이 3ms보다 크지 않도록 정해졌으며, 하나의 태스크는 최대 50%의 이용률을 넘지 않도록 하였다. 실험에서는 프로세서 이용률이 70%에서 100%까지 증가 시키면서 간섭 지터를 측정하였다. 또한 제어 시스템의 성능을 측정하기 위하여, ISE(Integral of Square Error)를 사용하였다.

$$ISE = \int_0^{\infty} e^2(t) dt \quad (1)$$

참조 입력 신호는 1.5초 단위로 반복하여 입력되었으며, ISE를 무한히 계산하면 그 값이 무한대가 되기 때문에, 실험 시간을 15초로 하고 ISE는 1.5초 단위의 평균으로 계산하였다. 그림 3은 TrueTime 에서의 시뮬레이션 세팅을 보여준다.

먼저, 세 가지 스케줄링 기법에서의 간섭 지터를 측정하였다. 그림 4는 태스크 1의 간섭 지터를 보여준다. 그래프들의 X축은 프로세서 이용률을 나타낸다. 예상할 수 있는 바와 같이, 선점형 DM 스케줄링에서 태스크 1의 우선순위가 가장 낮기 때문에 간섭 지터는 최대가

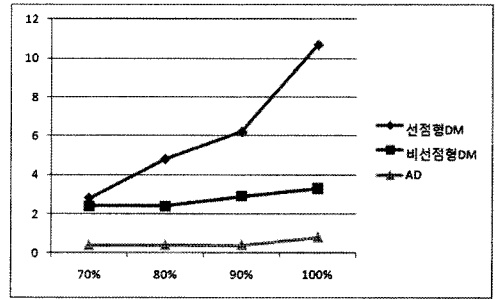


그림 4 태스크 1의 간섭 지터 (ms)

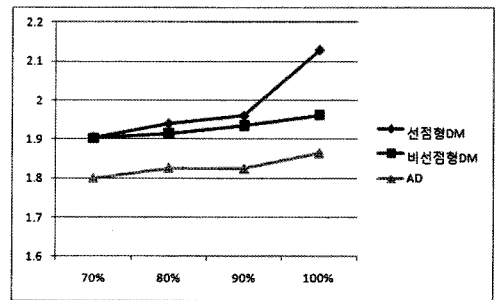


그림 5 태스크 1의 정규화된 ISE

된다. 본 논문에서 제안된 AD 기법(지연 한계에 의한 우선순위 할당 기법)에 의해 스케줄 되는 경우에 간섭 지터가 가장 작다.

태스크의 간섭 지터와 제어 성능의 연관성을 보이기 위하여, 태스크 1의 제어 오차를 측정하여 ISE를 계산하였다. ISE 값은 프로세서에서 한 태스크만 수행했을 때 얻을 수 있는 최대 성능을 1로 하여 정규화(normalize) 시켰으며, 그 결과는 그림 5에 표시하였다. 결과에서 볼 수 있는 바와 같이, 간섭 지터가 가장 작은 AD 기법의 ISE가 가장 작은 것을 볼 수 있다.

전체 시스템의 성능은 각 태스크의 간섭 지터와 ISE

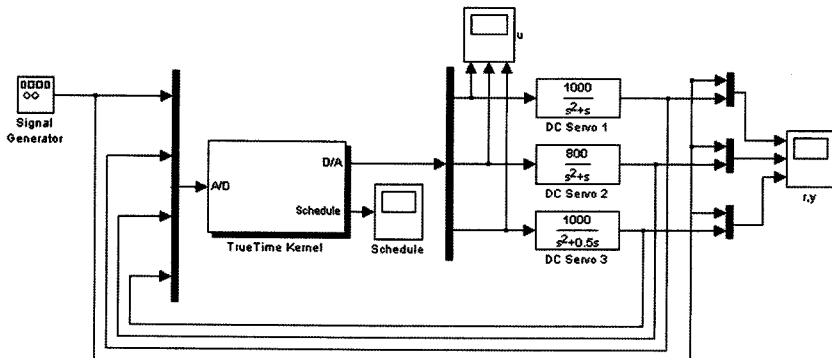


그림 3 TrueTime을 이용한 시뮬레이션 세팅

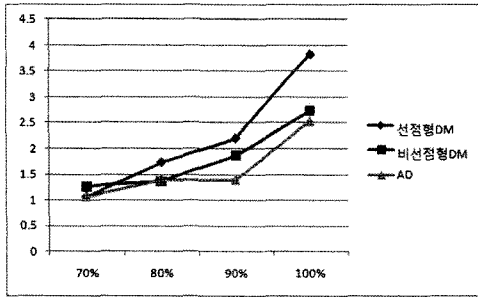


그림 6 평균 간섭 지터(ms)

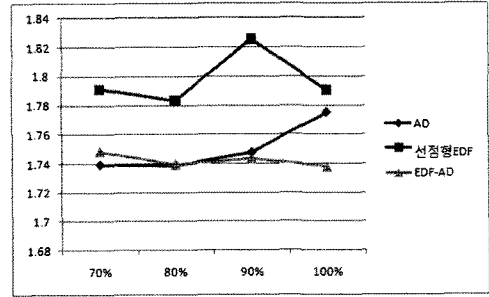


그림 8 EDF 기법과의 정규화된 평균 ISE 비교

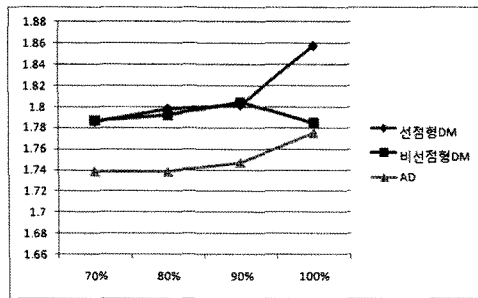


그림 7 정규화된 평균 ISE

평균으로 평가하였다. 그림 6은 태스크 간섭 지터의 평균, 그림 7은 태스크 ISE 평균의 최고 달성 가능 성능에 대한 정규화된 값을 표시한다.

평균적인 성능에 있어서도 지연 한계에 의한 우선순위 할당 기법이 다른 스케줄링 기법에 비하여 우수한 성능을 보이며, 실험에서 DM기법보다 낮은 성능을 보이는 경우는 없었다. 위의 실험 결과로부터, 본 논문에서 제시한 간섭 지터가 제어 성능과 강한 연관성이 있음을 알 수 있다.

마지막으로, 고정우선순위 스케줄링에서의 AD 기법을 동적우선순위 스케줄링 기법 중 최적인 EDF (Earliest Deadline First) 기법과 제어 성능 면에서 비교하도록 하였다. 비교대상 EDF 기법은 선점형 EDF 스케줄링 기법과, 지터 한계를 마감시한으로 할당하여 구현한 EDF-AD 기법[6]이다.

Buttazzo의 연구[9]에 의하면, EDF는 태스크 수가 많고 프로세서 이용률이 높을 때 고정우선순위 스케줄링 기법보다 작은 RTJ값을 가진다. 또한 IOJ는 EDF가 항상 RM보다 작다. EDF-AD는 EDF로 스케줄 되는 태스크들이 주기보다 빠른 마감시한을 가지도록 허용하는 것으로, EDF 스케줄링에서 가장 작은 지터를 가진다.

전체적인 성능 비교를 위하여 평균 ISE를 구하여 이를 최고 달성 가능 성능에 대하여 정규화시켰다. 이 실험의 결과는 그림 8에 보인다.

실험 결과에서 볼 수 있는 바와 같이, 고정우선순위에 구현된 AD 기법이 선점형 EDF 기법보다 우수한 제어 성능을 보이며, 90% 이하의 프로세서 이용률에서는 EDF-AD 기법과 유사한 성능을 보인다. EDF-AD 기법은 [1]의 실험에서 가장 좋은 제어 성능을 보인 기법이나, 마감시한 할당을 위하여 [6]에 기술된 바와 같이 convex optimization 등의 복잡한 계산을 필요로 한다. 본 논문에서 실험된 고정우선순위 AD 기법은 그러한 복잡한 계산 없이도 고정우선순위 기법을 이용하여 EDF-AD 기법과 유사한 성능을 얻을 수 있음을 알 수 있다.

5. 결론

본 논문에서는 기존의 지터 감소를 위해 스케줄링 정책들을 비교한 연구들에서 사용된 지터 측정기 상대적인 값으로서, 유용하지 않은 경우가 발생할 수 있음을 지적하고, 간섭 지터의 개념을 정의하여 이를 기존 측정값 대신 사용할 것을 제안하였다. 간섭 지터는 지터 한계에 기반을 둔 개념이며, 따라서 직접적으로 제어 성능을 반영할 수 있다.

기존의 연구들이 EDF와 같은 동적우선순위 스케줄링에 집중되어 있는 반면, 본 논문에서는 실제 상황에서 더 널리 사용되는 고정우선순위 스케줄링에 대하여 스케줄링 기법을 제시하고 실험을 실시하였다. 본 논문에서 제안된 우선순위 할당 방법은 태스크들의 제어 특성에 의존하기때문에 복잡한 계산 없이 할당을 수행할 수 있는 장점이 있다. 또한 실제로 마감시한을 할당하지 않아도 되므로 스케줄러의 구현이 간단해진다.

실험 결과는 새로운 우선순위 할당기법이 선점형 및 비선점형 DM보다 좋은 성능을 가짐을 보이고 있다. 이는 지터 한계가 적은 태스크에게 높은 우선순위를 부여함으로써 태스크의 긴급성을 반영한 결과일 것이다. 이 실험의 결과를 바탕으로, 향후 최적의 우선순위 할당기법을 찾기 위한 연구를 진행할 예정이다.

참고 문헌

- [1] K.-E. Arzen, A. Cervin, J. Eker, and L. Sha, "An Introduction of Control and Scheduling Co-design," *Proceedings of the 39th IEEE Conference on Decision and Control*, pp.4865-4870, 2000.
- [2] P. Marti, G. Fohler, K. Ramamritham, and J.M. Fuertes, "Jitter Compensation for Real-Time Control Systems," *Proceedings of the 22nd IEEE Real-Time Systems Symposium*, pp.39-48, 2001.
- [3] A. Cervin, B. Lincoln, J. Eker, K.-E. Arzen, and G. Buttazzo, "The Jitter Margin and Its Applications in the Design of Real-Time Control Systems," *Proceedings of International Conference On Real-Time and Embedded Computing Systems and Applications*, pp.9-18, 2004.
- [4] C. Liu and J. Layland, "Scheduling Algorithm for Multiprogramming in a Hard Real-Time Environment," *Journal of ACM*, vol.20, no.1, pp.46-61, 1973.
- [5] G. Buttazzo and A. Cervin, "Comparative Assessment and Evaluation of Jitter Control Methods," *Proceedings of the International Conference on Real-Time and Network Systems*, pp.163-172, 2007.
- [6] W. Yifan, E. Bini, and G. Buttazzo, "A Framework for Designing Embedded Real-Time Controllers," *Proceedings of the 14th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications*, pp.303-311, 2008.
- [7] N.C. Audsley, A. Burns, M.F. Richardson, and A.J. Wellings, "Hard Real-Time Scheduling: The Deadline-Monotonic Approach," *Proceedings of IEEE Workshop on Real-Time Operating Systems and Software*, pp.133-137, 1991.
- [8] A. Cervin, D. Henriksson, B. Lincoln, J. Eker, and K.-E. Arzen, "How Does Control Timing Affect Performance? Analysis and Simulation of Timing Using Jitterbug and TrueTime," *IEEE Control Systems Magazine*, pp.16-30, 2003.
- [9] G. C. Buttazzo, "Rate Monotonic vs. EDF: Judgment Day," *Real-Time Systems*, vol.29, pp.5-26, 2005.



콘텐츠

임 양 미

1993년 서울산업대학교 매체학과(공학사)
1998년 큐슈예술공과대학교 정보전달전공(예공학석사). 2009년 중앙대학교 첨단영상대학원 영상공학(공학박사). 2009년~현재 동국대학교 총무로영상센터 연구원. 관심분야는 디지털영상처리, 인터랙티브



박 문 주

1996년 서울대학교 조선해양공학과(공학사). 1998년 서울대학교 컴퓨터공학과(공학석사). 2002년 서울대학교 전기컴퓨터공학부(공학박사). 2002년~2006년 LG전자 단발연연구소. 2006년~2007년 IBM Ubiquitous Computing Lab. 2007년~현재 인천대학교 컴퓨터공학과 조교수. 관심분야는 실시간 시스템, 내장형시스템