

모바일 컴퓨팅 환경에서 실시간 스케줄링을 고려하며 평균 대기 시간을 최소화 시키는 브로드캐스트(Broadcast) 스케줄링 기법

°임성화, 김재훈

아주대학교 정보 통신 전문 대학원

Feasible Real-Time Data Broadcast Scheduling Algorithm Reducing Average Waiting Time in Mobile Computing Environments

°Lim, Sung-Hwa · Kim, Jai-Hoon
College of Information and Communication
Ajou University

요 약

이동컴퓨팅의 응용에서 대부분은 정보 송신측으로부터 정보 수신 측 방향으로의 통신량이 많으므로, 이런 비대칭 통신환경에서는 브로드캐스트(broadcasting)방법이 효과적이다. 본 논문에서는 기존에 제안된 마감시간을 갖는 주기적 정보전달과 빠른 액세스시간이 요구되는 비주기적인 정보전달이 동시에 요구되는 환경에서, 실시간 정보들의 실시간 제약을 만족시키면서 비실시간 정보들의 평균 대기시간을 최소화시키는 효율적인 채널 할당 방법을 제안하고 성능평가를 하였다.

1 서론

이동컴퓨팅의 응용중 Defense Message System (DMS), Digital Battlefield and Data Dissemination (BADD)[3] 등의 국방 분야 또는 각종 안내 정보(날씨, 환율, 주가, 주차정보 등)와 같은 일상 분야에서 사용자에게 여러 가지 정보를 효과적으로 전달하는 것은 중요한 문제이다. 이러한 시스템에서는 위성이나 기지국에서 이동 컴퓨터들이 공통으로 필요로 하는 정보를 전달한다. 이때, 정보 송신측(위성 또는 기지국)으로부터 정보 수신측(이동 컴퓨터 이용자들) 방향(downstream)의 통신량이 많고 반대 방향(upstream)으로의 통신량은 상대적으로 적다. 이러한 비대칭 통신 환경에서는 브로드캐스트(broadcast) 방법이 정보를 효과적으로 전달하는 방법이 된다. 이 방법에서는 많은 사용자들이 공통으로 필요로 하는 정보를 동시에 전송할 수 있는 장점이 있다.[7]

많은 종류의 애플리케이션에서 평균 대기 시간을 줄이는 것만이 사용자의 요구 사항을 극대화하는 것은 아니다. 즉 실시간 처리를 요구하는 주기적인(또는 비주기적인) 요구사항은 시간 제약이내에 서비스를 완료해야한다. 이 시간제약을 넘기면 사용자의 만족감은 거의 없어지고 경우에 따라서 전체 시스템이 제어가 불가능하거나 동작 불능의 상태가 될 수도 있다. 사용자의 만족도를 높이기 위해 시간제약이 중요한 실시간 요구사항과 평균 대기시간이 중요한 비실시간 요구사항이 공존하는 이동컴퓨팅 환경을 고려해야한다. 그러므로 주기적인(또는 비주기적인) 데이터 요구사항을 시간제약 이내에 제공받은 물론 비실시간 요구사항에 대해서는 평균 대기 시간을 최소화 시키는 데이터 브로드캐스트 스케줄링 전략이 요구된다.

기존의 실시간 스케줄링 전략은 실시간 및 비실시간 타스크가 공존하는 환경에서 어느 순서로 타스크에게 CPU를 할당할 것인가 하는 문제를 다루었다. 이 때 비실시간 타스크는 서로 독

립적인 일이며 보통 도착 순서에 따라 처리해주고, 되도록 대기 시간을 줄이기 위해 실시간 타스크를 시간제약 이내에 끝내는 한도까지 뒤로 연기시키는 방법을 사용한다. 실시간 통신에서도 기본적으로 비슷한 방법이 이용되는 데, 이때 CPU를 타스크에게 할당하는 문제 대신 통신 대역폭을 어느 메시지 또는 패킷 데이터 전송을 위해 할당할 것인가 하는 문제로 귀착된다. 즉, 여러 종류의 비 실시간 정보를 사용자의 요구 빈도에 따라 그리고 정보의 크기에 따라 효과적으로 브로드캐스트 해야하며 아울러 주기적 실시간 정보는 시간제약 이내에 끝내야 한다. [9]에서는 비실시간 정보들을 균등히 분포시킨후 실시간 정보들을 스케줄링 함으로써, 실시간 정보들이 마감시간을 넘지 않을 경우 비실시간 정보들에 대한 평균 대기시간이 기존의 기법들에 비해 월등히 줄어듦을 보였다. 그러나 제안된 방법은 채널 이용율이 100% 또는 어느 한계이상인 상황에서의 스케줄링 가능성을 항상 보장하지는 못한다. 그러므로 본 논문에서는 [9]에서 제안된 알고리즘에 기반하여, 실시간 정보들의 스케줄링 보장성을 갖는 실시간 브로드캐스트 알고리즘을 제안하고 그 성능을 분석하였다.

2. 관련연구

브로드캐스트 기법에 대한 연구는 그 동안 많이 이루어지고 있으며, 특히 여러 종류의 정보를 전송할 때 사용자들이 원하는 정보를 얻기 위하여 기다려야하는 평균 대기 시간 및 응답시간을 줄이기 위한 스케줄링 방법에 대해서도 연구가 이루어지고 있다 [1,2,4,5,6,7,9]. 즉, 여러 가지 종류의 정보를 사용자 가 동시에 요구할 때, 각 종류의 정보마다 정보의 크기, 정보의 요구 빈도를 고려하여 요구된 정보를 얻기까지 기다리는 평균시간을 최소화하기 위하여 정보의 브로드캐스트 순서를 어떻게 정할

것인가 하는 문제를 주로 다루었다. 또한 평균 대기시간을 최소화 하기 위해 제곱근 법칙(square root rule)에 제안 되었는데, 각 정보의 브로드캐스트 빈도는 사용자 요구 빈도의 제곱근에 비례하고 정보길이의 제곱근에 반비례 하게 유지 시키며, 동일 정보는 일정시간간격마다 브로드캐스트 할때 최적의 평균 대기 시간을 얻을수 있다[1,2,7]. 그 외에 Jain 과 Werth [6]은 연속적인 아이템들 사이의 공간의 편차를 줄이므로써 평균 액세스 시간을 줄일 수 있다는 연구를 하였다.

또한 비실시간 정보들에 대한 평균 대기시간을 줄이기 위하여 비실시간 정보를 균등하게 분포시킴으로써 대기시간을 최소화 시키는 기법이 제시되었다. 비실시간 정보를 균등히 할당된 뒤 실시간 정보들 스케줄링 할 경우 실제로 EDF나 슬랙 스틸링(slack stealing)기법에 비해, 상대적으로 적은 평균 대기시간을 소요하였다[9].

3 실시간 브로드캐스트 기법

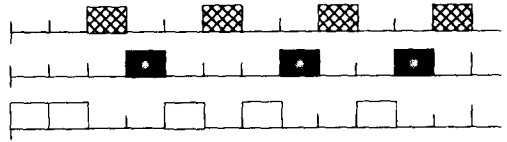
3.1 평균 대기시간을 최소화 하는 스케줄링 기법

본 논문에서 제안하는 실시간 브로드캐스트 기법은 실시간적 특성을 갖는 주기적 정보와 비실시간적 특성을 갖는 비주기적 정보가 공존하는 환경에서 여러 종류의 데이터를 효과적으로 스케줄링하는 것이다. 주기적 정보는 마감시간을 가지므로 마감시간의 준수가 주요 관점이고, 비주기적 정보는 평균 대기시간이 주요 관점이 된다. 기존의 스케줄링에서는 주기적 정보는 EDF(Earliest Deadline First)나 LRF(Latest Release time First)등의 실시간 스케줄링 기법으로 할당된 뒤, 남은 부분(slack)을 가지고 비주기적 정보를 스케줄 하였다. 그러나 [7]에서 제안된 제곱근 법칙에서와 같이 비주기적 정보는 동일 정보를 일정한 간격으로 균등히 할당하여야 평균 대기시간을 줄일 수 있다. 다음은 비실시간 정보의 평균 대기시간을 줄이기 위한 스케줄링 방법이며 이 논문에서는 알고리즘1이라 칭하였다.

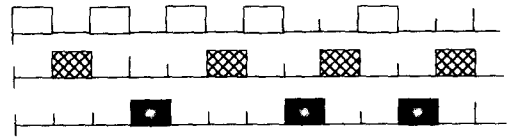
- step 1: 실시간 정보들의 주기 및 실행시간을 이용하여 실시간 정보들이 할당될 랑을 계산한 뒤, 남은 부분(slack)을 계산한다.
- step 2 : 슬랙을 전체 채널에 균등히 분포시킨 후, 슬랙에 비 실시간 정보들을 할당한다.
- step 3 : 남은 채널공간에 실시간 정보들을 preemptible EDF 알고리즘을 사용하여 스케줄링한다.

[그림 1]은 LRF 기법과 위에서 제안된 알고리즘1을 사용하여 주기적 정보가 {T₁ (3, 1), T₂ (4, 1)} 일 때, 슬랙에 비주기적 정보를 할당시킨 예이다. EDF 스케줄링의 경우 슬랙이 불균등하게 나타난 반면, 제안된 알고리즘을 사용했을 경우는 슬랙이 6까지는 1/2의 빈도로, 7~12까지는 1/3의 빈도로 균등하게 할당되었다. LRF기법의 경우 EDF 알고리즘으로 주기적 정보를 할당한 뒤 남은 부분(slack)을 앞당겨서 비주기적 정보를 할당 하였다(이런 과정을 slack stealing이라 한다). 제시된 예에서 EDF 기법은 1.33의 평균 대기시간을 소요한 반면 비주기적 정보를 균등하게 할당시킨 경우의 평균 대기시간은 1.25로, 제안된 기법이 더 좋은 성능을 나타내었다. 그러나 알고리즘1은 채널 이용율이 100% 또는 어느 한계 이상인 환경에서 실시간 정보들의 스케줄링 가능성을 항상 보장하지는 못한다. 그러므로 3.2절에서 실시간 정보들의 시간적 제약성을 고려한 실시간 브로드캐스트 스케줄링 기법을 제안한다.

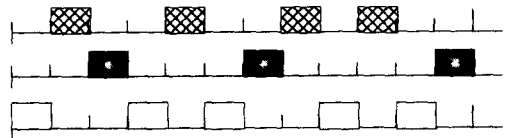
LRF(slack stealing)



제안 알고리즘 1 [9]



제안 알고리즘 2



□ : Slack ▨ : Task1 ★ : Task2

[그림 1] LRF와 제안 알고리즘1, 제안 알고리즘2의 비교

가정 및 용어

- 실시간 정보는 $T_i = (p_i, e_i)$ 로 표시된다.
(i 번째 정보의 주기는 P_i , 수행시간은 e_i , $1 \leq i \leq N$)
- N = 실시간 정보의 종류
- 비주기적 정보의 수행시간 e_n 는 주어진다.

스케줄링 알고리즘

- step 1. 비주기적 정보를 $N+1$ 번째의 실시간 정보로 가정한다.
- step 2. 비주기적 정보 $T_{N+1} = (p_{N+1}, e_{N+1})$ 이며, e_{N+1} 는 미리 주어진다. ($e_{N+1} = e_n$)
$$p_{N+1} = \frac{1}{1 - \sum \frac{e_i}{P_i}} \times e_{N+1}$$
- step 3. P_i ($1 \leq i \leq N$)로 Hyper period 계산
- step 4. T_i 에서 T_{N+1} 까지 EDF 알고리즘으로 스케줄
- step 5. 스케줄된 비주기적 정보와 실시간 정보를 swap하여 균등하게 분포시킨.

[그림 2] 제안 알고리즘 2

3.2 실시간 보장성을 갖는 브로드캐스트 알고리즘

알고리즘1은 채널 이용율이 100%환경에서 스케줄링 가능성을 보장할 수 없으므로, 실시간 환경에 적용하기에 적합하지 못하다. 본 장에서는 실시간적 제약성을 만족시키기 위하여, 선점가능한 EDF알고리즘이 채널 이용율 100%이하인 환경에서 스케줄이 가능함[8]을 이용하여 알고리즘2를 제안하였다.

[그림 2]는 실시간 제약성을 만족시키며 비실시간 정보들의 평균 대기시간을 최소화 시키는 브로드캐스트 스케줄링 알고리즘의 수행과정이다.

주기적 정보가 {T₁ (3, 1), T₂ (4, 1)} 일 때, 제안 알고리즘2를

이용하여 비주기적 정보들 공존하여 합당 시킨 경우의 예들 [그림 1]에 나타내었다. 제안 알고리즘 2를 이용한 경우에도 비주기적 정보의 평균대기시간이 1.25로 제안 알고리즘1과 같음을 알 수 있다. 위의 예에서 볼 수 있듯이 슬랙의 분포를 균등하게 하므로써 평균 대기 시간을 줄일 수 있으며, 제안 알고리즘2를 이용하면 스케줄링 가능성과 실시간적 제약을 만족시키면서도 비주기적 정보의 평균대기시간을 최소화 할 수 있음을 예상할 수 있다. 4장에서는 좀 더 다양한 경우에 대해 시뮬레이션을 통하여 제안 알고리즘을 평가하였다.

4 성능평가

제안된 스케줄링 기법을 좀 더 일반적으로 평가하기 위하여 N개의 실시간 정보들 슬랙의 양에 따른 여러 가지 경우로 대입시켜서 비실시간 브로드캐스트와의 스케줄링을 시뮬레이션 하였다. 다음과 같이 시뮬레이션 변수를 가정을 하였고 결과는 [표1]과 [그림2]에 나타내었다.

- 실시간 정보(주기적 정보)
N = 3
Type1 { (5,1), (6,1), (8,1) }
Type2 { (3,1), (5,1), (8,1) }
Type3 { (3,1), (4,1), (6,1) }
Type4 { (2,1), (5,1), (10,1) }
위와 같이 주기와 마감시간을 변화시켜서 실험하였다.

- 비주기적 정보
한 종류의 정보만 존재한다고 가정하였다.

실시간 정보 캐스트 / 스케줄링	Type1	Type2	Type3	Type4
	T ₁ =(5,1) T ₂ =(6,1) T ₃ =(8,1)	T ₁ =(3,1) T ₂ =(5,1) T ₃ =(8,1)	T ₁ =(2,1) T ₂ =(6,1) T ₃ =(13,1)	T ₁ =(2,1) T ₂ =(5,1) T ₃ =(10,1)
EDF, LRF	1.275	1.792	2.28	2.84
Proposed	1.00	1.5	2.00	2.59

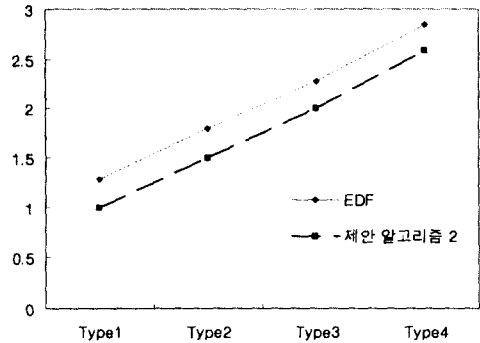
[표 1] EDF (LRF)와 제안된 스케줄링 알고리즘에서의 평균대기시간

[그림3]에 나타난 것과 같이 제안 알고리즘2를 이용하여 스케줄링을 할 경우, 스케줄링이 보장 됨은 물론 비주기적 정보가 비교적 균등하게 할당되어, 주기적인 정보들 먼저 할당하여 슬랙의 균일함이 보장되지 않은 경우보다 슬랙의 량과 상관없이 전체적으로 적은 평균 대기시간을 나타냄을 알 수 있다.

5 결과 및 향후과제

비대칭 통신 환경에서는 정보를 브로드캐스트(broadcast)하는 것이 효과적이므로, 현재 이동 통신분야에서는 브로드캐스팅 기법의 비중이 높다. 그러므로 브로드캐스트 채널에 대한 효율적인 스케줄링은 서비스의 질 향상에 필수적이다. 본 논문에서는 마감시간을 중요시 여기는 실시간 정보(주기적)와 평균 대기시간을 중요시하는 비실시간 정보가 혼합된 환경에서 비실시간 정보 데이터를 위한 대기시간을 최소화하는 스케줄링 방법

을 제안하였다. 슬랙의 분포를 균등히 할 경우 비실시간 정보들의 평균 대기 시간을 최소화 할 수 있으나, 실시간 정보들의 시간제약성을 동시에 만족시키기 위한 고려가 필요하다. 이용률이 100%이 아닌 환경에서 스케줄링 보장성을 갖는 선점가능한 EDF알고리즘의 특성을 이용하여 본 논문의 제안 알고리즘은 실시간 정보의 시간적 제약을 만족시키면서 비실시간 정보의 평균 대기시간을 기존의 기법에 비하여 줄일 수 있었다. 여러 종류의 비주기적 정보가 요구될 경우를 포함하여 실제 응용 환경을 고려한 다양한 시스템 변수를 이용한 실험이 향후에 이루어질 예정이다.



[그림 3] EDF (LRF)와 제안된 스케줄링 알고리즘2에서의 평균대기시간

참고문헌

- [1] S. Acharya, M. Franklin, and S. Zdonik, "Dissemination-based data delivery using broadcast disks." *IEEE Personal Communication*, pp. 50-60, December 1995.
- [2] M. H. Ammar and J. W. Wong, "On the optimality of cyclic transmission in teletext systems," *IEEE Transactions on Communications*, pp. 68-73, January 1987.
- [3] R. J. Douglass (Program Manager). "Battlefile awareness and data dissemination (BADD) program." program duration 1996-2000. Web site at <http://maco.dc.isx.com/iso/battle/badd.html>.
- [4] T. Imielinski, S. Viswanathan, and B. R. Badrinath, "Power efficient filtering of data on air," in *4th International Conference on Extending Database Technology*, March 1994.
- [5] T. Imielinski, S. Viswanathan, and B. R. Badrinath, "Data on the air- organization and access," manuscript.
- [6] R. Jain and J. Werth, "Airdisks and arraid : Modelling and scheduling periodic wireless data broadcast (extended abstract)," Tech. Rep. DIMACS Tech. Report 95-11, Rutgers University, May 1995.
- [7] N. H. Vaidya and S. Hameed, "Efficient Algorithms for Scheduling Single and Multiple Channel Data Broadcast," Tech. Report 97-002, Computer Science Dept, Texas A&M University, Feb. 1997.
- [8] Jane Liu, "Real-Time System," (draft).
- [9] 임성화, 김재훈, "이동 컴퓨팅을 위한 실시간 브로드캐스트 알고리즘," 한국 정보과학회 추계 학술발표 논문집, 제26권 1호, pp.364-366, 1999.