

<https://doi.org/10.7236/JIIBC.2021.21.5.61>

JIIBC 2021-5-9

질의 우선순위와 마킹에 기초한 충돌 회피 온디맨드 데이터 방송 스케줄링 기법

A Scheduling Scheme for Conflict Avoidance On-demand Data Broadcast based on Query Priority and Marking

권혁민*

Hyeokmin Kwon*

요약 온디맨드 방송은 모바일 컴퓨팅 환경에서 데이터를 효과적으로 전송하기 위한 기술이다. 본 논문은 온디맨드 방송 환경에서 다중 데이터 질의를 스케줄링하기 위한 주제를 연구하여, CASS라 명명된 새로운 방송 스케줄링 기법을 제안한다. 제안된 기법은 다중 데이터 질의의 특성을 반영하여 질의의 우선순위를 정하고, 질의의 우선순위에 따라 질의에서 가장 오랫동안 방송되지 않은 데이터를 방송 데이터로 선정한다. CASS 기법의 성능 평가를 위해 시뮬레이션이 수행되었다. 실험 결과에 따르면, 제안된 기법은 질의 데이터 인접성 및 데이터 충돌률 측면에서 매우 바람직한 특성을 보이기 때문에 평균 응답시간의 성능에서 다른 기법보다 우수한 성능을 보인다.

Abstract On-demand broadcast is an effective data dissemination technique in mobile computing environments. This paper explores the issues for scheduling multi-data queries in on-demand broadcast environments, and proposes a new broadcast scheduling scheme named CASS. The proposed scheme prioritizes queries by reflecting the characteristics of multi-data queries, and selects the data that has not been broadcast in the query for the longest time as the broadcast data according to the query priority. Simulation is performed to evaluate the performance of CASS. The simulation results show that the proposed scheme outperforms other schemes in terms of the average response time since it can show highly desirable characteristics in the aspects of query data adjacency and data conflict rate.

Key Words : broadcast schedule, mobile computing, on-demand broadcast, simulation

1. 서 론

모바일(mobile) 장치와 무선 네트워크 기술의 비약적인 발전으로 인하여 모바일 컴퓨팅 환경에서 정보 서비스가 점점 보편화 되고 있다. 이에 따라 모바일 클라이언

트로 데이터를 효과적으로 전송하기 위한 기법들이 큰 관심을 끌고 있는데, 방송(broadcast) 방식에 기초한 데이터 전달 기법은 모바일 컴퓨팅 환경에서 정보 전달을 위한 주요 기술로 인정받고 있다^[1-12]. 데이터 방송 시스템에서 방송 서버는 방송채널을 통하여 데이터들을 순차

*종신회원, 세명대학교 정보통신학부
접수일자 2021년 7월 19일, 수정완료 2021년 9월 13일
게재확정일자 2021년 10월 8일

Received: 19 July, 2021 / Revised: 13 September, 2021 /
Accepted: 8 October, 2021

*Corresponding Author: hmkwon@semyung.ac.kr
School of Information and Communication, Semyung
University, Korea

적으로 전파하고, 각 클라이언트는 자신이 원하는 데이터가 방송채널에 나타나면 이를 액세스한다. 데이터 방송 시스템에서 어떤 데이터의 방송은 그 데이터를 원하는 모든 클라이언트들의 요구를 충족시킬 수 있기 때문에 시스템의 확장성(scalability)이 매우 우수하다는 장점이 있어 방대한 규모의 클라이언트들을 지원해야 하는 응용 분야에 매우 적합하다.

데이터 방송 시스템의 주요 성능지수는 질의(query)의 평균 응답시간(response time)인데, 이는 클라이언트가 특정 질의를 제기한 순간부터 자신이 원하는 모든 데이터들을 수신할 때까지 걸린 대기시간을 의미한다. 이 응답시간 성능은 방송 스케줄링 기법에 크게 영향을 받는데, 방송 스케줄링 기법은 각 방송채널이 어느 순간에 어떤 데이터들을 방송할 것인가의 문제를 다룬다^{[1]-[11]}. 방송 스케줄링 기법들은 푸쉬기반(push-based)과 풀기반(pull-based) 기법으로 분류할 수 있는데^{[1]-[3]}, 푸쉬기반의 스케줄링 기법은 데이터 액세스 확률이나 이미 파악된 질의 프로파일 등을 이용하여 방송 스케줄을 미리 구성하여 정해진 순서대로 데이터들을 주기적으로 방송한다. 푸쉬기반의 기법은 데이터의 액세스 확률을 미리 파악할 수 있는 환경에서는 거의 최적에 근접하는 성능을 발휘할 수 있지만, 데이터 액세스 확률을 파악하기가 어렵거나, 데이터의 액세스 패턴이 자주 변하는 환경에서는 사용하기가 어렵다는 단점이 있다.

반면, 온디맨드(on-demand) 방송 스케줄링 기법으로 알려진 풀기반 기법에서는 클라이언트가 자신이 필요로 하는 데이터들을 방송 서버에 명시적으로 요청하고, 서버는 이 정보를 바탕으로 질의의 평균 응답시간이 최소화되는 방향으로 다음에 방송할 데이터들을 선정한다. 온디맨드 방송 스케줄링 기법은 푸쉬기반 기법에 비하여 데이터베이스의 규모에 따른 확장성이 우수할 뿐만 아니라, 클라이언트들의 데이터의 액세스 패턴이 자주 변하는 동적 워크로드(workload) 환경에 훨씬 더 적합한 기법이다^{[1]-[3]}. 그러나 많은 수의 클라이언트들이 존재할 경우에는, 이들의 요청 사항들을 취합하여 다음에 방송할 데이터를 선정하기 위한 실행부담이 매우 커지게 된다. 따라서 온디맨드 방송 시스템은 한 데이터의 방송시간 내에 다음에 방송할 데이터를 적시에 선정할 수 있는 효율적인 알고리즘이 필요하다.

온디맨드 방송 시스템에 적용 가능한 스케줄링 기법으로는 FCFS(First Come First Served), MRF(Most Requests First), 그리고 LWL(Longest Wait First) 등과 같은 전통적인 기법들이 있는데^[2, 4], 이들은 기본적인

로 단일 데이터를 액세스하는 질의에 최적의 성능을 발휘하도록 설계되었다. 다수의 데이터를 액세스하는 다중 데이터 질의의 경우에는 자신이 필요로 하는 데이터를 모두 수신하는 순간 응답시간이 결정되므로 기존의 단일 데이터 질의를 위한 스케줄링 기법을 그대로 적용하기에는 무리가 있다. 다중 데이터 질의가 제기되는 환경을 위하여 DUP(Dynamic Urgency and Productivity)^[5]와 MPFH(Most Popular First Heuristic)^[6] 기법이 제안되었는데, 이들은 동일 질의의 데이터들이 동시에 전송되는 데이터 충돌의 문제를 심도있게 고려하지 않았다. 일반적으로 클라이언트들은 한 순간에는 한 방송채널만을 수신할 수 있기 때문에 데이터 충돌이 많이 발생할 경우에는 응답시간의 성능이 크게 저하될 수 있다.

본 논문에서는 이와 같은 점을 고려하여 다중 데이터 질의 환경에서 온디맨드 방송 스케줄링 기법에 대한 연구를 진행하여, 질의의 응답시간의 성능을 크게 개선할 수 있는 새로운 기법을 제안한다. 본 논문이 제안하는 기법은 마킹 개념에 기초하여 데이터 충돌 가능성을 최소화하고, 질의의 방송 효율성과 대기시간 등을 고려하여 질의 우선순위를 결정하여 질의 중심으로 방송 데이터를 선정하는 정책을 채택하여 응답시간을 성능을 최적화한다. 본 논문의 구성은 다음과 같다. 2 장에서 기존에 개발된 기법에 대하여 기술하고, 3 장에서는 본 논문이 제안하는 새로운 방송 스케줄링 기법에 대하여 기술한다. 그리고 4 장에서는 제안된 기법의 성능을 평가, 분석하고, 5장에서 결론을 맺는다.

II. 관련 연구

이 장에서는 기존에 제안된 방송 스케줄링 기법에 대하여 기술한다. 방송 시스템을 위하여 DP(Dynamic Programming)^[7], PBA(Placement-Based Allocation)^[8]와 DAMA(Data Allocation using Moving Average)^[9] 등의 푸쉬기반의 스케줄링 기법들이 제안되었다. 그런데 기본적으로 푸쉬기반의 기법은 각 데이터의 액세스 확률이나 질의 프로파일 등을 미리 파악할 수 있어야 한다는 제약이 있고, 클라이언트의 액세스 패턴이 동적으로 변할 경우에는 적용하기가 어렵다는 단점이 있다. 이와 같은 환경에서는 클라이언트가 전송하는 데이터 요청 정보를 바탕으로 다음에 방송할 데이터를 선정하는 온디맨드 방송 스케줄링 기법을 적용하는 것이 바람직하다.

온디맨드 방송 스케줄링 기법들은 단일 데이터 질의 환경을 위한 기법과 다중 데이터 질의 환경을 위한 기법으로 분류할 수 있는데, 단일 데이터 질의 환경을 위한 스케줄링 기법으로는 FCFS, MRF, LWL, RxW 기법들이 있다^[2, 4]. FCFS 기법은 클라이언트가 요청한 순서대로 데이터를 방송하며, MRF 기법은 요청 횟수가 가장 많은 데이터를 방송한다. LWF 기법은 각 데이터마다 전체 대기시간을 관리하여, 전체 대기시간이 가장 큰 데이터를 선정하여 방송한다. 이 기법들 중에서 LWF 기법이 가장 우수한 성능을 발휘하는 것으로 알려져 있지만, 데이터 선정을 위한 실행 부담이 큰 편이다.

이 실행 부담을 완화하기 위하여 LWF 기법의 휴리스틱(heuristic) 알고리즘으로 볼 수 있는 RxW라고 명명된 기법이 제안되었는데^[4], RxW 기법은 클라이언트에서 요청된 각 데이터별로 요청 횟수 및 그 요청의 처음 도착 시간 정보를 유지 관리한다. 그리고 데이터의 요청 횟수 R과 그 데이터가 처음 요청된 시간에서 현재까지 흐른 시간 W의 곱으로 계산되는 RxW 값에 근거하여 다음에 방송할 데이터를 선정한다. 가장 큰 RxW 값을 가진 데이터 항목이 다음에 방송할 데이터로 선정된다. RxW 기법에서는 자주 요청되었거나 요청된 시간이 오래 경과된 데이터가 방송 데이터로 선정될 가능성이 높은 기법으로서 적절한 실행부담을 보이면서도 단일 데이터 질의 환경에서 매우 바람직한 성능 특성을 보인다^[4].

다중 데이터 질의 환경을 위한 온디맨드 스케줄링 기법에는 DUP^[5], MPFH^[6], MPLH(Most Popular Last Heuristic)^[6], UPF(Urgent and Popular First)^[10], 그리고 PVC(Profit Versus Cost)^[3] 기법들이 있다. DUP 기법은 각 질의가 액세스하는 데이터 중 전체 질의에서 가장 많이 요청된 데이터의 요청 횟수를 의미하는 질의의 요청 생산성(request productivity)과 질의의 대기시간을 서비스되지 않은 데이터의 수로 나누어 질의의 요청 긴급성(request urgency)을 계산한다. 그리고 요청 생산성의 가중치와 요청 긴급성 곱으로 표현되는 질의 우선순위를 계산한다. 그리고 가장 우선순위가 높은 질의를 선택하여 그 질의의 데이터 중 요청횟수가 가장 많은 데이터를 방송 데이터로 선정한다. 그리고 다음 우선순위의 질의를 선택하여 방송 데이터를 선정하는데, 방송채널의 수 만큼의 데이터를 선정할 때까지 이 과정을 반복한다. 비록 이 기법이 MRF와 RxW 기법보다 훨씬 우수한 응답시간의 성능을 보이지만^[5], 다중 데이터 질의 환경에서 발생하는 데이터 충돌에 대한 문제를 심도있게 고려하지 않았기 때문에 성능을 향상시키는 데는 한계가

있다.

MPFH^[6], MPLH^[6], 그리고 UPF^[10] 기법들은 서비스 종료까지 시간 제한(time constraint)을 가진 실시간 질의를 처리하기 위하여 제안되었다. MPFH 기법은 각 질의에서 가장 요청 횟수가 많은 데이터를 다음에 방송할 후보자로 선정하는 반면, MPLH 기법은 요청 횟수가 가장 적은 데이터를 방송 후보자로 선정한다. 방송 후보자를 선정할 때, 해당 질의의 서비스 제한시간과 서비스되지 않은 데이터의 수도 함께 관리되는데, 다수의 질의에서 동일 데이터가 방송 후보자로 선정되면 이들을 하나의 후보로 통합하는 과정을 거친다. 방송 후보자들이 방송채널의 수보다 적으면 이들을 모두 방송 데이터로 선정되며, 그렇지 않으면 서비스 종료 제한시간이 가장 적게 남아 있는 데이터들을 채널의 수만큼 선정하여 방송한다. 만일 종료 제한시간이 동률이면 서비스되지 않은 데이터의 수와 요청 횟수를 고려하여 방송 데이터를 선정한다. 이 기법은 방송 데이터를 선정하기 위한 실행부담이 비교적 적은 편이나 데이터 중심의 스케줄링 정책을 사용하기 때문에 데이터 충돌이 많이 발생한다는 단점이 있다. UPF 기법은 제한시간이 가장 적게 남은 질의를 선정해 그 질의의 모든 데이터들을 특정 채널에 연속적으로 할당하여 채널 전환이 적게 발생시키는 연구를 수행했다.

PVC^[3] 기법은 요청된 질의에 포함된 데이터들을 방송함으로써 얻는 이익(Profit)과 기회 비용(Opportunity Cost)를 계산하여 다음에 방송할 데이터를 결정한다. 이 기법은 바람직한 성능 특성을 보일수 있지만, 매우 복잡한 계산을 수행하며 방송 데이터를 선정하기 때문에 스케줄링을 위한 실행 부담이 매우 크다는 단점이 있다. HDAMA(Hybrid Data Allocation using Moving Average)^[11] 연구에서는 푸쉬기반과 풀기반의 장점을 결합할 수 있는 하이브리드 기법을 제안했다. HDAMA는 질의 크기에 따라 데이터의 액세스 확률이 방송 빈도에 미치는 영향력을 적절히 제어하며, 풀 데이터를 비교적 적시에 방송할 수 있기 때문에 우수한 성능을 발휘한다. 그러나 하이브리드 기법의 특성상 푸쉬채널에 할당된 데이터를 수신하기 위해서는 일정시간 대기해야 하므로 질의 요청률이 낮은 환경에서 상대적으로 낮은 성능을 보인다.

III. 새로운 방송 스케줄링 기법

이장에서는 본 논문이 제안하는 새로운 방송 스케줄링

기법에 대하여 기술한다. 기존에 제안된 온디맨드 방송 스케줄링 기법들은 대부분 다중 데이터 질의 환경에서 많이 발생하는 데이터 충돌에 대한 연구가 심도 있게 이루어지지 않았다. 본 논문에서는 이와 같은 점을 고려하여 질의 우선순위를 결정하는 방법을 제안하고, 이미 수신할 데이터가 있는 질의를 마킹하는 개념을 사용하여 데이터 충돌 가능성을 최소화할 수 있는 충돌 회피 스케줄링 기법(CASS: Conflict Avoidance Scheduling Scheme)으로 명명된 새로운 기법을 제안한다.

1. 기본 시스템 모델 및 용어

온디맨드 데이터 방송 시스템의 구조는 방송 서버, 다수의 모바일 클라이언트, 그리고 상향링크(uplink)와 하향링크(downlink)로 구성된다. 클라이언트들은 상향링크를 통하여 질의를 방송 서버에 전송하며, 서버는 이 질의 요청 정보들을 바탕으로 방송 데이터를 선정한다. 클라이언트는 질의를 방송 서버에 전송한 후, 질의의 데이터가 방송채널을 통하여 방송되면 이를 수신한다. 본 논문은 다음의 시스템 모델과 용어를 사용한다.

- 1) 전체 데이터들은 $d_k(1 \leq k \leq N)$ 로 표현되는 동일 크기의 N 개의 데이터들로 구성된다.
- 2) 방송 시스템에는 방대한 수의 클라이언트들이 존재하며, 각 클라이언트는 자신이 요청한 질의의 서비스가 완료되어야 다음 질의를 제기할 수 있다.
- 3) 질의는 Q 로 표현하며, 각 Q 는 자신이 액세스하려는 다수 개의 데이터로 구성된다.
- 4) 방송 서버에는 동일 대역폭을 갖는 K 개의 방송채널이 존재하는데, 방송 서버는 클라이언트에서 제기되는 질의 정보를 바탕으로 적절한 데이터를 선정하여 방송채널을 통하여 방송한다.
- 5) 클라이언트는 어느 한 순간에는 한 방송채널만을 수신할 수 있다. 만일 어떤 질의에 포함된 데이터들이 동시에 방송되면 이 중 하나의 데이터만을 수신하며, 이 경우 데이터 충돌이 발생했다고 표현한다.
- 6) 한 방송채널에서 하나의 데이터를 방송하는데 걸리는 논리적인 시간을 1 tick으로 정의하고, 방송주기 및 응답시간 성능의 단위로서 tick을 사용한다.

2. 충돌 회피 온디맨드 데이터 방송 스케줄링 알고리즘

본 논문이 제안한 CASS 기법은 다음 방송 구간에서 방송할 데이터들을 선정하기 위하여 시스템에 제기된 질의들의 우선순위를 결정하고, 질의 우선순위 순으로 각 질의에서 한 개의 방송 데이터를 선정한다. 온디맨드 데이터 방송 시스템에서 응답시간 성능을 높이기 위해서는 다음과 같은 사항을 고려하여 질의의 우선순위를 결정해야 한다. 첫째, 가급적 방송 효율성이 높은 데이터, 즉 많은 클라이언트들의 서비스 요구를 충족시킬 수 있는 데이터들을 포함하고 있는 질의일수록 우선순위가 높아야 한다. 둘째, 질의의 응답시간은 마지막 수신한 데이터에 의해 결정되므로 종료시점에 가까워진 질의에 우선권을 주어야 한다.

본 논문은 이와 같은 방식으로 질의 우선순위를 결정하기 위하여 전체 질의에서 각 데이터 d_i 의 요청 횟수를 $DataReqNum(d_i)$ 에 저장 관리하는데, $DataReqNum(d_i)$ 는 d_i 를 요구하는 새로운 질의가 제기되거나, d_i 가 방송될 때마다 다시 계산된다. 그리고 각 질의 Q_j 가 요청된 시간을 $ReqTime(Q_j)$ 에 저장하며, Q_j 의 데이터 중에서 아직 서비스되지 않은 데이터들을 $UnservSet(Q_j)$ 에 유지 관리한다. 질의 Q_j 의 방송 효율성(BE: Broadcast Effectiveness)은 식 (1)과 같이 아직 서비스되지 않은 데이터들의 요청 횟수의 평균으로 계산한다. 식 (1)에서 $|UnservSet(Q_j)|$ 는 Q_j 에서 아직 서비스되지 않은 데이터의 개수를 의미한다.

전술한 두 가지 측면만을 고려하여 질의 우선순위를 결정하면 비인기 데이터를 많이 포함하고 있는 질의는 기아(starvation) 현상이 발생할 수 있다. 본 논문은 이 기아 문제를 해결하기 위하여 오래 대기한 질의일수록 가점을 부여하는 정책을 부가적으로 사용한다. 질의 우선순위는 식 (2)에 의해 결정하는데, 이 값이 클수록 우선순위가 높은 것이다. 이 수식에 의하면 방송 효율성이 높고 오래 대기한 질의일수록, 그리고 서비스되지 않은 데이터의 개수가 적을수록 우선순위가 높게 된다. 이 수식에서 α 값은 질의의 방송 효율성의 가중치를 의미하는 성능 튜닝을 위한 조정 변수로서, 질의 크기, 질의 요청률, 방송채널의 수, 데이터 항목의 수 등의 다양한 입력

$$BE(Q_j) = (\sum_{d_k \in UnservSet(Q_j)} DataReqNum(d_k)) / |UnservSet(Q_j)| \quad (1)$$

$$QueryPriority(Q_j) = BE(Q_j)^\alpha \times (Current\ Time - ReqTime(Q_j)) / |UnservSet(Q_j)| \quad (2)$$

요인들에 의해 영향을 받는데, 이런 요소들을 모두 반영하여 최적을 α 값을 해석적으로 결정하는 것은 현실적으로 불가능하다. 또한 입력 요소들의 값이 변화됨에 따라 최적의 α 값도 변하게 마련이다. 본 논문에서는 α 값을 변화시키면서 이 값이 응답시간의 성능에 미치는 영향을 살펴 동적으로 변하는 최적의 α 값을 파악할 것이다.

식 (2)에 의하여 질의 우선순위를 결정하고 나면, 우선 순위가 높은 질의부터 시작해 K 개의 방송 데이터가 선정될 때까지 각 질의에서 하나의 방송 데이터를 선정한다. 질의 Q 에서 방송 데이터를 선정할 때는 $UnServedSet(Q)$ 에서 방송한 지 가장 오래된 데이터를 선정한다. 이와 같은 휴리스틱을 적용하는 이유는 동일 질의의 데이터 방송의 인접성을 높이기 위함이다. 데이터 방송의 인접성이란 질의의 데이터들이 서로 얼마나 근접해서 방송되는가를 의미하는데, 다중 데이터 질의는 가장 늦게 수신된 데이터에 의해 응답시간이 결정되기 때문에 동일 질의의 데이터들을 서로 인접해서 방송해야 응답시간의 성능을 향상시킬 수 있다. 오랫동안 방송되지 않은 데이터는 요청 횟수가 적은 비인기 데이터일 가능성이 높기 때문에, 그 데이터를 방송하고 나면 보통 그 질의의 우선순위는 더 높아지게 된다. 따라서 다음번 방송 데이터를 선정할 때 해당 질의가 다시 선택되어 데이터 방송의 인접성을 높일 수 있다. 또한 인기 데이터들은 다른 데이터들을 방송하는 동안 클라이언트들에 의해 다시 요청될 가능성이 높기 때문에 인기 데이터의 방송을 약간 지연함으로써 한 번의 방송으로 더 많은 클라이언트들의 요구를 충족시킬 수 있다는 장점도 있다.

CASS 기법의 전체 알고리즘이 그림 1에 제시되어 있는데, 이 기법은 우선 모든 질의의 마킹을 해제하고, 다음 방송구간에서 방송할 데이터를 저장하기 위한 $BroadcastDataSet$ 을 초기화한다. 그리고 방송 데이터 선정을 위하여 식 (2)에 의하여 각 질의의 우선순위를 계산한다(알고리즘 ① 부위 참고). CASS 기법은 우선순위가 가장 높은 질의 Q 를 선택하여, $UnServedSet(Q)$ 에서 가장 오랫동안 방송되지 않은 데이터를 다음에 방송할 데이터로 선정하여 $BroadcastDataSet$ 에 추가하고 Q 를 마킹한다. 마킹된 질의는 수신할 데이터가 결정된 질의를 의미한다. 그리고 다음 우선순위의 질의를 선택하여 이 과정을 반복하는데, K 개의 방송 데이터를 선정할 때까지 반복한다(알고리즘 ②의 반복문 부위 참고). 질의 선택 과정에서 마킹된 질의는 제외함으로써 데이터 충돌 가능성을 최소화시킨다.

```

//input ( $K$  : number of broadcast channels,
//       $m$  : number of queries)
//output ( $BroadcastDataSet$  : broadcast data items)
unmark all queries and empty  $BroadcastDataSet$ .
①for each query  $Q$  from  $i=1$  to  $m$ 
    calculate  $QueryPriority(Q)$  with equation 2.
end for
sort  $QueryPriority(Q)$  according to its value.
②Loop  $K$  times
    select an unmarked query  $Q_i$ 
    in the sorted  $QueryPriority(Q)$  order.
    find data  $d_k$  with the most time elapsed
    since its last broadcast in  $UnServedSet(Q_i)$ .
    add  $d_k$  to  $BroadcastDataSet$ 
    mark  $Q_i$ 
    remove  $d_k$  from  $UnServedSet(Q_i)$ 
    set  $DataReqNum(d_k)$  to zero
    for each query  $Q_j$  ( $j \neq i$ )
        if ( $d_k \in UnServedSet(Q_j)$ )
            if ( $Q_j$  is not marked)
                mark  $Q_j$ 
                remove  $d_k$  from  $UnServedSet(Q_j)$ 
            else
                increse  $DataReqNum(d_k)$  by 1
            end if
        end if
    end for
    if there is no further unmarked query
        break;
    end if
End Loop
return  $BroadcastDataSet$ 
    
```

그림 1. 충돌 회피 온디맨드 데이터 방송 스케줄링 알고리즘
 Fig. 1. Conflict Avoidance On-demand Data Broadcast Scheduling algorithm

어떤 데이터 d_k 가 방송 데이터로 선정되면, d_k 를 $UnServedSet(Q)$ 에 포함하고 있는 Q 는 이 데이터를 액세스하면 되기 때문에 Q 도 마킹되며 $UnServedSet(Q)$ 에서 d_k 를 제거한다. 그런데 이미 Q_j 가 마킹되어 수신할 데이터가 결정되어 있다면, Q 는 데이터 충돌이 발생하게 되어 d_k 를 수신할 수 없게 된다. 원래 d_k 를 방송하면, 이를 원하는 모든 질의들이 이를 수신할 수 있기 때문에 $DataReqNum(d_k)$ 를 0으로 설정해야 하지만, 이와 같이 충돌이 발생하는 질의를 고려하여 $DataReqNum(d_k)$ 를

적절히 계산해야 한다(알고리즘 ③ 부위 참고). BroadcastDataSet에 설정된 데이터들은 다음 방송구간에서 각 방송채널에 1 개씩 할당되어 방송된다.

IV. 성능 평가 모델 및 성능 평가

이 장에서는 본 논문이 제안한 CASS 기법의 성능을 평가하고 그 결과를 기술한다. 비교 대상으로는 RxW^[4] DUP^[5], MPFH^[6], 그리고 HDAMA^[11] 기법을 선정하였다.

1. 성능평가 모델 및 입력 변수

본 논문의 성능평가 모델 및 입력 변수들은 연구의 연속성을 위해 HDAMA^[11] 연구와 유사하게 설정했다. 성능평가 모델은 크게 방송 서버와 모바일 클라이언트들로 구성되는데, 클라이언트는 질의 확률 모델에 따라 질의를 제기한다. 클라이언트는 한 순간에는 한 방송채널만 수신할 수 있으며, 현재 질의가 필요로 하는 모든 데이터를 수신한 후에 다음 질의를 제기할 수 있다. 본 성능평가 모델은 CSIM^[13]을 이용하여 구현하였으며, 본 논문에서 사용된 입력변수들은 표 1과 같다. Q는 질의 크기로서 질의가 액세스하려는 데이터의 개수를 의미하며, QRT(Query Requests per Tick)는 1 tick라는 단위시간당 요청되는 질의의 개수를 의미한다.

표 1. 성능 평가를 위한 입력 변수

Table 1. Input parameters for performance evaluation

입력변수	의미	기본값	설정
N	데이터 항목의 수	5000	5000
K	방송채널의 수	10	1 ~ 20
QRT	질의 요청률	10	1 ~ 50
Q	질의 크기	10	1 ~ 20
θ	zipf 분포의 θ 값	0.95	0.0 ~ 1.0
α	질의의 방송효용성 가중치	-	0.0 ~ 1.0

서버가 방송하는 전체 데이터들은 동일 크기의 N 개의 데이터로 구성되며, 각 데이터의 액세스 확률은 불균등한 액세스 형태를 모델링하는데 많이 사용되는 zipf 분포 모델^[4, 6] 따른다고 가정한다. zipf 분포 모델에서 각 데이터의 액세스 확률은 식 (3)으로 표현되는데, 각 질의는 이 액세스 확률에 근거하여 질의가 액세스할 데이터를 결정한다. 여기서 θ 값은 각 데이터들의 액세스 확률이 어느 정도 편향되어 있는지를 나타내는 인수로서, θ 값이 커질수록 불균등한 액세스의 정도가 심화된다.

$$p_i = \frac{(1/i)^\theta}{\sum_{j=1}^N (1/j)^\theta} \quad (1 \leq i \leq N) \quad (3)$$

질의의 방송 효용성의 가중치는 표 1의 다양한 입력 변수들에 의해 영향을 받는데, 궁극적으로는 이런 요소들에 의해 시스템 내에 동시에 처리되고 있는 질의의 평균 숫자가 변하게 된다. CASS 기법에서는 시스템 내에 서비스되고 있는 질의 개수를 파악하면서, α 값을 0.0 ~ 1.0으로 변화시키면서 성능 추이를 살펴보았다. 그 결과 질의 개수가 1000 이하일 경우에는 α 값을 0.0, 1000 ~ 6000인 경우는 0.1, 6000 이상일 경우는 0.25로 설정할 때 응답시간의 성능이 우수한 편이다. 따라서 CASS 기법에서는 이와 같이 시스템 내의 질의 개수에 따라 동적으로 변화하는 α 값을 사용하여 실험을 실시하였다. 성능평가 결과에서 제시된 결과 값은 5 개의 서로 다른 임의의 수를 사용하여 질의를 생성한 후 이 질의를 제기하면서 실시한 실험 결과의 평균값으로, 각 실험은 천만 개의 질의들이 처리될 때까지 실시하였다.

2. 질의 요청률 변화에 따른 평균 응답시간의 성능

이 질의는 질의 요청률을 1~50으로 변경시키면서 응답시간의 성능 추이를 살펴보았다. 방송채널의 수를 10, 질의 크기를 10, 그리고 θ 값은 데이터의 편향적인 액세스를 모델링하기 위한 대표 값으로 많이 사용되는 0.95로 설정하고^[9, 11] 실시한 실험의 평균 응답시간의 성능이 그림 2에 제시되어 있다.

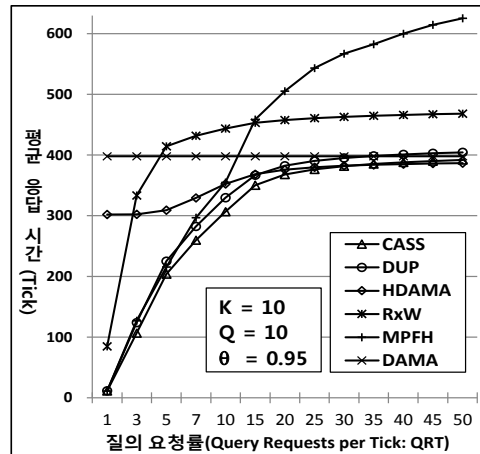


그림 2. 질의 요청률의 변화에 따른 평균 응답시간
Fig. 2. Query requests vs. average response time

이 실험에서는 질의 요청률이 높아짐에 따라 한정된 방송채널을 더 많은 질의들이 공유해야 하므로 각 기법

의 응답시간 성능은 저하되는데, 처음에는 급격하게 저하되다가 QRT가 30 이상이 되면 그 성능 저하가 미미한 편이다. 이는 온디맨드 방송 시스템은 QRT가 낮을 경우에는 질의 요청에 즉각 응답하는 것이 가능하여 성능이 우수하지만, 시스템에 질의가 어느 이상으로 포화 상태에 이르게 되면 푸쉬기반의 방송 시스템처럼 거의 일정한 성능을 보이는 특성이 있기 때문이다.

본 논문이 제안한 CASS 기법은 질의가 포화 상태에 이르는 경우 이외에는 다른 기법들에 비해 우수한 성능을 보인다. CASS 기법이 우수한 성능을 보이는 주된 이유는 이 기법은 질의에서 가장 오랫동안 방송하지 않은 데이터를 방송 데이터로 선정하는 정책을 사용하여 질의 우선순위에 따라 한번 선정된 질의는 연속해서 선정될 가능성이 높아 질의 데이터의 방송 인접성이 우수하기 때문이다. 또한 CASS 기법은 수신할 데이터가 있는 질의는 마감하여 질의 우선순위에 따라 제거함으로써 다른 기법들에 비해 데이터 충돌률이 낮기 때문이다. 참고로 질의 요청률이 10일 경우에 DUP 기법은 CASS에 비해 44% 정도의 더 많은 데이터 충돌을 야기하며, MPFH와 RxW 기법은 각각 295%, 254% 정도의 더 많은 데이터 충돌을 발생시킨다. DUP 기법은 질의 우선순위에 기초해 방송 데이터를 선정하므로 데이터 충돌을 어느 정도 방지하지만, 이미 수신할 데이터가 있는 질의도 그 질의의 다른 데이터를 방송 데이터로 선정할 수 있기 때문에 CASS 기법보다 더 많은 데이터 충돌을 야기한다. MPFH 기법은 각 질의에서 가장 요청횟수가 많은 데이터를 방송 후보자로 선정하고 이 중에서 방송 데이터를 선정하는 데이터 중심의 정책을 사용하므로 데이터 충돌률이 높다. RxW는 데이터 충돌 회피 전략이 없기 때문에 데이터 충돌을 많이 야기하는데, 오랫동안 방송되지 않은 데이터도 방송 데이터로 선정될 가능성이 있기 때문에 데이터 충돌을 어느정도 완화해 줄 수 있다.

푸쉬기반 기법 중 우수한 성능을 보이는 DAMA 기법의 성능을 그림 2에 참고로 제시하고 있는데, 푸쉬기반 기법은 질의 요청률에 관계없이 항상 일정한 성능을 보인다. CASS 기법은 전 QRT 구간에서 DAMA 기법보다 우수한 성능을 보인다. HDAMA 기법은 풀기반과 푸쉬기반 기술을 병합하여 액세스 확률이 높은 데이터는 푸쉬채널에 할당하여 주기적으로 방송하며, 액세스 확률이 낮은 데이터들은 요청에 응답하여 풀채널을 통해 방송한다. HDAMA 기법은 질의 요청률이 낮은 경우에는 다른 기법들에 비해 매우 낮은 상대적 성능을 보이는데, 이는 푸쉬채널에 할당된 데이터를 액세스하기 위해서 일정

한 시간을 필요로 하기 때문이다. 그러나 HDAMA 기법은 QRT가 35 이상이 되어 질의 포화 상태가 심화되면 가장 우수한 성능을 보이기 시작한다.

3. 질의 크기의 변화에 따른 평균 응답시간의 성능

이 절에서는 질의 크기를 1~15까지 변화시키면서 각 기법의 성능추이를 살펴보았다. 이 실험은 QRT를 질의의 포화가 시작되기 직전 상태인 15로, K 와 θ 값을 각각 10과 0.95로 설정하고 실시하였다. 질의 크기의 변화에 따른 평균 응답시간 성능이 그림 3에 제시되어 있는데, 질의 크기가 커질수록 모든 기법들은 성능이 저하된다. 이는 질의가 완료되기 위해서는 더 많은 수의 데이터를 액세스해야 하기 때문이다. 그림 3의 성능 추이를 살펴보면, CASS 기법은 질의 크기에 관계없이 다른 기법보다 항상 우수한 성능을 보인다. 이는 Q 가 변하여 시스템 내에 질의의 평균 숫자가 변화하더라도, CASS 기법은 질의의 방송 효율도의 가중치 α 값을 최적으로 가깝게 동적으로 변화시켜 질의 우선순위를 계산하는 정책을 사용하기 때문이다.

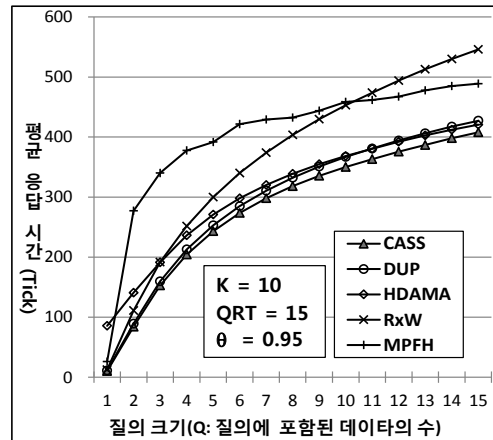


그림 3. 질의 크기에 따른 평균 응답시간
 Fig. 3. Query size vs. average response time

질의 크기가 1일 경우에는 데이터 충돌이 전혀 발생하지 않기 때문에 가장 인기 있는 데이터를 방송 후보자로 선정하는 MPFH 기법도 우수한 성능을 보일 것 같지만 실제로는 CASS나 DUP에 비해 2배 이상의 응답시간을 보인다. 이는 인기 데이터의 방송을 조금만 지연하더라도 그 사이에 그 데이터의 요청이 많이 쌓여 한 번의 방송으로 더 많은 질의의 요구를 충족시킬 수 있는데, MPFH는 인기 데이터를 너무 빈번하게 방송하여 방송 대역폭을 낭비하는 경향이 크기 때문이다. MPFH 기법

은 Q가 1인 경우 전체 5000개의 데이터 중 가장 요청이 많은 d_0 를 방송하는데 전체 방송채널 대역폭의 5%를 사용한다. 반면 RxW, DUP, CASS, 그리고 HDAMA 기법은 각각 2.1%, 1.6%, 0.9%, 그리고 0.1%의 대역폭을 사용한다. CASS 기법은 DUP나 RxW에 비해 데이터 요청 횟수의 가중치를 더 작게 설정하므로 d_0 를 위해 더 작은 대역폭을 사용한다. HDAMA 기법은 인기 데이터들을 푸쉬채널에 일정하게 할당하기 때문에 d_0 를 방송하는데 가장 작은 대역폭을 사용한다.

HDAMA 기법은 푸쉬채널에 할당된 데이터를 액세스하기 위해서는 일정시간 대기해야 하므로 질의 크기가 작을 경우에는 상대적으로 매우 낮은 응답시간 성능을 보인다. 그러나 Q가 커짐에 따라 상대적인 응답시간의 성능이 빠르게 개선된다. 이는 HDAMA 기법이 질의 크기에 따라 데이터의 액세스 확률이 방송 빈도에 미치는 영향력을 적절하게 제어하기 때문이다. 질의 크기가 작을 경우에는 데이터 충돌 확률이 작을 뿐만 아니라, 요청 횟수가 많거나 오래 대기한 데이터를 선정하여 방송하면 응답시간의 성능을 개선할 수 있기 때문에 RxW는 CASS와 DUP 기법과 유사한 성능을 보인다. 그러나 RxW 기법은 요청횟수의 가중치가 너무 높은 경향이 있기 때문에 액세스 확률이 낮은 데이터를 포함하고 있는 질의의 경우에는 응답시간이 매우 길어지게 된다. 질의 크기가 커짐에 따라 비인기 데이터를 포함하고 있는 질의 비율이 높아지기 때문에 Q의 증가에 따라 RxW 기법은 급격하게 성능이 저하되기 시작하여 Q가 11 이상으로 증가하게 되면 가장 낮은 성능을 보인다. MPFH 기법도 질의가 커짐에 따라 급격하게 성능이 저하되지만, 질의 크기가 6 이상이 되면 이미 시스템에 질의들이 포화상태에 이르기 때문에 그 성능 저하는 미미한 편이다.

4. 채널 수의 변화에 따른 평균 응답시간의 성능

이 절에서는 방송채널의 수를 1~20으로 변화시키면서 각 기법의 성능 추이를 살펴보았다. 이 실험에서 Q는 10, QRT는 10으로, 그리고 θ 값은 0.95로 설정하였다. 그림 4의 성능 결과를 살펴보면, 방송채널의 수가 증가함에 따라 모든 기법들의 질의 응답시간이 감소하는 것을 확인할 수 있다. 이는 K가 증가함에 따라, 이에 비례하여 각 tick당 방송할 수 있는 데이터의 수가 증가하기 때문이다. 이 실험에서 CASS 기법은 모든 채널 구간에서 다른 기법보다 우수한 응답시간의 성능을 보인다. 이는 CASS 기법은 다른 기법들에 비하여 데이터 충돌률이 낮을 뿐만 아니라 질의 데이터들의 방송 인접성이 우수

하기 때문이다. 참고로 채널의 수가 14일 경우에 CASS는 DUP와 MPFH 기법에 비해 각각 9%, 14% 더 우수한 응답시간의 성능을 보이는데, 데이터 인접성 측면에서는 각각 13%, 17% 정도의 우수한 특성을 보인다. HDAMA 기법은 일부 데이터들을 일방적으로 푸쉬채널에 할당하고, RxW 기법은 방송 데이터 선정시 질의가 관여하지 않기 때문에 인접성 측면의 특성은 큰 의미가 없다.

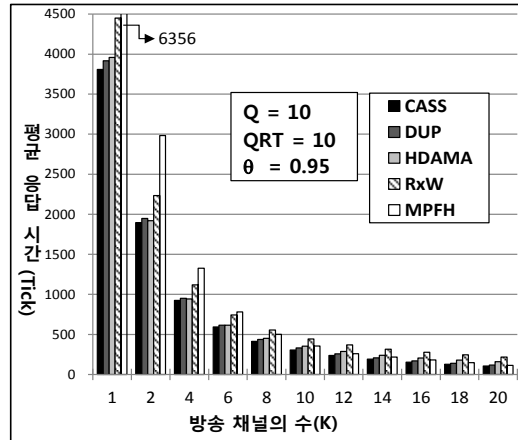


그림 4. 방송 채널의 수에 따른 평균 응답시간
Fig. 4. No. of channel vs. average response time

채널의 수가 증가하면, 방송 데이터 선정 정책의 차이가 성능에 더 큰 영향을 미칠 뿐만 아니라, 질의의 포화 상태가 더 완화되므로 더 큰 상대적인 성능 차이를 보인다. 그림 4의 결과를 살펴보면, K가 10일 경우에 CASS 기법은 DUP, HDAMA, RxW 기법에 비해 각각 7%, 15%, 45% 정도 우수한 성능을 보이는데, K가 20이 되면 그 성능 차이가 11%, 52%, 108% 정도로 더 커진다. MPFH 기법은 질의의 포화상태에 따라 가장 크게 영향을 받는데, K가 작을 경우에는 인기 데이터를 방송하느라 너무 많은 방송 대역폭을 사용하여 다른 기법에 비하여 매우 낮은 성능을 보이지만, K가 증가하여 질의의 포화 상태가 완화되면 성능이 상대적으로 더 빠르게 향상되어 K가 20이 되면 CASS를 제외한 다른 기법보다 더 우수한 성능을 보인다.

V. 결론

본 논문은 온디맨드 데이터 방송 시스템에서 다수의 데이터를 검색하는 다중 데이터 질의를 효율적으로 처리하기 위한 연구를 수행했다. 이 연구는 질의의 응답시간을 최소화시키는 것을 목표로 하여 다중 데이터 질의의 특성을 반영하여

질의의 우선순위를 정하고, 질의 내에서 가장 오랫동안 방송되지 않은 데이터를 방송 데이터로 선정하는 CASS라 명명된 새로운 데이터 방송 스케줄링 기법을 제안했다. 본 논문에서는 온디맨드 방송 시스템을 위한 사물레이션 모델을 설계 구현하여 성능평가를 실시하였다. 제안된 기법은 데이터 충돌 확률과 질의 데이터의 방송 인접성을 개선할 수 있기 때문에 대부분의 환경에서 다른 기법보다 우수한 평균 응답시간의 성능을 보였다.

본 논문의 미래 연구로서는 질의가 일정 시간 내에 처리되어야 하는 실시간 질의를 처리하기 위한 방송 스케줄링 기법에 대한 연구를 진행할 것이다.

References

- [1] M. Narvekar, H. Mukadam, and S.S. Mantha, "Comparison of Broadcast Scheduling Techniques in Mobile Computing Environment," *Procedia Computer Sciences* Vol. 45, pp. 706-711, 2015.
DOI: <https://doi.org/10.1016/j.procs.2015.03.136>
- [2] D. Aksoy and M.S. Leung, "Pull vs Push: A Quantitative Comparison for Data Broadcast," in *Proc. of the 47th IEEE Global Telecommunications Conference (GLOBECOM'04)*, pp. 1464-1468, 2004.
DOI: <https://doi.org/10.1109/GLOCOM.2004.1378225>
- [3] J. Lv, V.C.S. Lee, M. Li, and E. Chen, "Profit-based scheduling and channel allocation for multi-item requests in real-time on-demand data broadcast systems," *Data & Knowledge Engineering*, Vol. 73, pp. 23-42, 2012.
DOI: <https://doi.org/10.1016/j.datak.2011.09.005>
- [4] D. Aksoy and M. Franklin, "R×W: a scheduling approach for large-scale on-demand data broadcast," *IEEE/ACM Trans. on Networking*, Vol. 7, No. 6, pp. 846-860, 1999.
DOI: <https://doi.org/10.1109/90.811450>
- [5] Kai Liu and Victor C. S. Lee, "On-demand broadcast for multiple-item requests in a multiple-channel environment," *The Journal of Information Sciences*, Vol. 180, Issue 22, pp. 4336-4352, 2010.
DOI: <https://doi.org/10.1016/j.ins.2010.07.030>
- [6] C. M. Liu and T. C. Su, "Broadcasting on-demand data with time constraints using multiple channels in wireless broadcast environments," *Information Sciences*, Vol. 242, No. 1, pp. 76-91, 2013.
DOI: <https://doi.org/10.1016/j.ins.2013.04.026>
- [7] W.G. Yee, S. Navathe, E. Omiecinski, and C. Jermaine, "Efficient Data Allocation over Multiple Channels at Broadcast Servers," *IEEE Trans. on Computers*, Vol. 51, No. 10, pp. 1231-1236, Oct. 2002. DOI: <https://doi.org/10.1109/TC.2002.1039849>.
- [8] H.P. Hung, J.W. Huang, J.L. Huang, and M.S. Chen, "Scheduling dependent items in data broadcasting environments," *ACM SAC* 2006.
DOI: <https://doi.org/10.1145/1141277.1141555>
- [9] H.M. Kwon, "A Broadcast Data Allocation Scheme for Multiple-Data Queries Using Moving Average of Data Access Probability," *The Journal of the Institute of Internet, Broadcasting and Communication(IIBC)*, Vol. 14, No. 5, pp. 35-43, 2014.
DOI: <https://doi.org/10.7236/JIIBC.2014.14.5.35>.
- [10] P. He, H. Shen, and H. Tian, "On-demand Data Broadcast with Deadlines for avoiding conflicts in wireless networks," *The Journal of Systems and Software*, Vol. 103, pp. 118-127, 2015.
DOI: <https://doi.org/10.1016/j.jss.2015.01.022>
- [11] H.M. Kwon, "A Hybrid Data Broadcast Scheduling Scheme Using Moving Average of Data Access Probability," *The Journal of the Institute of Internet, Broadcasting and Communication(IIBC)*, Vol. 18, No. 5, pp. 131-140, 2018.
DOI: <https://doi.org/10.7235/JIIBC.2018.18.5.131>
- [12] J.W. Ha, J.H. Lee, and S.K. Lee, "Progressive Method for Efficient Skyline Query Processing in Data Broadcast Environments," *The Journal of KIIT*, Vol. 8, No. 2, pp. 127-135, 2010.
- [13] H. Schwetman, "CSIM19: A Power Tool for Building System Models," *CSIM Users' Guide for Use with CSIM Revision 16'*, in *Proc. of the 33rd IEEE Winter Simulation (WSC'01)*, pp. 250-255, 2001.
DOI:10.1109/WSC.2001.977280

저자 소개

권혁민(중신회원)



- 1984년 : 서울대학교 제어계측공학과 학사
- 1994년 : 한국과학기술원 정보및통신공학과 석사
- 1998년 : 한국과학기술원 정보및통신공학과 박사

- 1984년 ~ 1991년 : 대우전자 중앙연구소 컴퓨터개발부 선임연구원
- 1999년 ~ 현재 : 세명대학교 정보통신학부 교수
- 주관심분야 : 분산/병렬데이터베이스, 모바일 컴퓨팅, 데이터 방송시스템

※ 이 논문은 2019학년도 세명대학교 교내학술연구비 지원에 의해 수행된 연구임.