차량 네트워크 게이트웨이에서의 프리페치 성능 평가 시뮬레이터 구현1)

이정훈
제주대학교 전산통계학과
e-mail:jhlee@jejunu.ac.kr

Implementation of a performance evaluation simulator for the prefetch scheme in vehicular networks

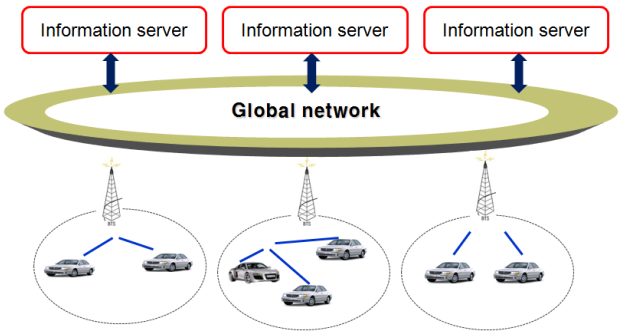
Junghoon Lee
Dept of Computer Science and Statistics
Jeju National University

요 약

본 논문은 차량형 네트워크에서 고급화된 루트 조회 요청의 응답시간을 측정하는 시뮬레이터를 구현하고 그 측정된 결과를 도시한다. 이산 이벤트 시뮬레이터에 기반하여 서버의 주기적 업데이트, 게이트웨이에서의 레코드당 리프레쉬, 차량으로부터의 조회 요청 발생 등의 사건들을 정의하고 이에 대한 동작을 구현한다. 시간 지역성을 대표하는 LRU 방식과 공간 지역성을 대표하는 FAR 방식의 성능을 측정하기 위해 참조시간을 기록하는 자료구조와 경위도 좌표를 거리로 변환하는 모듈을 작성하였으며 기존의 위치 이력 데이터로부터 조회 요청을 생성한다. 응답시간을 측정한 결과는 공간 지역성이 시간 지역성보다 응답시간에 더 많은 영향을 미치는 것으로 나타났으며 낮은 프리페치 부하에서 최대 4배 가량의 응답시간 차이를 보인다.

1. 서론

차량 네트워크는 무선통신의 다양성만큼 다양한 형태로 보급되고 있으며 이 네트워크는 차량들이 자율적으로 통신하는 차량 에드혹 네트워크와 셀룰라망과 같이 완벽한 연결성을 가진 인프라스트럭쳐 기반 네트워크 등으로 구성될 수 있다. 이들은 각각 연결성, 통신비용, 대역폭 등에서 장단점을 가지며 이를 결합한 형태로 인포스테이션, RSU (RoadSide Unit), WiFi 액세스 포인트 등을 설치하여 주변의 차량들과 통신하는 차량형 텔레매틱스 네트워크도 점차 시험되고 설치되고 있다[1].



(그림 1) 게이트웨이 기반 차량형 네트워크

각 차량들은 게이트웨이를 통해 실시간 교통정보, 날씨정보, 주변 공항이나 대중교통 스케줄 등에 대한 정보를 요청하게 된다. 게이트웨이들은 차량들에 대한 네트워크 인터페이스뿐만 아니라 글로벌 네트워크에 대한 연결을 갖고 있어서 이를 통한 각 서버들과의 상호작용에 의해 필요 정보를 서버에서 다운로드한 다음 차량에게 전달하게 된다. 만약 차량이 요청한 정보가 이미 게이트웨이에 다운로드가 되어있다면 이는 즉각적으로 차량에게 전달될 수 있기 때문에 게이트웨이는 차량이 요청할 가능성이 높은 정보들을 서버에서 프리페치해두면 응답시간을 크게 개선할 수 있다[3].

위에서 언급된 정보들은 각 서버에서 주기적으로 업데이트된다. 따라서 어떤 정보의 복사본이 게이트웨이에 저장되어 있다 하여도 서버에서 업데이트가 된다면 그 복사본은 차량에게 제공될 수 없다. 따라서 프리페치는 서버 업데이트 후 순차적으로 이루어져야 하는데 도로망에서 관련된 모든 정보를 각 게이트웨이에 복사시키는 것은 불가능하므로 향후 참조될 가능성이 높은 레코드들이 먼저 갱신되어야 한다.

데이터에 대한 요청 패턴은 서비스와 차량의 이동 패턴에 따라 결정이 되는데 본 논문에서 대상으로 하는 서비스는 특정 경로에 대해 단순한 경로 결정이나 추천이 아닌 주변의 식당, 식당의 좌석가용성, 실시간 쿠폰, 교통상황 등 다양한 동적정보를 제공한다[2]. 도로명 혹은 아이디가 키가 되며 이를 기준으로 정보들이 검색된다. 또, 차량의 이동 패턴 파악을 위해 실제 궤적 데이터에서 출발지와 목적지를 추출할 수 있으며 이 두 지점간에 A* 알고리즘을

1) 본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음 (NIPA-2009-C1090-0902-0040)

통해 차량이 이동하는 경로들의 모든 도로들을 추정하고 모든 도로들에 대한 정보가 검색된다고 가정한다.

2. 성능분석기의 구현

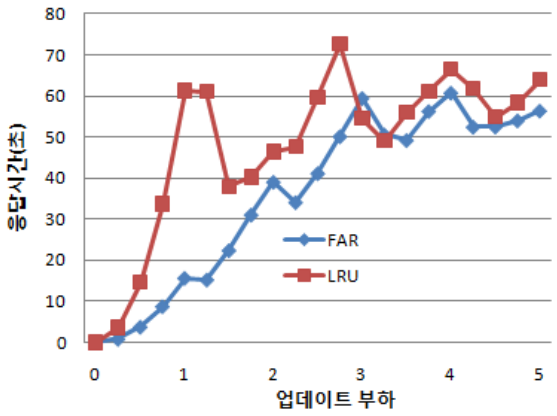
본 논문에서 구현된 시뮬레이터는 이산 이벤트 스케줄러인 SMPL에 기반하여 서버 업데이트, 레코드 리프레쉬, 루트 조회 요청 도착 등의 3 가지 사건으로 이루어져있으며 루트 조회 요청은 루트에 포함된 다수의 링크 레코드들에 대한 요청으로 구분되어 큐에 유지되고 있다[4]. 게이트웨이의 각 링크 레코드들에 대한 유효성을 나타내는 배열이 추가적으로 관리된다. 위치 이력 데이터에서 각각의 트립들을 분리하여 시작시간에 루트 조회 요청이 발생하도록 하며 24 시간 동안의 실제 요청을 재현한다.

이벤트에 대한 처리 과정은 다음과 같다. 서버 업데이트가 발생하면 각 게이트웨이에 포함된 복사본의 유효성이 취소되며 이 순간부터 각 링크 레코드들이 프리페치 정책이 결정한 순서에 따라 순차적으로 리프레쉬된다. 레코드 리프레쉬는 하나의 링크레코드를 리프레쉬할 때마다 요청큐를 검색하여 해당 링크의 데이터에 펜딩된 링크조회를 삭제한다. 모든 링크 대기가 삭제되면 루트 조회가 삭제될 수 있으며 이 시간에 루트조회 요청의 완료된 것으로 결정한다. 루트 조회 요청이 발생하면 루트에 포함된 각 링크별로 게이트웨이에 보관된 자료의 유효성을 검사하며 모든 링크에 대한 자료들이 이미 포함된 경우는 즉각적인 대답을 할 수 있다. 이외의 경우는 유효하지 않은 링크들의 리스트를 펜딩 큐에 삽입한다.

프리페치 정책은 캐쉬 관리정책과 쌍대적으로 LRU (Least Recently Used) first 대신 LRU last, FAR (Furthest Away Replacement) 대신 FAR (Furthest Away Refresh) last 등의 정책을 고려한다. 즉 가장 최근에 참조된 링크의 자료가 가장 먼저 프리페치되는 방식과 게이트웨이에서 가장 가까운 링크가 먼저 프리페치되는 방식을 따를 수 있다. LRU를 구현하기 위하여 각 링크별로 최종 액세스 시간을 저장하는 필드가 추가되며 서버의 업데이트 이벤트마다 프리페치 순서를 동적으로 계산한다. 이 과정은 최종 액세스 시간을 정렬하는 과정인데 참조가 한번도 안된 링크들인 경우 모두 같은 값을 갖게 되며 이는 프로그램의 수행시간을 단축하기 위해 대상에서 제외된다. 반면 거리 순서로 페치하는 방식은 시뮬레이션 루프 시작 이전에 미리 각 링크로부터 게이트웨이 위치까지의 거리를 계산하여 역시 정렬한다. 이 방식은 시간의 흐름에도 순서가 변하지 않는다.

3. 수행 결과 및 결론

(그림 2)는 본 논문에서 구현된 성능분석기를 이용하여 루트 정보 조회에 대한 응답시간을 측정한 결과이다. 의도로 네트워크의 경우 링크는 25000 여개이며 가장 통행량이 많은 지역은 제주시청 부근이다. 이 곳에 게이트웨이가 설치되어 있다고 가정을 하였으며 부근의 차량들은 출발전에 자신의 텔레매틱스 장치를 이용하여 루트를 결정한 후 이를 게이트웨이에게 전달하여 각 링크에 대한 교통정보 등을 조회한다고 가정한다. 본 실험에서는 프리페치 성능에 중점을 두기 위하여 단일한 서버가 모든 데이터를 유지하며 10분마다 데이터를 업데이트 한다고 가정하였다. (그림 2)는 그 응답시간을 측정한 결과이다. 이 그림에서 업데이트 부하는 서버 업데이트 주기 대 전체 링크의 업데이트 시간을 의미한다. 즉 업데이트 부하가 1.0 이면 10분에 25000 개 링크에 대한 업데이트가 게이트웨이에서 수행된다. 5.0 이면 5000 개 링크에 대한 업데이트가 수행됨을 의미한다.



(그림 2) FAR와 LRU의 응답시간 비교 결과

(그림2)에서 보는 바와 같이 FAR 방식이 전반적으로 응답시간이 우수하다. 이는 루트 조회 서비스에서는 시간적 지역성보다는 공간적 지역성이 성능에 더 큰 영향을 줌을 보이고 있다. 업데이트 부하가 커지게 되면 두 방식의 응답시간은 거의 차이가 없게 되는데 이는 프리페치 과정 중에 조회요청이 발생할 가능성이 많기 때문이며 페치 대상 링크의 집합을 축소시켜야 할 필요가 있음을 의미한다. 본 논문에서는 차량형 네트워크에서 고급화된 루트 조회 요청의 응답시간을 측정하는 시뮬레이터를 구현하고 그 측정된 결과를 도시하였다. 이산 이벤트 시뮬레이터에 기반하여 서버의 주기적 업데이트, 게이트웨이에서의 레코드당 리프레쉬, 차량으로부터의 조회 요청 발생 등의 사건들을 정의하고 이에 대한 동작을 구현하였다. 응답시간을 측정한 결과는 공간 지역성이 시간 지역성보다 응답시간에 더 많은 영향을 미치는 것으로 나타났다.

참고문헌

[1] B. Yu, J., Gong, C., Xu, "Data aggregation and roadside unit placement for a vanet traffic information system," ACM VANET pp.49-57, 2008.

[2] 이정훈, "SMPL을 이용한 노변 네트워크 캐쉬 성능 분석기의 구현," 한국정보처리학회 춘계학술대회, 2009년.

[3] E. Said, E. Omar, and L. Robert, "Data prefetching algorithm in mobile environments." European Journal of Scientific Research, Vol. 28, No. 3, pp. 478-491, 2009.

[4] M. H. MacDougall, Simulating Computer Systems: Techniques and Tools, MIT Press, 1987.