Journal of the Korea Academia-Industrial cooperation Society
Vol. 16, No. 12 pp. 8370-8377, 2015

http://dx.doi.org/10.5762/KAIS.2015.16.12.8370
ISSN 1975-4701 / eISSN 2288-4688

공간 데이터 스트림 질의 정확도 향상을 위한 다단계 부하제한 기법

정원일[1*]
[1]호서대학교 정보보호학과

Multi-level Load Shedding Scheme to Increase Spatial Data Stream Query Accuracy

Weonil Jeong[1*]
[1]Dept. of Information Security Engineering, Hoseo University

요 약 공간 데이터 스트림 관리 시스템에 실시간으로 입력되는 공간 데이터 스트림은 제한된 주기억장치의 용량을 초과할 수 있으므로 부하를 제한할 필요가 있다. 그러나 기존의 연구에서는 부하 제한을 위해 공간 데이터 스트림을 생성하는 데이터 소스의 특성이나 입력 변화, 그리고 공간 데이터 이용 정도를 효율적으로 적용하지 못함으로써 질의 처리의 정확도와 성능을 감소시키는 문제를 갖고 있다. 이에 본 연구에서는 공간 데이터 스트림 질의 관리 시스템에서 발생할 수 있는 부하를 제한하고 공간 질의 처리의 성능과 정확도를 높이기 위한 다단계 부하제한 기법을 제안한다. 제안 기법에서는 먼저 데이터를 수집하는 단계에서 데이터의 수량과 입력 빈도 변화를 이용하여 부하를 제한하고, 과부하 발생시 공간 이용도에 따라 질의 참여 확률이 낮은 데이터를 대상으로 추가적인 부하제한을 수행한다. 실험 결과에서 제안 기법은 기존 부하제한 기법에 비해 11% 이상의 부하 제한 발생 빈도를 감소시키면서 입력 데이터 스트림의 증가와 질의 영역에 증가에 따른 질의 처리 결과의 정확도는 0.04% 이상의 우위를 보였다. 또한, 질의 처리 성능에서도 기존 기법에 비해 3% 이상의 향상을 나타냈다.

Abstract In spatial data stream management systems, it is needed appropriate load shedding algorithm because real-time input spatial data streams could exceed the limitation of main memory. However previous researches, lack regard for input ratio and spatial utilization rates of spatial data streams, or the characteristics of data source which generates data streams with spatial information efficiently, can lead to decrease the performance and accuracy of spatial data stream query. Therefore, multi-level load shedding scheme for spatial data stream management systems is proposed to increase the spatial query performance and accuracy. This proposed scheme limits overloads in relation to the input rate and the characteristics of data source first, and then, if needed, query data representing low query participation probability based on spatial utilizations are dropped relatively. Our experiments show that the proposed method could decrease load shedding frequency for previous researches by more than 11% despite query results accuracy and query performance are superior at 0.04% and 3%.

Keywords : Spatial Data Stream, Load Shedding, Spatial Continuous Query

1. 서론

유비쿼터스 환경에서는 건물, 도로, 하천, 지하 시설물과 같은 지형·지물에 대한 공간 정보와 GeoSensor에서 생성되는 위치 정보가 포함된 실시간 데이터 스트림을 연계 처리하여 다양한 응용을 지원하는 u-GIS 플랫폼 기술이 널리 연구되고 있다[1-2]. GeoSensor는 고정되거나 이동하면서 시간에 따라 고유의 위치 및 센싱 정보를 실시간으로 대용량 데이터를 생성하는 데이터 스트림 소스로, 이러한 GeoSensor 데이터를 처리하기 위해서는 실시간 공간 데이터 스트림 처리 기술이 요구된다[3,4].

*Corresponding Author : Weonil Jeong(Hoseo University)
Tel: +82-41-540-5984 email: wnchung@hoseo.edu
Received August 20, 2015 Revised October 14, 2015
Accepted December 4, 2015 Published December 31, 2015

공간 데이터 스트림은 실시간으로 입력되는 대용량 데이터로 시간의 흐름에 따라 입력 데이터의 크기와 빈도가 매우 유동적인 특성을 나타낸다. 이는 서버의 제한된 주기억장치의 용량을 초과하는 상황을 야기함으로써 데이터가 손실되는 현상을 초래할 수 있다. 이러한 현상은 공간 데이터 스트림에 대한 질의 처리 결과의 정확도를 감소시킬 수 있어 부하 제한을 통해 문제를 해결하려는 기법들이 연구되고 있다[5-11].

Brian 기법은 데이터의 입력 속도나 튜플 처리 속도를 이용해 부하를 제한하고 있으나, 위치 정보를 포함한 공간 정보에 특성을 반영하지 않고 있어 공간 데이터 스트림에 대한 질의 정확도가 저하되는 문제를 야기하고 있다[5].

공간 특성을 반영한 부하 제한 연구로는 공간 질의의 영역 겹침 정도에 따라 공간 데이터 스트림이 질의에 참여할 확률이 낮은 데이터에 대해 샘플링을 통해 부하를 제한하는 연구[8], 불필요한 데이터를 사전에 필터링하고 공간 중요도와 데이터 중요도를 고려하여 추가적인 부하 제한을 수행하는 연구[9], 공간 데이터 스트림의 입력 변화량과 공간 데이터의 밀집도를 기반으로 부하를 제한하는 연구[11]가 있다. 그러나 [8]에서는 부하 제한을 위해 입력되는 공간 데이터 스트림의 유입되는 변화량과 튜플 처리율에 대해 고려하지 않고 있다. 이로 인해 유동적인 공간 데이터 스트림에 대한 부하 제한으로 인한 질의 처리 성능이 효율적이지 못한 결과를 보인다. [9]에서는 선-필터링 과정에서 데이터 스트림의 유입 변화율을 반영하지 않아 부하 제한의 효과가 저하되고, 부하 제한을 위해 데이터의 개수와 크기, 그리고 센싱 값을 이용하여 가중치를 산정하고 있으나 해당 데이터가 실제 질의에 이용되는 정도를 고려하지 않아 공간 질의 처리 성능과 정확도의 향상에 한계가 따른다. [11]은 중복되거나 불필요한 입력 데이터를 사전에 차단할 수 없어 공간 질의 처리의 성능 저하를 야기할 수 있고, 특정 질의 공간 영역에 대한 부하 정도를 산출하기 위해 연산 처리 비용에 대한 고려가 없어 질의 처리 결과의 정확도나 성능이 저하될 수 있는 문제점이 있다.

본 논문에서는 u-GIS 환경의 공간 데이터 스트림 관리 시스템에서 공간 데이터 스트림에 대한 질의 처리 결과의 정확도와 성능을 향상시키기 위한 다단계 부하 제한 기법을 제안한다. 제안 기법은 입력 데이터의 수량과 입력 빈도 변화 정도에 따라 공간 질의 처리에 불필요한 데이터를 적응적으로 사전에 차단하는 일차적인 부하 제한을 수행한 후 질의 공간 영역에 대한 연산 선택도와 입력 변화율, 그리고 연산 처리 속도를 반영하여 추가적인 부하 제한을 수행한다. 이를 위해 일차 부하 제한에서는 주기적으로 유입되는 데이터의 변화량과 데이터 특성을 분석하여 가중치를 산정하여 질의 처리 부하를 감쇄하기 위한 필터링을 수행한다. 이후 사용자에 의해 등록된 공간 질의에 대해 공간 이용도를 기반으로 해당 공간 영역에 대한 부하 정도를 계산하여 이차적인 부하 제한을 수행한다.

2. 관련연구

질의 정확도의 수준을 최대화하고, 부하 분산으로 인한 에러율을 균등하게 분산시키기 위해 Brian은 질의별로 적정 샘플링 비율을 선정하고 질의 처리에 있어 부하 제한을 실행할 수행할 위치를 결정하는 방법을 연구하였다[5]. 이 연구에서의 부하 제한은 임의 샘플링을 사용하여 주어진 샘플링 비율만큼 데이터를 감소시킨다.

UGLD 기법[8]은 활용 빈도가 높은 공간 영역에 가중치를 부여하여 샘플링을 수행하고, 빠른 부하 제한을 위해 해쉬 구조 기반의 스트림 구조를 이용하여, u-GIS 환경에 공간 데이터의 특성을 적극적으로 반영하여 부하 제한을 수행할 때 질의 정확도 및 처리 속도를 향상시키도록 한다. 그리고 공간 연산이 포함된 질의 계획을 생성할 때 공간 정보의 포함 여부에 상관없이 동시에 적용할 수 있는 질의 계획을 사용하여 샘플링을 수행함으로써 샘플링이 특정 공간 영역에 편중되는 것을 방지할 수 있다.

PFPLS 기법[9]은 GeoSensor 환경에서의 데이터스트림 관리 시스템으로 인해 발생할 수 있는 과부하를 제한하고 부하제한 수행 횟수의 감소 및 공간 질의 처리 결과의 정확도를 향상시키기 위해 1차적으로 선-필터링을 선행하여 데이터 유입의 폭증으로 인한 부하 발생에 대해 후-부하제한 기법을 수행한다. 이를 위해 GeoSensor에서 유입되는 데이터 특성을 분석하여 데이터 유입률에 따라 주기적으로 가중치를 갱신하여 필터링 범위를 결정한다. 후-부하제한에서는 공간 연속 질의의 공간 영역 이용도를 분석하고 센서에 의해 센싱된 값을 기초로 데이터의 중요도를 산출하여 중요도가 낮은 데이터부터 삭제

하여 부하 제한을 수행한다.

DSLD 기법[11]은 u-GIS 환경에서 효율적인 부하제한을 위해 질의 대상 영역을 공간 분할하여 구성한 그리드 해시 구조를 기반으로 공간 데이터스트림의 입력 빈도의 변화량과 공간 데이터의 분포 특성을 고려하여 동적으로 샘플링을 수행함으로써 공간적 특성을 고려한 부하제한 기법을 제안하였다. 제안 기법에서는 공간 연산 처리 속도를 향상을 위해 그리드 기반의 공간 분할 영역들에 대해 해시 구조를 이용하고, 질의 처리 결과에 대한 정확도 향상을 위해 공간 영역에 대해 저장 데이터의 밀집도를 통해 산출된 공간 이용도와 실시간으로 입력되는 공간 데이터스트림의 변화량을 이용해 동적인 부하 제한을 수행하였다.

3. 다단계 부하제한 기법

다단계 부하제한 기법은 데이터 소스로부터 입력되는 공간 데이터 스트림으로 인해 발생할 수 있는 부하를 제한하기 위해 먼저 데이터 수집 과정에서 스트림 큐에 과부하를 발생시킬 가능성이 높은 데이터를 사전에 제한하고, 이후 공간 질의 처리를 위한 스트림 큐에 부하가 발생하면 추가적인 부하 제한을 수행한다.

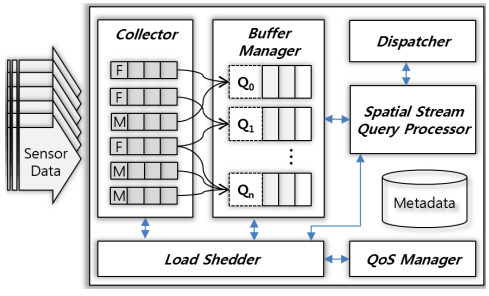

Fig. 1. Run-time Architecture for Spatial Data Stream Query Processing

Fig. 1은 제안하는 다단계 부하제한 기법을 적용하여 공간 데이터 스트림에 대한 질의를 처리하는 런타임 구조를 나타낸다. 데이터 스트림 소스에 따라 유입되는 센서 데이터의 수량과 크기 및 입력 변화에 따라 가중치를 산정하여 Collector에서 사전 필터링을 수행하며, 질의 처리를 위해 분할된 공간 영역을 대상으로 데이터의 중요도와 선택도를 기준으로 동적 샘플링을 수행하여 Buffer Manager의 스트림 큐에 추가적인 부하제한을 수행한다. 이러한 부하 제한은 Load Shedder에 의해 이뤄지며, 부하 제한 정도는 QoS Manager를 통해 조절된다.

3.1 유입 데이터의 가중치에 따른 부하제한

공간 데이터 스트림에 대한 질의 처리를 위한 스트림 큐에 많은 양의 데이터의 입력으로 인해 부하가 높아지게 되면 질의 처리 결과의 정확도와 성능이 급격히 감소한다. 또한, 각 스트림 큐에는 연동되는 데이터 스트림 소스로부터 입력되는 데이터의 유입량이나 속도가 일정하지 않다. 그러므로 각 스트림 큐에 입력되는 데이터의 상황을 고려하여 부하제한을 위한 가중치를 산정하고 적용한다.

데이터 스트림 소스로부터 스트림 큐로 유입되는 데이터의 유입율에 대한 산출식은 (Expr. 1)과 같으며, 이 산출식으로부터 스트림 큐에 대한 가중치를 도출한다.

$$R_i = \lceil F(D_i) \times S(D_i)/I_i \rceil \qquad \text{(Expr. 1)}$$

(Expr. 1)에서 $F(D_i)$는 i번째 데이터 스트림 소스 D_i에 대한 입력 변화의 정도를 나타내고, $S(D_i)$는 i번째 데이터 스트림 소스 D_i로부터 스트림 큐로 유입되는 데이터의 개수와 크기로부터 산출된 유입 데이터 량을 의미한다. I_i는 특정 데이터 스트림 소스에 설정되어 입력 데이터의 변화량이나 크기 산정하는 기준이 된다.

임의의 데이터 스트림 소스 D_i에 대한 입력 빈도 $F(D_i)$는 데이터 스트림의 데이터 생성 빈도의 변화율 $F_v(D_i)$와 현재 데이터 스트림의 입력 량을 나타내는 D_i^c의 합으로 유도된다. 여기서 $F_v(D_i)$는 과거 입력된 데이터 스트림의 변화량을 평균값으로 계산한 예측값에 대한 보정값인 $Avg(F^{p's}(D_i))$와 데이터 스트림의 직전 입력 빈도와 현재 입력 빈도에 대한 절대 수치를 나타내는 $|D_i^p - D_i^c|$을 합하여 계산된다.

(Expr. 1)로부터 데이터 스트림 소스 D_i에 대해 산출된 R_i로부터 부하제한을 위한 개별적 가중치는 아래 수식을 통해 결정된다.

$$W_i = \left\lceil R_i / \sum_{j=1}^{n} R_j \right\rceil \qquad \text{(Expr. 2)}$$

(Expr. 2)에서 개별적인 i번째 스트림 큐에 대한 가중치 W_i는 설정된 주기에 따라 각 스트림 큐에 대한 입력 빈도율을 전체 스트림 큐의 유입 빈도율의 합을 나누어 산출된다. 이로부터 유입되는 데이터 스트림의 유입 빈도가 높을수록 가중치도 높아진다.

특정 스트림 큐에 대한 가중치에 의한 부하 제한은 필터링 수행 단위를 결정해야 한다. 즉, 필터링 대상이 되는 데이터의 증감 단위가 필요하며, 이에 가중치 W_i는 최대 k이하의 자연수로 유도되고, 기본적인 증감 단위는 임의 설정 가능한 U값을 적용한다. 또한, 스트림 큐에 입력되는 데이터에 대한 값에 기반하여 해당 데이터를 필터링함으로써 중복되거나 무의미한 값의 입력들을 방지할 수 있어야 한다. 이에 갱신주기 동안 i번째 데이터 스트림 소스로부터 유입되는 데이터의 실제 값을 추출하여 평균값 M^i을 구하고, 이 평균값을 필터링의 기준값으로 활용한다. 이때, 가중치 W_i로부터 부하 제한을 위한 필터 최소값 L_i는 $\alpha(M_i - W_i)$로부터 산출되고, 필터 최대값 H_i는 $\alpha(M_i + W_i)$로 계산된다.

이러한 가중치에 대한 부하 제한 방법에서는 데이터 스트림 소스에 대응하는 스트림 큐에 독립적인 유입주기를 설정할 수 있고, 이 유입주기는 사용자 요구에 따라 변경될 수 있다. 이는 유입되는 데이터 스트림 처리에 따른 질의 결과의 정확도를 응용에서 요구하는 서비스의 수준에 맞추기 위한 것이다. 이동 객체와 같은 데이터 스트림 소스의 경우 위치가 빠르게 변화할 경우에는 질의 결과의 정확도 수준을 향상시키기 위해서는 유입 주기를 짧게 설정한다. 이와 같이 유입 데이터에 대한 가중치에 따른 부하제한 방법으로 질의 처리에 영향을 줄 수 있는 과부하를 미리 차단할 수 있다.

3.2 공간 이용도 기반의 부하제한

제안 시스템에서는 질의 공간 영역을 그리드 기반으로 분할하고, 입력되는 공간 데이터 스트림은 위치 정보를 갖는 객체와 공간 연산이나 조인 연산이 포함된 질의로 처리한다. Fig. 1에서는 질의 대상이 되는 공간 영역을 그리드 형태의 셀로 분할하여 관리하는 구조를 나타낸다.

Fig. 2에서 분할 영역의 크기는 임의로 설정할 수 있으며, 질의 처리를 위한 공간 영역에는 하나 이상의 분할 영역이 포함될 수 있다. 버켓 큐(Bucket Queue)에서는 질의 공간 영역에 대응하는 공간 데이트 스트림을 관리

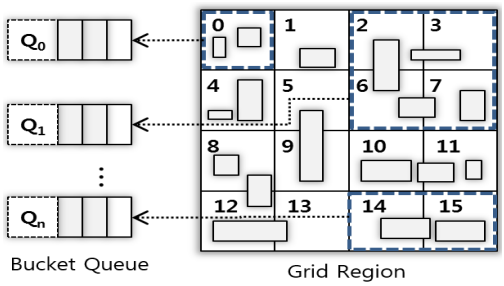

Fig. 2. Spatial query region management

하는 큐들로 관리된다. 버킷(Bucket)에는 식별자, 튜플 수, 가중치를 나타내는 정보를 포함한다. 각 버켓 큐별로 단위 시간에 입력되어 처리되는 데이터 스트림에 대한 입력 빈도, 연산 선택도와 처리시간을 이용하여 해당 질의 공간 영역에 대한 부하 정도를 계산한다.

버킷 B_i에 대한 데이터 스트림의 입력 빈도를 산출하기 위해서는 설정된 주기에 따라 누적된 해당 버킷에 대한 입력 빈도 변화율과 실제 입력 빈도를 이용한다. 입력 빈도 변화율을 계산하는 과정은 3.1절에서 기술한 임의의 데이터 스트림 소스에 대한 입력 빈도를 산출하는 과정과 그 대상이 버킷 큐라는 것을 제외하면 동일하다. 버킷 B_i에 대한 데이터 스트림 입력 빈도 변화율 $F_v(B_i)$을 구하는 수식은 Expr. 3과 같다.

$$F_v(B_i) = Avg(F^{p's}(B_i)) + (|B_i^p - B_i^c|) \qquad \text{(Expr. 3)}$$

Expr. 3에서 과거 입력 빈도 변화의 평균값인 $Avg(F^{p's}(B_i))$과 직접 입력 빈도와 현재 입력 빈도에 대한 절대 차이를 나타내는 $(|B_i^p - B_i^c|)$를 합한 결과로 $F_v(B_i)$은 산출된다.

버킷 B_i의 데이터 스트림 입력 빈도는 앞에서 계산된 입력 빈도 변화율과 현재 입력량을 이용하여 유도되며, 수식은 (Expr. 4)와 같다.

$$F(B_i) = F_v(B_i) + B_i^c \qquad \text{(Expr. 4)}$$

Expr. 4에서 버킷 B_i에 대한 입력 빈도 $F(B_i)$는 입력 변화율 $F_v(B_i)$와 현재 입력량을 나타내는 B_i^c의 합으로 표현된다.

위치 정보를 갖는 데이터 스트림은 연속 질의에 포함된 연산자에 의해 처리된다. 이때 연산 선택도는 특정 버

킷에 입력된 데이터 스트림에 대해 처리된 결과물의 비율을 의미한다[11]. 이러한 연산 선택도는 시간의 흐름에 따라 변화하게 되며, 질의 공간 영역에 대한 연산 선택도를 산출하기 위해서는 연산 선택도의 변화에 근간하여 구할 수 있다. 연산자 Op_i에 대한 선택도 변화율 $S_v(Op_i)$은 선택도 변화를 반영하는 변화율의 보정값 $Avg(S^{p's}(Op_i))$와 과거 선택도 $S^p(Op_i)$와 현재 선택도 $S^c(Op_i)$의 차이의 합으로 계산된다. 이로부터 연산자 Op_i에 대한 선택도는 아래 수식으로 표현된다.

$$S(Op_i) = S_v(Op_i) + S^c(Op_i) \quad \text{(Expr. 5)}$$

연산자 Op_i의 선택도 $S(Op_i)$는 연산자 선택도 변화율 $S_v(Op_i)$와 현재 선택도 $S^c(Op_i)$의 합으로 유도하고, 데이터 스트림이 연산에 처리되는 소요 시간을 산출하는 과정은 과거 데이터 처리 시간을 이용하여 계산한다.

$$T(B_i) = Avg(T^p(B_i^j)) \quad \text{(Expr. 6)}$$

Expr. 6에서 버킷 B_i에서 연산 처리에 소요되었던 과거 시간 $T^p(B_i)$들의 평균하여 산출하여 이후 부하 정도를 측정할 때 이용한다.

질의 공간 영역에 대한 부하는 위에서 기술한 버킷 B_i의 데이터 스트림 입력 빈도, 질의 공간 영역에 대한 연산 선택도, 그리고 연산을 처리하는 데 소요되는 시간 비용 $T(B_i)$을 주기적으로 측정하여 아래의 수식으로 산출한다.

$$L = \sum_{i=1}^{n} F(B_i) \times S(Op_i) \times T(B_i) \quad \text{(Expr. 7)}$$

Expr. 7에서 특정 질의 공간 영역에 대한 부하는 할당된 연산자의 수가 n개일 때, 각 연산자의 선택도 $S(Op_i)$와 데이터 스트림 입력 빈도 $F(B_i)$, 연산 처리 비용 $T(B_i)$에 대한 곱의 합으로 계산된다. 이는 임의 질의 공간 영역에 대한 데이터 입력 빈도, 공간 연산 선택도 및 연산 처리 비용의 곱에 비례한다는 것을 의미한다.

버킷에 입력되는 데이터 스트림에 대한 부하 정도는 단위는 사용자가 임의 설정할 수 있는 자연수 기반의 증감 단위를 활용한다[11]. 특정 버켓의 부하 정도 L_i는 사용자 요구에 따라 임의 설정 가능한 최대 k이하의 자연수 U를 기본적인 증감 단위로 이용한다. 이때 Fig. 1에서와 같이 질의 처리 대상인 공간 객체가 하나의 그리드 셀에만 포함되지 않고 다수의 그리드 셀과 겹치는 경우도 존재한다. 해당 공간 객체가 그리드 셀에 포함되거나 겹치는 횟수가 GN_i이고, GN_i 값들 가운데 가장 큰 값이 M이면 U는 M에 따라 조정될 수 있다.

$$U = \left\lceil \frac{M}{k} \right\rceil, \text{ if } kU < M \quad \text{(Expr. 8)}$$

$$L_i = \left\lceil \frac{GN_i}{U} \right\rceil, \ M = Max(GN_i) \quad \text{(Expr. 9)}$$

Expr. 8에서는 부하 정도의 증감 단위 U를 조정하는 수식을 나타낸다. 여기서 kU가 M보다 작은 경우에는 기설정된 U를 증감 단위로 L_i을 계산하며, L_i의 최대값은 k보다 클 수 없으므로 U의 범위를 M/k로 제한한다. Expr. 9는 해당 그리드 셀에서의 GN_i을 U로 나누고, 그 결과를 올림하여 L_i을 산출하는 수식이다. 높은 부하 정도는 해당 질의 공간 영역의 공간 객체가 연산 참여율이 높다는 것을 의미하고, 입력 빈도가 증가하는 경우에는 입력 데이터 스트림을 구성하는 데이터 중에서 적은 수의 데이터가 제한되어도 질의 처리 결의 정확도에 미치는 영향은 크지 않다는 것을 의미한다.

제안 기법에서는 부하 정도에 따른 부하 제한을 수행하기 위해 튜플 수를 기반으로 동적인 샘플링 기법을 적용한다. 각 공간 영역에 대한 부하 정도에 따라 샘플링하려는 튜플의 수 N_i^l는 아래 수식과 같이 유도된다.

$$N_i^l = N_i^t \times L_i \times (1-\alpha) \quad \text{(Expr. 10)}$$

Expr. 10에서 N_i^l는 모집단 N_i^t에 부하 정도 L_i와 샘플링 신뢰도를 의미하는 $(1-\alpha)$를 곱한 결과로 유도된다. 여기서 α는 부하 정도에 따라 샘플링 비율을 결정하는 상수로, α가 증가할수록 샘플링에 미치는 영향도 커지게 된다.

4. 성능 분석

4.1 평가 환경

본 실험은 CentOS 5.6 64비트 운영체제, 4GB 메모리, 펜티엄 3.0 GHz CPU, 512GB 디스크 시스템에서 수행하였다. 실험 데이터를 생성하기 위해서 고정 또는 이동 GeoSensor에 따라 데이터의 크기나 발생 주기 등을 설정할 수 있는 도구인 시티 시뮬레이터[12]를 사용하였다. 실험에 사용되는 기본 속성으로 공간 데이터 스트림을 생성하는 수는 총 10개, 유입되는 데이터 스트림을 저장하기 위한 스트림 큐의 크기는 8MB, GeoSensor 데이터는 초당 설정된 크기에 따라 발생하도록 설정하였다. 그리고 입력되는 GeoSensor 데이터와 공간 연산의 수행을 위한 지리정보 데이터는 TIGER/Line 2007 데이터[13]를 상용 데이터베이스관리시스템인 오라클에 구축하여 실험하였다.

4.2 성능 평가

제안 기법(MLLS)과 비교 대상은 공간 데이터 기반의 부하 제한 기법과의 평가를 위해 비공간 데이터를 위해 제안된 Brian[5], 공간 데이터에 대해 단일화된 부하 제한 기법으로 제안된 DSLD[11], 입력 공간 데이터에 대해 두 단계로 부하를 제한하는 PFPLS[9]로 선정하였다.

4.2.1 부하제한 발생 횟수 비교

과도한 부하제한 연산은 질의 처리 결과의 정확도를 저하시킬 수 있으므로 부하제한 발생을 최소화해야 한다. 실제 광범위한 지역에서 많은 센서로부터 수집된 데이터에는 중복이 발생할 확률이 높다[14]. 이러한 상황을 고려하여 이번 실험에서는 데이터의 중복 비율을 10%로 설정하고, 공간 데이터 스트림의 입력 증가에 따라 부하제한이 발생하는 횟수를 측정한다.

Fig. 3에서 Brian 기법은 공간 다른 기법들보다 평균 27.5%의 부하가 더 많이 발생하는 것으로 나타났다. 이는 Brian 기법은 부하 제한을 위해 샘플링 연산 등을 수행할 때 공간 데이터 스트림 기반의 공간 질의 처리에서 발생하는 부하를 반영하지 못하기 때문이다. 제안 MLLS 기법은 DSLD 기법보다 15% 정도로 부하 제한이 덜 발생하였는데, 이는 제안 기법에서는 중복되는 데이터에 대한 사전 필터링을 수행한 결과로 분석된다. 그리고 PFPLS 기법에 비해서는 평균 11%의 부하가 적게 발생하였으며, 이는 제안 기법에서는 사전 필터링 단계에서 공간 데이터 스트림의 입력 변화율을 반영하고, 또한 추가적인 부하제한 단계에서도 연산 선택도와 처리율을 적용한 것으로 분석된다.

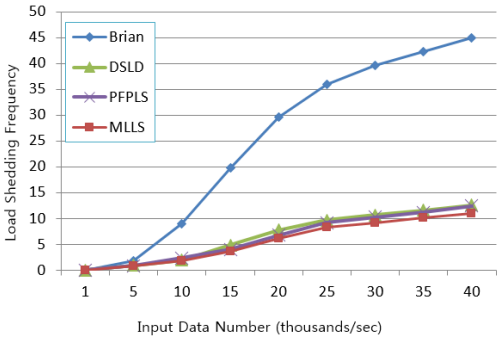

Fig. 3. Load shedding frequency according to increasing input data number

4.2.2 질의 처리 결과 정확도 비교

부하 제한을 위한 연산 수행은 공간 질의 처리에 적용되는 공간 데이터도 부하 제한의 대상으로 삭제될 수 있다. 이에 따라 공간 질의 처리 결과의 정확도를 저하시키는 요인이 된다. 이번 실험은 공간 데이터 스트림의 입력 증가에 따른 질의처리 결과의 정확도를 비교한다.

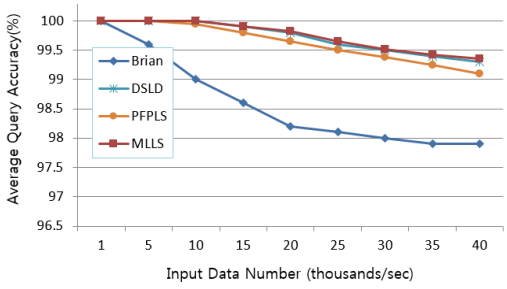

Fig. 4. Average query accuracy according to input data number

부하 제한 연산은 공간 질의 처리에 관련된 공간 데이터를 최대한 배제해야 한다. Fig. 4에서 Brian 기법은 비공간 데이터와 공간 데이터를 구분하지 않고 부하 제한을 수행하므로 다른 기법들에 비해 평균 1% 정도의 질의 처리 결과에 대한 정확도가 낮게 나타났다. PFPLS 기법은 제안 기법과 비교하여 평균 0.1% 질의 처리 정확도가 낮게 나타났으며, DSLD 기법은 제안 기법에 비해 거의 유사한 정확도를 보였다. 이는 Brian 기법을 제외하면 유의미한 차이를 나타내지 않음을 알 수 있다.

다음 실험은 공간 질의 영역의 변화에 따른 질의 처리

결과의 정확도를 비교한다. 공간 질의 처리를 위한 검색 범위가 증가하면 질의 처리 결과의 정확도는 감소한다.

Fig. 5에서 공간 질의 영역이 1%를 기점으로 Brian 기법에서 부하 제한의 발생으로 질의 정확도가 감소되기 시작하여 다른 기법들에 비해 평균 0.5% 정도의 평균 정확도가 낮게 나타났다. 이는 이전 실험에서 살펴본 바와 같이 Brian 기법에서는 부하 제한 의 대상에서 배제되어야 할 공간 데이터가 될 확률이 높기 때문이다. 그리고 제안 기법의 질의 처리 정확도는 DSLD 기법과 PFPLS 기법에 비해 약 0.04%의 미미한 차이를 보였다.

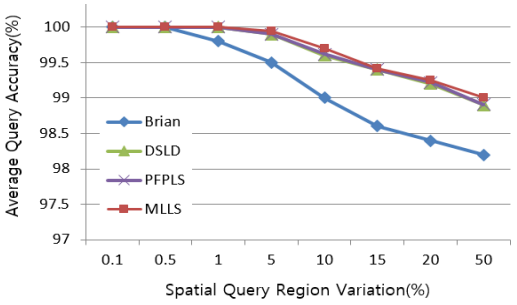

Fig. 5. Average query accuracy according to variation of spatial query region

4.2.3 질의 처리 성능 비교

공간 데이터 스트림 관리 시스템에서의 데이터 처리는 공간 데이터 뿐 아니라 비공간 데이터 처리도 포함한다. 이때 제안 기법에서 공간 연산에 사용되지 않는 속성 데이터들이 비공간 연산들의 공유로 인해 공간 데이터를 위한 연산을 불필요하게 수행하게 될 수 있다. 이러한 상황이 질의 처리 성능에 미치는 영향을 분석하기 위해 이번 실험에서는 비공간 질의가 공유될 때 공간 질의 영역의 변화에 따라 질의 처리 속도를 비교하였다. 이때 부하 제한 연산이 발생하지 않는 경우의 처리 속도를 100%로 가정했을 때 부하 제한 연산이 발생하는 경우의 처리 속도와 상대적인 비율로 실험하였다.

Fig. 6에서 비공간 연산이 공유된 상황에서 공간 검색 비율의 증가로 일부 비공간 데이터에 불필요한 연산이 발생하게 됨에도 불구하고 Brian 기법보다 다른 기법들의 평균 질의 처리 시간이 상대적으로 빠르게 나타났다. 이는 비공간 데이터에 대한 추가적인 연산 처리에도 불구하고 이어지는 공간 연산 처리에서의 속도 향상이 더 크다는 것을 의미한다. 또한, 제안 기법은 PFPLS 기법과 DSLD 기법에 비해 4% 및 3%의 비율로 질의 처리 비용의 이득을 보여주고 있다.

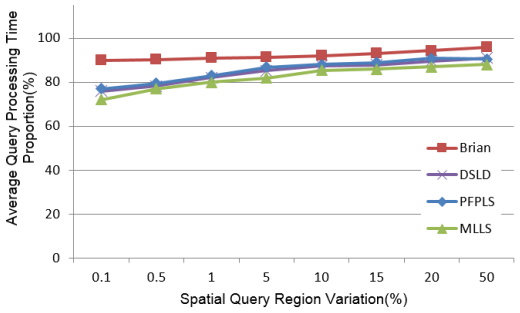

Fig. 6. Average query processing time according to the variation of query region

5. 결론

본 논문은 u-GIS 환경에서 GeoSensor로부터 수집되는 실시간 공간 데이터 스트림을 처리하는 시스템에서 발생할 수 있는 과부하를 효과적으로 제한함으로써 질의 처리 결과에 대한 정확성을 높이고 성능을 향상시키기 위한 다단계 부하 제한 기법을 제안하였다.

제안 기법에서는 우선 데이터 수집 단계에서 입력되는 공간 데이터 스트림의 크기와 빈도 변화를 분석하여 공간 질의 처리에 참여할 확률이 낮은 데이터를 적응적으로 사전에 차단하고, 질의 처리 과정에서 질의 영역에 대한 연산의 선택도와 입력 변화율 및 처리 속도를 이용하여 가중치를 산출하여 추가적인 부하 제한을 수행하였다. 실험을 통해 제안 기법이 기존의 기법들에 비해 질의 처리 결과의 정확도 향상에 있어서는 큰 차이를 보이지 않았으나, 부하제한 연산이 발생하는 횟수와 질의 처리 속도에서는 충분히 우수함을 증명하였다.

향후 연구로는 센서 네트워크, IoT 노드 등을 활용한 새로운 클러스터 환경에서의 부하 제한 기법들에 대한 연구가 필요하다. 또한 공간 및 비공간 연산이 포함된 질의 실행 계획에서 효과적인 부하 연산을 수행하기 위한 연산 스케줄링에 대한 연구도 수행되어야 한다.

References

[1] C. Lee, K. An, M. Lee, J. Kim, “Trends of u-GIS

Spatial Information Technology", ET Trends, Vol. 22, No. 3, pp.110-123, 2007.

[2] K. An, J. Kim, "Design of Mobile u-GIS Information Processing System", Proc. of 30th KIPS Fall Conference, Vol. 15, No. 02, pp. 315-317, November, 2008.

[3] W. Chung, and et. al., "GeoSensor Data Stream Processing System for u-GIS Computing", Journal of KSISS, Vol. 11, No. 1, pp. 9-16, 2009.

[4] H. Kang, C. Park, D. Hong, K. Han, "Development of a Spatial DSMS for Efficient Real-Time Processing of Spatial Sensor Data", Journal of KSIS, Vol. 9, No. 1, pp. 45-57, 2007.

[5] Brian B., Mayur D.,, and Rajeev M., "Load Shedding for Aggregation Queries over Data Streams", ICDE, pp. 350-361, 2004.
DOI: http://dx.doi.org/10.1109/ICDE.2004.1320010

[6] C. Basaran, K. Kang, Y. Zhou, M. Suzer, "Adaptive Load Shedding via Fuzzy Control in Data Stream Management Systems", SOCA, 5th IEEE International Conference, pp.1-8, Dec. 2012.
DOI: http://dx.doi.org/10.1109/soca.2012.6449438

[7] T. Pham, P. Chrysanthis, A. Labrinidis, "Self-managing load shedding for data stream management systems", ICDEW, IEEE 29th International Conference, pp70-76, Apr. 2013.
DOI: http://dx.doi.org/10.1109/icdew.2013.6547429

[8] S. Baek, D. Lee, G. Kim, W. Chung and H. Bae, "Load Shedding Method based on Grid Hash to Improve Accuracy of Spatial Sliding Window Aggregate Queries", Journal of KSIS, Vol. 11, No. 2, pp.89-98, 2009.

[9] H. Kim, S. Baek, D. Lee, G. Kim, H. Bae, "Pre-filtering based Post-Load Shedding Method for Improving Spatial Query Accuracy in GeoSensor Environment", Journal of KSISS, Vol. 12, No. 1, pp.18-27, 2010

[10] M. Ji, Y. Lee, G. Kim, and H. Bae, "A Dual Processing Load Shedding to Improve the Accuracy of Aggregate Queries on Clustering Environment of GeoSensor Data Stream", Journal of KSCI, Vol. 17, No. 1, pp. 31-40, 2012.
DOI: http://dx.doi.org/10.9708/jksci.2012.17.1.031

[11] W. Jeong, "Dynamic Load Shedding Scheme based on Input Rate of Spatial Data Stream and Data Density", Journal of KAIS, Vol. 16, No. 3, pp. 2158-2164, 2015.
DOI: http://dx.doi.org/10.5762/kais.2015.16.3.2158

[12] Kaufman, J., Myllymaki, J., and Jackson, J., "City Simulator", Alpha Works Emerging Technologies, Nov. 2001.

[13] "Tiger/LineShapefiles", census.gov/geo/www/tiger/tgrshp 2007/tgrshp2007.html, 2007.

[14] Konstantopoulos C. et al., "Effective Determination of Mobile Agent Itineraries for Data Aggregation on Sensor Networks," IEEE Transactions on Knowledge and Data Engineering, Vol. 22, pp. 1679-1693, 2010.
DOI: http://dx.doi.org/10.1109/TKDE.2009.203

정 원 일(Weonil Jeong) **[정회원]**

- 1998년 2월 : 인하대학교 전자계산공학과(공학사)
- 2004년 8월 : 인하대학교 컴퓨터정보공학과(공학박사)
- 2004년 7월 ~ 2006년 7월 : 한국전자통신연구원 선임연구원
- 2013년 1월 ~ 2014년 2월 : Univ. of Ohio Research Scholar
- 2007년 3월 ~ 현재 : 호서대학교 정보보호학과 교수

<관심분야>
공간데이터스트림, 클라우드보안, 시스템보안