스카이라인 질의 기법의 객관적 성능 평가를 위한 연구 조사

최종혁† · 나스리디노프 아지즈†
† 충북대학교 컴퓨터과학과

Survey for Objective Performance Evaluation of Skyline Query Methods

Jong-Hyeok Choi† · Aziz Nasridinov†
† Dept. of Computer Science, Chungbuk National University

요 약

스카이라인 질의는 데이터들 사이의 비교 연산을 통해 지배되지 않은 데이터들의 최소 집합을 스카이라인으로 탐색하며 이때 지배되지 않고 스카이라인으로 선택된 데이터들은 지배된 데이터들을 대표하게 된다. 이러한 특징은 금융, 네트워크, 웹서비스 등 다양한 분야에서 스카이라인의 활용을 이끌어냈다. 하지만 스카이라인 질의는 데이터의 양이나 차원의 수가 증가하는 경우 전체적인 성능이 크게 감소하는 문제를 야기하기 때문에 이를 해결하기 위한 다양한 기법들이 연구 및 제안되고 있다. 하지만 실제 스카이라인 질의를 활용하기 위해서는 객관적 성능 평가를 통해 주어진 상황에서 최적의 성능을 보일 수 있는 기법을 선택해야할 필요가 있지만 기존의 연구들은 성능 평가에 있어 각 기법이 목표한 문제들에 대한 단편적인 실험을 진행하고 있기 때문에 이들을 객관적으로 평가하기 위해서는 새로운 스카이라인 성능 평가 방법이 필요한 실정이다. 본 논문에서는 이러한 문제를 해결하기에 앞서 스카이라인 질의 기법의 객관적 성능 평가를 위한 품질 요소 선택 기준을 선택하기 위해 기존 연구들에 대한 조사와 분석을 진행한다.

1. 서 론

스카이라인 질의는 데이터베이스로부터 다른 데이터들에 지배되지 않은 데이터들의 최소 집합을 탐색한다. 이때 지배되지 않는다는 이야기는 다른 데이터들과 비교하여 모든 차원에서 같거나 최소 하나 이상의 차원에서 더욱 좋은 값을 지닌 것을 이야기한다. 이러한 특징으로 인해 스카이라인으로 선택된 데이터들은 데이터베이스에 저장된 데이터들을 대표할 수 있는 데이터들의 집합이라 이야기할 수 있다. 이러한 특징으로 인해 스카이라인은 금융, 네트워크, 웹서비스 등 다양한 분야에서 최적의 의사 결정을 지원하기 위해 활용되고 있다. 하지만 이러한 단순함과 반대로 스카이라인 질의는 데이터의 양이나 차원의 수가 증가하는 경우에 전체적인 성능 감소가 매우 두드러지는 문제가 발생한다. 이는 데이터의 양이나 차원의 수가 증가할수록 데이터 간의 단일 비교 연산의 비용과 그 횟수가 크게 증가하기 때문에 이를 해결하기 위해 다양한 스카이라인 질의 기법들이 제안되고 있다. 하지만 실제 스카이라인 질의를 활용하기 위해서는 해결하고자 하는 문제들의 요구사항을 면밀히 분석하고 각 기법들의 객관적인 성능 평가를 통해 최적의 성능을 보일 수 있는 기법을 선택해야할 필요가 있음에도 불구하고 기존의 연구들은 각 기법이 목표한 문제들에 대한 해결을 입증하기 위해 단편적이고 한정적인 실험을 실시하고 있기 때문에 연구가 아닌 스카이라인 질의를 실제 활용하고자 하는 측면에서는 각 기법의 단편적인 실험 결과만으로는 이들을 객관적으로 평가하기 매우 힘든 문제가 있다. 그렇기 때문에 다양한 스카이라인 질의 기법을 보다 객관적으로 평가할 수 있는 성능 평가 방법의 매우 요구되는 실정이다.

본 논문에서는 이러한 문제를 해결하기 위한 평가 방법의 제안에 앞서 다양한 스카이라인 질의 기법들을 조사 및 분류하고 이를 바탕으로 스카이라인 질의 기법들을 보다 객관적으로 성능 평가할 수 있는 품질 요소들을 탐색해 나간다.

이를 위한 본 논문의 구성은 다음과 같다. 2장에서는 다양한 스카이라인 질의 기법들을 각각의 특징에 따라 분류하고 이들로부터 기초적 품질 요소를 탐색한다. 3장에서는 본 논문의 내용을 요약하며 마친다.

2. 관련 연구

본 장에서는 스카이라인 질의 기법의 객관적 성능

평가 요소 탐색을 위해 기존 스카이라인 연구들을 전통적인 방식의 스카이라인, 공간 분할 기반의 스카이라인, 인덱스 구조 기반의 스카이라인, 근사 기법 기반의 스카이라인으로 구분하여 설명하고 각각의 연구로부터 객관적으로 평가가 수행되어야할 기초적인 품질 요소들을 탐색한다.

2.1 전통적인 방식의 스카이라인

초기의 스카이라인 질의는 모든 데이터들을 비교하는 방법이 주를 이루었다. 이러한 전통적인 방식의 스카이라인 중 가장 근간이 되는 방법은 Block Nested Loop(BNL)[1]라 불리는 기법으로, BNL의 가장 큰 특징은 스카이라인을 탐색하기 위해 스카이라인 후보 데이터들을 저장하기 위한 윈도우(window)라는 저장 공간을 사용한다는 점이다. BNL은 윈도우에 저장된 스카이라인 후보들과 데이터베이스로부터 읽어온 데이터들 사이의 비교를 통해 윈도우에 저장된 스카이라인 후보가 현재의 데이터를 지배하는 경우, 해당 데이터를 즉각 제거한 후 다음 입력 데이터와의 비교를 수행하는 방식으로 스카이라인을 탐색하였다. 하지만 BNL은 데이터들을 디스크 등에 저장된 순서대로 비교를 진행하는 방식이었기 때문에 스카이라인 후보로 선정된 데이터들은 단조 순서(monotone order)의 특성이 발휘되지 않았으며 이로 인해 스카이라인 후보로 선택되었던 데이터가 비교 과정에서 지배당하는 경우가 발생하는 등 윈도우가 효율적으로 운영되지는 못했다. 이러한 문제를 해결하기 제안된 Sort Filter Skyline(SFS)[2]는 데이터들의 위상 기대치가 갖게 되는 위상 수학적 특성을 이용하여 BNL이 갖는 문제점들을 해결하였다. SFS는 주어진 데이터의 정보량 점수(entropy score)가 좋을수록 스카이라인이 될 확률이 높다는 특징을 이용하여 주어진 데이터들을 정보량 점수를 기준으로 사전에 정렬하여 단조함수의 특성을 띄도록 하였으며 이로 인해 윈도우에 저장된 스카이라인 후보가 비교 대상 데이터에 의해 지배되는 경우가 발생하지 않았다. 하지만 해당 기법은 스카이라인을 탐색하기 위해 모든 데이터의 정보량 점수를 계산하거나 정렬하는 등 추가적인 연산을 수행해야했기 때문에 많은 데이터로부터 빠르게 스카이라인을 탐색하기에는 적합하지 못했다. 따라서 점차 대규모화 되는 데이터를 이용하여 스카이라인을 탐색하기 위해 SFS 보다 빠르게 스카이라인을 탐색할 수 있는 방법이 요구되기 시작하였으며 이를 위해 제안된 대표 방법들로는 Linear Elimination Sort for Skyline(LESS)[3]와 Sort and Limit Skyline Algorithm(SaLSa)[4]가 있으며 이들은 데이터를 조기에 제거하거나 비교를 조기에 종료시키는 방식으로 빠른 수행 시간을 확보하였다.

이처럼 전통적 방식의 스카이라인 기법들은 빠르고 보다 효율적으로 스카이라인을 탐색하기 위한 방법들이 주로 제안되어왔으며, 이를 평가하기 위해서는 실행 효율성을 바탕으로 평가를 수행해야할 필요가 있다. 특히나 스카이라인 질의의 특성상, 주어진 시간 이내에 질의 수행 결과를 사용자에게 전달해야 하기 때문에 스카이라인 질의 기법의 평가에 있어 시간 효율성은 매우 중요한 평가 요소라 할 수 있으며 이를 위해 소요되는 자원 활용성 또한 중요한 평가 요소라 하겠다.

2.2 공간 분할 기반의 스카이라인

공간 분할 기반의 방법들은 병렬 처리 또는 분산 처리 환경을 활용하여 스카이라인을 빠르게 탐색하기 위해 제안되었다. 특히나 병렬 처리 또는 분산 처리의 최종 성능은 데이터를 어떻게 분할하고 병합하는가에 따라 다양하게 변화하기 때문에 스카이라인에 적합한 공간 분할 기법의 선택은 매우 중요하다 할 수 있다. 이러한 공간 분할 기법의 적용 사례 중 가장 기본적은 공간 분할 기반의 스카이라인 질의 기법은 격자 분할에 기초한 방법이다. 격자 분할 기법은 데이터 분할 기법들 중에서 가장 전통적이면서도 다양한 분야에서 널리 사용되고 있는 분할 기법들 중 하나로써 주어진 데이터 공간을 격자 단위로 분할한다[5]. 이후 각각의 격자 분할 영역으로부터 해당 영역의 스카이라인을 탐색하게 되는데, 이를 로컬 스카이라인(local skyline)이라 부른다. 이러한 로컬 스카이라인 탐색이 모든 격자 영역에서 종료된 후에는 각각의 로컬 스카이라인들을 모두 비교하여 최종적인 스카이라인인 글로벌 스카이라인(global skyline)을 탐색한다. 하지만 격자 분할의 경우 글로벌 스카이라인이 발생할 수 없는 격자 영역에서도 1개 이상의 로컬 스카이라인을 탐색하고, 이들은 글로벌 스카이라인을 탐색하는 과정에서 불필요한 비교를 유발하는 문제가 있었다. 이러한 문제를 해결하기 위해 일부 방법은 일정한 간격을 기준으로 데이터를 분할하지 않고 특정한 데이터들을 기준으로 분할하는 방식을 선택하기도 하였다. 이 경우, 분할의 기준이 되는 데이터를 통해 조기에 많은 데이터들을 제거할 수 있기 때문에 불필요한 분할 영역의 발생을 최소화하여 효율적인 스카이라인의 탐색이 가능해지도록 하는 장점이 있다. 하지만 이 경우에는 분할의 기준이 되는 최적의 데이터를 선택하기 위해 우선적으로 해당 데이터를 탐색해야 하며 대상이 되는 데이터들이 단일 서버에 저장된 경우에는 기준 데이터 탐색을 분할된 자원들을 활용하여 처리할 수 없기 때문에 일정한 값을 기준으로 분할을 실시하는 경우보다 분할 작업에 요구되는 전처리 작업의 부하가 특정 자원에 크게 집중되어 경우에 따라 보통의 분할 기법 보다 느린 성능을 보였다.

이러한 문제를 해결하기 위해 Angle-based Space Partitioning (ABSP)[6]는 격자 분할 방식이 아닌 내각 기반의 새로운 방식을 새롭게 제안하여 많은 주목받았다. ABSP는 스카이라인의 위상적인 특성을 반영할 수

있는 내각 기반의 공간 분할을 통해 불필요한 분할 영역이 생성되지 않도록 하였으며 이로 인해 각 분할 영역에서 생성된 로컬 스카이라인은 최종적인 스카이라인 탐색 결과인 글로벌 스카이라인이 될 확률이 크게 증가되었다. 하지만 ABSP는 차원과 분할 영역의 수가 증가할수록 분할 공간 사이의 경계에서 글로벌 스카이라인이 될 수 없는 로컬 스카이라인의 발생이 크게 증가하는 문제가 있었다.

이와 같은 공간 분할 기반의 스카이라인 질의를 평가함에 있어 고려되어야 할 품질 요소는 병렬 처리에 사용된 프로세서의 수나 분산 환경에 사용된 시스템 수 대비 스카이라인 질의 처리의 시간 효율성을 기본적으로 평가해야 하며 공간 분할 기반 기법들 사이의 성능 비교를 위해서는 분할 기법의 기능 적합성을 파악하기 위해 각각의 분할 영역으로부터 생성되는 로컬 스카이라인의 기능 정확성을 판단해야할 필요가 있다.

2.3 인덱스 구조 기반의 스카이라인

스카이라인을 탐색함에 있어 가장 큰 문제는 스카이라인을 탐색하기 위해 모든 데이터에 대해 지배 연산을 진행해야 한다는 점이었다. 이러한 문제를 해결하기 위해 제안된 방법이 바로 Branch and Bound Skyline (BBS)[7]나 Z-SKY[8]와 같은 인덱스 구조 기반의 기법들이다.

BBS는 R-tree를 통해 주어진 데이터들의 인덱스를 생성한 후, 이후 스카이라인의 탐색이 요청되었을 때 R-tree의 인덱스 구성 요소인 Minimum Bounding Rectangle(MBR)의 공간적 특징을 이용하여 다른 MBR이나 스카이라인 후보에 의해 지배되는 MBR을 찾아 조기에 제거하는 방식으로 스카이라인이 될 수 없는 데이터들에 대한 접근을 최소화 하였다. 하지만 R-tree의 경우 차원이 증가할수록 MBR이 겹치게 될 확률이 증가하게 되고, 이는 지배 연산에서 지배가 발생할 확률을 크게 감소시기 때문에 스카이라인의 탐색의 효율성이 크게 감소하며 동시에 인덱스를 생성 및 유지하는데 매우 많은 연산을 요구하는 문제를 유발하였다.

Z-SKY는 BBS의 문제점을 해결하기 위해 B^+-tree와 데이터를 단일 차원으로 사상시키기 위해 제안된 Z-order curve를 조합한 ZBtree를 통해 스카이라인을 탐색하였다. Z-order curve는 주어진 다차원의 데이터들에 대해 Z-address라 불리는 주소를 생성한 후, 이에 기초하여 1차원의 공간으로 데이터들을 사상하는데, 이때 Z-address는 단조 순서의 특성과 함께 주어진 데이터들의 위치적인 정보를 역으로 추출할 수 있는 특징을 지니고 있었기 때문에 Z-SKY는 다른 방식들과 달리 실제 데이터에 대한 어떠한 접근 없이도 스카이라인을 탐색할 수 있는 특징을 지닌다. 또한 Z-SKY는 보다 효율적인 스카이라인의 탐색을 위해 B^+-tree와 Z-order curve의 단편을 이용한 클러스터인 RZ-region을 통해 조기에 스카이라인이 될 수 없는 데이터들이 속한 공간을 제거하는 것을 가능토록 하였다. 하지만 Z-SKY의 경우, Z-address에 구성에 있어 실수형 데이터와 같이 세밀한 간격을 갖는 데이터들을 사용하는 경우 Z-address의 길이가 매우 길어지는 문제가 있으며 데이터의 지배 유무를 확인함에 있어 생성된 스카이라인의 ZBtree를 큐를 통해 너비 우선 탐색하여 확인하기 때문에 노드의 적재량이 큰 경우, 매우 많은 수의 노드가 큐에 입력되고 이들 모두에 대해 지배 연산을 수행해야하기 때문에 탐색 성능의 감소하는 문제가 발생했다. 또한 RZ-region 사이에 명백한 지배가 발생하지 않는 경우, 해당 RZ-region에 속한 작은 크기의 RZ-region이나 데이터들을 다시금 큐에 저장한 후 이들과의 비교를 수행해야만 하기 때문에 데이터의 분포 및 생성된 RZ-region의 형태에 따라 성능의 차이가 매우 크게 발생하는 문제가 발생하였다.

이러한 인덱스 구조 기반의 스카이라인 질의를 평가함에 있어 고려되어야 할 품질 요소는 스카이라인 질의 처리와는 별도로 진행되는 인덱스 구축과 유지보수에 소요되는 시간 효율성을 기본적으로 평가해야 한다. 또한 스카이라인 질의 처리에서의 해당 인덱스 구조의 기능 적합성을 파악하기 위해서 기능 타당성을 판단해야할 필요가 있다.

2.4 근사 기법 기반의 스카이라인

정확한 스카이라인을 탐색하기 위해 스카이라인 질의를 수행하는 경우 발생하는 가장 큰 문제점은 모든 데이터가 비교 연산에 활용되어야 하기 때문에 스카이라인 질의의 비용이 매우 비싸다는 점이다. 이러한 문제를 해결하기 위해 몇몇 연구들은 보다 적은 비용으로도 실제 스카이라인의 결과와 매우 근사한 스카이라인을 탐색하기 위한 방법들을 제안하였다.

이를 해결하기 위해 BBS는 근사 기법 기반의 스카이라인 질의를 수행하기도 한다. 해당 기법의 특징은 R-tree를 탐색하는 과정에서 각각의 MBR로부터 일정한 공식에 따라 유도된 λ를 이용, 근사 스카이라인을 생성하고 이에 의해 지배되는 MBR 또는 데이터들을 조기에 제거하고, 지배되지 않은 실제 데이터들을 별도의 스카이라인으로써 저장하는 방식으로 실제 데이터를 이용하지 않더라도 근사 스카이라인과의 비교를 통해 빠른 스카이라인의 탐색을 가능케 하였다. 이러한 특징으로 인해 해당 기법을 통해 얻어진 스카이라인은 실제 스카이라인의 부분 집합으로 구성되는 특징이 있으며, 그 수는 항상 실제 스카이라인의 수보다 적은 특징을 지녔다.

이외에도 보다 빠르게 스카이라인을 탐색할 수 있도록 초평면 기반의 데이터 분할 기법과 병렬 처리를 조합한 방식의 스카이라인 질의 기법인 Plane Project Paralle - Skyline(PPPS)[9]도 제안되었다. PPPS의 경우, 주어진 데이터들을 대각의 초평면으로 1차 사상한

후 특정 차원과 수평을 이루는 초평면으로 2차 사상을 수행한다. 이후 일정한 기준에 따라 이들 데이터를 분할하고 이들을 병렬 처리하여 각각의 분할 영역으로부터 스카이라인을 탐색하고 이후 이들을 모두 병합하여 근사 스카이라인의 질의 결과로써 반환한다. 이러한 특징으로 인해 PPPS의 경우 단일 머신 환경에서 수행되는 BBS나 SFS에 비해 매우 빠른 속도를 보일 수 있었지만, 반대로 서로 다른 분할 영역 사이에 속한 데이터들 사이에서 발생할 수 있는 지배에 대한 확인을 거치지 않았기 때문에 실제 스카이라인이 될 수 없는 다수의 데이터들이 근사 스카이라인의 결과로 추가 선택되는 문제가 있었다.

따라서 근사 기법 기반의 스카이라인 질의를 평가하기 위해서는 근사 기법의 기능 적합성, 특히 근사 기법을 통해 생성된 스카이라인의 기능 정확성을 판단해야 할 필요가 있다.

2. 결론

본 논문에서는 스카이라인 질의 기법의 객관적인 성능 평가를 위한 품질 요소 선택 기준 새롭게 제안하기에 앞서 기존 스카이라인 관련의 연구들에 대한 조사 및 분류를 진행하였으며, 이로부터 기초적인 품질 평가 요소들에 대해 알아보았다. 차후 연구에서는 이러한 기초적 품질 평가 요소를 평가할 수 있는 방법들을 정의하고 이를 바탕으로 객관적 성능 평가를 수행할 수 있는 평가 모델을 새롭게 제안할 예정이다.

Acknowledgement

이 논문은 2017년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (NRF-2017R1D1A3B03035729).

참 고 문 헌

[1] Borzsony, S., Kossmann, D., & Stocker, K. (2001). The skyline operator. In *Proceedings of the International Conference on Data Engineering*, 421-430.

[2] Chomicki, J., Godfrey, P., Gryz, J., & Liang, D. (2005). Skyline with presorting: Theory and optimizations. In *Proceedings of the Intelligent Information Processing and Web Mining,* 595-604.

[3] Godfrey, P., Shipley, R., & Gryz, J. (2005). Maximal vector computation in large data sets. In *Proceedings of the International conference on Very large data bases,* 229-240.

[4] Bartolini, I., Ciaccia, P., & Patella, M. (2006). SaLSa: computing the skyline without scanning the whole sky. In *Proceedings of the ACM international conference on Information and knowledge management,* 405-414.

[5] Rocha-Junior, J. B., Vlachou, A., Doulkeridis, C., & Nørvåg, K. (2009). AGiDS: A grid-based strategy for distributed skyline query processing. In *Proceedings of the International Conference on Data Management in Grid and P2P Systems,* 12-23.

[6] Vlachou, A., Doulkeridis, C., & Kotidis, Y. (2008). Angle-based space partitioning for efficient parallel skyline computation. In *Proceedings of ACM SIGMOD International conference on Management of data,* 227-238.

[7] Papadias, D., Tao, Y., Fu, G., & Seeger, B. (2003). An optimal and progressive algorithm for skyline queries. In *Proceedings of the ACM SIGMOD International conference on Management of data,* 467-478.

[8] Lee, K. C., Lee, W. C., Zheng, B., Li, H., & Tian, Y. (2010). Z-SKY: an efficient skyline query processing framework based on Z-order. *The VLDB Journal, 19*(3), 333-362.

[9] PKöhler, H., Yang, J., & Zhou, X. (2011). Efficient parallel skyline processing using hyperplane projections. In *Proceedings of the ACM SIGMOD International Conference on Management of data,* 85-96.