직선 위에 중심을 갖는 *k*-센터를 계산하는 알고리즘

신찬수O 나현숙
한국외국어대학교 디지털정보공학, 숭실대학교 컴퓨터학부
cssin@hufs.ac.kr, hsnaa@ssu.ac.kr

Algorithms for computing *k*-centers on the line

Chan-Su ShinO Hyeon-Suk Na
Hankuk U. of Foreign Studies, Soongsil U.

기관위치설정 문제(Facility Location Problem)는 계산기하학 뿐만 아니라 Operational Research 분야에서도 매우 중요한 문제로 다루어지고 있다. 가장 대표적인 문제가 k-센터 문제이다. 이차원 평면에 n개의 점이 주어질 때, 이 점을 포함하는 k개의 디스크를 구하는 데, 가장 큰 디스크의 반지름이 최소가 되도록 하는 문제이다.

현재까지 k-센터 문제에 대한 많은 연구가 이루어졌다. 만약 $k=1$이라면, n개의 점을 포함하는 가장 작은 디스크를 찾는 문제가 된다. 이 문제는 n개의 점에 대한 FVD(Farthest Voronoi Diagram)을 구하여 다이어그램의 정점이나 에지위에 중심이 놓이는 경우를 따져서 $O(n \log n)$시간에 최소 디스크를 계산할 수 있다. 그러나 보다 효율적인 prune-and-search 기법을 사용하면 선형시간인 $O(n)$시간에 계산할 수 있다.[1] 그리고 $k=2$인 경우는 두 개의 디스크로 모든 점을 포함하는 것이고, 두 디스크의 반지름 중 큰 값을 최소화하는 것이다. 이 문제에 대한 다양한 알고리즘이 발표되었다. $O(n^2 \log^c n)$ 시간 알고리즘이 초기에 발표되었으나, 1996년에 $O(n \log^9 n)$ 시간에 해결할 수 있음이 알려졌다.[2] 이 알고리즘을 수정 보완한 T. M. Chan[3]에 의한 알고리즘은 $O(n \log^2 n \log^2 \log n)$ 시간에 동작하는 현재까지 알려진 가장 빠른 알고리즘이다. 그러나 $k \geq 3$인 경우에는 효율적인 알고리즘이 알려져 있지 않다.

이 외에도 다양하게 변형된 k-센터 문제가 연구되었다. 센터가 입력 점으로 제한된 이산 k-센터 문제, 점 이외의 선분이나 원과 같은 다른 기하 객체를 포함하는 센터 문제, 고차원 센터 문제, 다른 거리 기준에서의 센터 문제, 군집화(clustering) 문제 등 매우 다양하다.

본 논문에서는 디스크의 중심이 한 직선위에 놓일 때의 k-센터 문제를 다룬다. 이 문제를 <u>직선 k-센터 문제</u>라 부른다. 즉 디스크 중심이 모두 주어진 직선 위에 놓이고 디스크의 합집합이 입력 점을 모두 포함하는 k개 디스크의 중심을 결정하는 데 가장 큰 디스크의 반지름이 최소가 되도록 하는 문제이다. 대부분의 경우 직선의 방향을 하나로 고정한다. 본 논문에서는 수평선으로 제한한다. 이 직선(수평선) k-센터 문제는 다시 수평선이 고정된 경우와 움직이는 경우로 나뉜다. 움직이는 경우의 문제는 최대 디스크의 최소 반지름뿐 만 아니라 디스크의 중심이 놓이는 수평선을 구해야 한다. 본 논문에서는 두 가지 모두 다룬다.

이 문제는 무선통신을 위한 기지국 설치나 센서 네트워크 시스템에서 서버를 설치하는 응용분야에 활용될 수 있다. 예를 들어, 센서 네트워크 시스템에서는 작은 센서를 특정한 장소에 임의로 배치한 후, 센서가 수집한 다양한 종류의 데이터를 무선통신을 통해 서버에게 전달하여 후처리 과정을 수행한다. 센서는 자체 내장 전지에 의해 작동되지만 서버는 외부 전원으로 동작한다. 따라서 k개 서버의 설치 장소는 외부 전원을 쉽게 사용할 수 있는 곳이어야 하며, 모든 센서가 k개의 서버 중에서 최소 하나 이상의 서버의 통신범위 안에 놓이는 곳이어야 한다. 외부 전원은 직선 형태의 전선으로 표현되므로, k개 서버의 최적 위치는 모두 직선위에 놓이며, k개의 서버 영역(디스크)의 합집합이 n개의 센서(점)를 포함해야 하고, 서버 영역의 최대 반지름이 최소가 되도록 하는 것이 경제적으로 바람직하다. 이 문제는 바로 직선 k-센터 문제와 같다.

이전 연구 결과 중에는 k개 디스크의 반지름의 합을 최소로 하는 문제에 대한 결과가 있다[4]. 고정된

수평선의 경우엔 동적계획법(dynamic programming) 기법을 적용하였으며 $O(n^2 \log n)$시간에 해결하였다. (이 문제에서는 k가 입력의 일부로 주어지지 않는데, 그 이유는 반지름의 합이 최소가 되는 최적 디스크 집합에서의 디스크 개수는 자동적으로 결정되기 때문이다.) 반면에 수평선이 움직이는 경우엔, 디스크의 반지름이 합이 최소가 되는 수평선을 찾아야 한다. 이 문제는 다항시간에 풀기는 매우 어렵다는 것이 [4]에서 증명되었으며, 다양한 근사 알고리즘(approximation algorithm)이 제시되었다. 이 결과를 제외한 다른 관련 결과는 알려져 있지 않다.

본 논문에서는 수평선 k-센터 문제를 $O(\min\{n(\log n)^{\lceil \log k \rceil}, n^2 \log n\})$-시간에 해결하는 알고리즘을 제시한다. (여기서 $k \geq 2$이다. 만약 $k=1$이라면 $O(n \log n)$ 시간에 계산할 수 있다.) 또한 움직이는 직선에 대한 k-센터 문제는 $O(n^{\min\{4,k\}} \log n)$, ($k \geq 2$) 시간에 최적 수평선과 반지름을 찾는 알고리즘을 제시한다.

고정된 수평선에 중심을 갖는 k-센터

알고리즘의 개요만 간략히 설명한다. 입력으로 주어지는 점들의 집합을 $S=\{p_1, p_2, \cdots, p_n\}$라 하고, 고정된 수평선은 x-축이라 가정한다. 점들은 x-좌표에 대해 오름차순으로 정렬되어 있다고 가정한다. 임의의 점 집합 P에 대한 k-센터 문제의 최적 해(가장 큰 디스크의 최소 반지름)를 $R_k(P)$라 하자. S의 첫 i개의 점들로 구성된 S의 부분집합을 $S(i)$라 하자. $k_1+k_2=k$를 만족하는 임의의 두 자연수 k_1, k_2에 대해, $S_1 := S(i)$로 하고 $S_2 := S - S(i)$로 정의하면, $R_k(S) = \min_i \max\{R_{k_1}(S(i)), R_{k_2}(S-S(i))\}$을 만족하는 $i=i^\star$이 존재하며, 이를 이진탐색으로 찾을 수 있음을 보였다. 이 방법에 의해, 고정된 수평선에 중심을 갖는 k-센터 문제를 $O(\min\{n(\log n)^{\lceil \log k \rceil}, n^2 \log n\})$ 시간에 해결할 수 있다.

움직이는 수평선에 중심을 갖는 k-센터

정확한 문제 정의는, S를 포함하면서 디스크 중심이 특정 수평선 ℓ위에 있는 k개의 디스크 집합 $D_1, D_2, \cdots, D_k$을 찾는 것이다. 즉, 모든 수평선 ℓ 중에서 $\max_i\{r_i(D_i)\}$이 최소가 되는 최적 수평선 $\ell^\star$를 찾고 $\ell^\star$위에 중심을 갖는 디스크의 최소 반지름 $R_k(S)$을 찾는 것이 목표다. 고정된 수평선 $\ell(y)$에 대한 최적 디스크 집합을 $D_1^y, D_2^y, \cdots, D_k^y$라 하자. 이 중 가장 큰 디스크의 중심을 $c(y)=(x(y), y)$라 하고, 반지름을 $R_k^y(S)$로 하자. 그러면 $R_k(S) := \min_y R_k^y(S)$로 정의된다. $k=1$인 경우에는, S에 대한 Farthest Voronoi Diagram(FVD)의 기하학적 성질을 이용하여, $R_1^y(S)$는 $O(n)$개의 상수 차수의 곡선으로 이루어졌으며 strict unimodal 함수임을 보였다. 이 성질을 이용하여, $k=2$인 경우와 $k=3$인 경우의 최적 수평선과 최적 디스크 집합을 각각 $O(n^2 \log n)$ 시간과 $O(n^3 \log n)$에 계산할 수 있음을 보였다. $k \geq 4$인 경우는 최적 디스크 집합의 최대 크기의 디스크를 결정하는 경우의 수를 모두 고려해서 $O(n^4 \log n)$ 시간에 해를 구할 수 있음을 보였다. 이를 종합하면, 임의의 $k \geq 2$와 점 집합 S가 주어질 때, 최적의 수평선 $\ell(y^\star)$와 최적 반지름 $R_k^{y^\star}(S)$는 $O(n^{\min\{4,k\}} \log n)$시간에 계산 가능하다.

참고문헌

[1] N. Meggido, Applying parallel computation algorithms in the design of serial algorithms, *Journal of ACM*, 30, pp. 852--865, 1983.

[2] M. Sharir, A Near-Linear Algorithm for the Planar 2-Center Problem, Proc., 12th ACM *Symp. on Computational Geometry*, pp. 106--112, 1996.

[3] T. M. Chan, More planar two-center algorithms, *Computational Geometry: Theory and Applications*, 13, pp. 189--198, 1999.

[4] H. Alt, E. M. Arkin, H. Bronnimann, J. Erickson, S. P. Fekete, C. Knauer, J. Lenchner, J. S. B. Mitchell, K. Whittlesey, Minimum-Cost Coverage of Point Sets by Disks, 22th ACM *Symp. on Computational Geometry*, pp. 449--458, 2006.