# Breaking the Sorting Barrier for Directed Single-Source Shortest Paths

https://arxiv.org/abs/2504.17033

다익스트라가 깨졌다! SSSP(Single-Source Shortest Path)를 최단으로 구하는 새로운 방법의 발견

- 개요
  - 다익스트라(Dijkstra) 알고리즘은 그래프 내에서 두 정점 간의 최단거리를 구하는 알고리즘으로서 현재 두 대상 간의 이동에서 주된 구현 방법으로 사용됨
    - 다익스트라 알고리즘: 그래프가 가중치를 가지는 간선으로 이루어져 있을 때 한 정점에서 다른 정점까지의 최단 경로를 탐색하는 알고리즘
  - 다익스트라 알고리즘의 시간복잡도는 $O(V^2)=O(n^2)$, 우선순위 큐 사용 시 $O(V+ElogV)=O(nlogn)$이며 이는 현재까지 최단 시간으로 알려져 있음
    - V = 정점 수, E = 간선 수
  - 허나 이 논문은 벨만-포드(Bellman-Ford) 알고리즘을 기반으로 하여 시간복잡도를 $O(n \log^{(2/3)} n)$까지 줄일 수 있는 새로운 알고리즘을 제시함
    - 벨만-포드 알고리즘: 그래프가 가중치를 가지는 방향 있는 간선으로 이루어져 있을 때 한 정점에서 다른 정점까지의 최단 경로를 탐색하는 알고리즘
    - $\log^{(2/3)} n$: $\log n$의 값을 2/3 제곱한다는 의미
- 구조
  - 예비 지식
    - 상수→차수 변환: 각 정점을 0가중 사이클(간선들의 가중치 합이 0인 사이클, 경로가 유일하지 않게 만들 수 있음)로 분해하고 간선을 다시 배치하여 진입과 진출 차수가 2 이하인 그래프로 바꾸는 작업이 필요(최단 거리는 보존됨)

    - 비교 덧셈 모델: 가중치에 대한 연산은 비교/덧셈만 수행 = O(1)의 시간복잡도
    - 완료: 각 정점 v에 현재 추정거리 d^(v)를 유지하고 d^(v) = d(v)면 complete, 아니면 incomplete 처리(complete는 정점간의 거리가 확정됨을 의미, 전체 탐색 성공 X)
    - 경로에 대한 가정: 길이가 동일한 경우를 피하기 위해 하나를 일관되게 뽑기 위한 규칙을 설정(각 경로를 {길이 w, 정점 수 k, 정점 배열 v[]}의 튜플로 보고 순서대로 비교)
  - 우선 문제를 축소(Bounded Multi-Source SSP)
    - 현재 상태: 경계값 B와 B 미만의 최단경로가 반드시 완료 정점을 지난다는 불변식 아래의 피벗 후보 집합 F
    - 각 정점에 대한 거리인 d(v) < B인 모든 v의 참거리를 확정하는 것이 목표
    - 보조정리: 서브루틴 BMSSP는 경계를 앞으로 밀거나 작업량이 큰 경우 작업을 멈추고 불변식을 유지한 새 경계를 반환 → 최상위 호출에서는 전체 거리를 확정
  - 피벗 탐색
    - 벨만-포드 기법으로 이미 완성될 정점을 정리
    - 불완전한 정점의 최단경로 트리에 대해 크기가 일정 이상인 경로들만 피벗으로 남김
    - 이후 이 피벗들만 재귀 호출하여 피벗 후보 풀을 줄이기
  - 부분 정렬 자료구조
    - 완전히 정렬시키는 대신 필요한 부분만 꺼내는 방식으로 시간복잡도 줄이기
    - insert → 현재 구조보다 작은 값들을 배치로 앞에 추가 → 가장 작은 값 상위 k개와 나머지와의 경계값만 pull
  - 피벗 후보가 하나의 정점으로 수렴되면 이진 힙  다익스트라를 경계값까지만 실행하여 필요한 거리만 확정하고 다음 경계를 반환
  - 자료구조 초기화 → 피벗 탐색으로 후보 축소/분할 → 선택된 소수의 피벗만 재귀 호출 → 각 호출이 돌려준 경계(집합)으로 상태 갱신 → 성공 또는 부분 실행 종료 여부를 판단
  - 이러한 재귀 트리의 깊이는 로그의 거듭제곱 수준으로 제한되고 총합이 O(m log^(2/3) n)으로 수렴
- 장점

    - 기존 다익스트라 알고리즘의 한계를 깨고 시간복잡도를 단축시킴
    - 무작위 및 방향이 없는 결과가 아닌 방향이 있는 그래프에서 결정론적인 성과 달성
    - 완전한 정렬을 요구하지 않고 거리값만 계산하면 됨
- 단점
    - 거리 순서가 필요하면 여전히 다익스트라가 최적
    - 간선의 가중치는 항상 0 이상이어야만 함: 음수인 경우는 다루지 않음
    - 모든 정점이 시작점에서 도달 가능하다는 전제가 필요
- 결론
    - 알고리즘의 개선을 다룬 주제와는 별개로 현재 이해도를 벗어나는 부분이 많기 때문에 다시 읽어봐야 함
    - 부분적인 정렬을 사용함을 강조
    - 거리값만 필요할 때는 더 빠른 방법을 찾아내었다는 의의가 존재