

최적 경로 탐색을 위한 계층 경로 알고리즘의 탐색 영역 결정 기법의 구현

이현섭 · 윤상두 · 김진덕

동의대학교

An Implementation of Method to Determine Search Space of Hierarchical Path Algorithm for Finding Optimal Path

Hyoun-Sup Lee · Sang-Du Yun · Jin-Deog Kim

Dong Eui University

E-mail : lhskmj@naver.com

요 약

계층 경로 탐색 알고리즘의 탐색 기법에 대한 연구는 많은 논문에서 제시되어 왔다. 그러나 효과적인 영역의 분할에 대한 연구는 많지 않다. 본 논문에서는 계층 경로 탐색 알고리즘의 주 영역과 부 영역을 효과적으로 나눌 수 있는 다음과 같은 기법을 제안하고 평가한다: (1) 보로노이 알고리즘을 이용하여 하나의 주 노드를 중심으로 영역을 분할하는 방법, (2) 고정 그리드를 기반으로 주 영역과 부 영역을 설정하는 방법. 제안한 기법의 성능평가를 위해 전통적인 Dijkstra 알고리즘을 주 노드에서 탐색하여 가장 인접한 주 노드 4개를 통해 주 영역을 지정하는 방법과 비교하였다. 실험 평가 항목은 탐색된 경로의 정확도와 연산시간이며, 실험 결과 연산 속도 측면에서는 보로노이 기반 기법이 좋은 성능을 보였으며, 그리드 기반 기법이 정확도 측면에서 가장 좋은 성능을 보였다.

ABSTRACT

Many researches on hierarchical path search have been studied so far. Even though partitioning regions is essential part, the researches are not enough. This paper proposes two efficient methods to partition regions: 1) a method based on voronoi algorithm in which a major node is central point of a region, 2) a method based on fixed grid that partitions regions into major and minor. The performances of the proposed methods are compared with the conventional hierarchical path search method in which a region is formed by the boundary line of nearest 4 points of a major node in terms of the path search time and the accuracy. The results obtained from the experiments show that the method based on voronoi achieves short execution time and the method based grid achieves high accuracy.

키워드

최적경로, 계층 경로 탐색, voronoi, grid분할

1. 서 론

기존의 경로안내 시스템의 이슈는 정확한 경로를 탐색하는데 그 목적이 있었다면 현재의 경로안내 시스템은 앞서 언급한 목적은 기본으로 제공하며 client 시스템의 한정적인 자원에서 최대한 빠르게 경로를 탐색해 주는 것이 새로운 목적으로 이슈화 되고 있다. 이 두 가지 사항을 고려한 알고리즘들은 많은 연구 및 시스템에서 제시되어 지고 있으며 대표적인 경로 탐색 알고리즘으로 Dijkstra, A*, 계층 경로 탐색, 그리드 기반 경로 탐색 알고리즘 등이 있다. 이 중 연산 성능 향상을 위한 계층 경로 탐색 알고리즘은 도로 네트워크를 주 영역과 부 영역의 두 계층으로 분할하여 최적경로를 탐색하는 알고리

즘이다.

계층 경로 탐색 알고리즘을 통한 경로 탐색 기법에 대하여 여러 논문에서 제시되어 지고 있으나 주/부 영역을 나누는 기준이나 방법에 대해서는 구체적으로 제시되지 않았다. 효과적인 영역의 지정은 연산비용을 줄일 수 있으며, 탐색 결과의 정확도에서도 좋은 결과를 볼 수 있으므로 영역을 분할하는 기법은 계층 경로 탐색 알고리즘에서 중요한 기술이다.

영역 분할이란 도로 네트워크를 2계층으로 나누는 것을 의미 하는데 주 영역과 부 영역으로 구성된다. 주 영역의 경우 도로 네트워크에서 고속도로 이 동할 수 있는 노드의 집합이며 이러한 집합으로 형성되는 영역을 부 영역이라고 한다.

본 논문에서는 이러한 계층 경로 탐색을 위한 영역 분할 기법을 Dijkstra 알고리즘을 통한 근접 주 노드 탐색 분할과 보로노이 알고리즘을 사용한 분할, 고정 그리드 기반 영역 분할의 세 가지로 구현하였다. 또한 각각의 구현에 대한 결과를 비교하여 어떤 특징을 가지고 있는지를 분석하였다.

본 논문의 결과는 계층 경로 탐색 알고리즘을 이용한 경로 탐색 시스템을 구축할 경우 효과적인 영역 분할을 할 수 있는 기법을 제시한다.

본 논문은 II장에서 기존의 계층 경로 탐색 알고리즘의 특징과 한계에 대하여 언급한다. 그리고 III장에서는 여러 가지 영역 설정 기법의 특징들과 구현 내용을 살펴보고 IV장에서는 구현된 기법들에 대한 결과를 분석한다. 끝으로 V장 결론에서 정리한다.

II. 관련 연구

계층 경로 탐색 알고리즘은 연산 비용과 최적 경로 접근 정확도를 높이기 위하여 영역을 나누어 탐색하는 알고리즘으로 여러 방법론이 제시되어 왔다.

Karimi[1]는 출발지 및 목적지를 포위하는 영역을 만들어 검색 노드의 수를 제한하는 휴리스틱 탐색 방법을 제안하였다. 영역은 여러 개의 창에 의하여 나누어지며 그 창의 모양이나 크기는 네트워크의 크기에 따라 결정한다.

G. R. Jagadeesh[2]가 제시한 탐색 방법은 주 노드 4개를 이용해 영역을 만들어 그 영역 내부의 노드는 항상 주 노드 4개를 통과하여 경로를 탐색하는 알고리즘을 제안하였다. 전체 탐색 과정을 보면 선택 되어진 출발지 주 영역 4개의 노드와 목적지 주 영역 4개의 노드를 $\min(n*m)$ 의 형태로 경로 비용을 계산한다. 여기서 n은 출발지 주 영역에서 선택된 4개의 노드이며 m은 목적지 주 영역에서 선택되어진 4개의 노드이다. 즉 16개의 이동 경로를 탐색하여 최소 비용을 가지는 노드를 최종 결과로 가진다.

이현섭[4]의 계층 경로 탐색 알고리즘의 탐색 영역 결정 기법에서는 다중 노드를 이용한 영역 설정 및 단일 노드를 이용한 영역 설정에 대하여 제안하고 있다. 다중 노드는 전통적인 계층 경로 탐색 영역 설정과 고정 그리드를 이용한 영역 설정에 대해서 서술하였고 단일 노드의 경우 보로노이 알고리즘을 이용한 영역 결정 기법을 제안하였으며 본 논문에서는 이를 토대로 각각의 알고리즘을 구현하였다.

기존의 계층 경로 탐색 알고리즘은 그리드를 통하여 주 영역을 설정하는 부분에 대한 언급은 크게 고려를 하지 않고 방법론을 제시하는데 까지 제안하였다. 그러나 효과적인 영역의 분할은 효과적으로 탐색 시간을 감소시키면서 정확도 또한 최적의 경우와 비교하여(Dijkstra 경로 탐색) 손색없는 결과를 가져올 수 있다. 따라서

이러한 계층 경로 탐색의 영역 분할은 중요한 요소로 고려할 수 있다.

III. 영역 설정 기법

이 장에서는 관련 연구의 다중 노드와 단일 노드를 통한 영역 결정 기법에 대하여 제시한 내용에 대한 구현 방법을 자세히 설명한다.

3.1 인접 노드 탐색 영역 분할

인접 노드 탐색 영역 분할 기법은 대표적인 주 노드들을 선택하여 선택된 노드를 중심으로 Dijkstra 알고리즘을 실행하여 인접한 주 노드 3개를 찾아 하나의 영역으로 만드는 분할 기법이다. G. R. Jagadeesh가 제안한 논문에서 주/부 영역은 자연적으로 만들어지는 영역으로 설명하고 있는데 실제로 자연적으로 4개의 노드를 구성할 수 있는 방법은 인위적으로 영역을 나눌 수밖에 없다. 그러나 많은 노드가 존재하는 도로 네트워크에서 이 방법은 효과적이지 못하므로 본 논문에서는 고속도로 톨게이트 같은 대표 노드를 중심으로 인접한 주 노드를 탐색하는 방법으로 전통적인 계층 경로 탐색 알고리즘을 구현하였다.

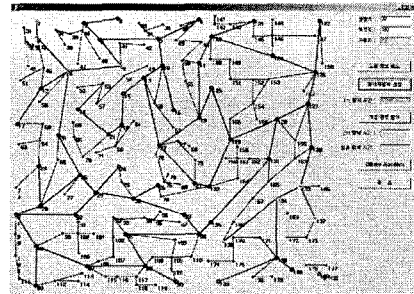


그림 1. 인접 노드 탐색 영역 분할

그림 1은 인접 노드 탐색 영역 분할의 구현 화면이다. 출발지와 목적지가 선택되면 각각의 주 노드 영역을 찾아내고 각 영역을 이루는 4개의 노드를 대상으로 $\min(n*m)$ 연산을 하여 경로를 탐색 한다. 실제 Dijkstra를 통한 인접 노드 탐색은 탐색 시점에 이루어지는 것이 아니라 미리 연산되어 Database에 저장되어 있다. 따라서 실제 탐색 시점에는 선택된 영역을 통한 경로 탐색만 이루어지므로 연산 속도는 효과적으로 줄어든다.

3.2 고정 그리드 기반 영역 분할

영역 설정 기법의 다른 방법으로 고정 그리드 기반 영역 분할 기법에 대하여 구현하였다. 그림 2는 탐색 영역을 4개의 고정 그리드로 분할하여 탐색 영역을 확보하는 내용이다[4].

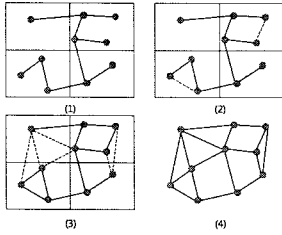


그림 2. 고정 그리드 기반 영역 분할

영역 탐색은 다음과 같은 순서로 이루어진다. 적용할 그리드 개수를 선택한 이후에 적용된 그리드 내에 포함되어 있는 주 영역 노드들을 연결한다. 다음으로 인접한 모든 그리드 내에 있는 노드와 기준이 되는 그리드 내의 노드를 직선으로 모두 연결한다. 노드의 수에 따라 직선의 수는 달라지는데 이 만들어지는 직선이 만약 먼저 이루어진 그리드 내의 영역 간선과 교차되면 직선을 삭제하는 순서의 작업을 진행한다. 나머지는 직선을 대상으로 만들어진 직선이 교차되는 부분이 생기면 이 직선도 삭제 한다.

다음의 표 1은 앞에서 언급한 그리드 영역 분할을 구현하면서 사용한 의사 코드이며 그림3은 구현된 결과 화면이다.

표 1. 그리드 기반의 영역 설정 알고리즘

```

Function : 그리드 기반 영역 설정
{
  For each Grid Cell
    if (그리드 내에 3개 이상의 노드가 존재)
    {
      존재하는 노드 모두 연결;
      교차 간선 삭제;
      남은 간선 = 영역의 경계
    }
  인접한 그리드의 모든 노드를 n*m 형태의 간선으로 연결
  if (선행 작업 시 만들어진 영역의 경계나 교차 간선 존재) 모두 삭제;
  else 남은 간선 = 영역의 경계
}
    
```

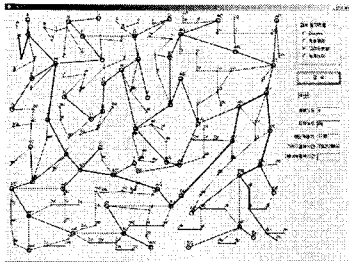


그림 3. 고정 그리드 기반 영역 분할

3.3 Voronoi 알고리즘 기반 영역 분할

마지막으로 구현한 영역 분할 기법은 Voronoi 알고리즘을 이용한 분할 방법이다.

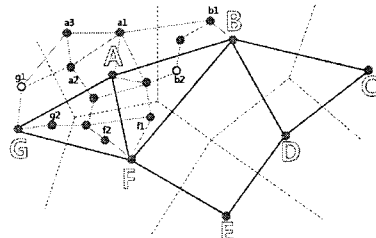


그림 4. Voronoi 알고리즘 기반 영역 분할

Voronoi 알고리즘은 그림 5와 같이 기하 및 단백질 세포 분열 및 여러 방면에서 사용되고 있는 알고리즘이다. 지정된 포인트에서 인접한 포인트와의 경계를 이룰 때 수직선을 그려서 영역을 분할하는 알고리즘으로 도로 네트워크의 주 노드를 대상으로 영역을 분할 할 때 이용하여 구현하였다.

그림 4는 Voronoi 알고리즘을 기반으로 한 도로 네트워크의 영역 분할을 나타내며 인접한 주 노드들 사이에 수직선(점선)을 그어 영역을 분할 하였다.

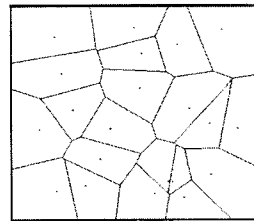


그림 5. 보로노이 알고리즘 구현

전체 도로 네트워크의 노드를 저장하고 있는 데이터베이스 테이블에서 주 영역 노드가 선택이 되면 이를 기준으로 하여 그림 4와 같이 부영역이 지정되면 표 2의 테이블이 데이터베이스에 생성된다.

표 2. 영역 분할 데이터베이스

주 영역 노드	영역 포함 노드	인접 영역
A	a1, a2, a3 ...	B, F, G
B	b1, b2 ...	A, C, D, F
F	f1, f2 ...	A, B, E, G
G	g1, g2 ...	A, F
:	:	:

테이블에서 인접 영역 필드는 현재 영역안의 도로 네트워크의 상황이나 목적지의 위치에 따라 반드시 자신의 주 영역 노드를 출발지 노드들이 선택을 하는 것이 최선의 경로를 보장 하

지 않을 경우 참조한다.

도로의 이동 속도는 가변적으로 변하기 때문에 현재 출발지에서 자신이 속한 주 영역 노드를 탐색하러 가는 구간의 가중치의 값이 클 경우 인접 영역으로 우회를 고려해야 한다.

IV. 구현 결과 분석

그림 6은 3장에서 언급한 세 가지 알고리즘의 영역 분할을 통해 경로를 탐색 했을 경우와 Dijkstra 알고리즘으로 영역을 탐색 했을 때의 정확도를 나타내는 그래프다. 그리드 기반 영역 분할의 경우 4개의 그리드와 16개의 그리드 두 가지를 가지고 테스트 하였다. 각각의 알고리즘은 C#으로 프로그래밍 하였다.

그림 6에서 보는 바와 같이 Dijkstra 알고리즘의 경우 최선의 경로를 탐색하는 것을 볼 수 있다. 4개의 그리드를 통한 영역 분할의 정확도가 높고 Voronoi 알고리즘 기반의 영역 분할의 정확도가 가장 낮다. 이런 결과가 나타나는 가장 큰 이유는 분할된 영역의 범위의 크기에 따라서 나타났다. 그리고 출발지 목적지의 위치 및 균등하지 못한 가중치에 따라서 결과차이가 나타났다.

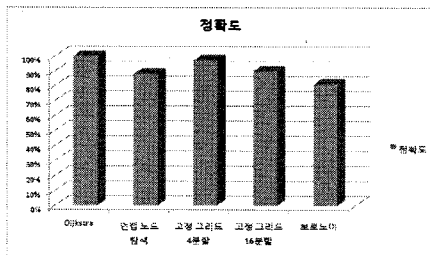


그림 6. 정확도 비교

그림 7은 세 가지 알고리즘을 통한 구현 결과 중에서 탐색 속도를 비교한 그래프이다.

탐색 속도의 경우 Voronoi 알고리즘이 가장 좋은 성능을 보였고 다음으로 인접 노드 탐색 영역 분할, 16개 그리드, 4개 그리드, Dijkstra 알고리즘 순으로 나타났다.

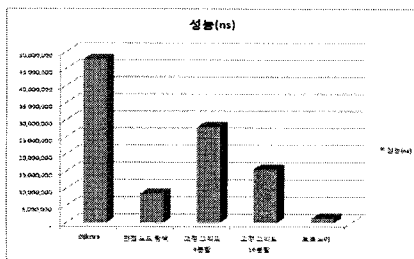


그림 7. 성능 비교

이런 결과의 원인을 분석하면 Voronoi 알고리

즘의 경우 어떠한 노드가 선택 되더라도 단 하나의 주 노드를 통과하여 경로를 탐색하므로 탐색 범위가 가장 짧은 경향을 보였으며 인접 노드 탐색 영역 분할과 16개의 그리드의 경우 거의 비슷한 성능을 보이는데 하나의 그리드 내에 있는 영역 범위와 인접 노드 탐색 영역으로 만들어진 범위가 비슷한 크기를 가짐으로써 나타나는 결과로 볼 수 있다.

V. 결론 및 향후 과제

최적경로를 탐색하기 위해서 고려할 사항 중에 가장 중요한 것이 정확도이지만 저 자원의 모바일 디바이스를 고려하면 연산 비용 또한 고려하지 않으면 안 되는 대상이다. 따라서 이러한 연산 비용을 최소화 하면서 최적경로의 정확도를 유지할 수 있는 탐색 알고리즘이 필요하다.

본 논문에서는 지금까지 도로 네트워크에 효과적으로 적용할 수 있는 계층 경로 탐색 알고리즘에서 영역을 지정하는 3 가지 기법에 대하여 제안하고 구현하였다. 또한 각각의 알고리즘에 대하여 동일한 데이터를 통해 비교한 결과를 통해 효과적인 영역 분할에 대한 정보를 얻을 수 있었다. 향후 연구 과제로는 도로네트워크의 정보와 시간대에 따른 도로 상황을 적용하면서 효과적으로 탐색 영역을 줄이기 위해 앞서 본 결과를 통해 새로운 영역 설정 기법을 만들어 정확도는 더욱 개선하며 성능은 최대한 증가시킨 영역 분할 기법에 대하여 고려해 본다.

참고문헌

- [1] H.A.Karimi, "Real-time optimal route computation: a heuristic approach," ITS J., vol. 3, no. 2, pp. 111-127, 1996.
- [2] G. R. Jagadeesh, T. Srikanthan, and K. H. Quek, "Heuristic Techniques for Accelerating Hierarchical Routing on Road Networks," IEEE Trns. Intelligent Transportation Systems. Vol. 3, No 4, pp.301-309, 2002
- [3] Improving Worst - Case Optimal Delaunay Triangulation Algorithms - Geoff Leach Department of Computer Science Royal Melbourne Institute of Technology Melbourne, Australia. June 15, 1992
- [4] Hyoun Sup Lee, "Optimal path search based on fixed grid for telematics applications" 동의대학교 학위논문
- [5] Hyoun Sup Lee, Jin-Deog Kim "A Method to determine Search Space of Hierarchical Path Algorithm for Finding Optimal Path," 한국해양정보통신학회 2007 추계 종합 학술대회, Vol.11 No.2, pp565-569, 2007