

# 탐색 영역 추출을 위한 추상 그래프 탐색 알고리즘 설계

김지수, 이지완, 문대진, 조대수

동서대학교

## A Path-Finding Algorithm on an Abstract Graph for Extracting Estimated Search Space

Ji-Soo Kim, Ji-Wan Lee, Dae-Jin Moon, Dae-Soo Cho

Dongseo University

E-mail : kimjisu29@gmail.com, wldhks85@gmail.com, wizardyk@gmail.com, dscho@dongseo.ac.kr

### 요약

실제 전체 도로망을 하나의 그리드로 간주하며, 그리드는 여러 개의 고정된 셀로 나누어진다. 경로 탐색 기법은 2단계 탐색으로 나누어진다. 1차 탐색은 실제 도로 네트워크와 고정 셀로부터 생성된 가상 정점과 가상 간선으로 이루어진 추상 그래프에서 우선 탐색한다. 추상 그래프에서 탐색된 가상 경로를 포함하고 있는 영역을 유효 셀이라 하며, 2차 탐색은 추상 그래프에서 제공하는 유효 셀 내에서 A\* 알고리즘 탐색을 한다. 즉, 유효 셀을 제외한 무효 셀을 제거함으로써 탐색 비용을 줄일 가능성이 있다. 이 논문에서는 추상 그래프 생성 방법과 탐색 영역 추출을 위한 추상 그래프 경로탐색 알고리즘을 제안한다.

### ABSTRACT

The real road network is regarded as a grid, and the grid is divided by fixed-sized cells. The path-finding is composed of two step searching. First searching travels on the abstract graph which is composed of a set of psuedo vertexes and a set of psuedo edges that are created by real road network and fixed-sized cells. The result of the first searching is a psuedo path which is composed of a set of selected psuedo edges. The cells intersected with the psuedo path are called as valid cells. The second searching travels with A\* algorithm on valid cells. As pruning search space by removing the invalid cells, it would be possible to reduce the cost of exploring on real road network. In this paper, we present the method of creating the abstract graph and propose a path-finding algorithm on the abstract graph for extracting search space before traveling on real road network.

### 키워드

추상 그래프, 탐색 영역 추출, A\* 알고리즘

## 1. 서론

최근 텔레매틱스 서비스가 급증하면서 경로 탐색에 대한 관심이 높아지고 있다. 이전의 경로 탐색 시스템의 가장 큰 이슈는 임의의 출발지와 목적지에 대해 정확한 경로를 탐색하는 것에 중점을 두었으나, 단말기 보급이 급증함에 따라 한정된 자원에서 보다 빠르게 경로를 탐색하는 방법에 관심이 더 높아지고 있다. 단말기 기반의

경로 탐색의 경우, 상대적으로 저사양의 PDA 등의 단말기에서 자체적으로 경로 탐색을 수행해야 하므로 일반적으로 최적 경로탐색이 아닌, 준 최적 경로 탐색 알고리즘을 사용한다. 이 논문에서는 단말기 기반에서 준최적 경로를 탐색하는 알고리즘을 다룬다[1,2,3].

이 논문에서 제안하는 방법은 다음과 같다. 추상 그래프 탐색을 이용하여 탐색 영역을 추출하여 경로를 탐색한다. 실제 도로 네트워크를 추상

그래프로 표현하기 위해 전체 도로 네트워크를 하나의 그리드로 간주한다. 추상 그래프는 가상 정점과 가상 간선으로 이루어져 있으며, 가상 정점과 가상 간선을 생성하기 위해 그리드는 여러 개의 고정된 셀로 나누어진다.

실제 경로를 탐색하기 전 추상 그래프에서 우선 탐색함으로써 출발지와 목적지 사이의 탐색 경로 방향과 범위를 추출할 수 있다. 추상 그래프에서 탐색된 경로를 가상 경로라 하며, 가상 경로가 포함된 셀들을 유효 셀이라 한다. 실제 경로 탐색은 추상 그래프에서 제공되는 유효 셀 범위 내에서 경로 탐색한다. 실제 경로 탐색이 적용될 탐색 영역을 추출하여 불필요한 노드 접근 횟수를 줄임으로서 탐색 비용을 줄일 가능성이 있다.

이 논문의 구성은 다음과 같다. 2장에서는 기존의 길 찾기 알고리즘에 대해 알아보고, 3장에서는 추상 그래프 생성 방법 및 추상 그래프 경로 탐색 알고리즘을 제안하고, 마지막 4장에서는 결론 및 향후 과제에 대해 설명 하겠다.

## II. 관련 연구

A\* 알고리즘[2]은 가장 범용 적으로 사용되고 있으며, 특히 AI분야에서 많이 쓰이는 방법이다. A\* 알고리즘은  $g$ ,  $h$ ,  $f$  의 3가지 노드 평가 함수를 사용하며,  $g$ 는 현재위치에서 목적지까지의 실제비용,  $h$ 는 현재위치에서 목적지까지의 가상 비용,  $f$ 는  $g+h$ 이다. 경로 탐색 시  $f$ 값을 사용하여 수행한다.  $g$ 는 경로 탐색전 이미 결정되어 있는 값이지만,  $h$ 는 경로 탐색의 방법에 따라 조작되어지는 값이다. 따라서 탐색된 경로는  $h$ 값에 매우 의존적이며  $h$ 을 어떻게 설정하는가에 따라 경로의 길이나 탐색비용이 달라진다.

고정 그리드를 이용한 탐색 알고리즘은 가변 휴리스틱을 적용하기 위해 고정 그리드 기법을 사용한다. 그러나 단말기 기반의 시스템일 경우 모든 노드에 대하여 실시간으로 동적 휴리스틱을 적용하기에 어렵다. 따라서 실제 도로 네트워크를 일정한 크기의 그리드를 덮어 여러 개의 노드가 하나의 그리드 안에 들어가게 하는 기법이다. 하나의 그리드에 여러 개의 노드를 포함시켜 그리드별로 휴리스틱 가중치를 적용한다[4].

계층 경로 탐색 알고리즘은 도로를 작은길(마이너 로드)와 큰길(메이저 로드)로 구분하여 각각의 계층에서 경로를 탐색하는 방법이다. AI분야에서는 그리드 기반의 추상 그래프를 생성하여 이를 이용한 계층 구조 경로 탐색 알고리즘을 제안하였다[3,5,6].

## III. 추상 그래프 탐색

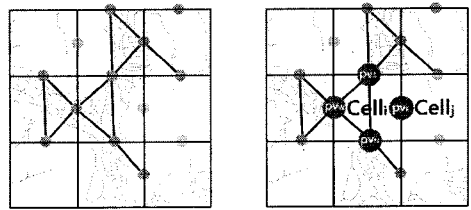
단말기 기반의 경로 탐색을 위한 많은 알고리즘이 제안되고 있으며, 특히 A\* 또는 변형A\* 알고리즘에 대해 연구 되었다[1]. 이러한 알고리즘은 휴리스틱에 매우 의존적이기 때문에 휴리스틱에 의해 결정된 추정 경로에 실제 경로가 존재하지 않을 경우, 불필요한 노드 접근으로 인한 탐색 비용이 커지는 경향이 있다. 추상 그래프에서 우선 탐색으로 탐색 방향을 결정함으로써 휴리스틱 의존성에 따른 문제점을 줄일 수 있으며, 실제 경로 탐색 시 적용되는 범위가 제한되어 있기 때문에 탐색 비용을 줄일 수 있다.

### 3.1 추상 그래프 생성

추상 그래프는 실제 도로 네트워크를 추상적으로 표현한 그래프로서 추상 그래프  $AG = (PV, PE)$ 는 가상 정점 PV와 가상 간선 생성 기준에 따라 전체 탐색 영역을 연결한 가상 간선 PE의 모임으로 이루어진다.

추상 그래프 생성은 다음과 같이 이루어진다.

첫째, 맵 전체를 고정된 크기의 셀로 구성된 하나의 그리드  $Grid = \{Cell_1, Cell_2, \dots, Cell_n\}$ 로 나뉘인다.



(a) 추상 그래프 (b) 가상 정점과 간선  
 그림 1. 추상 그래프 예제

둘째, 각각의 고정 크기 셀을 구성하는 네 개의 경계간선(boundary edge)에 대해 하나의 가상 정점을 생성한다. 인접한 셀이 존재하지 않을 경우 경계간선에 대해서는 가상 정점을 생성하지 않는다. 가상 정점은 실제 도로에 존재하는 노드가 아니며, 추상 그래프 생성을 위해 만들어진 가상의 노드이다.

하나의 셀은 최소 2개에서 최대 4개의 인접 셀이 존재한다. 임의의 셀  $Cell_i$ 에 존재하는 인접 셀들을  $Cell_{i,E}$ ,  $Cell_{i,W}$ ,  $Cell_{i,N}$ ,  $Cell_{i,S}$ 로 정의하며,  $Cell_i$ 에 존재하는 4개의 경계면을  $Cell_{i,L}$ 로 표기한다. 생성된 가상 정점은 각 경계간선의 중앙에 표현하며,  $Cell_i$ 에 존재하는 가상 정점들을

Cell<sub>i</sub>.PV로 표기한다. Cell<sub>i</sub>와 하나의 인접 셀 간에는 하나의 경계간선을 공유하고 있으며, 경계간선에 존재하는 가상 정점 또한 공유하게 된다.

셋째, 그리드에 존재하는 가상 정점들을 연결시켜, 가상 간선  $pe = \langle pu, pv \rangle$ 를 생성한다. 가상 간선은 셀 단위로 생성된다. 즉, 하나의 셀 내에 존재하는 최대 4개의 가상 정점들 간에만 연결이 가능하다.

예를 들어 그림 1(b)에서 나타나듯이 Cell<sub>i</sub>에 존재하는 가상 정점  $pv_s$ 와  $pv_N$ 사이엔 연결된 가상 간선을  $\langle pv_s, pv_N \rangle$ 로 표기한다.  $\langle pv_s, pv_N \rangle$  간선이 존재한다는 것은 Cell<sub>i,S</sub>에서 Cell<sub>i,N</sub>으로 갈수 있는 실제 경로가 최소 1개 이상 존재하는 것이다. 즉, 가상 간선은 현재 셀에서 인접 셀로 가는 경로를 나타내는 것이 아닌 인접 셀에서 다른 인접 셀로 가는 경로를 나타내는 것이다. 가상 정점과 가상 간선의 조건 및 정의는 다음과 같다.

- Cell<sub>i</sub>.L = { Cell<sub>i,E</sub>, Cell<sub>i,W</sub>, Cell<sub>i,N</sub>, Cell<sub>i,S</sub> }
- Cell<sub>i</sub>.PV = { Cell<sub>i,pv<sub>E</sub></sub>, Cell<sub>i,pv<sub>W</sub></sub>, Cell<sub>i,pv<sub>N</sub></sub>, Cell<sub>i,pv<sub>S</sub></sub> }
- adj(Cell<sub>i</sub>, Cell<sub>j</sub>) = True  
 ⇔ Cell<sub>i,E</sub> = Cell<sub>j,W</sub> ,  
 Cell<sub>i,pv<sub>E</sub></sub> = Cell<sub>j,pv<sub>W</sub></sub> ,  
 ⇔ adj(Cell<sub>j</sub>, Cell<sub>i</sub>) = True  
 ※ Cell<sub>j</sub>은 Cell<sub>i</sub>의 왼쪽에 존재한다고 가정
- BG(Cell<sub>i</sub>, Cell<sub>j</sub>) = Cell<sub>i,E</sub>  
 = Cell<sub>j,W</sub>  
 ※ BG는 Boundary Edge이다.
- $\langle pu, pv \rangle, pu \in \text{Cell}_k.PV, pv \in \text{Cell}_j.PV$   
 $pu \in \text{Cell}_{k,j}.PV, pv \in \text{Cell}_{k,j}.PV$   
 ⇔ Cell<sub>k</sub>에 존재하는 임의의 노드에서 출발하여 Cell<sub>k</sub>만을 경유하며 Cell<sub>j</sub>에 존재하는 임의의 노드에 도착하는 경로가 존재

추상 그래프 탐색을 위해 연결된 가상 간선에 대해 가중치 부여가 필요하며, 설정될 가중치는 표 1과 같다. 추상 그래프에서는 가로, 세로로 이동하는데 드는 비용을 14로 하고 대각선으로 이동하는데 드는 비용을 10으로 설정한다. 이렇게 하는 이유는 가로, 세로로 이동하는 것은 가상 정점을 기준으로 할 경우 대각선 이동이 된다. 대각선으로 이동하는 것은 가로 세로로 이동하는 것에 비해  $\sqrt{2}$ 의 비용이 더 들기 때문이다. 물론 소수점을 사용하여 연산할 경우 더욱더 정확한 계산을 할 수 있으나 연산 비용이 커지

게 된다. 따라서 연산을 빠르게 하기 위해 단순한 10, 14의 값을 사용한다[2].

표 1. 가상 간선에 대한 가중치

	$pv_E$	$pv_W$	$pv_N$	$pv_S$
$pv_E$		∞	∞	∞
$pv_W$			10	10
$pv_N$				14
$pv_S$				

3.2 추상 그래프 탐색

추상 그래프 탐색은 고정 셀을 기반으로 추상 그래프를 생성하여 이를 이용한 2단계 경로 탐색을 수행한다. 1차 탐색은 추상 그래프에서 경로를 탐색하여 탐색 방향과 탐색 영역을 추출하며, 2차 탐색은 추출된 탐색 영역 범위 내에서 경로를 탐색한다.

추상 그래프에서 경로를 탐색하기 위해 시작점 S 와 목적지 D는 추상 그래프에 포함되어야 한다. 그러나 S(D)는 실제 도로에 존재하는 노드이지만 가상 정점은 추상 그래프 생성을 위해 임의로 생성된 노드이다. 따라서 두 노드를 연결하는 방법에 따라 탐색 성능이 저하 될 수도 있으며, 경로를 찾지 못할 수도 있다. 이 논문에서는 2가지의 S(D) 삽입 방법을 제안한다.

첫 번째, 실제 도로에서 Dijkstra 알고리즘 탐색으로 S(D)와 가상 정점을 연결하는 방법이다. Dijkstra 알고리즘 탐색이 적용되는 범위는 S(D)가 삽입된 셀뿐만 아니라, 모든 인접 셀들도 포함한다. 단, 인접 셀 내부에 가상 간선이 하나라도 존재하지 않을 경우 범위 대상에서 제외된다. 가상 간선의 의미는 현재 셀에서 인접 셀로 갈수 있는 경로를 표현한 것이 아닌 인접 셀에서 다른 인접 셀로 가는 경로를 표현한 것이기 때문에 Dijkstra 탐색에 포함 되어야 한다. 따라서 S(D)에서 갈수 있는 모든 경로를 탐색하여 정확한 간선을 생성 할 수 있다. 그러나 탐색 비용이 너무 높을 수 있다. 셀의 크기와 셀 내부에 존재하는 총 노드 개수에 따라 탐색 비용이 매우 가변적이게 된다.

두 번째, 가상 정점은 실제로 존재하는 노드가 아니기 때문에 S(D)에서 가상 정점으로 가는 경로가 존재한다.라고 가정하며, S(D)와 S(D)가 삽입된 셀 내부에 존재하는 모든 가상 정점을 연결시키는 방법이다. 이 방법의 장점은 S(D)에서 가상 정점까지의 탐색 비용이 존재 하지 않

때문에 삽입 비용이 매우 작다. 그러나 실제 탐색에서 길을 찾지 못하는 경우가 발생하게 된다.

제안하는 방법은 경로 탐색의 정확성과 성능에 많은 영향을 미칠 수 있으며, 효과적인 방법을 적용하기 위해 많은 연구가 필요 할 것이다.

동일한 셀 내부에 S와 D가 함께 삽입될 경우 추상 그래프 탐색은 무의미하므로, S(D)가 삽입된 셀 정보만 제공하면 된다. 따라서 동일한 셀에 S와 D가 삽입될 경우 추상 그래프 경로 탐색을 수행하지 않는다.

S(D)가 삽입된 후 추상 그래프에서 경로 탐색이 수행하게 된다. 추상 경로 탐색으로 결정된 경로를 가상 경로라 하며, 가상 경로를 구성하는 가상 간선을 포함하고 있는 모든 셀을 유효 셀이라 한다. 유효 셀은 실제 경로 탐색에 이용되며, 추상 그래프 탐색이 종료되었을 경우 S(D)는 추상 그래프에서 제거 된다.

실제 경로 탐색은 추상 그래프에서 제공하는 유효 셀 범위에서 경로 탐색을 수행하여 경로를 제공한다. 따라서 불필요한 영역을 제거하여 실제 경로가 존재할 수 있는 범위 내에서 경로를 탐색하기 때문에 불필요한 노드 접근 횟수를 줄일 수 있다.

지금까지 설명한 추상 그래프 탐색 알고리즘은 그림 2에서 간략하게 나타난다.

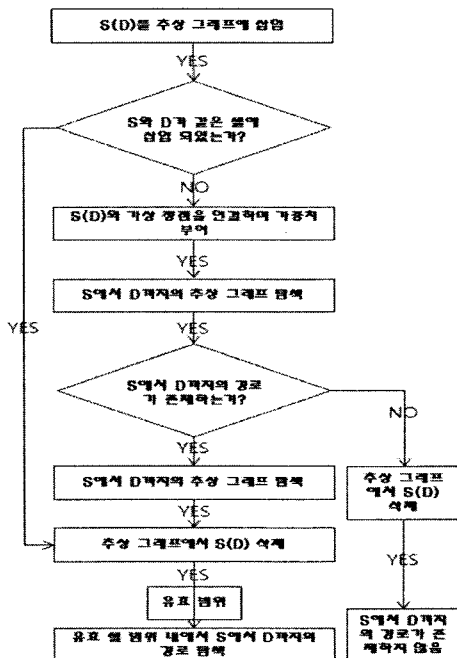


그림 2. 추상 그래프 경로 탐색 알고리즘

#### IV. 결론

이 논문에서는 추상 그래프 생성 방법과 이를 이용하여 경로를 탐색하는 알고리즘을 제안하였다. 제안하는 기법은 추상 그래프 탐색으로 추출된 유효 셀을 이용하여 제한된 범위에서 경로를 탐색하는 것이다. 이를 통해 실제 경로를 탐색을 수행할 때, 불필요한 노드 접근 횟수를 줄일 가능성이 있다.

논문에서 제안하는 2가지의 S(D) 삽입 방법 중 어떤 것이 더 적합한 것 인가에 대한 확인을 할 수 없다. 또한 보다 적합성이 높은 방법이 존재할 수 있다. 따라서 S(D)를 가상 정점에 연결하는 방법에 대한 연구가 필요할 것이다.

향후, 이 논문에서 제안하는 알고리즘으로 프로그램을 구현하고, 기존의 경로 탐색 기법과 성능 평가를 해야 할 필요가 있다. 또한 제안하는 2가지의 S(D) 삽입 방법에 대해서도 평가 되어야 할 것이다.

#### 참고문헌

- [1] 문대진, 조대수, "최저 속력 동적 휴리스틱을 이용한 경로 탐색", 한국 공간 정보 시스템 학회 2008 춘계 종합학술대회
- [2] 이현섭, 김진덕, "고정 그리드 기반 가변 휴리스틱을 이용한 최적경로 탐색", 한국해양정보통신학회 2005 추계 종합학술대회. Vol 9, No.2, pp.137-141, 2005
- [3] Peter E. Hart, Nils J. Nilsson and Bertram Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths in Graphs," IEEE Trans. on Systems Science and Cybernetics, Vol. SSC-4, No.2, pp 100-107, 1968
- [4] G. R. Jagadeesh, T. Srikanthan, and K. H. Quek, "Heuristic Techniques for Accelerating Hierarchical Routing on Road Networks", IEEE Trans. Intelligent Transportation Systems, Vol. 3, No 4, pp.301-309, 2002
- [5] A. Botea, M. Müller, and J. Schaeffer, "Near optimal hierarchical path-finding" J. of Game Develop. Vol. 1 No 1, pp.7-28, 2004
- [6] L. M. Duc, A. S. Sidhu, and N. S. Chaudhari, "Hierarchical Pathfinding and AI-Based Learning Approach in Strategy Game Design," International Journal of Computer Games Technology, 2008