
탐색 영역 확장 기법들을 활용한 추상 그래프 기반의 탐색 알고리즘 성능 개선

조대수*

Enhanced Methods of Path Finding Based on An Abstract Graph with Extension of Search Space

Dae-Soo Cho*

요 약

이 논문에서는 추상 그래프 기반의 경로 탐색 알고리즘에서 탐색된 경로의 비용이 증가하는 문제점을 보완하기 위해 탐색 영역 확장 기법들을 제안한다. 제안하는 기법들은 버퍼링 셀을 추출하여 유효 셀과 함께 탐색 영역으로 설정하는 기법으로, 단순 버퍼링 기법, 속도 제한 버퍼링 기법, 거리제한 버퍼링 기법을 제안하고 성능 평가하였다. 단순 버퍼링 기법은 유효 셀의 인접 셀들을 버퍼링 셀로 추출하며, 속도 제한 버퍼링 기법과 거리 제한 버퍼링 기법은 단순 버퍼링 기법을 통해 추출된 버퍼링 셀을 속력과 거리에 대해 제한하여 임계값을 미치지 못하는 버퍼링 셀을 제외하는 기법이다. 성능 평가 결과 탐색 영역을 확장함으로써 탐색된 경로의 비용을 줄일 수 있었다. 제안한 기법은 경로탐색, 물류관리 등 텔레매틱스 응용 서비스의 개발에 활용될 수 있을 것으로 기대된다.

ABSTRACT

In this paper, we propose enhanced methods of path finding based on an abstract graph with extension of search space to improve the quality of path. The proposed methods that are called simple buffering method, velocity constrained method and distance constrained method are to extract buffering-cells for using search space with valid-cells. The simple buffering method is to extract adjacent cells of valid-cells as buffering-cells. velocity constrained method and distance constrained method are based on simple buffering method, these eliminate buffering-cells through each of threshold. In experiment, proposed methods can improve the quality of path. The proposed methods are applicable to develop various kinds of telematics application, such as path finding and logistics.

키워드

동적 휴리스틱, 탐색 영역 가지치기, 추상 그래프

Key word

Dynamic Heuristic, Pruning Search Space, Abstract Graph

* 정회원 : 동서대학교 컴퓨터정보공학부 부교수
(교신저자, dscho@dongseo.ac.kr)

접수일자 : 2010. 07. 26
심사완료일자 : 2010. 09. 16

I. 서 론

일반적으로 단말기 기반의 경로 탐색 시스템의 경우 단말기의 하드웨어 성능을 고려하여 도로의 길이와 같은 정적인 정보를 사용하며, A*알고리즘과 같은 준최적 알고리즘[1,2,3,4,5,6,7,8]을 사용한다. 최근 단말기 기반의 경로 탐색 시스템에서도 경로 탐색에 활용할 수 있는 부과적인 정보(동적 정보), 즉 도로 상황을 나타낼 수 있는 도로의 속력, 차선의 수, 교통량 정보 등을 TPEG [9,10]채널을 통해 제공받을 수 있게 되었다.

동적정보를 활용한 연구[그리드, 최고, 최저, 추상] [2,3,4,5,6]가 있었으며, 동적인 교통정보의 갱신 비용 문제로 인해서 그리드 셀 단위의 속력정보를 활용한다. 그리드 셀 단위의 속력이라 각 그리드 셀 내에 존재하는 도로들의 평균 속력을 의미한다. 기존에 연구되어진 세 가지 연구[그리드, 최고, 최저][2,3,4]는 동적 정보를 활용함에 따라 탐색비용이 증가하는 문제가 있다.

가변적인 그리드 속력을 이용한 경로탐색 기법[2]은 실시간 교통정보를 활용할 경우를 대비한 경로탐색 기법이다. 이 경로탐색 기법은 경로 탐색 시 우선권을 가지는 노드를 탐색할 때 고정된 데이터인 거리만으로 평가하지 않고 가변적인 데이터인 각 도로의 속력정보를 고려한 탐색을 한다. 즉, 시간(T)=거리(S)/속력(V)를 이용하여 이동 시간(T)을 노드 평가시 가중치로 사용하였다. 속력은 그리드 셀 내의 모든 도로들의 평균 속력을 활용한다.

최저속력 동적 휴리스틱[3]은 그리드 셀의 최저속력을 동적 휴리스틱으로 사용한다. 그러나 경로 탐색에서 제외된 그리드내에 주요 교차로나 강 또는 하천 등을 가로지르는 다리가 있는 경우 크게 돌아가는 경로를 탐색해야 하므로 탐색비용이 증가하는 단점이 있다.

최고 속력 동적 휴리스틱[4]을 이용한 경로 탐색 알고리즘은 A*알고리즘을 기반으로 하지만, 열린 목록에서 노드 추출 방법을 2단계 노드 추출로 변형하였다. 동적 휴리스틱을 이용함으로써, 경로의 길은 높일 수 있었으나, 2단계 노드 추출을 이용하기 때문에 탐색 비용이 증가하는 단점이 있다.

추상 그래프 기반의 경로 탐색 알고리즘[5,6]은 경로 탐색에서 탐색 비용이 증가하는 문제점을 보완하기 위해 제시되었다. 추상 그래프란 실제 도로망을 간략한 그

래프로 표현한 것으로, 그리드 셀과 도로의 연결정보를 기반으로 생성된다. 추상 그래프 탐색을 통해 탐색 영역을 추출하고, 추출된 탐색 영역에서 실제 탐색을 수행한다. 그러나 제한된 범위 내에서 실제 탐색이 수행되기 때문에 탐색된 경로의 질이 좋지 못한 문제점이 있다.

이 논문에서는 추상 그래프 활용에 따른 문제점을 보완하기 위해 탐색 영역 확장 기법 제안한다. 추상 그래프 기반의 경로 탐색 알고리즘에서 실제 경로 탐색이 수행되는 탐색 범위는 추상 경로 탐색의 결과인 추상 경로를 포함하는 유효 셀의 범위가 실제 탐색 영역이 된다. 제안하는 기법들은 탐색 영역을 확장하기 위해 버퍼링 셀을 추출하는 기법으로 추출된 버퍼링 셀은 유효 셀과 함께 탐색 영역으로 설정된다. 이 논문에서는 단순 버퍼링 기법, 속력 제한 버퍼링 기법, 거리제한 버퍼링 기법을 제안하고 성능 평가하였다. 단순 버퍼링 기법은 추출된 유효 셀과 인접한 버퍼링 셀을 추출하는 기법이며, 속력 제한 버퍼링 기법과 거리제한 버퍼링 기법은 단순 버퍼링 기법을 기반으로 추출된 버퍼링 셀을 각각 속력과 거리로 제한하여 임계값을 미치지 못하는 버퍼링 셀을 제외하는 기법이다.

이 논문의 구성은 다음과 같다. 2장에서는 이론적 배경으로서 추상 그래프를 활용한 경로탐색 알고리즘에 대해 알아본다. 3장에서는 추상 그래프를 기반으로 탐색 영역 확장 기법에 대해 자세히 설명하고, 4장에서는 제안하는 기법의 성능 평가 실험을 수행하여 결과를 분석하고, 마지막 5장에서 결론을 짓는다.

II. 이론적 배경

추상 그래프란 실제 도로망을 간략한 그래프로 표현한 것으로, 그리드 셀과 도로의 연결정보를 기반으로 생성된다. 추상 그래프 탐색은 실제 탐색이 수행되기 전 추상 그래프에서 우선 탐색을 수행하며, 추상 그래프의 결과로 추상 경로를 포함하고 있는 유효 셀을 추출한다. 실제 경로 탐색은 유효 셀 범위 내에 존재하는 노드들을 대상으로 수행하게 된다. 즉, 추상 그래프를 통해서 실제 탐색 영역을 제한하기 위한 1차 필터링 역할을 수행하며, 실제 그래프에서의 2차 정제를 통해 제한된 범위 내에 존재하는 노드들을 대상으로 경로 탐색 수행을 수행

한다.

[5]에서는 그리드 기반의 경로 탐색 기법에 비해서 추상 그래프 경로 탐색 기법이 노드 접근 횟수에서 최대 약 35% 좋은 성능을 보임을 실험을 통해 증명하였다. 그러나 추상 그래프 기반의 경로 탐색 알고리즘은 탐색 영역을 제한함으로써 탐색 비용을 줄일 수 있었으나, 탐색된 경로의 질은 좋지 못하는 문제점이 있다. 즉, 노드 접근 횟수 측면에서 탐색 영역이 좁아질수록 탐색 대상이 되는 노드의 수가 줄어들기 때문에 탐색 비용을 줄일 수 있으나, 탐색된 경로의 비용 측면에서는 추상 그래프에서의 1차 필터링에서 너무 많은 탐색 공간을 제거함으로써 잠재적으로 좋은 해가 사라질 확률이 높아진다. 따라서 탐색 영역 범위에 따라 탐색 비용과 탐색된 경로의 비용 간에 **tradeoff** 발생한다.

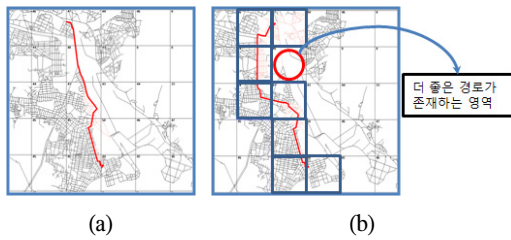


그림 1. 탐색 영역을 제한함으로써 발생하는 문제점
 (a) A*(Grid)에 의해 탐색된 경로
 (b) A*(AG)에 의해 탐색된 경로
 Fig 1. Problem of using constrained search space
 (a) Path by A*(Grid) (b) Path by A*(AG)

그림 1은 탐색 영역을 제한함으로써 발생하는 문제점의 예를 보여주고 있다. 그림 1.(a)에서는 그리드 기반 경로 탐색 알고리즘에서 탐색된 경로를, 그림 1.(b)은 추상 그래프를 이용한 탐색된 경로를 굵은 간선으로 각각 보여주고 있다. 그림 1.(b)에서 굵은 사각형으로 표시한 것은 유효 셀을 의미하며, 실제 경로 탐색은 굵은 사각형 범위 내에서 수행하게 된다. 그러나 원으로 표시한 영역에 그림 1.(a)과 같이 경로가 존재하나 유효 셀 범위 내에 포함되어 있지 않기 때문에 탐색 대상에서 제외된다. 즉, 제한된 범위 내에서 경로 탐색을 수행하기 때문에 탐색시 노드 선택의 경우의 수가 줄어들며, 제외된 영역에 좋은 경로들이 존재하더라도 경로 탐색의 대상이 되지 않는 문제점이 발생하였다.

III. 추상 그래프 기반의 탐색 알고리즘 개선

이 장에서는 추상 그래프를 기반으로 탐색된 경로의 질을 높이기 위한 탐색 영역 확장 기법들을 제안한다. 탐색 영역을 확장하기 위해 버퍼링 셀을 추출하며, 유효 셀과 함께 탐색 영역으로 설정된다. 제안하는 기법은 단순 버퍼링 기법, 속력 제한 버퍼링 기법, 거리 제한 버퍼링 기법이다.

3.1 단순 버퍼링 기법

단순 버퍼링 기법은 탐색 영역 확장만을 고려하는 기법으로 유효 셀에 대해 인접한 그리드 셀을 버퍼링 셀로 추출하는 기법이다. 인접한 그리드 셀이란 하나의 그리드 셀을 둘러싸는 이웃 그리드 셀을 의미한다. 단순 버퍼링 기법에서는 버퍼링 셀의 범위를 설정하는 k -파라미터가 존재하며, k 의 값이 커질수록 추출되는 버퍼링 셀의 수는 증가하게 된다. $k=1$ 일 경우 하나의 유효 셀을 둘러싸는 그리드 셀을 추출하게 되며, $k=2$ 일 경우 $k=1$ 에 의해 추출된 버퍼링 셀을 둘러싸는 버퍼링 셀을 추출하게 된다. 즉, $k=n$ 일 때, 추출되는 버퍼링 셀 B 는 $\cup B_n = B$ 가 된다. 그림 2.(a)와 그림 2.(b)는 추출된 유효 셀을 기반으로 각 $k=1, k=2$ 일 때 추출된 버퍼링 셀을 보여주고 있다.

3.2 속력 제한 버퍼링 기법

속력 제한 버퍼링 기법은 그리드 셀의 속력 정보를 고려하는 기법으로 단순 버퍼링 기법을 기반으로 한다. 추상 그래프 기반의 경로 탐색에서 노드 평가시 거리/그리드 셀의 속력을 이용하여 노드를 평가하기 때문에 그리드 셀의 속력 정보가 낮을수록 노드가 선택될 확률이 낮아지게 된다. 따라서 단순 버퍼링 기법을 통해 추출된 버퍼링 셀 중 한계 값 이하의 속력을 가진 버퍼링 셀은 제외시키는 기법이다. 설정되는 한계 값은 유효 셀들의 평균 속력으로 설정되게 된다. 예를 들어 그림 3.(a)의 유효 셀의 평균 속력은 $4(=4+4+6+5+4+3+2/7)$ 이며, 버퍼링 셀의 속력이 4보다 큰 버퍼링 셀은 제외된다. 그림 3.(b)은 속력 제한을 통해 추출된 버퍼링 셀을 보이고 있다.

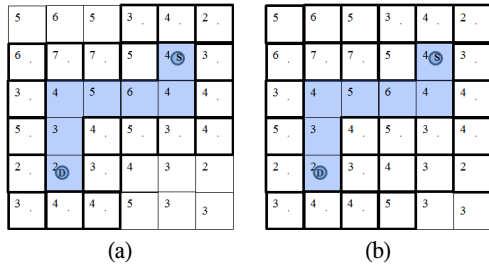


그림 2. 단순 버퍼링 기법
(a) k=1추출된 버퍼링 셀 (b) k=2추출된 버퍼링 셀
Fig 2. Method of simple buffering. (a) k=1 extracted buffering cell (b) k=2 extracted buffering cell

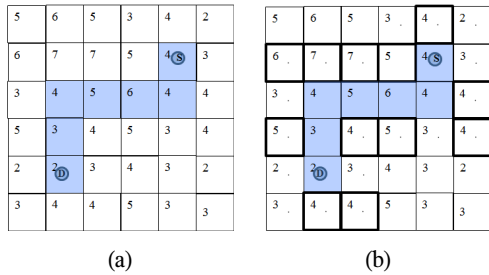


그림 3. 속력 제한 버퍼링 기법
(a) 추출된 유효셀 (b) 추출된 버퍼링 셀
Fig 3. Method of Constrained Velocity. (a) extracted valid cell (b) extracted buffering cell

3.3 거리 제한 버퍼링 기법

거리 제한 버퍼링 기법은 그리드 간의 거리 정보를 고려하는 기법으로 단순 버퍼링 기법을 기반으로 버퍼링 셀을 제한하는 기법이다. 버퍼링 셀을 제한하기 위해 기준이 되는 기준 거리는 버퍼링 셀을 파생시킨 그리드 셀의 중심과 목적지를 포함하고 있는 그리드 셀(목적셀)간의 거리이다.

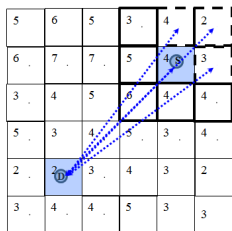


그림 4. 버퍼링 셀 제외 예제
Fig 4. Example of elimination of buffering cell

즉, 하나의 유효 셀에 대해 파생된 버퍼링 셀을 대상으로 목적 셀과의 거리를 측정하여 기준 거리보다 클 경우 버퍼링 셀에서 제외된다. 예를 들어 그림 4와 같이 기준 거리보다 큰 3개의 버퍼링 셀을 버퍼링 셀에서 제외되며, 그림 5.(b)는 그림 5.(a)의 유효 셀을 기반으로 추출된 버퍼링 셀을 보이고 있다.

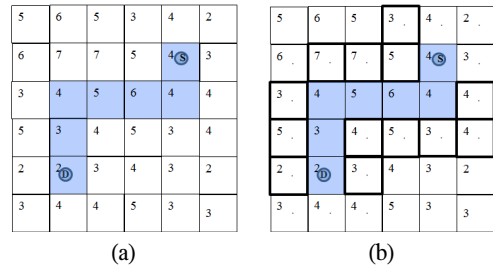


그림 5. 거리 제한 버퍼링 기법
(a) 추출된 유효셀 (b) 추출된 버퍼링 셀
Fig 5. Method of Constrained Distance. (a) extracted valid cell (b) extracted buffering cell

IV. 성능평가

이 장에서는 이 논문에서 제안하는 기법들에 대하여 성능을 평가한다. 속력 제한 버퍼링 기법과 거리 제한 버퍼링 기법은 단순 버퍼링 기법을 기반으로 하기 때문에 단순 버퍼링 기법에서 버퍼링 셀 범위를 제한하는 k-파라미터에 영향을 받기 때문에 k=1로 설정하여 성능 평가 하였다. 경로 탐색 성능 비교 모델은 버퍼링 기법을 사용하지 않는 추상 그래프 탐색 알고리즘(AGNB)과 이 논문에서 제안하는 단순 버퍼링 기법(AGSB), 속력 제한 버퍼링 기법(AGVCB), 거리 제한 버퍼링 기법(AGDCB)을 활용한 추상 그래프 탐색 알고리즘을 사용한다. 경로 탐색 성능 비교는 노드 접근, 탐색된 경로의 비용을 비교 하며, 탐색 영역 범위를 비교하여 탐색 성능과 탐색 영역 범위의 연관성을 알아보았다.

4.1 실험 데이터 및 질의 셋

실험은 실제 부산시내 도로 데이터로서, 42,357개의 노드와 106,254개의 간선으로 구성되어 있다. 하나의 간선은 방향성을 가지는 하나의 도로를 뜻하며, 각 간선들은 주변의 여러 간선들과 연결되어 있다. 그리드 생성을

위해 전체 지도 크기는 부산시내 데이터를 구성하는 노드의 최소 좌표와 최고 좌표를 고려하여 가로 45,487m, 세로 39,966m로 설정하였다. 탐색 성능 실험에 사용된 질의 셋은 다음과 같다. 출발지와 목적지까지의 직선거리를 5km미만, 5~10km, 10~15km, 15~20km, 20~30km, 30~40km으로 6등급으로 나누어, 각각 200개의 임의의 출발지와 목적지를 추출하여 각각의 알고리즘으로 탐색을 수행한다. 또한 크기가 다른 4개의 그리드 셀별로 동일한 질의 셋으로 탐색 성능을 비교하였다.

4.2 탐색 영역 범위 비교

실제 탐색이 수행되는 탐색 영역 범위는 탐색 성능에 영향을 미치기 때문에 각 알고리즘에 대해 탐색 영역 범위를 비교해 보았다. 실험결과 제약조건이 존재하지 않은 AGSB가 가장 넓은 탐색 범위를 형성하였으며, AGVCB, AGDCB순으로 탐색 범위를 형성하였다.

4.3 경로 탐색 성능 비교

각 탐색 영역 확장 기법에 대한 탐색 성능을 평가하기 위해 비교 대상으로 AGNB를 사용하였다. 비교 내용은 경로 탐색 성능을 나타낼 수 있는 노드 접근 횟수와 탐색된 경로 비용을 비교하였다.

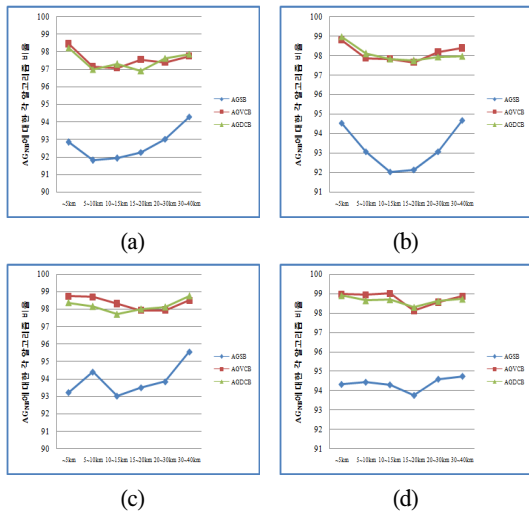


그림 6. 탐색된 경로의 비용 비교
(a) 500m (b) 1,000m (c) 1,500m (d) 2,000m
Fig 6. Comparison of cost of path searched
(a) 500m (b) 1,000m (c) 1,500m (d) 2,000m

그림 6은 그리드 셀의 크기별로 AGNB에 대한 AGSB, AGVCB, AGDCB의 탐색된 경로의 비용의 비율을 보이고 있다. 탐색된 경로의 비용을 비교하였을 때 탐색 범위가 가장 넓은 AGSB가 가장 좋은 경로를 탐색함을 알 수 있었으며, 약 4~8% 좋은 경로를 탐색하였다. AGVCB와 AGDCB는 AGNB에 비해 약 1~2% 좋은 경로를 탐색함을 알 수 있다.

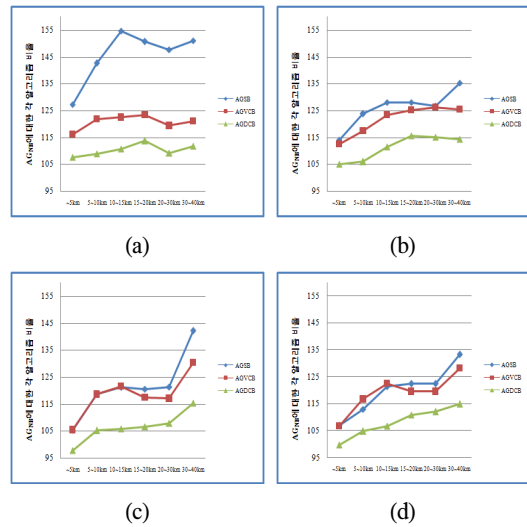


그림 7. 노드 접근 횟수 비교
(a) 500m (b) 1,000m (c) 1,500m (d) 2,000m
Fig 7. Comparison of cost of traveling
(a) 500m (b) 1,000m (c) 1,500m (d) 2,000m

그림 7은 그리드 셀의 크기별로 AGNB에 대한 AGSB, AGVCB, AGDCB의 노드 접근 횟수의 비율을 보이고 있다. 탐색 범위가 가장 작은 AGDCB가 모든 구간에서 가장 적은 노드 접근 비율을 보이고 있으나, AGNB에 비해 최대 15% 노드 접근 횟수가 증가하였다. 탐색 범위가 가장 넓은 AGSB의 경우, 최대 약 55% 노드 접근 횟수가 증가하였으며, AGVCB의 경우 최대 약 30% 증가하였다.

V. 결 론

이 논문에서 제안한 추상 그래프 활용에 따른 탐색된 경로의 비용이 증가하는 문제점을 보완하기 위해 실제

탐색이 수행되는 탐색 영역 확장 기법들을 제안하였다. 탐색 영역 확장을 위해 버퍼링 셀을 추출하여 유효 셀과 함께 탐색 영역으로 설정된다. 제안하는 기법은 단순 버퍼링 기법과 단순 버퍼링 기법을 기반으로 하는 속력 제한 버퍼링 기법, 거리 제한 버퍼링 기법이다. 단순 버퍼링 기법은 유효 셀의 인접 셀들을 버퍼링 셀로 추출하며, 속력 제한 버퍼링 기법과 거리 제한 버퍼링 기법은 단순 버퍼링 기법을 통해 추출된 버퍼링 셀을 속력과 거리에 대해 제한을 두어 임계값을 미치지 못하는 버퍼링 셀을 제외하는 기법이다.

실험 결과 탐색된 경로의 비용을 비교하였을 때, 탐색 범위가 가장 넓은 AGSB가 AGNB에 비해 약 4~8% 좋은 경로를 탐색하였으며, AGVCB와 AGDCB는 약 1~2% 좋은 경로를 탐색하였다. 그러나 노드 접근 횟수 측면에서는 탐색 영역을 확장하였기 때문에 AGSB는 AGNB에 비해 최대 50%까지 증가 하였다. 즉, 탐색 영역 범위에 따라 탐색 비용과 탐색된 경로의 비용 간에 tradeoff 발생함을 보였다. 성능 개선된 추상그래프 기반의 경로 탐색 알고리즘은 실시간 정보를 반영하고, 빠른 경로 탐색이 요구되는 분야에서 널리 활용될 수 있을 것으로 기대된다.

참고문헌

[1] 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

[2] 이현섭, 김진덕, "고정 그리드 기반 가변 휴리스틱을 이용한 최적경로 탐색," 한국해양정보통신학회 2005 추계 종합학술대회, Vol.9, No.2, pp.137-141, 2005

[3] 문대진, 조대수, "최저 속력 동적 휴리스틱을 이용한 경로 탐색", 한국공간정보시스템학회 2008 Vol.19, No.2, 2008

[4] 문대진, 조대수, "실시간 도로 정보를 이용한 최고 속력 동적 휴리스틱의 설계", 한국해양정보통신학회 2008 춘계 종합학술대회, Vol.12, No.1, pp.827-830, 2008

[5] 김지수, 이지완, 조대수, "동일 특성 노드 제거를 통한 추상 그래프 기반의 경로 탐색 알고리즘", 한국공간정보시스템학회 논문지, 제11권, 제4호, 2009. 12.

[6] 김지수, 이진완, 조대수, "추상 그래프를 활용한 경로 탐색 알고리즘의 구현 및 성능 평가", 한국해양정보통신학회 논문지, 제13권, 제11호, 2009.11.

[7] Jagadeesh, G. R., Srikanthan, T., and Quek, K. H., "Heuristic Techniques for Accelerating Hierarchical Routing on Road Networks," IEEE Trns. Intelligent Transportation Systems, Vol. 3, No 4, 2002, pp. 301-309.

[8] Botea, A., Müller, M., and Schaeffer, J., "Near Optimal Hierarchical Path-Finding," Journal of Game Development, Volume 1, Issue 1, 2005, pp 7-28.

[9] <http://en.wikipedia.org/wiki/TPEG>

[10] EBU B/TPEG, "Transport Protocol Experts Group (TPEG) TPEG specifications - Part 1: Introduction, Numbering and Versions", TPEG-INV/002, draft, October 2002.

저자소개

조대수(Dae-Soo Cho)

한국해양정보통신학회논문지
제13권 제8호 참조