VTA* 알고리즘: 가변적인 턴 휴리스틱을 적용한 A* 경로탐색 알고리즘

김지수* · 조대수**

VTA* Algorithm: A* Path-Finding Algorithm using Variable Turn Heuristic

Ji-Soo Kim* · Dae-Soo Cho**

요 약

차량을 타고 이동할 경우 좌회전, 우회전, U턴 등의 방향 전환은 차량의 속력 감소의 주요한 요인이 된다. 즉, 같은 거리를 이동할 경우 방향전환이 잦은 경로보다 직진 구간이 많은 경로가 보다 빨리 목적지에 도착할 수 있다. 이 논문에서는 직진성이 높은 경로를 탐색하기 위해서 방향전환 비용을 고려한 턴 휴리스틱과 이를 적용한 경로탐색 알고리즘(TA* 알고리즘)을 제안한다. 또한 TA* 알고리즘의 탐색비용을 개선하기 위해서 일부 구간에서만 턴 휴리스틱을 적용하는 가변적인 턴 휴리스틱(VTA* 알고리즘)을 제안한다.

ABSTRACT

In driving a car, turns such as left turns, right turns, or u-turns, make the speed of the car decrease considerably. A more straight path, therefore, is probably faster to arrive at the destination than zig-zag path with same distance. In this paper, we have newly proposed the turn heuristic to make more straight path. The path navigation algorithm with turn heuristic(called as TA* algorithm) could enhance the straightness of a path by putting the turned-edges to the turn cost. It requires higher cost to use TA* algorithm than traditional A* algorithm because the straight-edge first searching have increased the search space. We have improved the TA* algorithm into the variable TA* algorithm(called as VTA* algorithm) which adopt the turn-heuristic during the a portion of the whole path.

키워드

방향전환, A*, 턴 휴리스틱, 경로탐색

Key word

Turns, A*, Turn Heuristic, Path-Finding

* 동서대학교 컴퓨터정보공학부

** 동서대학교 컴퓨터정보공학부 조교수 (교신저자)

접수일자 : 2009. 10. 13

심사완료일자 : 2010. 01. 13

Ⅰ. 서 론

차량에 탑재되는 내비게이션 단말기의 보급이 늘어남에 따라서 복잡한 도로 네트워크상에서 임의의 출발지와 목적지에 대해 최적경로에 관한 다양한 연구가 진행되어 왔다[1]. 최적경로는 다양하게 해석될 수 있다. 즉, 출발지에서 목적지까지 거리상으로의 짧은 경로로 해석될 수도 있으며, 실제 차량을 운행할 경우에 가장 빠른 경로로 해석될 수도 있다. 또한, 운전자가 운전하기에 가장 편하고 안전한 경로로 해석될 수도 있다. 이 논문에서는 빠른 경로로서의 최적경로를 다루고자 한다.

빠른 경로에 대한 연구는 크게 두 가지로 나눌 수 있다. 첫째는 실시간 교통정보를 반영하여 이동속도가 가장 빠를 수 있는 경로를 탐색하는 방법이다. 이 방법은 주로 실시간 도로내의 차량들의 평균속력을 이용한다. 둘째는 도로의 폭, 이동방향등을 고려하여 이동속도가 빠를 것으로 추정되는 경로를 탐색하는 방법이 있다. 이 논문에서는 두 번째 방법을 대상으로 한다.

실제 도로에서의 빠른 경로를 찾기 위해서는 많은 변수들을 고려해야한다. 그 중 하나는 교차로에서의 대기시간 및 이동방향이다. 일반적으로 교차로는 교통신호등으로 통제되므로, 교차로를 지나갈 경우 신호 대기시간이 있기 때문에 교차로가 없는 도로보다 주행속도가 느릴 수 있다. 또한, 교차로에서의 좌회전, 우회전, U턴 등의 경우에는 신호대기 시간이 없는 경우에도 방향전환을 위한 속력 감소가 필요하다. 이와 같이 사람이 걸을 때와 달리 차량이 이동할 때에 방향전환이 생기면 직진할 때 보다 이동시간이 느릴 수 있다.

이 논문에서는 방향전환이 생기는 구간에 대해 방향전환 가중치를 부여하여 경로의 직진성을 높인 경로탐색으로서 TA* 알고리즘과 VTA* 알고리즘을 제안한다. TA* 알고리즘은 전체 경로 탐색 구간에서 방향전환 가중치를 적용하는 반면에 VTA* 알고리즘은 일부 구간에서만 방향전환 가중치를 선택적으로 적용하는 것을 의미한다.

이 논문의 구성은 다음과 같다. 2장에서는 기존의 경로탐색 알고리즘에 대해 알아보고, 3장에서는 TA* 알고리즘에 대해 다룬다. 4장에서는 TA* 알고리즘을 개선한 VTA* 알고리즘에 대해 설명한다. 5장에서는 제안하는 알고리즘의 성능실험 및 결과를 분석하고, 마지막 6장에서 결론을 짓는다.

Ⅱ. 관련연구

경로탐색 알고리즘은 탐색 결과의 품질에 따라서 준최적경로 탐색방식과 최적경로 탐색방식으로 분류될 수 있다. 첫째, 준 최적해를 탐색하는 방법은 휴리스틱을 이용하고 있다. [2]에서는 Major Road, Highways, Freeways 등의 도로는 매크로 레벨로, City Streets, Frontage Roads 등의 도로는 마이크로 레벨로 구분하고 있다. 매크로 레벨 도로는 빠른 속도로 이동할 수 있으므로, 일반적으로 경로 탐색은 (1)마이크로 레벨에서의 매크로 레벨까지의 검색, (2)매크로 레벨에서의 검색, (3) 매크로 레벨에서 다시 마이크로 레벨로의 검색과 같은 세 단계를 거치게 된다.

둘째, 최적해를 탐색하는 방식에서는 경로 뷰(path view)[3,4,5]라고 불리는 미리 계산되어진 패스 정보를 활용함으로서 검색비용을 줄이려는 연구들이 많이 수행되었다. 즉, Dijkstra 알고리즘[6]과 같이 최적의 해를 찾는 알고리즘은 많이 있으나, 도로네트워크를 구성하는 노드의 수가 커짐에 따라 검색비용이 매우 커지기 때문에, 실제 응용에서는 사용할 수 없다. 따라서 검색비용(시간)을 줄이기 위해서 미리 계산되어진 정보가 필요하다. 플랫 패스 뷰(FPV: Flat Path View)[3,4]는 모든 노드 쌍에 대해서 미리 최적의 경로를 계산해 놓는 방식이다. 패스 뷰는 미리 계산되어진 패스라는 의미로 실체화된 패스 뷰라는 용어에서 출발하였다. 패스 뷰를 사용하는 경우에 경로 탐색비용은 줄어들지만 FPV를 유지하는 비용이 매우 커지는 단점이 있다. 즉, 도로네트워크 정보에 변경이 발생하면, 패스 뷰를 재계산해야 하는 단점이 있다. 따라서 계층구조를 활용하여, 패스 뷰의 양을 줄이는 연구들[5,7,8]이 다양하게 수행되었다.

Ⅲ. TA* 알고리즘

TA*에서는 경로탐색 시 방향전환비용을 포함하여 다음 노드를 선택한다. 이 논문에서는 서로 인접한 두 간선 $e_{i,j}$, $e_{j,k}$간의 방향전환 비용은 다음과 같이 정의한다.

$$turncost(e_{i,j}, e_{j,k}) = 1 - \frac{\overrightarrow{e_{i,j}} \bullet \overrightarrow{e_{j,k}}}{|\overrightarrow{e_{i,j}}||\overrightarrow{e_{j,k}}|} = 1 - \cos\theta \quad (1)$$

단, Θ는 인접한 두 간선이 이루는 각을 의미한다. 두 간선이 서로 직진하는 경우(Θ=0° 또는 360°)에는 방향전환 비용은 0이고, 좌회전 또는 우회전과 같이 직각인 경우(Θ=90° 또는 270°)에는 방향전환 비용은 1이며, U턴과 같이 서로 반대방향인 경우(Θ=180°)에는 방향전화 비용이 2가 된다. 즉, 경로상의 두 간선이 이루는 각이 직진에 가까울수록 방향전환 비용이 줄어들고, 반면에 서로 반대방향에 가까울수록 방향전환 비용이 늘어나게 된다. 일반적으로 차량 이동시 직진일 경우에는 방향전환에 따른 감속이 발생하지 않으면, U턴일 경우 가장 방향전환에 따른 감속이 많이 발생하므로, 수식(1)은 차량의 이동방향에 따른 방향전환 비용과 유사하다.

기존의 경로탐색에서 쓰이는 비용모델은 산출된 경로를 구성하는 모든 간선들의 거리의 합으로서, 시작노드 s에서 목적 노드 d까지의 경로 $P_{s,d}$의 비용은 다음의 수식(2)와 같다. 이 논문에서는 모든 간선들의 길이뿐 아니라, 경로상에서 간선의 방향전환 비용을 고려하여 경로 $P_{s,d}$의 비용은 수식(3)과 같이 계산한다.

$$Cost(P_{s,d}) = \sum_{e \in P_{s,d}} e.length \quad (2)$$

$$Cost(P_{s,d}) = \sum_{e \in P_{s,d}} (e.length + e.turncost)) \quad (3)$$

단, $e.turncost = turncost(e,\ e.next)$이며, 경로 $P_{s,d}$에서 최초 간선 e_i와 최종 간선 e_l에 대한 방향전환 비용은 $e_i.turncost = e_l.turncost = 0$이다.

예를 들어, 다음의 그림 1과 같은 그래프에서 출발 노드 s와 도착 노드 d 간에는 수식(2)에 의하면 다음과 같은 두 가지 최단 경로가 존재하게 된다. 즉, 두 경로의 거리는 18로서 동일하다.

s-A-B-C-d ····· ①
s-A-B-D-E-F-d ····· ②

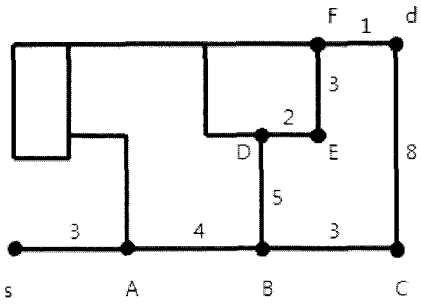

그림 1. TA* 알고리즘 예제
Fig 1. Example of TA* Algorithm

그러나 방향전환 비용을 고려하면 두 경로의 비용은 서로 달라진다. 경로 ①은 A노드에서 직진, B노드에서 직진, C노드에서 좌회전으로 방향전환 비용은 1이 발생하지만, 경로 ②는 A노드에서 직진, B노드에서 좌회전, D노드에서 우회전, E노드에서 좌회전, F노드에서 우회전이 필요하므로 방향전환 비용 4가 발생된다. 따라서 경로 ①은 총 19의 비용이, 경로 ②는 총 22의 비용이 발생하므로 서로 동일한 길이의 경로이더라도, 방향전환이 적은 경로 ①이 최적경로가 된다.

Ⅳ. 개선된 TA* 알고리즘: VTA*

TA* 알고리즘은 직진인 간선을 우선 탐색함으로 인해, 목적지가 직진만으로 갈 수 없는 곳일 경우 탐색비용이 높은 문제가 있다. 특히, 기존 TA* 알고리즘의 성능평가에 의하면 거리가 짧을 때 보다 먼 곳을 탐색할 경우 탐색비용이 더욱 높았다.

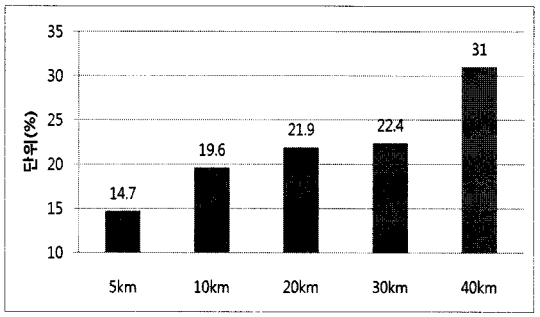

그림 2. 경로의 길이에 따른 TA* 알고리즘의 노드접근 증가량
Fig 2. Ratio of Node Access Increase by applying TA* Algorithm According to the Length of Paths

그림 2는 경로의 길이에 따른 TA* 알고리즘의 노드 접근 증가량을 도표로 나타낸 것이다. 5km 미만의 짧은 거리를 탐색할 경우 A* 알고리즘의 노드 접근양 보다 14.7% 가량 높지만, 40km 이상의 먼 거리를 탐색할 경우 노드 접근양은 30%이상 차이가 난다.

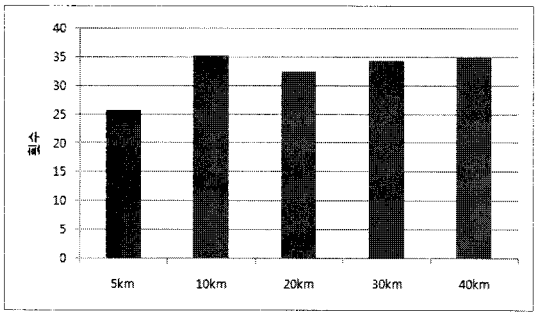

그림 3. 경로 길이에 따른 TA* 알고리즘의 방향전환 횟수 감소량
Fig 3. Ratio of Turn Reduction by applying TA* Algorithm According to the Length of Paths

그림 3은 경로 길이에 따른 TA* 알고리즘의 방향전환 회수 감소량을 나타낸 것으로 노드 증가량과 대조적인 모습을 보이고 있다. 그림에서 보이는 것처럼 10km ~ 40km 구간에서 방향전환 횟수의 감소량은 큰 차이가 없다. TA* 알고리즘으로 경로를 탐색하면 A* 알고리즘 보다 넓은 구역을 탐색하고, 경로의 질이 높아진다. 그러나 먼 거리를 탐색할 수 록 경로 품질이 증가 하는 량에 비해 탐색비용의 증가량이 더 커진다.

먼 거리를 탐색할 때 탐색비용이 높아지는 TA* 알고리즘의 단점을 보완하기 위해 턴 휴리스틱을 가변적으로 적용하는 방법을 고려할 수 있다. 이 논문에서는 일부 구간에서는 TA*알고리즘으로, 나머지 구간에서는 기존의 A* 알고리즘으로 경로탐색을 수행하는 가변적인 TA* 알고리즘(VTA*: Variable TA*)을 제안한다.

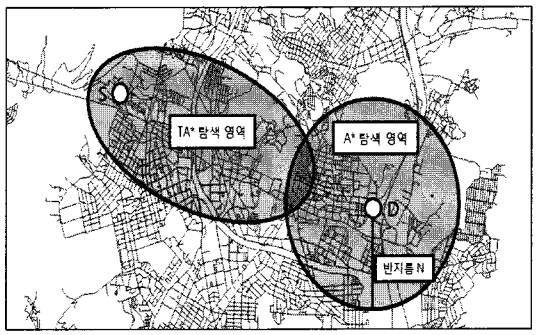

그림 4. VTA* 알고리즘의 탐색 영역
Fig 4. Search Space of VTA* Algorithm

그림 4는 VAT* 알고리즘에서 A* 알고리즘과 TA* 알고리즘의 대략적인 탐색 범위를 표현한 것으로, 출발지 부근에서는 TA* 알고리즘을 사용하며, 목적지 부근의 일정 거리 내에서는 A* 알고리즘을 적용한다. VTA* 알고리즘을 사용하는 경로탐색은 다음의 구성 순서를 따른다.

1) 출발지 S와 목적지 D 입력
2) S에서 최초 경로탐색이 진행되면 우선 TA* 알고리즘으로 탐색
3) 지속적으로 주변 노드를 탐색하면서 목적지 D의 반경 N 거리 이내에 노드를 방문하게 되면 A* 알고리즘으로 변경
4) D를 찾으면 경로탐색 종료

3)의 N은 S와 D의 직선거리의 일정 비율을 뜻한다. 예를 들어 S와 D 사이의 거리가 1km 이고 N이 30이면, D 주변 반경 300m 이내에서는 A* 알고리즘으로 경로를 탐색한다. VTA* 알고리즘의 기본 아이디어는 S 부근에서 D 방향으로 대략적인 방향을 잡으며 직진 위주의 경로를 탐색한 후, D 부근에 도달하면 방향전환을 고려하지 않고 최대한 빠르게 D를 찾는 것이다. 이와 같은 탐색을 하는 이유는 D부근에서 경로를 탐색할 때 D까지의 거리가

짧기 않기 때문에 방향전환 가중치로 인해 D로의 방향과는 다른 직진의 길을 탐색할 가능성이 더 높기 때문이다.

VTA* 알고리즘은 TA* 알고리즘 탐색을 가변적으로 사용하며, 탐색범위는 N에 의존한다. N이 0이면 VTA* 알고리즘은 TA* 알고리즘과 완전히 동일하며, N이 클수록 A* 알고리즘에 가까운 탐색을 한다. 예를 들어, 그림 5는 VTA*(50) 알고리즘의 탐색범위로 S와 D의 거리가 10km일 때 A* 탐색범위는 D를 원점으로 반경 5km 이내이고, 이 외의 영역에서는 TA* 알고리즘을 사용한다.

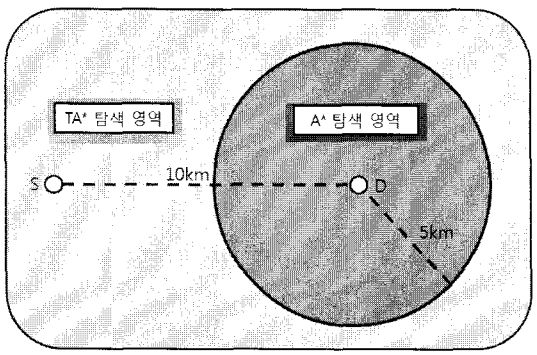

그림 5. VTA*(50) 알고리즘의 탐색범위
Fig 5. Search Space of VTA*(50) Algorithm

V. 성능 실험

이번 장에서는 VTA*(N) 알고리즘의 N 값을 변경하여 이에 따른 성능 평가 실험의 내용을 다룬다. 성능 실험은 최대한 실제 도로와 비슷한 환경으로 결과를 얻기 위해 실제 도로 네트워크와 유사한 그래프에서 실행하였다. 실험에 쓰인 그래프는 실제 부산시내 도로 데이터를 기반으로 하는 데이터이며, 정점의 개수는 42,357개이고, 간선의 개수는 106,254개이다. 하나의 간선은 방향성을 가지는 하나의 도로를 뜻하며, 평균 간선의 길이는 66m이고, 최대 길이는 4,432m, 최소 길이는 4m이다.

VTA*(N) 알고리즘의 N은 0, 20, 40, 60, 80, 100의 값을 가지고 실험하였으며, 각 알고리즘은 출발지와 목적지 간의 거리를 7등급으로 나누어, 각각 200개의 임의의 S와 D를 뽑아 실험을 하였다. 즉, 알고리즘별로 1400회의 경로탐색을 실시하였으며, 총 8400회의 경로탐색을 수행하였다.

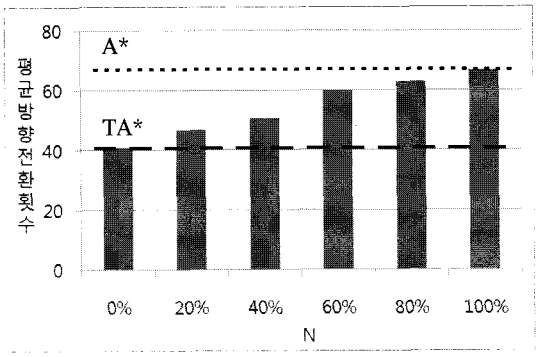

그림 6. VTA*(N) 경로탐색의 방향전환 횟수
Fig 6. Number of Turns of VTA*(N)

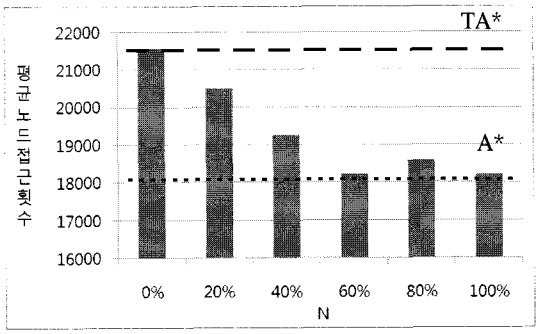

그림 7. VTA*(N) 경로탐색의 노드접근 횟수
Fig 7. Number of Node Accesses of VAT*(N)

그림 6과 7은 VTA*(N) 알고리즘으로 경로를 탐색한 결과를 그래프로 나타낸 것이다. 대략적으로 N이 60일 때 가장 좋은 성능을 보이고 있지만 그 차이는 크지는 않다. 실험결과에 보이듯이 N 값이 낮을수록 방향전환 횟수가 감소하여 직진성이 높은 경로를 탐색하고, 탐색 비용이 증가한다. 반면 N 값이 높으면 직진성은 낮고, 탐색 비용은 감소한다. 두 그래프를 유추해 볼 때 VTA* 알고리즘은 N 값에 따라 TA* 알고리즘과 A* 알고리즘의 결과를 모두 가질 수 있으며, N 값을 적절히 조절할 경우 경로의 품질과 탐색 비용 사이에 선호하는 탐색을 할 수 있다.

VI. 결론

VTA* 알고리즘은 기존의 TA* 알고리즘을 보완한 알고리즘으로, 탐색되는 모든 구간에 걸쳐 턴 휴리스틱을 적용하는 TA* 알고리즘과 달리, 턴 휴리스틱이 적용되는 구간을 가변적으로 선택하는 탐색이 가능하다.

가변적인 TA* 알고리즘을 적용하기 위해 VTA*(N) 알고리즘의 N 값을 조절하여 성능 평가 실험을 한 결과 N 값을 0으로 주면 TA* 알고리즘과 같은 결과를, 높이면 A* 알고리즘에 가까운 결과를 보였다. N이 60일 때 가장 효율적인 탐색을 할 수 있지만, 그 차이는 크지는 않았다. 경로의 직진성을 높이고 탐색비용도 함께 낮출 수는 없지만 VTA*(N) 알고리즘의 N 을 조절하면 경로의 품질과 탐색 비용 사이에 사용자가 선호하는 탐색을 할 수 있다.

이 논문에서는 경로탐색시 이미 알고 있는 정적인 정보만을 가지고 경로를 탐색 하였다. 향후 실시간 도로 교통정보를 이용하여 보다 경로의 질을 높일 수 있는 탐색에 대해 연구할 계획이며, VTA* 알고리즘을 융합하여 이전 보다 나은 알고리즘을 연구하겠다.

참고문헌

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

[2] T. A. Yang, S. Shekhar, B. Hamidzadeh and P. A. Hancock, "Path planning and evaluation in IVHS databases,"VNIS, pp.283-290, 1991

[3] R. Agrawal and H.V. Jagadish, "Materialization and Incremental Update of Path Information," Proc. Fifth Int'l Conf. Data Eng., pp. 374-383, 1989.

[4] Y.-W. Huang, N. Jing and E. A. Rundensteiner, "A Semi-Materialized View Approach for Route Guidance in Intelligent Vehicle Highway Systems," Pro. Second ACM Workshop Geographic Information System, pp.144-151, 1994

[5] Y.-W. Huang, N. Jing and E. A. Rundensteiner, "Hierarchical Encoded Path Views for Path Query Processing: An Optimal Model and Its Performance Evaluation," IEEE Trans. Knowledge and Data Eng., Vol. 10, No. 3, pp.409-432, 1998

[6] E. W. Dijkstra, "A note on two problems in connection with graphs," Numerische Mathematik, Vol. 1, pp.260-271, 1959

[7] R. Goldman, N. Shivakumar, S. Venkatasubramanian and H. Garcia-Molina, "Proximity Search in Databases," VLDB, 1998

[8] S. Jung and S. Pramanik, "An Efficient Path Computation Model for Hierarchically Structured Topographical Road Maps," IEEE Trans. Knowledge and Data Eng., Vol. 14, No. 5, pp.1029-1046, 2002

저자소개

김지수(Ji-soo Kim)

한국해양정보통신학회 논문지
제13권 제8호 참조

조대수(Dae-Soo Cho)

한국해양정보통신학회 논문지
제13권 제8호 참조