

방향성 및 동적정보를 이용한 능동적 CNS에 대한 연구

이 종 현* · 김 영 민** · 이 상 준***

A study on the Self-adjusting CNS Using the Direction and Dynamic Information

Jong-Heon Lee*, Young-Min Kim** and Sang-Joon Lee***

ABSTRACT

The shortest path search algorithm is the most important part in the CNS(Car Navigation System). Most of CNS use dijkstra or A* algorithm for searching the path. But those algorithms spend many memory and time to search the path through the real digital map. So we propose an improved algorithm which apply the direction information and the dynamic informations such as traffic jams, accidents and terrible weather. The proposed algorithm showed us it can reduce the search area, the number of search nodes and time to calculate. And the simulation system could be able to self-adjust the path immediately, when the dynamic informations reach to the system while the car driving.

Key Words : CNS, Shortest path search algorithm, Dynamic routing algorithm

1. 서 론

지속적인 도로망 확충에도 대도시의 자동차 정체현상은 날로 심화되고 있는 추세이다. 이러한 교통문제를 해결하기 위한 여러 가지 해결방법이 절실히 요구되고 있으며 차량항법 시스템이라고 불리는 CNS(Car Navigation System)가 가장 많이 연구되고 있는 분야이다.²⁾

CNS에서 가장 중요한 것은 경로탐색 알고리즘이다. 경로탐색의 기법중에서도 최적경로는 무선 통신망을 이용해 실시간으로 시시각각 변하는 교통상황 정보를 전달받아 경로선택에 적용하는 방법이다. 갑작스런 날씨 변화에 의한 교통체증, 교통 사고에 의한 교통 통제, 이외 돌발 사고 등 운전자가 미리 예측할 수 없었던 여러 가지 사고들은 교통체증 및 사고를 유발하기 쉽다. 따라서 이러한 실시간 기상정보, 도로상태, 도로정보, 사고 정보 등을 실시간을 수신하여 경로 선택에 적용함으로써 최적 경로를 찾아낼 수 있게 되는 것이다.³⁾

최단경로문제를 찾아내기 위한 알고리즘으로는 가장 대표적이며 Dijkstra 알고리즘⁷⁾과, 휴리스틱함수^{4,5,6,8)}를 이용한 A* 알고리즘³⁾이 있으며 Dijkstra, A*를

* 제주대학교 대학원 정보공학과 석사과정
Master course of Engineering Information, Cheju Nat'l Univ.

** 제주대학교 대학원 정보공학과 박사과정
Doctor course of Information Engineering, Cheju Nat'l Univ.

*** 제주대학교 통신컴퓨터 공학부, 첨단기술연구소
Faculty of Communication & Computer Engineering, Res. Insti.
Adv. Tech., Cheju Nat'l Univ.

변형한 Bi-directional Dijkstra 와 Bi-directional A*들^{4,5,6,8)}이 있다.

Dijkstra 알고리즘은 확실히 최단경로를 탐색할 수 있다는 장점이 있으나 목적지의 방향에 관계없이 모든 방향의 노드를 검색하게 하여 때로는 무시할 수 없을 정도의 많은 탐색 소요시간을 갖는다는 단점이 있으며, A*는 탐색시간은 줄일 수 있으나 인접한 노드들에서 목적지까지의 거리계산을 추가로 해야하는 부담을 안고 있다. 또한 선정된 경로가 출발지에서 목적지까지의 탐색경로가 최단경로라는 보장을 할 수 없다.

그리고 Bi-directional dikstra, A*는 탐색을 출발지에서 목적지 방향으로 탐색하는 전방탐색과 목적지에서 출발지 방향으로 탐색하는 후방탐색을 병행하여 서로의 탐색이 중간에 교차될 때 탐색을 완료하게 된다. 이론적으로는 양방향 탐색은 단방향 탐색에 비해 전체 탐색노드 수의 절반을 줄일 수 있는 장점이 있다. 그러나 만일 양방향 탐색은 양단에서 탐색이 교차하지 않을 경우 오히려 단방향 탐색에 비해 최고 두 배의 노드를 탐색해야 하는 단점이 있다.

또한 대단히 규모가 크고 복잡한 조건을 갖는 도로망이 대상이므로 기존의 최단경로 알고리즘을 적용하는 것은 여러 문제점을 발생시킨다. 먼저, 기존 최단경로 알고리즘을 실제 도로망에 적용할 때 많은 계산량과 메모리가 필요하기 때문에 빠른 응답시간을 요구하는 실제 시스템의 구현에는 적합하지 않다. 이 문제를 해결하기 위하여 보다 빠르고, 정확한 탐색알고리즘이 필요하다.

본 논문에서는 이러한 문제에 대한 해답을 얻기 위하여 기존의 dijkstra 알고리즘에 목적지까지의 방향각을 적용하여, 탐색 영역 및 탐색 노드수를 줄임으로써 탐색에 필요한 메모리량과 탐색 시간을 줄일 수 있었으며, 또한 차량 운행중 발생할 수 있는 사고, 기후악화 등의 동적인 도로 교통정보를 활용하여 고정된 경로가 아닌 능동적으로 탐색 경로를 재 수정할 수 있는 시스템을 구현하였다.

II. 제안하는 탐색 알고리즘

2.1. 방향정보를 이용한 최단경로 탐색알고리즘

최적경로탐색을 위한 디지털 지도에는 위도와 경도에 의한 위치정보가 수록되어있다. 본 연구에서는 이에 착안하여 출발지로부터 목적지에 이르는 방향정보를 최적경로 탐색 알고리즘의 설계에 이용하였다. 즉 기존의 Dijkstra 알고리즘이 방향에 상관없이 현재의 위치에서 인접한 모든 노드를 검사하기 때문에 많은 계산량과 이에 따른 과도한 메모리의 소요 및 계산시간의 과다문제가 발생한다. 따라서 인접한 노드 중 목적지의 도달에 필요 없음직한 노드에 대한 불필요한 탐색을 사전에 제거함으로써 계산시간과 메모리의 효율적 이용이 가능해 지게 된다.

본 연구에서 제안하는 알고리즘은 Fig. 1에서와 같이 출발지 u에서 목적지 d 노드까지 인접한 모든 v 노드를 찾을 때 직선 u, D 와 u, v의 방향각 h을 이용한 방향성을 이용한다. 이때 방향각을 설정하는 휴리스틱 함수를 사용한다. 이때 사용 거리에 따라 휴리스틱함수를 조정함으로써 목적지의 방향과 일정 정도 이상의 일치하지 않는 노드는 탐색을 하지 않는다. 따라서 새로운 노드의 확장을 위한 평가함수는 식 (1)과 같이 표현할 수 있다.

$$f = g + h \tag{1}$$

h값은 직선(u,d)와 직선(u,v)의 삼각함수 Cosine 제 2법칙을 이용하여 계산한다. g는 dijkstra알고리즘을 말한다. 이 둘을 결합하여 새로운 알고리즘 f를 구현하였다. 이에 대한 구체적 알고리즘은 Fig. 2와 같다.

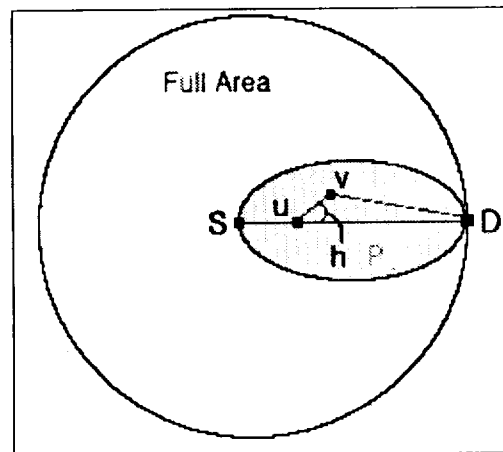


Fig. 1. Search area of proposed algorithm.

```

1. Initialization :
   T = {s}  s:source vertex
   L(n) = w(i,j) for n≠s , w(i,j) = link cost
           : least cost from vertex s to n

2. Get next vertex
   insert adjacent vertex v into T.
   v is not exist in T and Cos(uv,uD) is
   smaller than h

3. Update least-cost paths
   L(n) = min[L(n), L(x)+w(x,n)]
   add the link cost of new vertex of T to
   L(n)
    
```

Fig. 2. Proposed algorithm.

Fig. 1과 같이 목적지의 방향과는 방향각이 일정크기 이상 어긋나는 노드에 대한 불필요한 계산은 하지 않는다. Fig. 1의 영역 P에 대해서만 검색할 경우 전체적인 검색 속도가 증가되리라고 예측할 수 있다. 식 (1)에서 최적의 성능을 보이는 h의 값은 도로의 특성에 따라 가변적이며 거리가 멀어짐에 따라 각도도 작게 잡아도 됨을 알 수 있다.

2.2. 실시간 교통 정보의 반영

주행거리, 도로폭, 예상 주행시간, 교차로수 이 중에서 선택한 모든 코스트 요소에 수치를 할당함으로써 서로 다른 경로의 총 코스트를 비교한다. 이렇게 함으로써 가장 경제적(low cost)인 경로를 선택할 수 있다.

그리고 운행중 예기치 않는 돌발상황이 발생할 경우를 고려하여 사고발생, 자연재해(눈, 비), 교통체중, 행사, 도로공사 등과 같은 상황을 분류하였고, 각각의 상황 또한 상황의 정도를 2가지 등급으로 분류하여 적용하였다. 그리고 이에 따른 다음 목적지까지의 최적의 경로를 다시 계산한다.

1) 초기정보

교차로수, 차선폭, 노선의 길이에 따라 weight 값을

적용했다. 다음은 Weight 계산식이다.

$$Weight = \frac{\text{노선의길이}}{1000} + (1 - \frac{\text{차선폭}}{10}) \quad (2)$$

2) 동적정보

차량사고, 각종행사, 자연재해, 교통혼잡, 각종공사 등의 교통정보에 따라 Weight 값을 적용해서 실시간으로 최적의 경로로 갱신했다.

각각의 교통정보는 그 심각성에 따라 상, 중, 하로 나누어 Weight를 적용 했는데 그 식은 다음과 같다.

$$Weight = \text{초기 Weight값} + (\frac{\text{초기 Weight값}}{3} * C) \quad (3)$$

C는 심각성 : 3에서 1의값

3) 탐색 각도

기본적으로 시작지점과 목적지점사이의 탐색 노드를 결정하기 위한 방향각 $\angle\theta$ 은 90° 로 설정을 했으며 특정한 도로상황에서는 90° 의 범위 안에서 최적 경로를 찾지 못할 수도 있어 이럴 때는 찾는 각도를 거리에 비례하여 일정 크기만큼씩 증가시키면서 찾도록 하였다.

III. 시스템 구현 및 고찰

3.1. 시스템 환경

전체 구성요소들은 델파이(Delphi4.0)를 이용하여 구현하였으며 맵엔진은 ESRI사의 맵오브젝트(Map objects2.0)을 사용하였다. 수치지도로는 제주도와 신제주의 2차선 이상 도로를 벡터방식의 데이터를 사용하였으며 도로의 모든 정보(위치좌표, 도로길이, 차선폭, 교차로표시, 웨이트값, 등...)를 Dbase III plus 와 MS-ACCESS에 담고 있다.

IV. 시뮬레이션

다양한 최단 경로 탐색 알고리즘들의 성능을 비교하기 위하여 실제의 디지털 도로 지도 데이터 베이스

를 가지고 시뮬레이션한 결과와 그 결과를 바탕으로 분석한 내용을 정리한다.

4.1. 시뮬레이션 내용 및 결과

차량 항법 시스템에서 경로 탐색 알고리즘의 성능을 평가하는 기준은 탐색된 최단 경로의 정확성과 탐색에 소요된 시간이다. 본 논문에서는 탐색 알고리즘들이 선정한 경로의 정확성을 측정하기 위하여 각 알고리즘을 이용하여 탐색된 최단 경로의 길이를 측정하였으며 탐색 속도를 측정하기 위해서 탐색 과정에서 방문했던 전체 노드 수와 각 경로 탐색 알고리즘 모듈의 호출에 소요된 시간을 측정하였다. 탐색에 소요된 시간의 정확성을 위하여 경로의 선택을 랜덤하게 선택하여 실험을 500회 반복 실시하여 소요 시

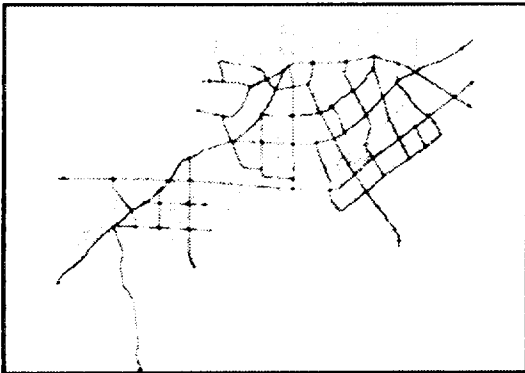


Fig. 3. Target digital map for search.

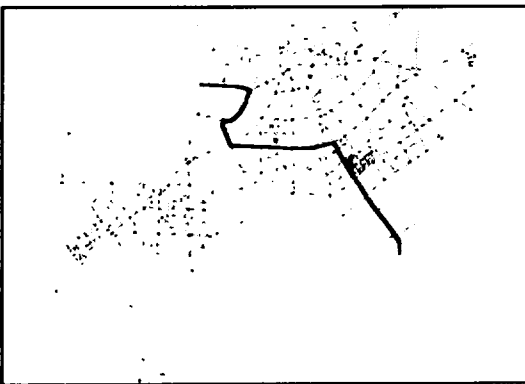


Fig. 4. Search for least cost path.

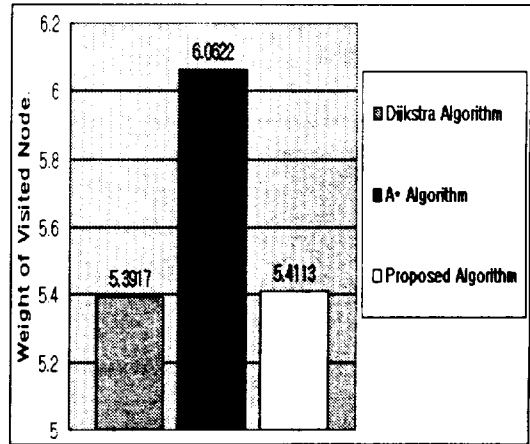


Fig. 5. Average weights.

간의 평균을 해당 경로의 탐색에 소요된 시간으로 설정하였다.

Fig. 3은 실험에 사용된 맵(map)의 한 예와 각 알고리즘에 대한 최단 경로 탐색 결과이다. 맵은 차량 항법 시스템에서 제주시와 신제주 지역을 표현한 것인데 노드 수가 70개이며 링크 수는 110개를 포함하고 있다. 노드는 내부적으로 번호로 구분하였다. 또한 실험의 편의를 위하여 편도2차선 이상의 도로만을 대상으로 실험하였다.

Fig. 4는 Dijkstra 알고리즘과 A* 알고리즘, 제안 알고리즘으로 찾아낸 최단거리 경로를 보여준다. Fig. 5는 각 알고리즘 별로 500회에 걸쳐 시작지점과 목적지점을 랜덤하게 발생시켰을 때 선정된 최단 경로의 비용의 합을 보여준다. 실험 결과에 의하면 제안 알고리즘이 Dijkstra 알고리즘과 거의 근접함으로 제안 알고리즘에 의해 선정된 경로가 최단경로임을 볼 수 있다.

Fig. 6은 각 알고리즘들이 전체 실험을 통해 방문한 노드 수의 평균을 그래프로 나타낸 것이다.

Fig. 7, Fig. 8, Fig. 9은 실제 시뮬레이션 화면을 보여준다. 각 그림에서 녹색 원은 운행중인 차량의 위치이고 파란색 원은 목적 지점이다. 그리고 빨간선은 차량이 운행할 최단 경로를 보여주고 있으며, 빨간색 원은 일정 주기적으로 발생한 사고 지점을 보여준다.

Fig. 7은 최초 시작 지점에서 목표지점까지의 탐색된 최소 비용 경로를 보여주고, Fig. 8은 1차 사고 발

생후 기존 경로의 비용 과다로 인해 새롭게 탐색된 탐색경로를 보여준다.

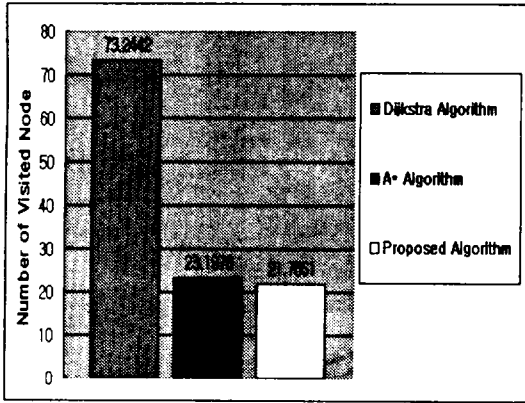


Fig. 6. Average nodes.

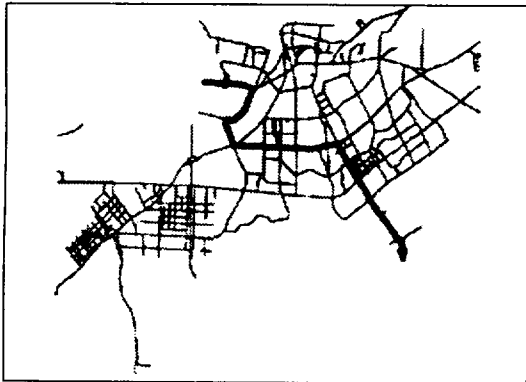


Fig. 7. First search result.

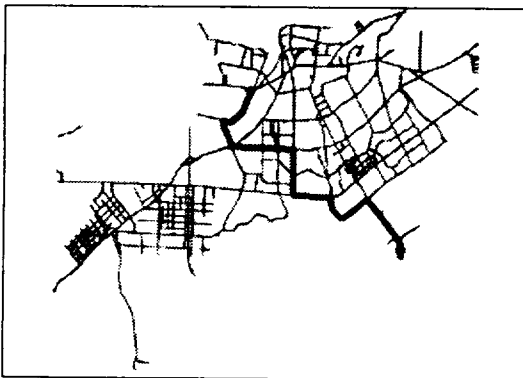


Fig. 8. Updated path after first event.

Fig. 9는 다시 2차 사고 발생후 또다시 기존 경로의 비용과다로 인해 새로운 경로를 탐색후, 최종 수정된 경로를 보여주고 있다. 이 시뮬레이션에서 사고는 10초 간격으로 랜덤하게 탐색 경로상에 발생하도록 하였다.

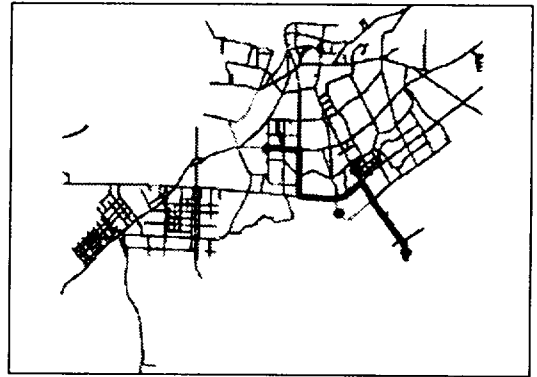


Fig. 9. Updated path after second event.

V. 결론 및 향후 연구

본 논문에서는 차량 항법 시스템에 적용하기 위한 빠른 경로 탐색을 할 수 있는 알고리즘을 제안하였다. 탐색 범위를 줄이기 위하여 출발지와 목적지간에 만나는 각 노드에서 방향각(90°) 안에 포함되는 노드들을 대상으로 탐색하였을 경우, 기존의 dijkstra 알고리즘과 유사한 정확성과 빠른 탐색 시간을 보여주었다.

또한 운행중 발생할 수 있는 동적정보를 무선통신을 통하여 수신할 수 있다는 가정하에, 동적정보 수신할 때 마다 탐색 경로를 능동적으로 재 수정할 수 있게 하였으며, 효과적으로 동작함을 볼 수 있었다.

추후 알고리즘의 좀 더 세밀한 평가를 위해 좀더 광범위한 디지털 맵을 대상으로 적용하고 또한 좀더 다양한 도로 교통정보를 활용할 수 있는 시스템으로의 업그레이드가 필요하다. 그리고 시각적인 차량항법장치는 운전자로 하여금 전방주시 태만을 일으키게 함으로써 심각한 안전의 위험을 제공할 수 있다. 따라서 이러한 방법의 해결책으로 음성인식기술등 다양

한 인터페이스의 연구가 되어져야할 것이다.

참고문헌

- [1] 김기섭, 1999, 실시간 차량경로결정을 위한 동적 최단경로 알고리즘, 홍익대학교 산업공학과 석사 논문.
- [2] 전홍석, 1998, 차량 항법 시스템의 경로 탐색을 위한 탐색 알고리즘들의 성능 비교, 한국정보교육학회 논문지, 제2권 2호, pp.252-259.
- [3] 남상우, 박문성, 1997, 방향성을 고려한 우편경로 최적화 시스템의 최단 경로 생성 알고리즘 연구, 한국정보처리학회, 정보처리 논문지, 제4권 2호, pp.491-498.
- [4] Dennis de Champeaux, 1993, Bidirectional Heuristic Search Again JACM 30(1), pp.22-32.
- [5] E. van Dijk, 1982, Heuristic Search with Partial Node Expansion and Bi-Directional Search in Product Space. pp.180-182.
- [6] Leni Sint, 1997, An Improved Bidirectional Heuristic Search Algorithm, JACM 24(2), pp.177-191.
- [7] Andrea Borella, Franco Chiaraluce, 1998, Multicast routing in BMSNs through greedy, weighted greedy, and Dijkstra algorithm: a comparative analysis. SYBEN, pp.444-454.
- [8] P.Hart, N. Nilsson, and B. Raphael, 1968, A Formal Basis for the Heuristic Determination of Minimum Cost Paths, IEEE Transactions on Systems, Science and Cybernetics, SCC-4(2), pp.100-107.