

논문 2013-50-8-21

안전운전을 위해 위험지역을 회피하는 내비게이션 경로탐색 (Searching a Navigation Path to Avoid Danger Area for Safe Driving)

이 용 후*, 김 상 운*

(Yong-Hu Lee and Sang-Woon Kim[©])

요 약

내비게이션은 차량 운전자에게 경로탐색과 길안내 서비스의 제공을 기본으로 한다. 그러나 현재의 경로탐색에서는 운전자의 안전을 지키기 위해 다양한 도로 상황을 고려하지 않고, 안전운전 DB만을 이용하여 최단경로를 찾는 서비스만을 제공하고 있다. 따라서 안전운전을 위해서는 단순한 최단경로 검색 서비스를 벗어나, 위험지역이나 위험시간대를 고려한 경로탐색이 필요하다. 본 논문에서는 A* 알고리즘을 이용하여 안전운전을 위해 위험지역을 회피할 수 있는 경로탐색 방법을 제안한다. 이 경로탐색에서 탐색노드의 경로 적합도를 측정할 때 위험지역 종류별로 서로 다른 휴리스틱 가중치를 적용하였다. 특히 경로 탐색에서 회피할 위험지역으로는 사고가 자주 발생하는 사고 다발구간, 어린이 보호구역, 교차로지역, 그리고 위험시간대를 고려하여 동적으로 위험지역별 가중치를 부여하였다. 컴퓨터 시뮬레이션 결과, 안전운전을 위해 위험지역을 회피할 수 있는 경로탐색 서비스를 제공할 수 있는 가능성을 확인하였다.

Abstract

The primary function of navigation system is to provide route search and road guidance for safe driving for drivers. However, the existing route search system provides a simple service that looks up the shortest route using a safe driving DB without considering different road characteristics for the safety of the drivers. In order to maintain the safe driving, rather than searching the shortest path, a navigation system, in which the danger areas and/or the dangerous time zones have been considered, is required. Therefore, in this paper we propose a strategy of searching a navigation path to avoid danger areas for safe driving by using the A* algorithm. In the strategy, when evaluating the path-specific fitness of the navigation nodes, different heuristic weights were assigned to different types of risk areas. In particular, we considered three kinds of danger areas, such as accident-prone sections where accidents occur frequently, school zones, and intersection regions, as well as the time slots when the probability of danger is high. From computer simulation, the results demonstrate that the proposed scheme can provide the way to avoid danger areas on the route searching and confirm the possibility of providing the actual service.

Keywords : Route Search, Safe Driving, Navigation-path, GPS

I. 서 론

차량용 내비게이션 시스템은 출발지점에서 목적지점

까지의 최적경로를 탐색하고, 탐색된 경로를 이용하여 길을 안내하는 시스템이다. 내비게이션은 차량위치를 파악하기 위해 GPS(global positioning system)^[1]를 활용한다. GPS 기술은 1990년대부터 산업, 운송, 레저, 스포츠 분야에서 광범위하게 사용되기 시작하였으며, 2000년대부터 차량용 내비게이션으로 빠르게 성장하고 있다.

내비게이션은 GPS와 안전운전 DB를 이용하여 차량 운전자에게 안내서비스를 제공하지만, 출발지부터 목적지에 이르는 운행경로 상에 존재하는 위험지역이나 위

* 정희원, 명지대학교 컴퓨터공학과
(Department of Computer Engineering, Myongji University)

© Corresponding Author(E-mail: kimsw@mju.ac.kr)

※ "이 논문은 2012년 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (NRF-2012R1A1A2041661)"

접수일자: 2013년3월25일, 수정완료일: 2013년7월21일

험시간대를 고려하지 않기 때문에 해당 운전자에게 맞는 위험관련 서비스를 제공하지 못하고 있는 실정이다.

한편, 경로탐색 알고리즘에서 가장 기본이 되는 알고리즘에 *Dijkstra*^[2]와 *A**^[3]가 있다. 특히 *A** 알고리즘을 활용하는 방법에 대한 다양한 연구^[4]가 진행되었으며, 검색 속도, 검색 장비에 따른 메모리용량 최소화, 실시간 검색 등의 응용으로 대별할 수 있다. 이 중에서 내비게이션 경로탐색 알고리즘은 *Dijkstra*와 *A**를 기반으로 경로탐색 시간을 단축하기 위해 양방향 검색 방식을 사용하는 Bi-directional *Dijkstra*와 Bi-directional *A**를 사용한다^[5]. 또한 검색 영역을 작은 영역으로 분할하여 검색하는 Stochastic time-dependent planning 방식^[6]을 사용하였으며, 그 밖에 다양한 차량운행 환경에서 최단 경로탐색 알고리즘 연구^[7-8]가 진행되었다.

본 논문에서는 *A** 알고리즘을 이용하여 위험지역을 회피한 경로를 탐색하여 안전운전 경로를 서비스하는 방법을 연구한다. 즉 기존의 단순한 최단경로 탐색 방식과는 달리, 운전자가 원하는 운행구간에서 위험지역을 회피하면서 목적지에 빠르게 도착할 수 있도록 하는 안전운전 경로탐색 방법을 검토한다.

본 논문의 향상된 안전운전 경로탐색은 위험지역과 위험시간대를 회피하는 경로탐색으로, 위험지역은 사고 다발구간, 어린이 보호구역, 교차로지역이 포함된다. 이 중에서 어린이 보호구역은 위험시간대에 따라 서로 다르게 취급된다. 이러한 위험지역 데이터는 교통사고 통계데이터에 기반하고 있으며, 이를 효과적으로 활용할 경우 사고 발생을 최소화시킬 수 있는 최단경로의 탐색을 할 수 있다.

본 논문에서는 *A** 알고리즘의 탐색방향 데이터(이를 *h*라 함)를 사용하여 출발지점에서 목적지점까지 최단 경로를 검색한다. 이 때 문헌^[5]에서와 같이 *A** 알고리즘에 교통이 혼잡한 시간과 검색 영역을 세부 분리한 후 경로를 탐색하는 방법을 고려하여, 최단 시간 내에 목적지점에 도착할 수 있는 경로를 제시한다. 또한 본 논문에서는 사고발생이 높은 위험지역과 위험빈도가 높은 시간대를 고려하는 가중치를 사용하며, 지역과 시간대에 따라 동적으로 위험지역을 회피할 수 있는 탐색 방식을 사용한다.

모의실험 결과, 제안한 방법으로 기존 경로탐색인 *Dijkstra*와 *A** 알고리즘보다 향상된 경로탐색으로 위험지역을 회피할 수 있음을 보이고, 위험지역의 유형과

위험시간대에 따라 경로탐색 결과가 달라질 수 있음을 보인다. 즉, 운전 경로를 탐색할 때 사고발생 가능성이 높은 지역과 위험시간대를 회피하는 방법으로 안전운전에 도움을 줄 수 있도록 한다.

이하, 제 II장에서는 기존의 탐색 알고리즘을 이용한 경로탐색과 안전운전을 위한 교통사고 자료를 설명한다. 제 III장에서는 안전운전 경로탐색 시스템에 대한 구조 및 알고리즘을 설명하며, 제 IV장에서는 컴퓨터 시뮬레이션 결과를 고찰한 후, 제 V장에서 결론을 맺는다.

II. 관련 연구

1. 경로탐색 알고리즘

최단경로의 탐색은 어떤 네트워크에서 두 노드 사이에 여러 개의 경로가 존재할 때 가장 짧은 경로를 찾는 방법이다. 시작노드에서 도착노드까지의 최단경로 탐색은 전통적인 그래프 탐색의 문제로서, 소요시간이나 메모리용량 등에서 큰 차이를 보이는 다양한 방법이 제안되어 있다. 이하 대표적인 경로탐색 기법인 *Dijkstra* 알고리즘과 *A** 알고리즘을 간단히 설명한다.

먼저, *Dijkstra* 알고리즘은 단일 출발지점 최단경로 탐색을 위한 알고리즘으로, 철도건설, 통신 네트워크의 경로설계, 항공기 운항계획 등의 응용분야에서 많이 사용되었다. *Dijkstra* 알고리즘이 해를 찾는 과정은 다음과 같이 세 단계로 이루어진다.

단계 1: 시작지점에서 가장 인접한 지점을 찾는다. 지금까지 최단거리로 알려진 지점은 자기 자신과 지금 찾은 지점 두개가 되며, 최단거리로 알려진 지점들의 집합을 *S*라 한다.

단계 2: 집합 *S*에 포함되지 않은 지점 중에서 시작지점에서 가장 가까운 지점을 찾는다. 이 새로운 지점은 집합 *S*에 바로 이웃한 지점들 중의 하나이고, 그 지점까지 거리는 최단거리이며, 그 지점을 *S*에 포함시킨다.

단계 3: 모든 지점이 집합 *S*에 포함될 때까지 단계 2로 돌아가 반복한다.

한편, *A** 알고리즘은 시작노드에서 목표노드까지 최우선탐색을 적용하여 최단경로를 산출한다. 즉 가장

적절한 탐색방향을 평가하며, 탐색이 잘못되었을 경우 다시 뒤로 돌아와서 다른 방향을 탐색하는 목표 지향적인 알고리즘이다. 이 때 n 번째 노드의 적합도(fitness)는 $f(n)$ 으로, 다음과 같이 평가한다.

$$f(n) = g(n) + h(n) \quad (1)$$

여기서, $g(n)$ 은 시작노드에서 현재노드까지의 최단 비용이고 $h(n)$ 는 현재노드에서 목표노드까지 최단 추정비용이다. 따라서 $h(n)$ 추정치가 정확 할수록 $f(n)$ 의 값은 실제 최단경로의 값에 가까워지며 A^* 알고리즘은 최단거리로 목표지점에 도달할 수 있게 된다. A^* 알고리즘이 해를 찾는 과정은 다음과 같이 네 단계로 이루어진다.

단계 1: 시작지점을 노드 P 라고 하며 P 에 f, g, h 값들을 배정한다. P 를 열린 목록에 추가하며 이 시점에서 열린 목록에는 P 뿐이다.

단계 2: 열린 목록의 노드 중 f 값이 가장 작은 노드를 B 라고 한다. 만약 B 가 목표 노드이면 탐색은 종료되며, 열린 목록에서 검사할 노드가 하나도 없다면 경로가 없는 것으로 간주한다.

단계 3: B 에 연결된 유효한 노드를 C 라고 하고, C 에 f, g, h 값들을 배정한다. C 가 열린 목록이나 닫힌 목록에 들어 있는지를 점검하여, 새 경로의 f 값이 더 작은지 검사한 후 작을 때 열린 목록에 추가한다. 이번 단계의 과정은 B 에 연결된 모든 노드에 반복된다.

단계 4: 단계 2로 돌아가 반복한다.

A^* 알고리즘을 이용한 다양한 응용이 연구되었으며, 이들을 용도에 따라서 incremental, memory-concerned, parallel, anytime 및 real-time의 다섯 가지로 분류한다. A^* 알고리즘의 응용에 대한 설명은 생략하며, 자세한 내용은 서베이 논문^[4]을 참조할 수 있다.

2. 위험지역과 위험시간대

위험지역은 크게 사고 다발구간, 어린이 보호구역, 그리고 교차로지역의 세 종류로 나눌 수 있다.

첫째, 사고 다발구간은 사망 교통사고가 자주 발생하는 구간을 선정해 ‘특별도로 교통안전진단’을 실시하여 선정된 구간^[9]이다. 예를 들어 2007년 서울특별시의 경

우, 혜화동 로터리 (5건), 여의도공원 앞 교차로 (4건), 영등포역 교차로~남부주유소 (4건), 동대문구 동부시장~교보생명 (3건) 등이다.

둘째, 어린이 보호구역은 교통사고의 위험으로부터 어린이를 보호하기 위한 지역으로, 어린이집, 유치원, 초등학교, 특수학교 등이 해당되며, 자동차 등의 통행속도를 시속 30km 이내로 제한한다.^[10]

셋째, 교차로지역은 교차로 내 또는 교차로 후방 30m 이내의 부분에서 발생한 교통사고 건수를 조사하여 선정한 위험지역이다.

도로형태별 교통사고 발생건수^[11]에서는 단일차로와 교차로 교통사고 건수를 보이며, 단일차로에서 발생한 교통사고는 감소하고 있지만, 교차로에서 발생한 교통사고는 증가하고 있는 실정이다.

어린이 보호구역에서 위험시간대 별로 교통사고를 조사한 결과이다^[12]. 시간대별 어린이 교통사고의 구성비를 보면 하교시간대인 12시부터 급격히 증가하기 시작하여 저녁시간대(16:00~19:59)에서는 사망자 최고치를 차지하며, 특히 하교 후 오후와 저녁시간대(12:00~19:59)의 사망자 발생률은 전체의 61.9%를 차지한다.

위험지역의 어린이 교통사고 원인조사^[13]에서 운전자의 ‘안전운전의무 불이행’이 71.6%를 차지하며, 과속에 의한 사고는 매우 적은 3.1%에 해당한다. 이는 위험지역에서 차량 속도를 줄인 경우에도 교통사고가 많이 발생함을 보이며, 운전자가 안전운전 및 방어운전 등을 하여야 하는 이유이다. 그러나 이러한 모든 위험요소를 해결하지 못하는 경우, 위험지역을 회피하도록 하는 것이 가장 좋은 대안이라고 볼 수 있다.

III. 제안된 경로탐색 알고리즘

1. 경로탐색 알고리즘

내비게이션 경로탐색은 출발지점에서 목적지점까지 위험지역을 회피한 루트를 사전에 파악하여, 안전운전에 도움을 줄 수 있도록 하는 것으로, 본 논문에서는 탐색엔진으로 A^* 알고리즘을 이용한다. 위험지역으로는 사고 다발구간, 어린이 보호구역 및 교차로지역 등 세 종류를 고려한다. 또한 어린이 보호구역은 위험시간대 별로 경로탐색 회피여부를 결정한다.

먼저 A^* 알고리즘에서 위험지역을 회피하기 위한 적합도 계산은 다음과 같다.

$$f(n) = g(n) + w(n) + h(n) \quad (2)$$

여기서 $g(n)$ 과 $h(n)$ 는 식(1)의 정의와 같고, $w(n)$ 은 현재노드가 위험지역인 경우의 가중치 값으로 계산은 다음과 같다.

$$w(n) = A(n) + S(n) + DAS(n) \quad (3)$$

여기서, $A(n)$ 는 현재노드가 사고 다발구간인 경우 가중치이며, $S(n)$ 는 어린이 보호구역인 경우 가중치이다. 그리고 $DAS(n)$ 는 현재노드가 교차로지역인 경우 차량의 감속 및 가속 지연시간에 따른 가중치이다. 따라서 현재노드의 거리비용을 $C(n)$ 이라 할 때 $A(n)$ 과 $S(n)$ 은 $C(n)$ 에 비례하며, 각각 다음과 같이 설정할 수 있다.

$$A(n) = \alpha_A C(n), \quad (4)$$

$$S(n) = \alpha_S C(n). \quad (5)$$

여기서 문제를 단순화시키기 위해 두 비례상수를 각각 $\alpha_A = 2$, $\alpha_S = (Y_{time} - Y_{min}) / Y_{max}$ 로 설정할 수 있다. 여기서 Y_{time} , Y_{max} , Y_{min} 는 시간대별 어린이 교통사고 통계조사에 기반 한 실험상수이다. 즉 $S(n)$ 은 현재 노드의 거리비용인 $C(n)$ 과 어린이 보호구역의 위험관련 정도를 나타내는 비례상수 α_S 에 의해서 결정된다. 예를 들어, 문헌^[12]에 의하면 어린이 교통사고가 가장 많이 발생하는 오후 세시(15:00)의 경우 세 실험상수는 각각 $Y_{time} = 29.4$, $Y_{max} = 32.5$, $Y_{min} = 2.4$ 이고, 반대로 교통사고가 거의 없는 새벽 세시(03:00)의 경우에는 각각 $Y_{time} = 3.2$, $Y_{max} = 32.5$, $Y_{min} = 2.4$ 가 된다*1). 따라서 노드 n 의 거리비용을 $C(n) = 5$ 라 할 때, 사고율이 높은 오후 세시의 $S(n)$ 은 거리비용의 두 배에 해당하는 9.153으로 매우 높은 반면, 새벽 세시의 $S(n)$ 은 0.123으로 매우 낮은 가중치가 된다. 즉 어린이 보호구역인 경우 노드의 가중치 $S(n)$ 이 시간에 따라 변하게 된다.

끝으로, 식 (3)의 세 번째 항인 교차로지역의 가중치 $DAS(n)$ 은 감속 및 가속 지연시간에 기인한 것으로 이

의 계산은 다음과 같다.

$$DAS(n) = ds[speed] + as[speed] \quad (6)$$

여기서 ds 와 as 는 각각 감속 및 가속 지연시간으로 모두 차량의 주행속도인 $speed$ 로 결정된다. 예를 들어, 두 차량 운행속도가 각각 100km와 10km라 할 때 100km 차량이 10km 차량보다 감속과 가속 시간이 더 많이 필요하다. 또한 교차로 지역에선 직진이나 좌-우 회전 등과 같이 차량의 이동방향에 따라 차량의 감속과 가속 지연 시간이 달라진다.

이상에서 고찰한 바와 같이 식 (3)의 가중치의 합 $w(n)$ 을 산출할 수 있으며, 이를 식 (2)에 대입하여 위험지역을 회피하기 위한 적합 도를 계산할 수 있다. A^* 알고리즘에서 식 (2)을 이용하여 위험지역을 회피하는 경로탐색은 다음과 같이 네 단계로 수행된다.

단계 1: 시작지점을 노드 P 라고 하며 P 에 f , g , w , h 값들을 배정한다. P 를 열린 목록에 추가하며 이 시점에서 열린 목록에는 P 뿐이다.

단계 2: 열린 목록의 노드 중 f 값이 가장 작은 노드를 B 라고 한다. 만약 B 가 목표 노드이면 탐색은 종료되며, 열린 목록에서 검사할 노드가 하나도 없다면 경로가 없는 것으로 간주한다.

단계 3: B 에 연결된 유효한 노드를 C 라고 하고, C 에 위험지역(어린이 보호구역, 사고 다발구간, 교차로지역)이 포함되는 경우, 해당 위험 데이터 w 값을 산출한다. 또한 C 에 f , g , w , h 값들을 배정한다. C 가 열린 목록이나 닫힌 목록에 들어 있는지를 점검하여, 새 경로의 f 값이 더 작은지 검사한 후 작을 때 열린 목록에 추가한다. 이 단계의 과정을 B 에 연결된 모든 노드에 반복 적용한다.

단계 4: 단계 2로 돌아가 반복한다.

IV. 컴퓨터 시뮬레이션

이 장에서는 위험지역을 회피하는 경로탐색에 대한 컴퓨터 시뮬레이션 내용을 설명한다. 먼저 간단한 예를 들어 탐색과정을 설명한 다음, 보다 큰 규모의 지도 데이터를 대상으로 시뮬레이션 한 결과를 제시한다.

*1) 이 실험상수와 관련된 데이터 중 차량 위치정보는 GPS로부터, 그 이외는 국토정보지리원의 DB로부터 모두 리얼타임으로 취할 수 있으며 자세한 내용은 문헌[14]을 참조할 수 있다.

1. 위험지역을 회피하는 경로탐색

먼저 그림 1은 도로 상에 위험지역인 사고 다발구간, 어린이 보호구역, 교차로지역을 포함한 테스트용 지도 데이터이다. 여기서 타원 안의 $N_i, (i = 1, 2, \dots, 36)$ 은 노드 번호로서, $N1$ 과 $N36$ 이 각각 출발지점 노드(start node)와 목적지점 노드(end node)에 해당한다. 그리고 h 는 현재 노드에서 목적지 노드까지의 직선거리를 나타내고, c 는 노드와 노드 사이에 거리비용을 나타낸다.

표 1은 그림 1의 테스트 지도를 대상으로 *Dijkstra*, *A** 알고리즘, 제안 알고리즘으로 경로를 탐색한 결과이다. 앞의 두 알고리즘은 (위험지역을 회피하지 않는) 기존 경로탐색 알고리즘으로, 제안 알고리즘과의 비교를 위하여 실험하였다. 본 논문에서 제안한 알고리즘의 경우(이하 $A+S+DAS$ 로 표기함) *Dijkstra*와 *A** 알고리즘과 달리 세 가지 위험도 가중치 A, S, DAS 를 결합하여 경로를 탐색한 결과이다. 여기서 p_{n1n2} 은 그림 1에

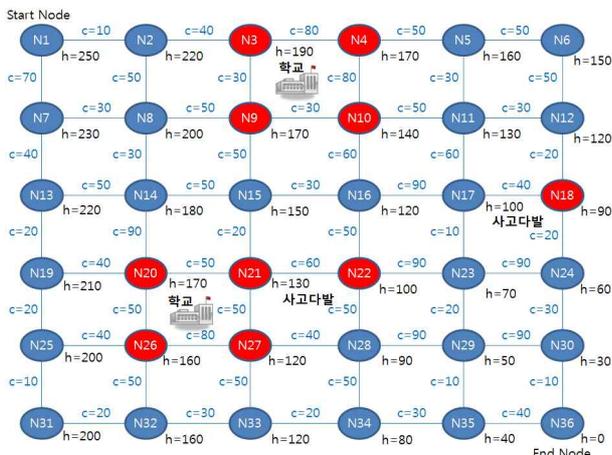


그림 1. 사고 다발구간, 어린이 보호구역, 교차로지역을 포함한 테스트 지도

Fig. 1. A testing map that includes accident prone areas, school zones, and intersections.

표 1. 세 알고리즘으로 탐색한 최단경로
Table 1. The shortest routes obtained with the three algorithms.

탐색알고리즘	탐색된 최단경로
<i>Dijkstra</i> 및 <i>A*</i>	$P_{n1n2} \rightarrow P_{n2n3} \rightarrow P_{n3n9} \rightarrow P_{n9n10} \rightarrow P_{n10n11} \rightarrow P_{n11n12} \rightarrow P_{n12n18} \rightarrow P_{n18n24} \rightarrow P_{n24n30} \rightarrow P_{n30n36}$
$A + S + DAS$	$P_{n1n7} \rightarrow P_{n7n13} \rightarrow P_{n13n19} \rightarrow P_{n19n25} \rightarrow P_{n25n31} \rightarrow P_{n31n32} \rightarrow P_{n32n33} \rightarrow P_{n33n34} \rightarrow P_{n34n35} \rightarrow P_{n35n36}$

서 표시된 노드 $N1$ 에서 노드 $N2$ 까지의 경로를 나타낸다.

표 1에서 *Dijkstra*와 *A** 알고리즘의 경우 모두 최단거리만을 탐색기준으로 삼았기 때문에 동일한 경로가 탐색된 반면, $A+S+DAS$ 의 경우에는 위험지역을 회피한 최단경로가 탐색되는 것을 확인할 수 있다.

예제의 테스트 지도에서 사고 다발구간은 두 군데 ($N18, N22$)이고, 어린이 보호구역은 여덟 곳이다. 표 2는 그림 1에서 위험지역을 통과하는 횟수를 파악한 것이다. 여기서 첫 번째 컬럼의 A, S, DAS 는 각각 경로를 탐색할 때 사고 다발구간, 어린이 보호구역, 교차로지역 가중치 하나씩만을 고려한 *A** 알고리즘이다.

여기서 어린이 보호구역을 통과하는 시간대는 오후 6시로 선정하였다. 또한 *Arc*는 사고 다발구간을 통과한 횟수이고, *Src*는 어린이 보호구역을 지나친 횟수이다. 그리고 *DASrc*는 교차로지역에서 주행 방향을 변경한 횟수이다. 기존 탐색방법인 *Dijkstra*와 *A** 알고리즘은 위험지역 검출이 불가능하며, 세 가중치 하나씩만 고려하는 A, S, DAS 방법으로도 위험지역을 효과적으로 회피하지 못하고 있다. 그러나 $A+S+DAS$ 는 위험지역을 가장 잘 회피할 수 있는 결과로 나타난다. 표 3은 표 1의 $A+S+DAS$ 동일 경로탐색 조건에서 운행시간에

표 2. 알고리즘별 위험지역 결과
Table 2. Results of risk area on different algorithms.

탐색 방법	<i>Arc</i>	<i>Src</i>	<i>DASrc</i>	계
<i>Dijkstra</i>	1	3	3	7
<i>A*</i>	1	3	3	7
A	-	3	4	7
S	-	-	4	4
DAS	1	3	3	7
$A + S + DAS$	-	-	1	1

표 3. 시간대별 위험지역 결과
Table 3. Results of risk area for each time.

시간대	<i>Arc</i>	<i>Src</i>	<i>DASrc</i>	계
00:00 ~ 03:59	1	3	3	7
04:00 ~ 07:59	1	3	3	7
08:00 ~ 11:59	-	-	1	1
12:00 ~ 15:59	-	-	1	1
16:00 ~ 19:59	-	-	1	1
20:00 ~ 23:59	-	-	1	1

표 4. 알고리즘별 연산비용과 거리비용 결과
Table 4. Results of operational cost and distance related cost on different algorithms.

비용 종류	연산비용						거리 비용
	g	w			h	계	
		Ac	Sc	DASc			
<i>Dijkstra</i>	60	-	-	-	-	60	270
<i>A*</i>	37	-	-	-	37	74	270
<i>A</i>	43	4	-	-	43	90	290
<i>S</i>	47	-	6	-	47	100	340
<i>DAS</i>	46	-	-	16	46	108	300
<i>A+S+DAS</i>	43	2	6	19	43	113	300

따른 위험지역을 통과한 횟수를 표현한 것이다.

새벽과 아침 시간대(00:00~07:59) 사이에 어린이 보호구역에 대한 위험도가 낮아짐으로 어린이 보호구역에 차량운행이 가능함을 보인다. 그러나 어린이 교통사고가 자주 발생하는 나머지 시간대(08:00~23:59)에서는 어린이 보호구역에 대한 가중치가 높아짐으로, 어린이 보호구역을 회피하여 경로를 안내하게 된다.

표 4는 그림 1의 테스트 지도에서 경로탐색을 위한 알고리즘별 연산비용과 거리비용을 측정한 결과이다. 여기서 연산비용은 경로탐색을 위한 CPU-처리량을 측정한 것으로, 출발지점에서 목적지점까지의 최종경로를 찾기까지 방문하게 되는 노드 수에 해당한다. *Dijkstra*는 방향성 정보를 이용하지 않고 주변의 모든 경로를 방문하기 때문에 연산 비용이 증가하는 반면, *A**는 방향성이 다른 경로는 방문하지 않기 때문에 경로 비교 횟수가 줄어들고 따라서 연산비용이 감소한다.

거리비용은 표 1에서 확인된 경로를 이용하여 출발지점 노드에서 목적지점 노드까지의 거리비용을 더한 값이다.

또한 표 4에서 A_c 는 사고 다발구간의 가중치 연산비용이며, S_c 는 어린이 보호구역의 가중치 연산비용이다. 그리고 DAS_c 는 교차로지역의 가중치 연산비용이다. 표 4에서 *A** 알고리즘의 연산비용은 74이며, *A+S+DAS* 알고리즘에서 $g+w+h$ 연산비용은 113이다. 따라서 그림 1에서 링크의 수가 60개 이므로 *A** 알고리즘의 연산비용은 123% 증가한 반면 *A+S+DAS* 연산비용은 188% 증가함을 알 수 있다.

2. 시뮬레이션 결과

그림 2는 실제의 도로 정보와 비슷한 환경을 구축하기 위한 시뮬레이션 초기화면이다. 시뮬레이션 화면은 크게 탐색경로를 나타내는 화면영역(그림의 왼쪽영역)과 실험환경의 설정을 위한 파라미터 입력영역(그림의 오른쪽영역)으로 구분되어 있다. 왼쪽의 탐색경로 화면영역은 최대 1600(40*40)개의 노드를 표기할 수 있고, 노드마다 그림 1과 같은 위험지역 데이터를 갖고 있다. 그리고 오른쪽의 파라미터 입력 영역은 시작노드(S)와 목표노드(E)를 비롯하여 실험환경을 설정할 수 있다.

그림 3은 그림 2의 시뮬레이션을 이용하여 경로를 설정하고, *Dijkstra*, *A** 및 제안한(*A+S+DAS*) 알고리즘을 이용한 경로탐색 결과를 나타낸다. 여기서 위험지

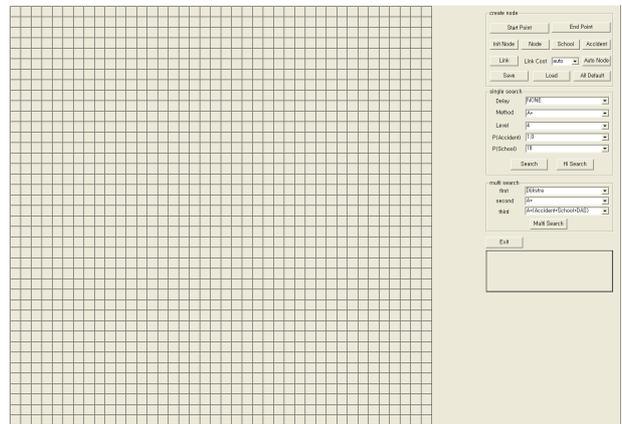


그림 2. 최단경로 탐색 시뮬레이션을 위한 초기화면
Fig. 2. An initial view of the simulation for the shortest route.

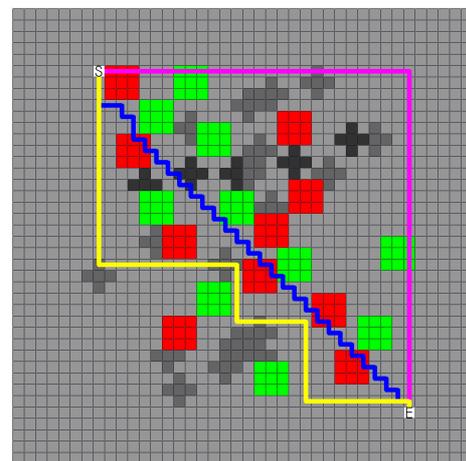
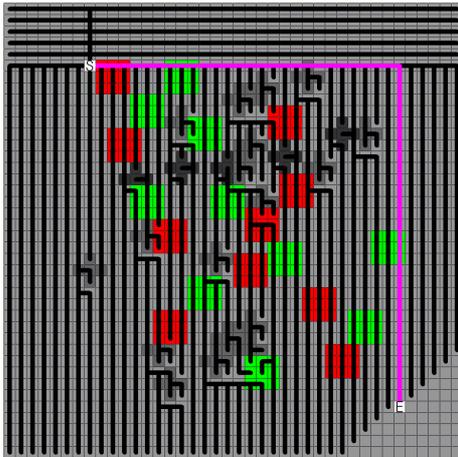
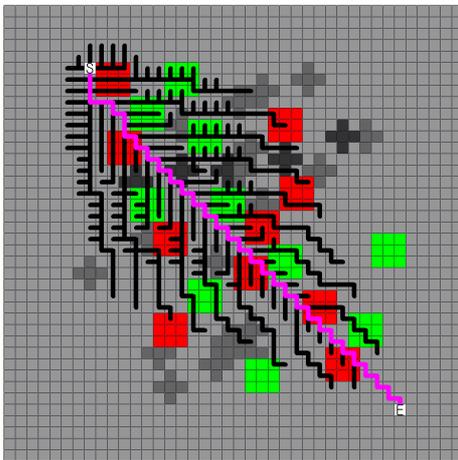


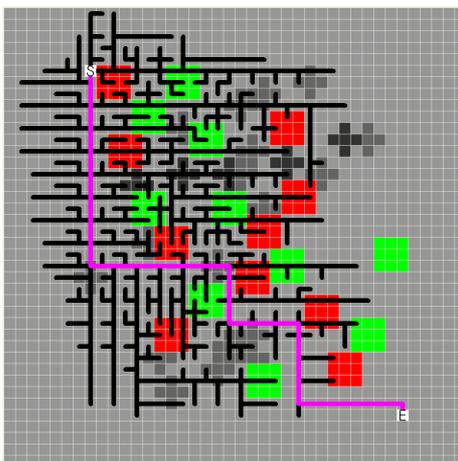
그림 3. 시뮬레이션에서 얻은 세 개의 최단경로
Fig. 3. Three shortest routes obtained from the simulation.



(a) *Dijkstra*



(b) *A**



(c) *A+S+DAS*

그림 4. 세 가지 최단경로에 대한 탐색영역(비용)
Fig. 4. Searching areas(cost) for the three shortest routes.

역 중에서 사고 다발구간과 어린이 보호구역을 각각 빨강색과 초록색으로 표시하였고, 노드사이의 거리비용은 서로 다른 그레이 값(엷은 회색에서부터 검정색까지)으로 표시하였다. 또한 그림 3에서 *Dijkstra*와 *A** 알고리즘으로 탐색한 최단경로는 각각 분홍실선과 파란실선으로 표기하였고, *A+S+DAS* 알고리즘으로 위험지역을 회피한 최단경로는 노란실선으로 표기하였다.

그림 3에서 노란색으로 표시된 제안 알고리즘의 탐색결과는 위험지역을 회피한 최단경로임을 보인다.

마지막으로, 그림 4 (a), (b), (c)는 각각 그림 3의 최단경로를 찾기 위한 연산비용(탐색영역)을 나타낸다. 여기서 세 알고리즘으로 탐색(방문평가)하였을 경우의 탐색영역(노드)를 검은 실선으로 표기하였으며, 최종 목적지 도착까지 경로는 분홍색실선으로 표시하였다.

그림 4는 *Dijkstra*와 *A**는 위험지역 중에서 사고 다발구간과 어린이 보호구역을 통과하는 반면 제안 알고리즘은 위험지역을 회피한 경로를 탐색할 수 있음을 보인다. 또한 *A**는 교차로지역(*DAS*)의 검출이 불가능한 반면 제안 알고리즘 *A+S+DAS*에서는 통과 교차로지역을 최소화할 수 있음을 보인다.

그림 3과 그림 4의 최단 탐색경로에 대한 연산비용을 정리한 결과는 표 5와 같다. 여기서 *Dijkstra*와 *A**는 사고 다발구간과 어린이 보호구역을 다수 통과한 반면 제안 알고리즘으로는 위험지역을 회피하여 경로를 탐색하였음을 나타낸다. 또한 제안 알고리즘의 탐색한 노드 수는 2909개로, *A**의 1059개에 비하여 두 배 이상 높은 이유는 위험지역에 따른 탐색비용(가중치 계산)이

표 5. 알고리즘별 연산비용 비교

Table 5. A comparison of computational cost per algorithm.

종류 \ 비용	<i>Dijkstra</i>	<i>A*</i>	<i>A+S+DAS</i>
통과한 사고다발구간	3개	12개	0개
통과한 어린이보호구역	6개	2개	0개
방향을 바꾼 교차로지역	1개	52개	6개
탐색한 노드 수(연산 비용)	3039	1059	2909
운행 시간*	31분 07초	57분 28초	34분 32초

*이 실험에서 노드와 노드 사이에 거리에 따른 주행 시간이며, 교차로 방향전환 시간은 속도에 따라서 1초에서 20초까지 서로 다른 감속 및 가속지연을 적용하였다.

포함되었기 때문이다. 끝으로 운행 시간은 노드와 노드 사이를 주행하는데 필요한 시간과 교차로 지역에서 방향변경에 기인한 감속 및 가속 지연시간을 합한 값이다. 따라서 방향을 바꾼 교차로 수를 크게 줄인 제안 알고리즘이 A^* 보다 더 빨리 도착지점에 도달할 수 있음을 보인다. 그러나 *Dijkstra*보다는 다소 증가하게 된다.

3. 가중치 비례상수에 대한 고찰

위 시뮬레이션에서 탐색되는 경로는 사고다발 구간과 어린이 보호구역에서 사용되는 가중치 비례상수 α_A 와 α_S 값에 영향을 받게 된다. 그림 5와 그림 6은 두 비례상수 값을 달리하였을 경우의 실험결과를 보인다.

먼저 그림 5는 사고다발 구간에서 α_A 값에 따른 위험지역 통과 횟수이다. 여기서 수평축은 α_A 의 값 1.0에서 3.0까지를 나타내고, 수직축은 α_A 에 따른 위험지역 통과 횟수의 합이다.

그림 5에서 α_A 값이 1.8에서 통과할 위험지역의 개수가 다소 감소하지만, 2.3 이후부터 다시 소폭으로 증가함을 보인다. 이는 α_A 값이 일정 범위보다 커지는 경우, 사고 다발구간을 회피하는 대신 어린이 보호구역이나 교차로지역을 통과하게 되는 경우가 발생하기 때문이다. 즉, 사고다발 구간의 가중치 값이 다른 위험지역의 가중치에 영향을 줄 수 있음을 알 수 있다. 이와 같은 실험결과에 따라 본 연구에서는 $\alpha_A = 2$ 로 선정하였다. 그리고 *Dijkstra*와 A^* 는 α_A 값을 사용하지 않기 때문에 α_A 값이 증가하여도 통과한 위험지역 횟수가 동일하다. 또한, A^* 의 경우 통과한 교차로지역 수가 *Dijkstra*보다 매우 크기 때문에 높은 값을 갖게 된다.

끝으로, 그림 6은 시간대에 따른 위험지역 통과 횟수

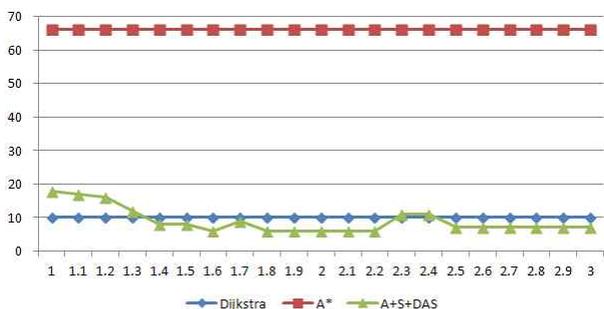


그림 5. α_A 와 위험지역 통과 횟수의 관계
 Fig. 5. A relationship between α_A and the number of the passed risk areas.

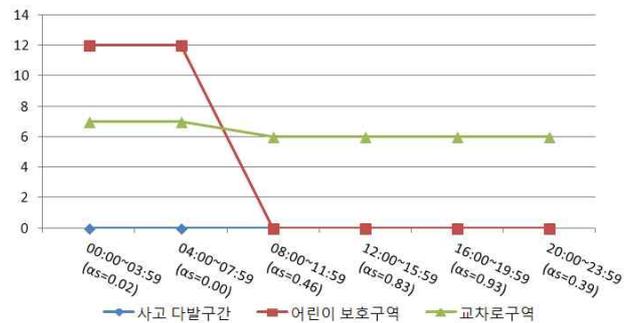


그림 6. α_S 와 위험지역 통과 횟수의 관계
 Fig. 6. A relationship between α_S and the number of the passed risk areas.

이다. 여기서 수평축은 어린이 보호구역에서 가중치에 계산에 필요한 시간과 비례상수 α_S 값이며, 수직축은 시간에 의해 산출된 α_S 에 따른 위험지역 통과 횟수의 합이다.

그림 6의 결과는 어린이 보호구역에서 사고 발생이 적은 00:00~03:59 및 04:00~07:59 시간대에선 차량이 운행이 가능하도록 위험지역 통과 횟수가 높게 계산되었음을 보인다. 그러나 08:00~11:59 시간대에선 학생들의 통학 등으로 어린이 보호구역이 회피되어 통과한 위험지역 수가 감소하는 것을 보인다. 즉, 사고 발생이 높은 시간대에 어린이 보호구역을 회피하도록 할 수 있다. 또한 어린이 보호구역을 회피함으로써 발생하는 다른 위험지역이 일부 변경됨을 알 수 있다. 특히, 08:00~11:59 시간대에선 교차로지역의 개수가 소폭 감소하였다.

V. 결 론

지금까지 다양한 환경에서 최단경로 탐색을 이용한 많은 연구가 진행되었다. 특히 내비게이션 환경에서 최단경로 탐색 알고리즘들이 제안되었다. 그러나 이들 연구는 차량사고가 많이 발생하는 위험지역을 회피하지 못하는 문제점을 가진다. 본 논문에서는 안전운전을 위해 위험지역을 회피하는 내비게이션 경로탐색 기법을 연구하였다. A^* 알고리즘을 기본 탐색방식으로 사용하였으며, 위험지역과 위험시간대에서 가중치를 사용한 경로탐색을 사용하였다. 먼저, 위험지역은 사고 다발구간, 어린이 보호구역, 교차로지역에서 분류하여 사용하며, 위험시간대는 어린이 보호구역에서 사용한다.

시각적인 관찰이 가능한 컴퓨터 시뮬레이션을 통해서 1600(40*40)개의 노드와 세 종류 위험지역을 표기할 수 있는 지도 데이터를 생성하였고, 이 지도 데이터를 대상으로 기존의 대표적인 경로탐색 알고리즘(Dijkstra 와 A*)과 제안된 알고리즘(A+S+DAS)에 대한 최단 경로 탐색과 연산비용 및 실행시간을 비교 분석하였다. 본 시뮬레이션 결과로부터 A+S+DAS가 기존 알고리즘에 비해 교통사고가 빈번히 발생하는 위험지역을 회피하면서 효율적인 탐색비용으로 출발지점에서 목적지점까지의 최단경로를 탐색할 수 있음을 보였다.

본 논문에서는 사고 다발구간의 가중치 계산에 사용되는 비례상수 α_A 값을 2로 선정하였다. 그러나 주어진 α_A 를 달리하였을 경우 사고 다발구간의 가중치 변화 때문에 최종 경로탐색 결과가 달라진다. 따라서 실제의 도로 상황을 염두에 둔 최적의 α_A 값을 산출하기 위한 추가적인 연구가 필요하다. 또한 최단경로 탐색을 위한 연산비용과 주행시간의 산출에서 선형 식을 모든 노드에 일률적으로 적용하였다. 그러나 지역에 따라 또 시간대에 따라 비선형 관계식이 적용될 수 있으며, 앞으로 이에 대한 연구도 필요하다.

REFERENCES

[1] Hofmann-Wellenhof, B., Lichtenegger, H., Collins, J., *Global Positioning System: Theory and Practice*, Springer-Verlag, New York, 2001.

[2] Dijkstra, E. W., "A note on two problems in connexion with graphs", *Numerische Mathematik*, vol. 1, no. 1, pp. 269-271, 1959.

[3] Hart, P. E., Nilsson, N. J., Raphael, B., "A formal basis for the heuristic determination of minimum cost paths", *IEEE Transactions on System Science and Cybernetics*, vol. 4, no. 2, pp. 100-107, 1968.

[4] Rios, L. H. O., Chaimowicz, L., "A survey and classification of A* based best-first heuristic search algorithms", *Proceeding of the 20th Brazilian Symposium on Artificial Intelligence (SBIA 2010)*, vol. 6404, pp. 253-262, 2010.

[5] 이재무, 김종훈, 전홍식, "차량 항법 시스템의 경로 탐색을 위한 탐색 알고리즘의 성능 비교", 정보교육학회 논문지, 제2권, 제2호, 252-259쪽, 1998년

[6] Flinzenberg, I. C. M., *Route Planning Algorithms for Car Navigation*, PhD Thesis,

Technische Universiteit Eindhoven, The Netherlands, 2004.

[7] 이봉우, 최우경, 전홍태, "퍼지 로직을 이용한 지능형 네비게이션 시스템", 전자공학회논문지, 제43권 CI편, 제4호, 67-72쪽, 2006년

[8] 옥승호, 안진호, 강성호, 문병인, "선호도기반 최단 경로탐색을 위한 휴리스틱 융합 알고리즘", 전자공학회논문지, 제47권 TC편, 제8호, 74-84쪽, 2010년

[9] 장영채, "교통사고 다발지역 개선사업 현황과 발전 방향: 서울특별시를 중심으로", 손해보험협회, 통권 제468호, 35-48쪽, 2007년

[10] 이호원, 주두환, 현철승, 김동호, 박부희, 이철기, "어린이 보호구역내 교통안전시설이 구간통행속도에 미치는 효과 분석", 한국ITS학회논문지, 제11권, 제3호, 124-132쪽, 2012년

[11] 통계청, "2007년 사망원인 통계연보", 통계청, 14-23쪽, 2008년

[12] 박재영, 김도경, "어린이 보호구역에서의 차량 속도위반 특성 분석", 한국도로학회지, 제12권, 제2호, 63-69쪽, 2010년

[13] 경찰청, 도로교통안전관리공단, 2006년 어린이 교통사고 종합분석, 범신사, 2007년

[14] 박관동, 김혜인, 원지혜, "국토지리정보원 GPS 상시 관측소 관측환경 분석", 한국측량학회지, 제25권, 제4호, 337-345쪽, 2007년

저 자 소 개



이 용 후(정회원)

2000년 울산대학교 컴퓨터공학과 학사 졸업.

2002년 명지대학교 컴퓨터공학과 석사 졸업.

2013년 명지대학교 컴퓨터공학과 박사 재학.

2008년~현재 삼성 SNS 주식회사 Senior Engineer.

<주관심분야 : 패턴인식, GPS, HCI>

김 상 운(정회원) - 교신저자

전자공학회 논문지

제 47권 CI편 제 2호 참조