

장애물 접점과 경유점 트리를 이용한 지능형 항로계획

Intelligent Route Planning Using Obstacle's Point of Contact and Waypoint-Tree(IRCT)

지민수*, 박정선*, 이영일*, 최종락**, 김용기*
경상대학교 컴퓨터과학과*, 국방과학연구소**

Min-Su Ji*, Jeong-Seon Park*, Young-il Lee*, Joong-Lak Choi**, Yong-Gi Kim*
Department Computer Science, Gyeongsang National University*

Agency for Defense Development**

e-mail : jms@ailab.gsnu.ac.kr

요 약

운항경비의 절감과 같은 경제적 제한과 3D직종 회피현상과 같은 사회적 제한으로 인해 선박의 운항에 소요되는 인원을 최소화하는 요구가 증가되고 있다. 이러한 요구에 부합하기 위해 선박의 자동화 및 지능화 기술이 요구된다. 선박의 자동화를 위해서는 GPS와 전자해도에 기반한 지능형 항해시스템의 항로계획이 선행되어야 한다. 본 논문에서는 지능형 항해시스템의 자동화된 항로계획을 위해 S-57에 정의되어 있는 좌표쌍을 기반으로한 접점산출기법과 이를 이용한 트리생성 기법, 그리고 생성된 트리구조를 기반으로한 항로산출기법을 제안한다. 제안한 항로산출기법의 효율성 검증을 위해 최적화 관점에서 대표적 항로계획 기법인 A*와 이를 개선한 RPAP 기법과 비교·분석한다.

1. 서론

선박업계에서는 최근 두드러지게 나타나는 인력부족 현상을 해결하기 위해 선박의 자동화 및 지능화에 관한 연구가 활발히 진행되고 있다. 선박의 자동화 및 지능화를 위해서는 지능형 항해시스템(intelligent navigation system)이 요구되는데, 지능형 항해시스템은 항해전문가의 지식을 저장, 관리하여 그것을 기반으로 항해전문가의 임무를 대신할 수 있는 항해 시스템을 의미한다.

선진국을 중심으로 활발히 진행되고 있는 선박의 자동항해시스템에 관한 연구는 다양한 기능을 보유하고 신뢰도가 높은 ECDIS(Electronic Chart Display Information System)의 개발을 목표로 이루어지고 있다. 현재까지 각국에서 개발된 ECDIS는 GPS, 전자해도(ENC, Electronic Navigational Chart), 항로계획, 항로감시, 항로관리, 유도, 제어 기능 등을 보유하는데, 이들 다양한 기능 중 항로계획은 항해전문가의 경유점 입력에만 의존하여 계획되고 시스템은 단지 입력된 정보만을 관리하고 있는 상황이다.

항로계획은 선박의 운항전 또는 선박이 항로를 크게 이탈했을 시에 항해전문가의 오랜 항해지식을 기반으로 하여 최단거리와 선박 운항 안전을 목표로 이루어진다. 항로계획의 자동화를 위해서는 효과적인 검색기법의 적용이 가장 필요하다. 자동항로계획을 위한

대표적인 검색기법으로는 A*탐색기법을 이용한 항로계획 기법[1][2]과 폴리선기능을 응용한 항로계획(RPAP, Route Planning Applying the Poly-line)[3]기법 등이 있다. A*탐색기법을 이용한 항로계획은 구현이 간단하다는 장점이 있으나 격자를 기반으로 함으로써 격자의 크기에 따라 전자해도에 표시된 원래 장애물의 모양 및 크기가 왜곡되어 항로계획 결과가 다르게 나타나는 단점이 있다. 그리고 RPAP기법은 장애물의 모양이 유지되고 항로계획 결과가 동일하게 산출된다는 장점을 가지지만 항로계획 시 한 장애물에 대한 항로를 결정하기 위해 $m \times n$ (m : 2보다 큰 짝수, n : 장애물을 구성하는 좌표의 수)개의 좌표를 탐색해야하기 때문에 항로계획 시 소요되는 시간이 너무 길다는 단점을 가진다.

본 논문에서는 A*탐색기법과 RPAP기법을 적용한 기존의 자동항로계획 기법의 단점을 보완할 수 있는 새로운 지능형 항로계획 기법인 장애물 접점과 경유점 트리를 이용한 지능형 항로계획(IRCT, Intelligent Route Planning Using Obstacle's Point of Contact And Waypoint-Tree)기법을 제안한다. IRCT기법에서는 항로계획시 이루어지는 장애물 회피를 위해 장애물과 임시 항로 사이의 접점을 산출하며, 이를 위해 전자해도의 표준인 S-57(Standard Publication 57)[4]에 정의되어 있는 좌표쌍[5]을 이용한다. 그리고 출발지점과

목표지점 사이에 계획된 경유점을 연결하는 트리(tree)[6]를 생성하며, 생성된 경유점 트리(WT, Wapoint Tree)를 기반으로 최적항로를 산출한다.

제 2절에서는 기존의 자동항로계획 기법인 격자기반 A*탐색기법을 이용한 항로계획 기법과 RPAP기법에 대해 다루고, 제 3절에서는 본 논문에서 제안하는 IRCT기법에 대해 다룬다. 그리고 제 4절에서는 A*탐색기법을 이용한 항로계획기법 및 RPAP기법과 IRCT기법을 비교 및 평가하며 마지막으로 제 5절에서는 결론 및 향후 연구과제에 대해 살펴본다.

2. 기존의 자동항로계획 기법

지능형 항해시스템의 자동항로계획을 위해 다양한 인공지능 기법이 응용되고 있다[7][8][9][10][11][12]. 본 절에서는 자동항로계획 분야에서 가장 널리 쓰이는 A*탐색기법을 이용한 항로계획과 벡터형 전자해도에 적합한 자동항로계획 기법인 RPAP기법에 대해 다룬다.

2.1. 격자좌표기반 A*탐색기법을 이용한 항로계획

격자좌표 기반으로 A*탐색기법을 이용한 항로계획에서는 그림 1과 같이 전자해도상의 장애물의 실제 크기나 모양이 격자에 의해 왜곡되고, 그로 인해 항로산출 결과가 격자의 크기에 따라 다르게 나타난다는 문제점이 발생한다. 그리고 그림 2와 같이 현재의 노드에서 인접한 8방향의 노드들만을 고려하여 탐색하므로 최적항로를 탐색하지 못하는 문제도 발생한다.[2]

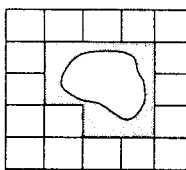
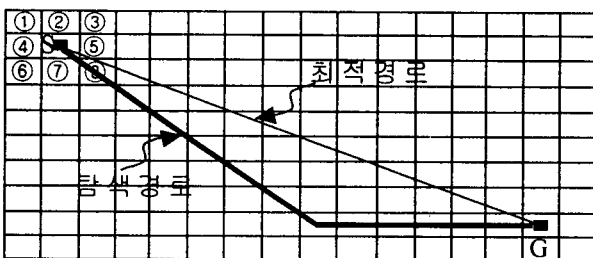


그림 1. 장애물 확장(격자기반)



* S 주위의 숫자는 S 위치에서 존재하는 각 후보 노드들에 번호를 부여한 것임

그림 2. 격자좌표기반 A*탐색기법 적용시 항로계획의 문제점

위에서 언급한 문제 이외에도 격자좌표기반 A*탐색기법을 이용하여 항로계획 시 격자의 크기가 너무 작게 설정되었거나 골이 깊은 장애물을 만나는 경우 불필요한 노드의 탐색이 많아져 메모리 부족현상이 발생하고 수행시간이 연장된다는 문제가 발생한다.

2.2. 폴리선기능을 응용한 항로계획

RPAP 기법은 출발지점과 목표지점을 지나는 임의의 직선 l_1 과 장애물이 교차하는 두 점 $(X_{11}, Y_{11}), (X_{12}, Y_{12})$ 를 찾고, 찾은 두 점의 중점을 지나고 직선 l_1 과 직교를 이루는 직선 l_2 를 찾는다. 그리고 직선 l_2 와 장애물이 교차하는 두 점 (X_{21}, Y_{21}) 과 (X_{22}, Y_{22}) 중에 출발지점과의 거리가 짧은 점을 선택하는 과정을 반복하여 목표지점을 찾아가기 때문에 역행(back-tracking)기능을 수행하지 못하는 항로계획이 된다. 따라서 결과로 산출된 항로가 최적항로가 되지 못한다는 단점을 가지며 앞에서 언급했듯이 RPAP기법에서는 장애물이 복잡하고 장애물을 구성하는 좌표의 수가 많은 경우 항로산출시간이 매우 길어진다는 단점을 가진다. 또한 RPAP기법은 그림 3과 같은 특정 모형의 장애물에 대해서는 그 출발지점이나 목표지점의 위치에 따라 최적항로를 산출하지 못하는 단점도 가진다.

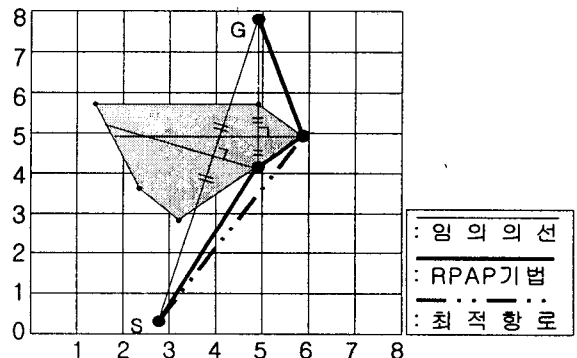


그림 3. RPAP기법에서 최적항로를 산출할 수 없는 경우

3. 장애물 접점과 경유점 트리를 이용한 지능형 항로계획(IRCT) 기법

IRCT기법을 적용하여 항로계획을 수행하기 위해서는 그림 4와 같은 장애물확장(Obstacle-growing)개념을 적용시킨 가상장애물[13]을 채용한다.

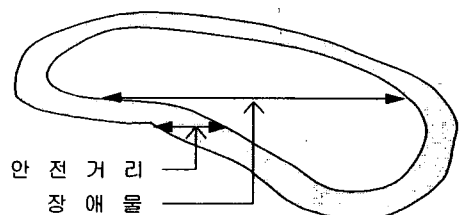


그림 4. 가상장애물

IRCT기법은 전자해도 상에 표현된 한 장애물을 회피할 때 출발지점 S를 지나는 직선과 장애물이 접했을 때 생기는 좌우 접점 Ls와 Rs, 목표지점 G를 지나는 직선과 장애물이 접했을 때 생기는 좌우접점 Lg와 Rg를 산출한다. 다음으로 Ls와 Lg사이의 경유점을 이용한 좌측향로, Rs와 Rg사이의 경유점을 이용한 우측향로를 산출한다. 그림 5는 장애물 접점을 이용하여 계획된 좌측과 우측 향로의 예를 보인다.

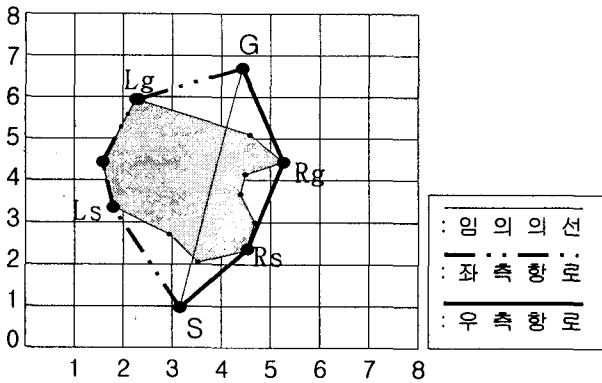


그림 5. 접점을 이용한 계획된 좌우 향로

장애물의 접점을 산출하는 방법은 다음과 같다.

첫째, 장애물을 구성하는 첫 번째 좌표 O_1 이 S와 G를 지나는 직선의 좌측에 위치하면 O_1 이 Ls와 Lg가 된다.

둘째, O_2 가 S와 G를 지나는 직선의 좌측에 위치하면서 O_2 가 S와 Ls를 지나는 직선보다 좌측에 위치하면 O_2 가 Ls로 되고 또한 O_2 가 G와 Lg를 지나는 직선보다 좌측에 위치하면 O_2 가 Lg로 된다.

셋째, 장애물을 구성하는 마지막 좌표 O_n 까지 둘째를 반복하면 Ls, Lg가 산출된다. Rs와 Rg도 역시 동일한 방법으로 산출할 수 있다.

IRCT 기법에서는 항로계획시 둘 이상의 장애물을 접할 때 각 장애물의 좌우를 지나는 항로를 이용하여 그림 6과 같은 경유점 트리(WT, Waypoint Tree)를 생성하며, WT의 각 노드는 그림 7과 같은 구조를 가진다.

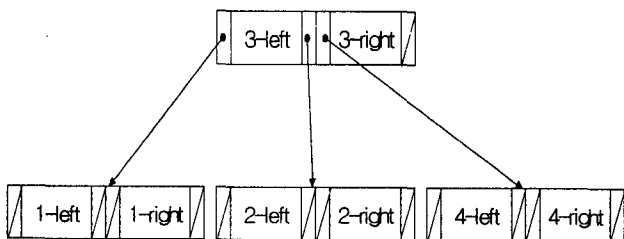


그림 6. 경유점 트리(WT)

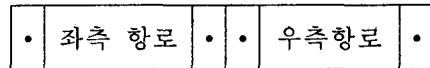


그림 7. 노드 구조

본 논문에서는 생성된 WT를 깊이우선탐색(depth first search)[6]기법으로 탐색하여 최적항로를 산출한다. 표1은 본 논문에서 최적항로 산출을 위해 적용한 탐색 알고리즘을 보인다.

```

1: PROGRAM make_route_using_search_tree
2: FUNCTION MakeRoute(S, G : coordinate,
3:                   nodenum, flag : Integer) : waypoint;
4: BEGIN
5:   link = GetLeftLink(nodenum, flag);
6:   if link <> NULL then begin
7:     tmp_g = GetFirstCoordinate(nodenum, flag);
8:     route_left := MakeRoute(S, tmp_g, link, LEFT);
9:     route_right := MakeRoute(S, tmp_g, link, RIGHT);
10:    if Length(S, tmp_g, route_left) <
11:       Length(S, tmp_g, route_right) then begin
12:      route_best := AppendRoute(route_best, route_left);
13:    end;
14:  else begin
15:    route_best := AppendRoute(route_best, route_right);
16:  end;
17: end;
18: route_best :=
19:   AppendRoute(route_best, GetRoute(nodenum, flag));
20: link = GetRightLink(nodenum, flag);
21: if link <> NULL then begin
22:   tmp_s = GetLastCoordinate(nodenum, flag);
23:   route_left := MakeRoute(tmp_s, G, link, LEFT);
24:   route_right := MakeRoute(tmp_s, G, link, RIGHT);
25:   if Length(tmp_s, G, route_left) <
26:      Length(tmp_s, G, route_right) then begin
27:    route_best := AppendRoute(route_best, route_left);
28:  end;
29:  else begin
30:    route_best := AppendRoute(route_best, route_right);
31:  end;
32: end;
33: MakeRoute := route_best;
34: END.
    
```

표 2. 경유점 트리(WT) 탐색 알고리즘

그림 8은 본 논문에서 제안하는 IRCT기법을 통해 산출된 최적항로를 보인다.

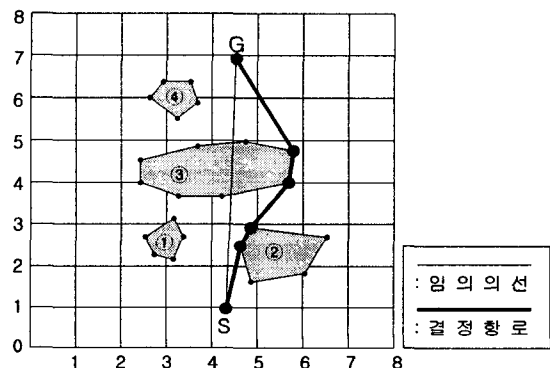


그림 8. IRCT 기법을 이용하여 산출된 최적항로

4. 비교 및 평가

본 절에서는 격자기반 A*탐색기법 및 RPAP기법과 IRCT 기법을 적용하여 좌표상의 특정 출발지점에서 목표지점까지의 항로를 계획하고 그 결과를 비교함으로써 IRCT기법의 최적성(Optimality)을 증명한다.

각 기법들을 비교 평가하기 위해 동일한 출발지점과 목표지점을 가지는 네 가지 종류의 시나리오를 설정하였으며 이들 시나리오 중 두 종류의 시나리오는 한 개의 장애물을 가지며 나머지 두 종류의 시나리오는 각각 두 개, 세 개의 장애물을 가진다. 그리고 항로계획 결과로 산출되는 총 항로 길이 합과 경유점 수를 기준으로 각 기법을 비교 평가한다. 표 2은 각 기법을 적용하여 항로계획을 한 결과이다.

시나리오 (장애물 수)	적용 기법	총 항로 길이 합	경유점 수
시나리오#1 (1)	A*	88.78	8
	RPAP	75.12	4
	IRCT	75.12	4
시나리오#2 (1)	A*	68.32	8
	RPAP	65.51	6
	IRCT	61.25	4
시나리오#3 (2)	A*	104.28	10
	RPAP	93.87	6
	IRCT	89.45	4
시나리오#4 (3)	A*	93.64	8
	RPAP	83.24	6
	IRCT	77.51	5
비교		소수점 두자리 반올림	

표 2. 항로계획 결과 비교

5. 결론 및 향후과제

본 논문에서는 지능형 항해시스템의 자동화된 항로 계획을 위해 장애물 점점과 경유점 트리를 이용한 지능형 항로계획(IRCT) 기법을 제안하였다.

IRCT기법을 이용한 항로계획은 격자기반 A*탐색기법과 RPAP기법을 이용한 항로계획보다 총 항로 길이의 합이나 경유점 수에서 더 최적화된 항로가 산출된다는 장점을 지닌다.

본 연구에 이어 향후에 이루어져야 할 과제는 다음과 같다. 첫째, 실제 전자해도 상에 본 기법을 적용하여 보다 신뢰성 높은 결과를 산출하도록 유도하여야 할 것이다. 둘째, 전자해도 상의 장애물인식에 대한

다양한 연구가 이루어져야 할 것이며 셋째, 지능형 항해시스템과의 연동 수행에 관한 연구가 이루어져야 할 것이다.

참 고 문 헌

- [1] Szczerba, Robert J., Robust Algorithm for Real-Time Route Planning, IEEE Transactions on Aerospace and Electronic Systems Vol. 36, No. 3 July 2000.
- [2] 하희천, 전자해도를 이용한 최적항로결정시스템에 관한 연구, 한국해양대학교, 1997, 2.
- [3] 박정선, 김창민, 김용기, "지능형 항해를 위한 응용 폴리선 항로계획," 한국정보처리학회 춘계 학술발표논문집, Vol. 8, No.1, pp.321-324, 2001.
- [4] S-57 Maintenance document, IHO, 2000.
- [5] 박필성, 전자해도의 교환용 표준에 관하여(1) -S-57 제 2판-, Ocean Research 19(1): 43-53, 1997.
- [6] Horowitz, E., Sahni, S., Data Structures in Pascal(Fourth Edition), Computer Science Press.
- [7] Nearchou, Andreas C., Adaptive Navigation of Autonomous Vehicles Using Evolutionary Algorithms, Artificial Intelligence in Engineering 13, 1999.
- [8] Louis, Sunshil J., Multiple Vehicle Routing With Time Windows Using Genetic Algorithms, IEEE, 1999.
- [9] 민중수, 김창민, 김용기, "자율무인잠수정의 휴리스틱 경로탐색을 위한 자세 기반 상대적 격자모형," 한국정보처리학회 춘계 논문집, 1999.
- [10] 이영일, 김창민, 김용기, "퍼지관계급 기반 평가 함수를 이용한 개선된 AUV의 항행 휴리스틱 탐색기법," 한국정보처리학회 춘계 논문집, 1999.
- [11] Kallgren, Laban, Real-time replanning of mission routes based upon threats, Master thesis, Linkoping University, 5th of January 2001.
- [12] Ong, Seow Meng, A Mission Planning Expert System with Three-Dimensional Path Optimization for the NPS Model 2 Autonomous Underwater Vehicle, Naval Postgraduate School, Monterey, California June 1990.
- [13] Lozano-Perez, T., and Wesley, M. A., "An Algorithm for Planning Collision Free Paths among Polyhedral Obstacles," Communications, v. ACM-22(10), pp. 560-570, 1979.