

# RRT\*를 활용하여 향상된 이종의 개미군집 기반 경로 계획 알고리즘

## Improved Heterogeneous-Ants-Based Path Planner using RRT\*

이준우<sup>†</sup>  
Joonwoo Lee<sup>†</sup>

**Abstract:** Path planning is an important problem to solve in robotics and there has been many related studies so far. In the previous research, we proposed the Heterogeneous-Ants-Based Path Planner (HAB-PP) for the global path planning of mobile robots. The conventional path planners using grid map had discrete state transitions that constrain the only movement of an agent to multiples of 45 degrees. The HAB-PP provided the smoother path using the heterogeneous ants unlike the conventional path planners based on Ant Colony Optimization (ACO) algorithm. The planner, however, has the problem that the optimization of the path once found is fast but it takes a lot of time to find the first path to the goal point. Also, the HAB-PP often falls into a local optimum solution. To solve these problems, this paper proposes an improved ant-inspired path planner using the Rapidly-exploring Random Tree-star (RRT\*). The key ideas are to use RRT\* as the characteristic of another heterogeneous ant and to share the information for the found path through the pheromone field. The comparative simulations with several scenarios verify the performance of the improved HAB-PP.

**Keywords:** Ant Colony Optimization, Heterogeneous Ants, Rapidly-exploring Random Tree, Path Planning

### 1. 서론

이동 로봇은 최근 다양한 분야에서 그 수요가 급증하였다. 또한 자율주행에 대한 사회적 관심이 높아지고 있어 그 수요는 더욱 더 증가할 것이다. 이에 맞춰 이동 로봇과 자율주행 분야에서 가장 중요하고 활발하게 연구되어 왔고, 연구 중은 부문이 경로 계획(Path Planning)이다. 경로 계획은 특정한 공간에서 움직이는 이동 로봇이 출발점에서 도착점까지 안전하면서도 사용자의 여러 가지 요구 조건들을 충족시키는 최적 경로(Optimal Path)를 찾는 문제를 푸는 것이다.

경로 계획은 크게 전역 경로 계획(Global Path Planning)<sup>[1-4]</sup>과 지역 경로 계획(Local Path Planning)<sup>[5-7]</sup>으로 구분한다. 전역 경로 계획의 주목적은 출발점에서 도착점까지 이동 로봇이 주행 가능한 최적의 경로를 찾는 것이다. 따라서 전역 경로 계획

에서는 지역 경로 계획과는 달리, 경로 계산 시간보다는 경로의 질을 더 우선순위에 두고 탐색을 하는 경우가 흔하다. 반면, 지역 경로 계획은 이동 로봇에 탑재된 센서들(예를 들면, 초음파, 적외선, 레이더, 라이다, 카메라 등)을 통해 실시간으로 획득한 정보를 바탕으로, 갑자기 나타난 장애물을 회피하는 경로를 빠른 시간 내에 찾아내는 것을 그 목표로 한다. 본 논문은 개미 군집 최적화(Ant Colony Optimization, ACO) 알고리즘에 기반한 전역 경로 계획 방법을 제안한다.

전역 경로 계획 문제를 풀기 위해서는 이동 로봇이 속하여 이동하려고 하는 주변환경을 이해하기 쉽게 표현해야 한다. 일반적으로는 4가지 유형의 지도 표현 방법<sup>[8]</sup>이 존재하지만, 그 중에서도 공간을 사각 셀로써 표현하는 composite-space map 방법이 가장 널리 사용된다. 이는 분할한 각각의 셀들이 장애물과 비장애물을 나타냄으로써 보다 효율적으로 지도를 활용할 수 있다. 많은 전역 경로 계획 알고리즘들이 이 방법을 활용한다; A-star (A\*) 탐색 알고리즘<sup>[9-11]</sup>, 유전자 알고리즘(Genetic Algorithms, GAs)<sup>[12-15]</sup>, 개미 군집 최적화(ACO) 알고리즘<sup>[16-19]</sup> 등. 하지만, 이 방법을 활용한 알고리즘들은 이산(discrete) 상태

Received : Sep. 11. 2019; Revised : Oct. 10. 2019; Accepted : Oct. 13. 2019

※ This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2017 R1C1B5017992)

† Assistant Professor, Corresponding author: Electrical Engineering, Kyungpook National University, Daegu, Korea (jwl@knu.ac.kr)

이동을 사용하기 때문에, 찾은 경로를 통한 이동에 있어서, [Fig. 1]과 같이, 45도의 배수의 각도로만의 이동으로 제약 조건이 발생하게 된다. 이는 진정한 최단 경로보다는 더 긴 근접 최적 경로(Near-optimal Path)이다. 또한, 이러한 경로를 따라 이동하는 이동 로봇은 부드러운 움직임에 제약을 받으며, 경우에 따라서는 잘못된 방향 전환이 발생하여 효율적인 에너지 소비 측면에서 손해를 입게 된다.

이러한 단점을 극복하고 보다 부드러운 최적 경로 생성을 위해, 우리는 지난 연구에서 이종의 개미 군집 기반 경로 계획 (Heterogeneous-Ants-Based Path Planner) 알고리즘을 제안하였다<sup>[19]</sup>. HAB-PP는 대표적으로 개미의 종의 다양화(Diversification of Ant Species), 경로 교차(Path Crossover), 동적 시야 확장 원리(Dynamic Sight Expansion Principle) 기술들을 기반으로, HAB-PP의 성능 향상에 도움이 될 수 있는 다른 기술들을 추가함으로써, 잘못된 방향 전환은 줄이고, 보다 최적 경로에 가까우며, 후처리 작업없이도 부드러운 최적 경로를 생성하였다. 이는 몇 가지 전역 경로 계획 문제들을 가지고, A\* 탐색 알고리즘과 같은 다른 대표적인 전역 경로 계획 알고리즘들과 비교를 통해 그 성능을 입증하였다.

HAB-PP는 알고리즘이 시작된 이후, 일단 첫번째 경로를 찾지만 한다면, 그 경로 정보를 바탕으로 경로 최적화 작업은 우수한 성능으로 실행하지만, 그 첫번째 경로를 찾는데 다소 많은 시간이 소요된다는 단점을 가지고 있다. 물론, 전역 경로 계획에서는 앞선 언급한 바와 같이, 경로를 찾는 시간보다는 경로의 질이 더 중요한 고려사항이기는 하다. 하지만, 지도의 크기가 커지거나 지도 내부에 장애물이 복잡하게 다수 존재하는 경우에는 첫번째 경로 탐색의 시간이 그 중요성을 가질 수도 있다. 또한, HAB-PP는 앞서 언급한 경우와 같이, 지도의 크기가 크거나 복잡한 장애물이 다수 존재하는 경우, 다른 전역 경로 계획 알고리즘이 가지는 문제점 중 하나인 국소 최적해 (Local Optimal Solution)에 빠지는 일이 종종 발생한다. 이러한 문제점을 해결하기 위해 이번 연구에서는 Rapidly-exploring Random Tree-star (RRT\*)를 활용하고자 한다. 자세한 내용은 이후 4장에서 언급한다.

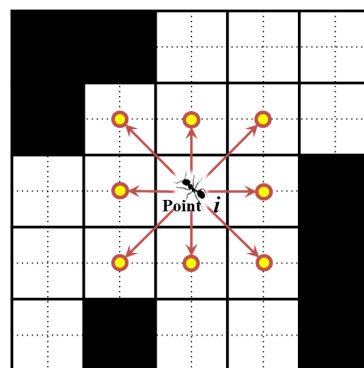
Rapidly-exploring Random Tree (RRT)<sup>[20]</sup>는 탐색 영역 전체에 대해 임의의 위치에 점을 생성하고, 출발점부터 트리를 성장시켜 나가면서 도착점까지 신속하게 경로를 생성하는 알고리즘이다. RRT는 국소 최적해에 빠지지 않으면서도 고차원 환경에도 적용이 가능한 샘플링 기반 알고리즘으로 그 활용성이 높다. 하지만, 한번 생성된 트리를 수정하는 과정이 없기 때문에, 경로의 최단거리에 대한 보장이 되지 않는다. 이를 개선하여 제안한 알고리즘이 바로 RRT\*<sup>[21]</sup>이다. 탐색 영역 내에 생성한 임의의 점이 이미 트리를 구성하고 있는 노드를 비용을 줄이면서 대체가능한 경우, 기존 노드를 대신한다. 이러한 과

정을 재귀적으로 반복하면서 최적 경로를 찾게 된다. RRT\*에 대해서도 많은 파생 알고리즘들이 존재하지만, 본 연구에서는 지난 연구에서 제안한 HAB-PP의 성능 개선에 RRT\*를 활용하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 지난 연구에서 제안한 HAB-PP에 대해 특징들을 중심으로 간략하게 설명한다. 3장에서는 본 연구에서 활용할 RRT\*에 대해 간략하게 살펴보고, 4장에서는 본 연구에서 제안하는 RRT\*를 활용하여 향상된 HAB-PP 알고리즘에 대해 설명한다. 5장에서는 몇 가지 전역 경로 계획 문제를 풀어 기존 HAB-PP로부터 성능 개선 여부를 확인한다. 마지막으로 6장에서는 본 논문의 결론을 맺는다.

## 2. 이종의 개미군집 기반 경로 계획 알고리즘(HAB-PP)

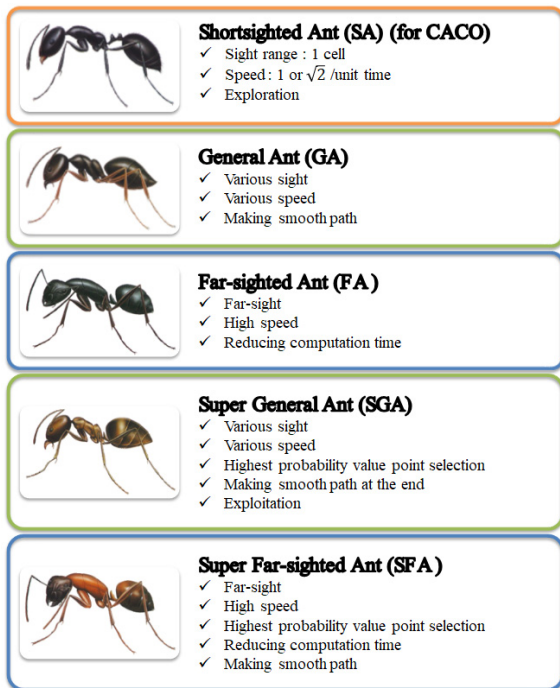
지난 연구에서 제안했던 HAB-PP가 기존의 ACO 알고리즘을 활용한 전역 경로 계획 방법과의 가장 큰 차이점이라 한다면, 이종의 개미군집(Heterogeneous Ants)을 활용했다는 점이다. 기존의 ACO 기반 알고리즘들에서는 개미가 이동시, [Fig. 1]에서와 같이, 현 위치 주변 8개의 점(장애물이 존재하지 않을 시) 중에서만 확률적 선택에 의해 한번에 한 칸씩만 이동할 수 있었다. 하지만, HAB-PP에서는 [Fig. 2]에서 보는 것과 같이, 한번에 다양한 거리를 여러가지 특징을 가지고 이동 가능한 것을 확인할 수 있다. 이를 통해, 계산 시간(Computation time)을 줄이고, 좀 더 부드럽고 잘못된 방향 전환이 줄어드는 직선 경로를 얻을 수 있었다. 물론 이를 위해 단순히 개미의 종류만 늘린 것은 아니다. 이를 실현하기 위해서는 [Table 1]에 보인 바와 같이, 경로 교차(Path Crossover), 동적 시야 확장 원리(Dynamic Sight Expansion Principle) 기술 등 다양한 기술들의 개발 및 도입이 필요했고, 이들 간에 유기적으로 동작해야 비로소 HAB-PP가 제대로 된 성능을 발휘할 수 있었다.



[Fig. 1] Example of movement constrained to multiples of 45 degrees (black cell is obstacle)

[Table 1] Characteristics of HAB-PP<sup>[19]</sup>

Characteristic	Scheme	Expected Effect
Modified Transition Probability Function (MTPF)	Factor Normalization	Instinctive parameter value selection
	Modified Visibility Field (MVF)	Reduce computation time, Consideration of robot size
	Added Angle Change Factor	Straight-line path
Modified Pheromone Update Rule (MPUR)	Change of Timing for Pheromone Update	Fast convergence
	Limit of Cumulative Pheromone Quantity	Path diversity and exploration
	Path Crossover (PC)	Fast convergence, Additional path obtained without search
	Normalized Pheromone Trail Value	Instinctive parameter value selection
Heterogeneous Ants (HA)	Various Species of Ants	Reduce computation time, Straight-line path
	Dynamic Sight Expansion Principle	Straight-line path, Support path crossover scheme
	Pheromone Interpolation between Points	Support path crossover scheme



[Fig. 2] Heterogeneous Ants for HAB-PP

[Table 2]는 HAB-PP의 pseudocode를 나타낸 것이다. 먼저, 격자지도(Grid map)로부터 출발점, 도착점, 장애물의 위치와 같은 정보를 추출한 뒤, 사용자 정의 매개변수(parameter)들에 값을 설정한다. 여기서  $N_{SA}$ ,  $N_{GA}$ ,  $N_{SGA}$ ,  $N_{FA}$ ,  $N_{SFA}$ 는 각각 이종의 개미들의 수를 나타내며,  $N_p$ 는 경로 교차에 이용할 경로의 수를,  $N_R$ 은 지금까지 가장 좋은 성능을 낸 경로들을 저장할 공간의 크기를 나타낸다. 매개변수  $\alpha, \beta, \rho$ 는 ACO 알고리즘이 가진 고유 변수들이며,  $\gamma$ 는 추가한 각도 변화에 대한 가중치를 결정하는 매개변수이다. 매개변수  $k_{mix}$ 는 수정된 시야 정보 저장소(Modified Visibility Field, MVF)의 가중치 조절 매개변수이고,  $d_0$ 는 시야 정보 저장소에서 장애물의 영향의 정도를 결

[Table 2] Pseudocode of HAB-PP<sup>[19]</sup>

**Algorithm 1** HAB-PP for Global Path Planning

```

1: GetInformation(Map);
2: SetParameters( $N_{SA}, N_{GA}, N_{FA}, N_{SGA}, N_{SFA}, N_p, N_R$ );
3: SetParameters( $\alpha, \beta, \gamma, \rho, k_{mix}, d_0, w_L, w_A, w_P$ );
4: Initialize( $\bar{\tau}, \bar{\eta}, \mathbb{I}_A, \mathbb{L}_R$ );
5:  $n_I \leftarrow 0$ ; ▷  $n_I$ : Iteration Number
6: while Termination condition doesn't meet do
7:    $n_I \leftarrow n_I + 1$ ;
8:    $n_R \leftarrow 0$ ; ▷  $n_R$ : Rank List Number
9:   for  $k = 1$  to  $N_A$  do ▷  $N_A$ : # of Total Ants
10:    // Ant  $k$ 's travel start.
11:     $j \leftarrow \text{RouletteSelect}(\bar{p}^k)$ ;
12:     $\mathbb{I}_{A,k} \leftarrow j$ ;
13:    PathPruning( $j$ );
14:    if  $j = g$  then ▷  $g$ : Goal Point
15:       $n_R \leftarrow n_R + 1$ ;
16:       $\mathbb{L}_R(N_R + n_R) \leftarrow \mathbb{I}_{A,k}$ ;
17:      Initialize( $\mathbb{I}_{A,k}$ );
18:    end if
19:  end for
20:  if Arrived ant(s) exist then ▷  $n_R > 0$ 
21:    PathCrossover( $\mathbb{L}_R(1), \dots, \mathbb{L}_R(N_p), \mathbb{L}_R(N_R + 1), \dots, \mathbb{L}_R(N_R + n_R)$ );
22:  end if
23:  // Sort and Cut  $\mathbb{L}_R$  by  $N_R$ .
24:  Sort( $\mathbb{L}_R(1), \dots, \mathbb{L}_R(N_R + n_R), \dots, \mathbb{L}_R(N_R + n_R + N_p)$ );
25:   $\mathbb{L}_R \leftarrow \text{Cut}(\mathbb{L}_R(1), \dots, \mathbb{L}_R(N_R))$ ;
26:  // Update pheromone.
27:  UpdatePheromone( $\tau$ );
28: end while
29: return  $\mathbb{L}_R(1)$ ;

```

정하는 매개변수이다. 마지막으로, 매개변수  $w_L, w_A, w_P$ 는 HAB-PP가 찾은 경로를 평가하는 평가함수에서 각 항목별 가중치를 결정하는 매개변수이다. 매개변수 선정이 모두 완료되면, 다음에는 시야 정보 저장소(Visibility Field) 값을 계산하고, 이를 행렬  $\tilde{\eta}$ 에 저장한다. 페르몬 정보 저장소(Pheromone Field)의 값은 행렬  $\tilde{\tau}$ 에  $\tilde{\eta}$ 의 값으로 초기화한다. 개미들의 이동 정보를 저장하는 배열  $\mathbb{I}_A$ 는 출발점의 위치 정보로 초기화하고, 개미들이 찾은 경로를 순위별로 저장할 배열  $\mathbb{L}_R$ 을 선언하면 모든 사전 작업이 완료되고, 본격적인 개미들의 여정이 시작된다.

총  $N_A$ 마리의 개미들이 매번 반복마다 한 스텝씩 이동을 하며, 이러한 이동에는 다음 점을 선택함에 있어, 다음 식 (1)의 확률에 기반한 룰렛 휠 선택(Roulette wheel selection)방식을

따라 선택하게 된다. 식 (1)은 점  $i$ 에서 점  $j$ 로 이동시  $k$ 번째 개미가 사용하는 수정된 이동 확률 함수(Modified Transition Probability Function, MTPF)이다. 이는 기존의 ACO 기반 알고리즘과는 다르게 HAB-PP를 위해 수정되었다.

$$p_j^k(n_I) = \begin{cases} \frac{[\tilde{\tau}_j(n_I)]^\alpha [\tilde{\eta}_{ij}]^\beta [\tilde{\xi}_j]^\gamma}{\sum_{c \in \tilde{C}} [\tilde{\tau}_c(n_I)]^\alpha [\tilde{\eta}_{ic}]^\beta [\tilde{\xi}_c]^\gamma}, & \text{if } j \in \tilde{C} \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

여기서,  $\tilde{C} = \{c = (\text{points in sight} - \text{window around point } i)\}$ 으로 *sight-windows*는 동적 시야 확장 원리에 의해 결정된다<sup>[9]</sup>. 변수  $\tilde{\xi}_j$ 는 각도 변화에 대한 정보를 담고 있으며, 방향 전환이 적을수록 선택될 확률이 높아지도록 설계되어 있다. 만약  $k$ 번째 개미가 이번 이동에서 도착점에 도착한다면, 이  $k$ 번째 개미의 여정 경로 정보는 배열  $L_R$ 에 저장되며,  $k$ 번째 개미는 모든 정보가 초기화 되고, 다시 출발점에서부터 새로운 여정을 시작한다.

$N_A$  마리의 모든 개미가 한 스텝씩 이동을 완료한 후에는, 만약 한 마리라도 이번 이동에서 도착한 개미가 존재한다면, 경로 교차를 진행한다. 경로 교차는 유전자 알고리즘(GA)의 교배(Crossover)의 아이디어에 착안하여, 경로를 새로이 탐색하지 않고도 기존의 개미들이 찾은 양질의 경로들로부터 새로운 경로를 만들어내는 과정이다.

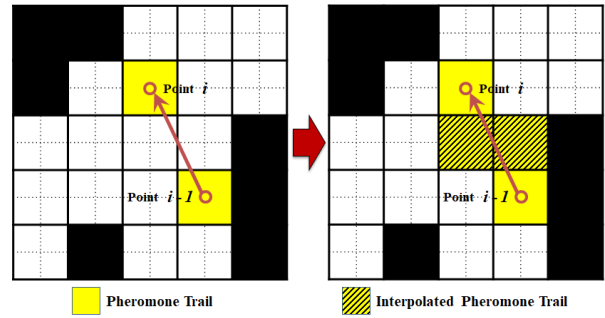
이러한 경로 교차 과정까지 완료하면, 개미들이 찾은 경로와 경로 교차로 새롭게 만들어진 경로, 모두를 배열  $L_R$ 에 넣고는 다음의 평가 함수에 따라 평가하여, 순위를 정한 후, 순위 밖, 즉, 미리 정한 배열  $L_R$ 의 크기 안에 들지 못한 경로는 버린다. 평가 함수(Evaluation Function)는 다음 식 (2)와 같다.

$$f_{Eval}(n_R) = w_L \cdot \frac{f_L(1)}{f_L(n_R)} + w_A \cdot \frac{f_A(1)}{f_A(n_R)} + w_P \cdot \frac{f_P(1)}{f_P(n_R)} \quad (2)$$

$$\text{여기서, } \begin{cases} w_L + w_A + w_P = 1, \\ f_L(n_R) = \sum_{i,j \in L_R(n_R)} d_{ij}, \\ f_A(n_R) = \sum_{i \in L_R(n_R)} \theta_i, \\ f_P(n_R) = \sum_{i \in L_R(n_R)} 1, \end{cases}$$

아래 첨자  $L, A, P$ 는 각각 경로 길이, 경로 상의 각도 변화량, 경로를 구성하는 점의 수와 관련된 것들임을 나타내며, 이들 각각의 값은 정규화(Normalization)이 이루어지고, 그 중요도를 사용자가 선택할 수 있도록 각각의 요소에 가중치가 곱해져 그 중요도를 사전에 설정할 수 있도록 하였다.

모든 순위가 결정이 되고 나면, 이제 이 정보를 바탕으로 페



[Fig. 3] Pheromone interpolation between points

르몬 정보 저장소의 페르몬 양을 업데이트 한다. 전체 페르몬 정보 저장소 중,  $n_I + 1$  번째 반복 때 사용될 점  $j$ 의 페르몬의 양  $\tilde{\tau}_j$ 은 다음 식으로 결정한다.

$$\tilde{\tau}_j(n_I + 1) = (1 - \rho) \cdot \tilde{\tau}_j(n_I) + \Delta \tilde{\tau}_j \quad (3)$$

여기서,  $\rho$ 는 (0, 1)사이의 값을 갖는 페르몬 감쇠 매개변수이다. 또한,

$$\Delta \tilde{\tau}_j = \sum_{n_R=1}^{N_R} \Delta \tilde{\tau}_j^{n_R} \quad (4)$$

여기서,  $\Delta \tilde{\tau}_j^{n_R}$ 는  $n_R$ 번째 순위 경로의 점  $j$ 에 놓인 페르몬의 양이다. 여기까지는 기존의 ACO 기반 전역 경로 계획 알고리즘들과 유사하나, 제안한 HAB-PP의 경우, 개미가 움직이는 한 스텝의 크기가 한 칸 이상인 경우도 존재하기 때문에, [Fig. 3]과 같은 페르몬 보간법이 필요하다. 즉, 찾은 경로 상에서는 점과 점 사이가 비어있지만, 페르몬 정보를 업데이트 할 때는 두 점을 연결해서 닿는 모든 격자에 양 끝점에 쌓을 페르몬의 양과 동일한 양의 페르몬의 양으로 업데이트 한다.

이상의 과정이 종료 조건을 만족할 때까지 반복된다. HAB-PP의 경우, 경로 순위 정보를 담은 배열  $L_R$ 의 내용이 변화가 없을 때를 종료 조건으로 하고 있다.

### 3. RRT-star (RRT\*)

Rapidly-exploring Random Tree (RRT)<sup>[20]</sup>는 탐색 영역 전체에 대해 임의의 위치에 점을 생성하고, 출발점부터 트리를 성장시켜 나가면서 도착점까지 신속하게 경로를 생성하는 알고리즘이다. RRT는 국소 최적해에 빠지지 않으면서도 고차원 환경에도 적용이 가능한 샘플링 기반 알고리즘으로 그 활용성이 높다. 하지만, 한번 생성된 트리를 수정하는 과정이 없기 때문에, 경로의 최단거리에 대한 보장이 되지 않는다. 이를 개선하여 제안한 알고리즘이 바로 RRT\*<sup>[21]</sup>이다. 탐색 영역 내에 생성한 임의의 점이 이미 트리를 구성하고 있는 노드를 비용



[Table 3] Pseudocode of RRT\*[21]

**Algorithm 2** RRT\* for Global Path Planning

```

1:  $\mathbb{V} \leftarrow \{x_{start}\};$ 
2:  $\mathbb{E} \leftarrow 0;$ 
3: SetParameters(N);
4: for  $i = 1$  to  $N$  do  $\triangleright N_A: \# \text{ of Iterations}$ 
5:    $x_{rand} \leftarrow \text{Sampling}(i);$ 
6:    $x_{nearest} \leftarrow \text{Nearest}(x_{rand}, \mathbb{V});$ 
7:    $x_{new} \leftarrow \text{Steer}(x_{nearest}, x_{rand});$ 
8:   if  $\text{CollisionFree}(x_{nearest}, x_{rand})$  then
9:      $x_{min} \leftarrow x_{nearest};$ 
10:     $c_{min} \leftarrow \text{Cost}(x_{nearest}, \mathbb{V}) + \text{Cost}(x_{nearest}, x_{new});$ 
11:     $x_{near} \leftarrow \text{Near}(x_{new}, \mathbb{V});$ 
12:    for all  $x_{near} \in X_{near}$  do  $\triangleright \text{Choose parent}$ 
13:      if  $\text{CollisionFree}(x_{near}, x_{new})$  then
14:         $c_{new} \leftarrow \text{Cost}(x_{near}, \mathbb{V}) + \text{Cost}(x_{near}, x_{new});$ 
15:        if  $c_{new} < c_{min}$  then
16:           $x_{min} \leftarrow x_{near};$ 
17:           $c_{min} \leftarrow c_{new};$ 
18:        end if
19:      end if
20:    end for
21:     $\mathbb{V} \leftarrow \mathbb{V} \cup \{x_{new}\};$ 
22:     $\mathbb{E} \leftarrow \mathbb{E} \cup \{(x_{min}, x_{new})\};$ 
23:    for all  $x_{near} \in X_{near} \setminus \{x_{min}\}$  do  $\triangleright \text{Rewire near nodes}$ 
24:      if  $\text{CollisionFree}(x_{near}, x_{new})$  then
25:         $c_{near} \leftarrow \text{Cost}(x_{near}, \mathbb{V}) + \text{Cost}(x_{near}, x_{new});$ 
26:        if  $c_{near} < \text{Cost}(x_{near}, \mathbb{V})$  then
27:           $x_{parent} \leftarrow \text{Parent}(x_{near});$ 
28:           $\mathbb{E} \leftarrow \mathbb{E} \setminus \{(x_{parent}, x_{near})\};$ 
29:           $\mathbb{E} \leftarrow \mathbb{E} \cup \{(x_{new}, x_{near})\};$ 
30:        end if
31:      end if
32:    end for
33:  end if
34: end for
35: return  $\mathbb{V}, \mathbb{E};$ 

```

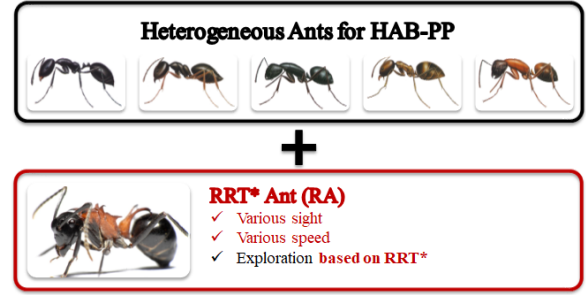
줄이면서 대체가능한 경우, 기존 노드를 대신한다. 이러한 과정을 재귀적으로 반복하면서 최적 경로를 찾게 된다.

[Table 3]은 일반적인 RRT\*의 pseudocode이다. Algorithm 2의 줄 22까지는 일반 RRT와 동일하지만, 그 다음 과정에서 일정 반경 내의 주변 노드들 중에서 비용을 줄일 수 있는 기존 노드가 존재한다면 이를 임의의 새로운 노드가 대체하는 방식을 반복적으로 수행하면서 최적의 경로를 찾는다라는 점에서 다르다.

#### 4. RRT\*를 활용하여 향상된 HAB-PP

이번 장에서는 이전에 제안했던 HAB-PP의 성능 향상을 위해 RRT\*를 활용하는 방법에 대해 설명한다. 주안점은 HAB-PP가 가진 기존의 성능을 훼손하지 않는 것이다. 따라서 본 논문에서는 RRT\*를 활용한 성능 향상 방법으로, [Fig. 4]과 같이, RRT\*의 탐색 방법을 기반으로 한 개미의 종을 하나 새롭게 추가하는 것이 주된 아이디어이다. 이는 단순한 종의 추가로서 완료되는 것이 아니라, RRT\* 기반의 개미들이 찾은 경로의 정보를 기존의 HAB-PP의 개미들이 활용하고, 반대로, HAB-PP가 찾은 경로의 정보를 RRT\* 기반의 개미들도 활용할 수 있어야 한다. 이를 실현하기 위해, 앞선 연구에서 사용한 페르몬 정보 저장소를 통한 정보 공유 및 교환 방법을 제안한다.

먼저, RRT\*의 트리 전개는 원래의 HAB-PP에서 이종의 개미들이 한 스텝을 이동하는 주기를 따라 함께 새로운 노드를 하



[Fig. 4] Heterogeneous Ants for Improved HAB-PP using RRT\*

나씩 트리 전개하는 것으로 하였다. 이 때 새로운 노드를 선정하는 과정에서, 기존의 RRT\*는 랜덤 생성을 했지만, 향상된 HAB-PP에서는 페르몬 정보 저장소의 정보에 기반한 룰렛 휠 방식의 확률적 선택을 통해 선정하도록 하였다. RRT\*의 트리 구성에 있어, 단독으로 시행한 알고리즘 보다 이동 간격을 크게 했으며, 또한 보다 적은 반복 횟수를 통해 덜 최적화되지만 빠른 수렴이 가능하도록 매개변수 값들을 조정했다. 이는 향상된 HAB-PP에서 RRT\*가 이루어야 할 빠른 수렴이라는 역할에 충실하도록 하기 위한 조치이며, 경로를 최적화하는 작업은 보다 나은 성능을 가지는 기존의 HAB-PP가 담당하도록 하였다.

또한, 기존의 HAB-PP가 이산적 격자 지도를 활용하고 있기에 향상된 HAB-PP에서 RRT\*의 활용에 있어, 기존의 RRT\*의 연속적인 실수 좌표가 아닌 이산적인 정수 좌표를 활용할 수 있도록 하였다.

RRT\*와 HAB-PP 간의 서로 찾은 경로 간의 정보 공유 및 교환을 위해서는 기존의 HAB-PP가 활용하던 페르몬 정보 저장소를 통해 서로 간 정보 공유를 하도록 하였다. RRT\*의 경우도 결국은 HAB-PP 내의 일반 개미(General Ant, GA)와 원시 개미(Far-sighted Ant, FA)처럼 찾은 경로에 대한 노드 정보만을 가지고 있기 때문에, 앞서 HAB-PP에서 경로 정보 공유를 위해 활용하던 [Fig. 3]의 페르몬 보간법을 이용하면 어렵지 않게 이를 구현할 수 있고, 또 향상된 HAB-PP에서 활용할 수 있다.

#### 5. 실험 결과

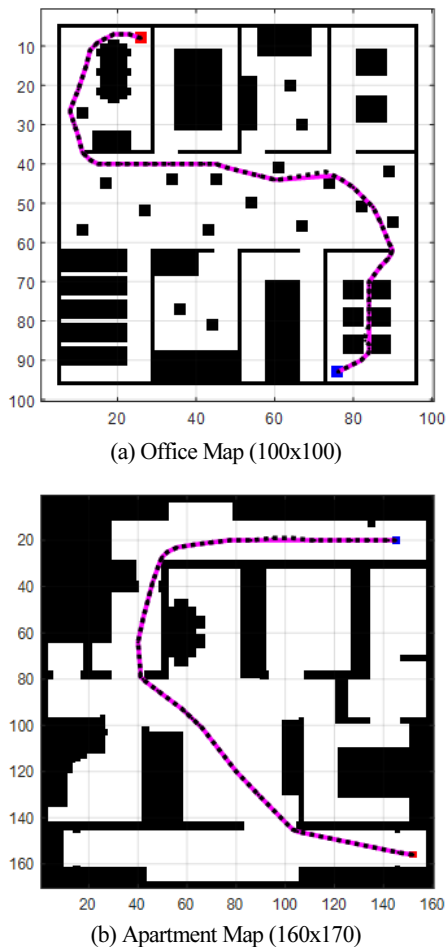
이번 장에서는 RRT\*를 활용하여 HAB-PP의 첫 경로 탐색 시간 축소 및 종종 국소 최소해에 빠지는 현상 탈피 등의 성능 향상이 이루어졌는지에 대한 성능 평가를 한다.

실험 환경은 다음과 같다. Intel Core i5 CPU 760 @ 2.80 GHz 에 4 GB 램이 장착된 PC에서 모든 시뮬레이션은 MATLAB으로 구현하였고 실행하였다. 시뮬레이션 대상은 이전에 제안한 HAB-PP와 이번의 RRT\*를 활용하여 향상된 HAB-PP(이하, Improved HAB-PP로 표기), 이 두 가지 알고리즘에 대해 비교 실험을 수행하였다.

시뮬레이션에 사용된 전역 지도는 각각 실제 사무실(Office

Map)과 아파트(Apartment Map)의 평면도에서 추출한 지도를 사용하였다. 지도의 크기는 사무실 지도가 100x100과 아파트 지도가 160x170이며, 지도에서 과란 사각형은 출발점, 빨간 사각형은 도착점을 나타낸다. 경로계획을 위해 설정한 매개변수 값들은 다음과 같이 선정하고 시뮬레이션을 진행하였다.  $N_{SA} = 100$ ,  $N_{GA} = 30$ ,  $N_{EA} = 30$ ,  $N_{SCA} = 5$ ,  $N_{SEA} = 5$ ,  $N_{RA} = 5$ ,  $N_P = 5$ ,  $N_R = 30$ ,  $\alpha = 2$ ,  $\beta = 1$ ,  $\gamma = 1$ ,  $\rho = 0:1$ ,  $k_{mix} = 0.8$ ,  $d_0 = 2$ ,  $s_{RRT} = 30$ ,  $r_{RRT} = 30$  and  $w_L = w_A = w_P = 0.333$ . 시뮬레이션은 각 지도마다 30번을 수행하였으며, 결과 분석은 이 자료들을 가지고 통계적인 분석을 실시하였다.

먼저 [Fig. 5]는 각 전역 지도에 대해, HAB-PP와 Improved HAB-PP가 찾은 경로를 표시한 그림이다. 그림에서 분홍색 실선은 Improved HAB-PP가 찾은 경로이며, 검은색 점선은 HAB-PP가 찾은 경로이다. 표기한 경로는 30회 실험 중 최단 거리 경로이다. 두 알고리즘 모두 이산 격자 지도 정보를 활용하다 보니, 경로 길이는 매번 동일하거나 약간의 차이가 존재

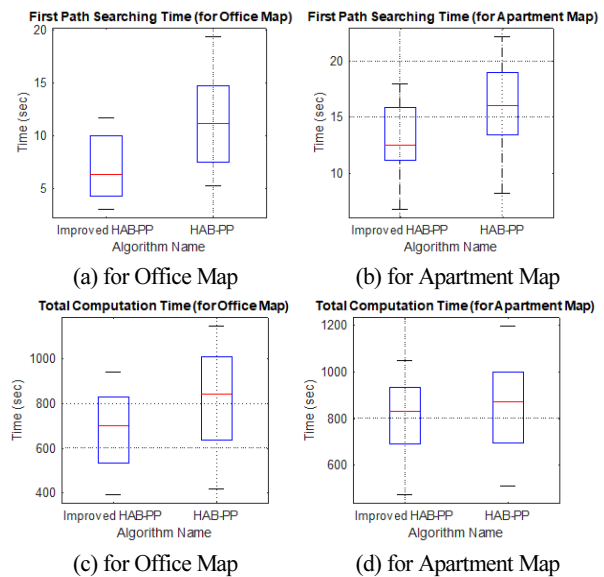


[Fig. 5] Simulation Results - Path Graph, HAB-PP (Black dot line), Improved HAB-PP (Pink solid line); (a) Office Map, (b) Apartment Map

할 뿐이라 통계적인 수치에는 비교에 있어 큰 의미가 없어 따로 표시하지 않았다. 다만, 각각의 최단거리는 Office Map의 경우, Improved HAB-PP가 169.9823, HAB-PP가 170.3482이고, Apartment Map의 경우, Improved HAB-PP가 291.1896이고, HAB-PP가 291.3073이었다. 통계적으로 유의미한 차이는 아니지만, 절대적인 수치나 그림의 경로를 보았을 때, Improved HAB-PP가 더 짧고 각도 변화량 측면에서도 유리한 경로를 생성하는 것을 확인할 수 있다.

다음으로, [Fig. 6]은 시간적인 측면의 실험 결과에 대해 통계적 수치 분석 결과를 한눈에 알아보기 좋도록 Box Plot으로 표현한 것이다. 본 논문에서 개선하고자 했던 첫 경로 탐색 시간과 국소 최소해 회피에 대한 성능 향상 부분을 확인할 수 있는 데이터이기도 하다.

먼저 [Fig. 6]의 상단의 두 그래프는 두 알고리즘의 각 지도에 대한 첫 경로 탐색 시간을 표현한 것이고, 하단의 두 그래프는 최적 경로를 찾는 데 걸린 총 계산 시간이다. 그래프에서 보여지는 바와 같이, RRT\*를 활용한 Improved HAB-PP가 더 빠른 시간에 첫번째 경로를 찾아내는 것을 확인할 수 있으며, 또 그로 인해 전체적인 경로 탐색 시간에서도 시간을 단축한 것을 확인할 수 있다. 수치적으로 표현하면, 첫 경로 탐색 시간에 있어서는, Office Map의 경우, 약 40.3%, Apartment Map의 경우, 약 18.5%의 개선 효과가 발생했다(각각 평균값 기준). 전체 경로 계산 시간에 있어서는, Office Map의 경우, 약 14.9%, Apartment Map의 경우, 약 6.0%가 개선되었다. 이로써 RRT\*의 활용이 HAB-PP의 첫 경로 탐색 시간 단축 및 전체적인 탐



[Fig. 6] Simulation Results - BoxPlots for First Path Searching Time (sec), Total Computation Time (sec); First Path Searching Time (a) for Office Map, (b) for Apartment Map; Total Computation Time (c) for Office Map, (d) for Apartment Map

색 시간 단축에 유의미한 결과를 가졌었다는 것을 확인할 수 있다. 아울러, 국소 최소해 회피의 문제는 직접적인 확인은 불가능하지만, 전체적인 계산 시간이 단축된 것이 국소 최소해에 머무는 시간이 줄어들었기 때문에 나타난 효과라고 추측해 볼 수 있다. 또한 또 다른 증거로 전체 탐색 시간에 있어, 최대 시간의 발생 횟수가 기존의 HAB-PP의 경우보다 현저하게 줄어든 것으로부터도 국소 최소해 회피가 이루어져, 성능이 향상되었음을 유추할 수 있다.

## 6. 결 론

본 논문은 앞선 연구에서 제안한 이중의 개미군집 기반 경로 계획(HAB-PP) 알고리즘의 개선점인 첫 경로 탐색 시간 단축, 국소 최소해 회피라는 성능 향상 목적을 달성하기 위해 RRT\*를 활용하는 방안에 대해 연구하고, 그 결과 성능 향상된 HAB-PP를 제안하였다. 그 방법론에 있어서는 기존의 HAB-PP가 가진 우수한 성능을 훼손하지 않도록 하기 위해, RRT\*의 탐색 방법에 기반에 경로 탐색을 실시하는 새로운 개미의 종을 하나 추가하였다. 이로써 기존 알고리즘의 구조적인 변화는 최소화하였다. 하지만, 두 알고리즘이 서로 탐색한 정보를 공유하고, 서로 활용하는 방법이 필요한데, 이는 기존의 ACO 기반 경로 계획 알고리즘들이 활용하는 페르몬을 매개로 한 정보 공유 방법을 활용하였다. 페르몬 정보 저장소를 통해 이중의 개미군집들이 도착점에 도착할 때마다 평가함수에 의해 평가받은 성적에 따라 페르몬 정보 저장소에 그에 해당하는 페르몬 자취를 남김으로써 기존의 이중의 개미군집 뿐만 아니라, 새로이 도입한 RRT\*기반 개미군집도 이 정보에 기반하여 다음 트리 생성점을 선택하도록 하였다.

그 결과, 기존의 HAB-PP 알고리즘 대비, 첫 경로 탐색 시간에 있어서 약 20~40%가량의 탐색 시간 단축 효과를 확인했으며, 전체적인 최적 경로 탐색 시간에 있어서도 6~15% 가량의 시간 단축 효과를 확인함으로써, RRT\*를 활용하여 향상된 HAB-PP의 성능을 검증하였다.

## References

- [1] G.-H. Kim, H. Kim, B.-S. Kim, and S.-G. Lee, "Path Planning for an Intelligent Robot Using Flow Networks," *Journal of Korea Robotics Society*, vol. 6, no. 3, pp. 255-262, Sept., 2011.
- [2] J.-H. Suh and S.-H. Oh, "A Cost-Aware RRT Planning Algorithm," *Journal of Korea Robotics Society*, vol. 7, no. 2, pp. 150-159, Jun., 2012.
- [3] C.-B. Noh, M.-H. Kim, and M.-C. Lee, "Path Planning for the Shortest Driving Time Considering UGV Driving Characteristic and Driving Time and Its Driving Algorithm," *Journal of Korea Robotics Society*, vol. 8, no. 1, pp. 43-50, Mar., 2013.
- [4] Y. Park and H. Ryu, "Improved Path Planning Algorithm based on Informed RRT\* using Gridmap Skeletonization," *Journal of Korea Robotics Society*, vol. 13, no. 2, pp. 142-149, Jun., 2018.
- [5] D.-J. Kang and D. G. Kim, "Optimal Path Planning of Mobile Robot for Multiple Moving Obstacles," *Journal of Korea Robotics Society*, vol. 2, no. 2, pp. 183-190, Jun., 2007.
- [6] S.-J. Kim, J.-W. Kang, and M.-J. Chung, "Efficient Coverage Path Planning and Path Following in Dynamic Environments," *Journal of Korea Robotics Society*, vol. 2, no. 4, pp. 304-309, Dec., 2007.
- [7] J.-Y. Moon, H.-W. Chae, and J.-B. Song, "Planning of Safe and Efficient Local Path based on Path Prediction Using a RGB-D Sensor," *Journal of Korea Robotics Society*, vol. 13, no. 2, pp. 121-128, Jun., 2018.
- [8] S. Bandi and D. Thalmann, "Space discretization for efficient human navigation," *Computer Graphic Forums*, vol. 17, no. 3, pp. 195-206, 1998.
- [9] P. Melchior, B. Orsoni, O. Lavielle, A. Poty, and A. Oustaloup, "Consideration of obstacle danger level in path planning using A\* and Fast-Marching optimization: comparative study," *Signal Processing*, vol. 83, no. 11, pp. 2387-2396, Nov., 2003.
- [10] J. Yao, C. Lin, X. Xie, A. J. Wang, and C.-C. Hung, "Path planning for virtual human motion using improved A\* algorithm," *2010 Seventh International Conference on Information Technology: New Generations*, Las Vegas, USA, pp. 1154-1158, 2010.
- [11] J.-H. Seok, J.-W. Lee, J.-H. Wang, J.-J. Lee, and H. J. Lee, "A temporal path planner for solving information inconsistency in an integrated path planner," *International Journal of Control, Automation and Systems*, vol. 11, no. 6, pp. 1232-1240, 2013.
- [12] T. W. Manikas, K. Ashenayi, and R. L. Wainwright, "Genetic Algorithms for Autonomous Robot Navigation," *IEEE Instrumentation and Measurement Magazine*, vol. 10, no. 6, pp. 26-31, Dec., 2007.
- [13] J. Zhao, L. Zhu, G. Liu, and Z. Han, "A modified genetic algorithm for global path planning of searching robot in mine disasters," *2009 International Conference on Mechatronics and Automation*, Changchun, China, pp. 4936-4940, 2009.
- [14] J. Berger, K. Jabeur, A. Boukhtouta, A. Guitouni, and A. Ghanmi, "A hybrid genetic algorithm for rescue path planning in uncertain adversarial environment," *IEEE Congress on Evolutionary Computation*, Barcelona, Spain, pp. 1-8, 2010.
- [15] S. C. Yun, V. Ganapathy, and L. O. Chong, "Improved Genetic Algorithms based Optimum Path Planning for Mobile Robot," *2010 11th International Conference on Control Automation Robotics & Vision*, Singapore, pp. 1565-1570, 2010.
- [16] D. Zhao and J. Yi, "Robot planning with artificial potential field guided ant colony optimization algorithm," *International Conference on Natural Computation*, Xian, China, pp. 222-231, 2006.
- [17] M. A. P. Garcia, O. Montiel, O. Castillo, R. Sepulveda, and P. Melin, "Path planning for autonomous mobile robot navigation with ant colony optimization and fuzzy cost function evaluation," *Applied Soft Computing*, vol. 9, no. 3, pp. 1102-1110, Jun., 2009.

- [18] K. Ioannidis, G. Ch. Sirakoulis, and I. Andreadis, "Cellular ants: A method to create collision free trajectories for a cooperative robot team," *Robotics and Autonomous Systems*, vol. 59, no. 2, pp. 113-127, Feb., 2011.
- [19] J. Lee, "Heterogeneous-ants-based Path Planner for Global Path Planning of Mobile Robot Applications," *International Journal of Control, Automation and Systems*, vol. 15, no. 4, pp. 1754-1769, Aug., 2017.
- [20] S. M. LaValle and J. J. Kuffner, "Randomized Kinodynamic Planning," *International Journal of Robotics Research*, vol. 20, no. 5, pp. 378-400, 2001.
- [21] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *International Journal of Robotics Research*, vol. 30, no. 7, pp. 846-894, 2011.



### 이준우

2007 부산대학교 전자전기통신공학전공(학사)

2009 KAIST 로봇공학학제전공(석사)

2014 KAIST 전기 및 전자공학과(박사)

2015~현재 경북대학교 전기공학과 조교수

관심분야: Swarm Intelligence, Evolutionary Computation, Swarm Robotics, Evolutionary Robotics, Intelligent Systems