

외판원 순회 문제 해결을 위한 효과적인 알고리즘

이유진, 정규용
선린인터넷고등학교

{proqk2, jeong000307}@gmail.com

Effective Algorithm for the Traveling Salesman Problem

Yujin Lee, Gyuyong Jeong
Sunrin Internet High School

요약

외판원 순회 문제(Traveling Salesman Problem, TSP)는 다항 시간내에 해가 구해지지 않는 NP-Complete 문제로 널리 알려져 있으며 합리적인 시간 내에 문제에 접근하는 다양한 휴리스틱 알고리즘이 많은 연구자들에 의해 제안되었다. 본 논문에서는 TSP 문제의 최적 해를 구하는 방법을 구하기 위해 여러 휴리스틱 알고리즘 중 개미 집단 시스템(Ant Colony System, ACS)을 고찰 및 기술했으며 실험을 통해 그 결과를 도출하였다.

1 서론

TSP는 한 도시에서 출발한 판매원이 다른 도시들을 빠짐없이 한 번씩만 방문한 뒤 다시 처음 도시로 돌아오는 최단 경로를 구하는 문제이다. 즉 n 개의 노드 집합과 거리가 주어지고, 모든 노드를 중복없이 방문하는 최소 거리를 구해야 한다. TSP 같은 경우엔 도시가 하나씩 늘수록 연산의 양이 기하급수적으로 증가한다. 이 때문에 완전 탐색이나 동적계획법(Dynamic Programming, DP)을 이용한 풀이 같은 경우에는 굉장히 오랜 시간이 걸리게 되어 TSP 해결에는 바람직하지 않다. 이에 본 논문에서는 TSP 해결을 위하여 ACS를 사용하기로 하였다.

2 개미 집단 시스템

2.1 기존의 개미 시스템

개미 시스템(Ant System, AS)은 1992년 Marco Dorigo가 창안해낸 휴리스틱 알고리즘이다. 모든 개미들을 탐색시키면서 각 개미의 페로몬을 업데이트 하고, 찾은 해를 비교하며 최적 경로를 찾는다. 하지만 기존 AS의 특성상 한 번의 반복이 끝나야 페로몬이 업데이트 되어 더 좋은 경로를 찾은 개미가 있더라도 그 개미의 선택이 다른 개미들에게 영향을 끼치지 않는다. 또 다음 도시로 이동할 때 확률적으로 경로를 선택하기 때문에 여행을 계속한다고 개미가 이전보다 더 나은 경로를 찾는다는 보장이 없다. 이같은 경우를 보완한 것이 ACS다.

2.2 개요

ACS는 개미가 다음 여행할 도시를 선택할 때 확률적으로 랜덤한 길을 선택함으로써 페로몬 값이 낮은 최적 경로도 발견할 수 있다. 그리고 개미가 이동할 때마다 페로몬을 업데이트 시켜 개미들 중 가장 좋은 경로로 이동한 개미가 다른 개미들에게도 영향을 미친다.

2.3 개미 여행 범위

개미 집단 시스템에서 사용하는 양방향 그래프는 노드(정점)와 노드와 노드를 잇는 엣지(간선)의 집합이다. 여기서는 노드를 도시, 엣지를 도시 간 거리라고 칭한다. 1번부터 N 번까지 번호가 매겨져 있는 도시들이 있고, 도시들 사이에는 길이 있을 수도 있고 없을 수도 있다. 있을 경우 각 도시를 이동하는데 드는 비용이 행렬 M_{ij} 형태로 주어지며 없을 경우 $M_{ij} = 0$ 이다.

2.4 개미 여행

개미는 먹이 수집 활동을 할 때 최적의 경로를 찾기 위해 시각뿐만 아니라 페로몬을 사용한다. ACS 알고리즘에선 노드 i 에 위치한 k 번째 개미는 다음과 같은 수식 (1)을 통하여 다음 노드 j 를 확정하게 된다.

$$j = \begin{cases} \tau_{iu}^{\alpha} \eta_{iu}^{\beta}, (u \in N_i), & (q \geq q_0) \\ P_{ij}, & (q < q_0) \end{cases} \quad (1)$$

(1)에서 N_i 는 i 에서 갈 수 있는, 즉 개미가 한번도 가본 적 없는 이동 가능한 노드의 집합이며 τ_{iu} 는 시간 t 일 때, 노드 i 와 u 를 연결하는 링크의 페로몬 양이다. η_{iu} 는 휴리스틱 함수로써, 일반적으로 노드 i 와 u 간의 거리의 역수로 정의한다. $\alpha, \beta (\alpha > 0, \beta > 0)$ 는 페로몬과 거리의 상대적 중요도를 결정하는 파라미터로써 α 가 페로몬, β 가 거리의 중요도를 결정하게 된다. q_0 는 $[0, 1]$ 사이의 노드 선택 확률을 결정하는 파라미터이며, q 는 $[0, 1]$ 사이의 무작위 파라미터이다.

만약 $q \geq q_0$ 라면 오직 선행적 정보에 의존하여 노드를 선택하지만, $q < q_0$ 라면 (2)의 확률 함수를 통하여 노드를 결정하게 된다.

$$P_{ij} = \frac{\tau_{ij}^{\alpha} \eta_{ij}^{\beta}}{\sum_{u \in N_i} \tau_{iu}^{\alpha} \eta_{iu}^{\beta}} \quad (2)$$

(2)의 확률 함수를 통해서는 페로몬이나 휴리스틱 값이 큰 노드를 선택할 가능성이 높다. 하지만 (1)와는 달리 다양한 경로를 탐

색할 수 있어 최적해를 찾는 데 도움이 된다.

2.5 페로몬 갱신

$$\tau_{ij} = (1 - \rho)\tau_{ij} + \rho\tau_{0ij} \quad (4)$$

(4)에서 ρ 는 $[0,1]$ 사이의 파라미터이며 증발률을 나타낸다. τ_{0ij} 은 일반적으로 노드 i, j 간의 거리와 총 노드 갯수의 곱을 역수로 취한 값을 사용한다.

(4)를 통하여 개미가 이동한 경로들의 페로몬을 전부 갱신 시키는 것은 AS와 동일하다. 하지만 ACS는 기존의 AS와는 달리 최단 경로를 찾은 개미만 특별히 전역 페로몬 갱신 규칙 (3)을 사용해 최단 경로에 페로몬을 증가시킨다.

$$\tau_{ij} = (1 - \rho)\tau_{ij} + \Delta\tau_{ij}^{best} \quad (3)$$

(3)에서 $\Delta\tau_{ij}^{best}$ 는 최단 경로에서 노드 i, j 간 거리의 역수를 나타낸다.

(3)를 통하여 최단 경로는 한 번 더 페로몬을 갱신시켜 페로몬의 양을 더욱 늘리게 되며 갱신된 최단 경로로 개미들을 더 많이 보내게 되는 것이다.

3 TSP 최적 알고리즘 탐색

개미 집단이 여행을 할수록 길이가 짧은 길은 페로몬이 더 많아지고 길이가 긴 길은 개미가 가지 않기 때문에 자연스럽게 도태된다. 이러한 과정에서 개미는 가장 짧은 경로를 찾게 되는데, 모든 노드를 한 번씩만 순회하는 개미 집단을 만든다면 ACS에서 찾은 가장 짧은 경로가 결국 TSP의 해가 된다.

4 실험 결과

ACS에서 파라미터 α 와 β 값에 따라 수렴 속도를 빠르게 하는 것에 중점을 둘 것인지, 좋은 해를 찾는 것에 중점을 둘 것인지가 결정이 된다. 여기서는 α 값이 높아지면 페로몬 양이 많은 길을 선택할 확률이 높아지며 β 값이 높아지면 짧은 길을 선택할 확률이 높아진다. TSP의 최적해를 구하는 시간에 대해 전체를 탐색하는 경우, Dynamic Programming을 사용한 경우, ACS에서 α 와 β 값과 ρ 을 서로 다르게 한 9가지 경우를 비교하면 표 1과 같다.

5 결론

본 연구에서는 TSP 문제의 최적 해를 구하는 방법으로 ACS가 적당한지 고찰하였다. 10개 도시 이하에서는 빠르지만 도시가 많아질수록 한계가 있는 Dynamic Programming이나 최적 해를 찾지 못하는 확률이 높은 AS에 비해 ACS는 많은 도시도 비교적 빠른 시간 안에 최적 해를 찾을 수 있었으며, 파라미터 값을 변경해 알고리즘을 다양화 할 것인지 강화할 것인지 선택이 가능하여 성능이 우수함을 알아내었다.

5.1 참고 문헌

옥승호, 안진호, 강성호, 문병인. (2010). 선호도 기반 최단 경로 탐색을 위한 휴리스틱 융합 알고리즘. 전자공학회논문지-TC, 47(8), 74-84.

이승관, 최진혁. (2011). 개미 집단 최적화에서 강화와 다양화의 조화. 한국콘텐츠학회논문지, 11(3), 100-107.

Dorigo, M., Maniezzo, V., & Colomi, A. (1996). Ant system: optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 26(1), 29-41.

알고리즘	α	β	ρ	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	Case 7	Case 8	Case 9	Case 10
완전탐색	-	-	-	0.1787	12.8760	0.0017	13.8424	13.4087	0.0009	-	-	-	-
DP	-	-	-	0.0026	0.0072	0.0007	0.0018	0.0021	0.0017	0.0969	0.2594	0.5429	71.5173
ACS	0.5	0.5	0.05	0.0786	0.4890	0.0330	0.1491	0.1602	0.0338	0.2920	0.3450	1.5160	8.8453
	3	0.5	0.05	0.1007	0.1079	0.0292	0.1086	0.1336	0.0265	0.2971	0.3004	1.3562	6.3512
	5	0.5	0.05	0.0750	0.1248	0.0324	0.1131	0.1203	0.0286	0.2665	0.3035	1.3479	17.6484
	0.5	1	0.05	0.0834	0.1229	0.0345	0.4429	0.1078	0.0392	0.2637	0.3048	1.3499	13.3496
	0.5	3	0.05	0.0756	0.1078	0.0309	0.2276	0.1261	0.0276	0.3464	0.3450	0.8365	6.1265
	0.5	5	0.05	0.1118	0.1056	0.0485	0.2505	0.1248	0.0763	0.3158	0.3100	1.4216	3.5385
	1	1	0.05	0.0703	0.1451	0.0343	0.2426	0.1332	0.0296	0.2653	0.3094	0.7277	87.8601
	1	1	1.5	0.0860	0.1273	0.0314	0.3411	0.1209	0.0276	0.2607	0.3156	1.0080	43.3649
	1	1	3	0.0694	0.0992	0.0319	0.3008	0.1093	0.0266	0.2485	0.2714	0.9207	171.1054

표 1: 완전탐색, DP, ACS에서의 TSP 실험 결과