Ant Colony Optimization

Algorithm and Approaches in Robot Path Planning

Katinka Böhm

Universität Hamburg
Fakultät für Mathematik, Informatik und Naturwissenschaften
Fachbereich Informatik
Technische Aspekte Multimodaler Systeme

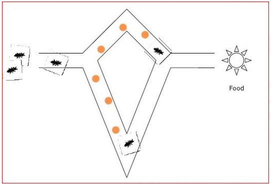
January 4th, 2016

Structure

Motivation

Natural Inspiration

→ based on the the behavior of ants seeking a path between their colony and a source of food



Stigmergy

Unorganized actions of individuals serve as a stimuli for other individuals by modifying their environment and result in a single outcome .

In short: A group of individuals that behave as a sole entity.

Motivation (contd.)

- Swarm Intelligence method
- probabilistic technique $\rightarrow$ non-deterministic
- solve hard combinatorial optimization problems

Definition

Combinatorial Optimization Problem $P = (S, \Omega, f)$
S ... finite set of decision variables,
Ω ... constraints,
f ... objective function to be minimized

Prominent example: Traveling Salesman

Metaheuristic

Ant Colony Optimization (ACO)

Set parameters
Initialize pheromone trails

while termination condition not met **do**

ConstructAntSolutions

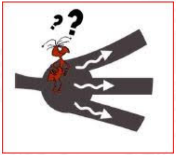
DaemonActions (optional)

UpdatePheromones

endwhile

Ant System (AS)

- oldest most basic algorithm
- by Marco Dorigo in the 90s



Ant Movement

Probability for ant k to move from i to j in the next step:

$$p_{ij}^{k} = \frac{\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}}{\sum_{\substack{\forall c_{il} \\ c_{il}\, feasible}} \tau_{il}^{\alpha} \cdot \eta_{il}^{\beta}}$$

where α and β control importance of pheromone τ vs. heuristic value η

Standard heuristic: $\eta_{ij} = \frac{1}{d_{ij}}$ where d_{ij} is the distance between i and j

Ant System (AS) (cont.)

Pheromone Update

Pheromone update for all ants that have built a solution in that iteration:

$$\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}^{k}$$

where ρ is the evaporation rate and $\Delta\tau_{ij}^{k}$ is the quantity of pheromone laid on edge (ij) with

$$\Delta\tau_{ij}^{k} = \frac{Q}{L_{k}}$$

where Q is a constant and L_k is the total length of the tour of ant k

Max-Min Ant System (MMAS)

- pheromone values are bound
- only the best ant updates its pheromone trails after solutions have been found

Pheromone Update

$$\tau_{ij} \leftarrow \left[(1-\rho)\cdot\tau_{ij} + \Delta\tau_{ij}^{best}\right]_{\tau_{min}}^{\tau_{max}}$$

where $\Delta\tau_{ij}^{best} = \frac{1}{L_{best}}$

L_{best} can be the iteration best or global best tour

Ant Colony System (ACS)

- diversify the search through a local pheromone update
- pseudorandom proportional rule for ant movement

Local Pheromone Update

Performed by all ants after each construction step to the last traversed edge

$$\tau_{ij} = (1 - \psi) \cdot \tau_{ij} + \psi \cdot \tau_0$$

where $\psi \in (0, 1]$ is the pheromone decay coefficient
and ψ_0 is the initial pheromone value

Pheromone Update

$$\tau_{ij} \leftarrow \begin{cases} (1 - \rho) \cdot \tau_{ij} + \rho \cdot \Delta\tau_{ij} & \text{if } (i, j) \text{ belongs to the best tour} \\ \tau_{ij} & \text{otherwise} \end{cases}$$

Theoretical Approach

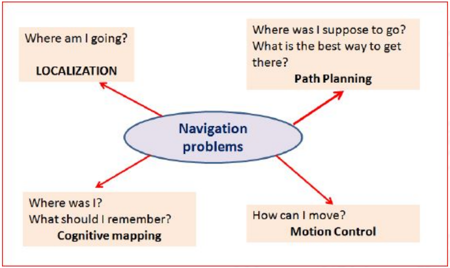
Overview

Algorithm	Ant Movement	Pheromones	Update Evaporation
Ant System (AS) 1991	random proportional	$\tau_{ij} \leftarrow (1-\rho)\cdot\tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}^{k}$	all paths
Max-Min Ant System (MMAX) 2000	random proportional	$\tau_{ij} \leftarrow \left[(1-\rho)\cdot\tau_{ij} + \Delta\tau_{ij}^{best}\right]_{\tau_{min}}^{\tau_{max}}$	best-so-far tour min/max bound
Ant Colony System (ACS) 1997	pseudorandom proportional	local: $\tau_{ij} = (1-\psi)\cdot\tau_{ij} + \psi\cdot\tau_0$ global: $\tau_{ij} \leftarrow \begin{cases}(1-\rho)\cdot\tau_{ij} + \rho\cdot\Delta\tau_{ij}\\ \tau_{ij}\end{cases}$	last step best-so-far tour

Problem Types

- Routing Problems
  → Traveling Salesman, Vehicle Routing, Network Routing
- Assignment Problems
  → Graph Coloring
- Subset Problems
  → Set Covering, Knapsack Problem
- Scheduling
  → Project Scheduling, Timetable Scheduling
- Constraint Satisfaction Problems
- Protein Folding

Robot Path Planning



- $\mathcal{NP}$-complete problem
- static vs. dynamic environment
- known vs. unknown environment
- rerouting on collision
- shortest path

Robot Path Planning Alg1

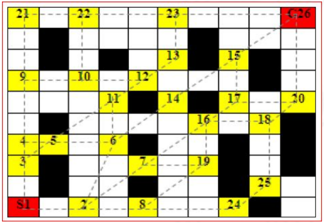
Mohamad Z. et al. [8]

Shortest Path in a **static environment**

Map Construction

Generate a global free space map where the robot can traverse between the yellow nodes

Free space nodes (white) can be traversed by the robot



Robot Path Planning Alg1

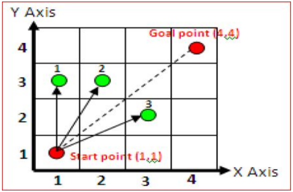
Mohamad Z. et al. [8]

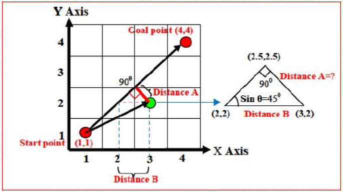
Ant Movement

Probability

$p_{ij} = \eta_{ij}^{\beta} \cdot \tau_{ij}^{\alpha}$ with $\alpha = 5, \ \beta = 5$

$$Heuristic \ \eta = \frac{1}{\text{distance between next point with intersect point at reference line}}$$





Robot Path Planning Alg1

Mohamad Z. et al. [8]

Pheromone Update

local $\rightarrow$ after each step from one node to another

global $\rightarrow$ after path calculation is finished

Local Evaporation

prevents accumulation of pheromone

$\tau_{ij} = (1 - \rho) \cdot \tau_{ij}$ with $\rho = 0.5$

Global Reinforcement (AS)

$\tau_{ij} = \tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}^{k}, \quad \Delta\tau_{ij} = \frac{Q}{L_k}$

where

Q ... number of nodes

L_k ... length of path chosen by ant k

Robot Path Planning Alg1

Mohamad Z. et al. [8]

Results

- comparison to a standard GA algorithm
- ACO faster with smaller number of iterations (due to good state transition rule - distance to baseline)

No of run	*Optimal path*	*Distance*	*Time(sec)*	*Iteration*
1	1.2.6.14.15.26	13.6476	13.3536	3
2	1.2.6.14.15.26	13.6476	18.7286	4
3	1.2.6.14.15.26	13.6476	10.0510	3
4	1.2.6.14.15.26	13.6476	8.4564	2
5	1.2.6.14.15.26	13.6476	23.6816	5
6	1.2.6.14.15.26	13.6476	18.5721	4
7	1.2.6.14.15.26	13.6476	8.9377	2
8	1.2.6.14.15.26	13.6476	18.4917	4
9	1.2.6.14.15.26	13.6476	22.0616	5
10	1.2.6.14.15.26	13.6476	11.7273	3
Total Average			**15.4062**	**3.5**

RPP algorithms		*GA*		*ACO*	
No of run	*Optimal path & path cost*	*Time*	*Iteration*	*Time*	*Iteration*
1	1.2.6.14.15.26 (13.6476 cm)	111.838	10	104.606	4
2		147.958	7	44.4	4
3		114.362	8	73.552	6
4		310.464	7	43.635	4
5		101.278	8	49.297	4
Total Average		**157.18**	**8**	**63.098**	**4.4**

Robot Path Planning Alg2

Michael Brand et al. [2]

Shortest path in a **dynamic environment**

- grid world of 20x20, 30x30 and 40x40
  four possible movement directions: left, right, up, down
- basic AS approach
- re-routing after obstacles are added
- focus on re-initialization of pheromones

Global Initialization

$\tau_{ij} = 0.1$ for every transition between blocks

Local Initialization

Gradient of pheromones around every object
Pheromone levels are decreased in a cyclic fashion by a certain fraction (50%)

Robot Path Planning Alg2

Michael Brand et al. [2]

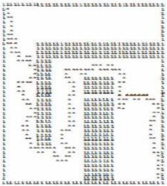
Results

Global Initialization

Map Size	20x20	30x30	40x40
Iterations	151	277	148
Path Length	39	66	138

Local Initialization

Map Size	20x20	30x30	40x40
Iterations	122	84	69
Path Length	39	64	128



Local Initialization: 1st iteration

Local Initialization: 1000th iteration

Comparison to other meta-heuristic techniques

- other techniques:
  Genetic Algorithms (GA), Simulated Annealing (SA), Particle Swarm Optimization (PSO), Tabu Search (TS)
- hard to compare in general → dependent on specific problem instance, algorithm implementation and parameter settings (*No free lunch theorem*)
- slow convergence compared to other approaches
  → long runtime for small easy instances and fast, pretty good results for complex instances
- ACO often performs really bad or really good

Traveling Salesman Problem

results for a small TSP instance with 20 nodes over multiple runs

Measures	ACO	GA	SA	PSO	TS
Parameters	pheromone evaporation	population, crossover, mutation	temperature annealing rate	population size, velocity	tabu list length
Convergence	slow due to pheromone evaporation	rapid	avoids trapping by deterioration moves	less rapid	tabulist avoids trapping in local optima
Intensification Diversification	ant movement, pheromone update	crossover, mutation	cooling, solution acceptance strategy	local search, fitness	tabulist, neighbor selection
CPU Time(s)	250	200	101	220	140
Path Length	300	200	99	250	97

Advantages and Drawbacks

Advantages

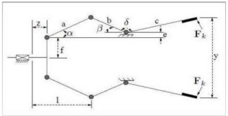
- inherent parallelism
- easy to implement on a basic level $\rightarrow$ few parameters
- possible to solve $\mathcal{NP}$-hard problems
- fast in finding near optimal solutions in comparison to classical approaches
- robust $\rightarrow$ suitable for dynamic applications

Drawbacks

- randomness $\rightarrow$ not guaranteed to find the optimal solution
- slow convergence
- theoretical research is hard $\rightarrow$ mostly rely on experimental results

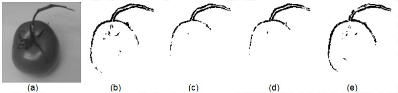
Interesting Applications

Dexterous Manipulation: Gripper Configuration



Determine forces extracted by robot grippers to guarantee stability of the grip without causing defect or damage to the object.
Non-linear problem containing five objective functions, nine constraints and seven variables.

Image Processing: Edge Detection



Ants move from one pixel to another and are directed by the local variation of the images intensity values stored in a heuristics matrix. The highest density of the pheromone is deposited at the edges.

Recap

- Swarm Intelligence
- Inspired by ant colony movement
- Three basic approaches
  - Ant System
  - Min-Max Ant System
  - Ant Colony System
- Application: Robot Path Planning
  - Shortest path in static environment with free space map
  - Shortest path in dynamic environment
- slow convergence but fast good solutions for complex problems

References

[1] Toolika Arora and Yogita Gigras.
A Survey of Comarison Between Various Meta-Heuristic Techniques for Path Planning Problem.
International Journal of Computer Engineering & Science, 3(2):62–66, November 2013.

[2] Michael Brand, Michael Masuda, Michael Masuda, Nicole Wehner, and Xiao-Hua Yu.
Ant Colony Optimization Algorithm for Robot Path Planning.
International Conference On Computer Design And Appliations (ICCDA 2010), 3:436–440, 2010.

[3] Marco Dorigo and Thomas Stützle.
Ant colony optimization.
MIT Press, 2004.

[4] Marco Dorigo and Thomas Stützle.
International Series in Operations Research & Management Science 146, Ant Colony Optimization: Overview and Recent Advances, chapter 8, pages 227–263.
Springer Science+Business Media, 2010.

[5] Yogita Gigras and Kusum Gupta.
Ant Colony Based Path Planning Algorithm for Autonomous Robotic Vehicles.
International Journal of Artificial Intelligence & Applications (IJAIA), 3(6), November 2012.

[6] Mauro Birattari Marco Dorigo and Thomas Stützle.
Ant Colony Optimization Artificial Ants as a Computational Intelligence Technique.
IEEE Computational Intelligence Magazine, 1556-603X(06):28–39, 2006.

References (cont.)

[7] Seedarla Moses Mullar and R. Satya Meher.
Optimizing of Robot Gripper Configurations Using Ant Colony Optimization.
International Journal of Engineering Research & Technology (IJERT), 2(9), September 2013.

[8] Buniyamin N., Sariff N., Wan Ngah W.A.J., and Mohamad Z.
Robot global path planning overview and a variation of ant colony system algorithm.
International Journal of Mathematics and Computers in Simulation, 5(1), 2011.

[9] Alpa Reshamwala and Deepika P Vinchurkar.
Robot Path Planning using An Ant Colony Optimization Approach: A Survey.
International Journal of Advanced Research in Artificial Intelligence, 2(3), 2013.