Swarm Intelligence Project

Ant colony algorithms for the Closest String Problem

Samuel Buchet

Swarm Intelligence
Université Libre de Bruxelles

June 13, 2017

Plan

1 Introduction

2 Ant colony algorithms for the CSP

3 Results

4 Conclusion

The Closest String Problem

- An alphabet
- A set of strings
- Find a string that minimizes the maximum hamming distance to the problem's strings
- NP-hard combinatorial optimization problem

Basic ACO elements

- Population of solutions
- Pheromones and heuristic information
- Solutions made using biased probabilites
- Pheromones are deposited and evaporated

Previous work

- Ant-CSP algorithm by (Faro & Pappalardo, 2010)
- No heuristic information
- Elitist strategy
- Amount of pheromone deposited: $1 - \frac{distance}{string\_length}$

Heuristic information for the CSP

- Hunch: choose a character which is close to all the strings in the instance set
- $\rightarrow$ for each position, the score of a character is his frequency
- Problem: the frequencies can be very closed which has no effect on the biased probabilities
- Alternative: $score_{alt} \leftarrow e^{score*5}$
- After that, the score is reduced to avoid large numbers

Algorithms implemented

Max Min Ant System

- Bounds on the pheromones
- Pheromones initialized to $+\infty$
- $upper\_bound = 1 - \rho * \frac{best}{m}$
- $lower\_bound = \frac{upper\_bound}{a}$
- Only the best iteration ant deposits pheromones
- Pheromones can be re-initialized

Ant Colony System

- The best ant so far updates the pheromones
- Pheromones are also deposited during the construction
- Two decision rules: intensification and diversification

Local search

- Apply a small change on the solution
- Change a character $\rightarrow$ all possible modifications are tried, the best solution is selected (best improvement rule)
- Fast evaluation of a solution: the distances to all the strings of the instance are saved and updated

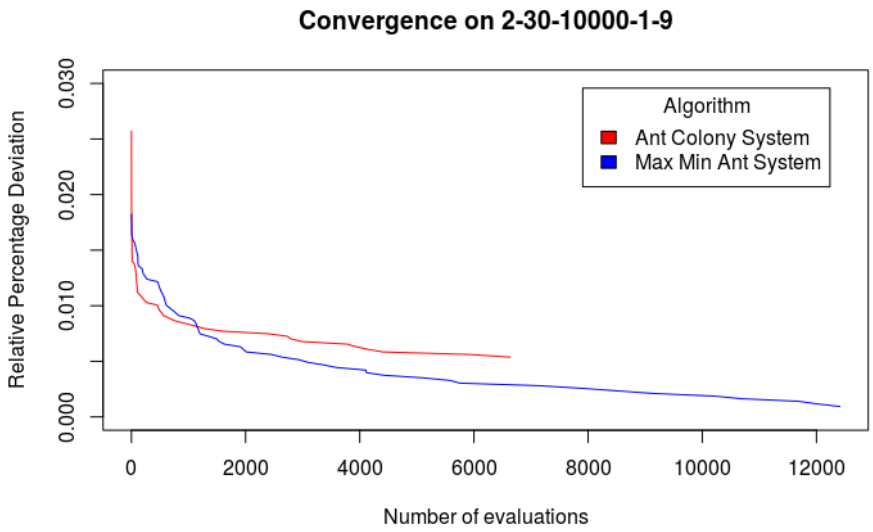
This process remains slow. It is only applied once, at the end of the ant algorithm execution

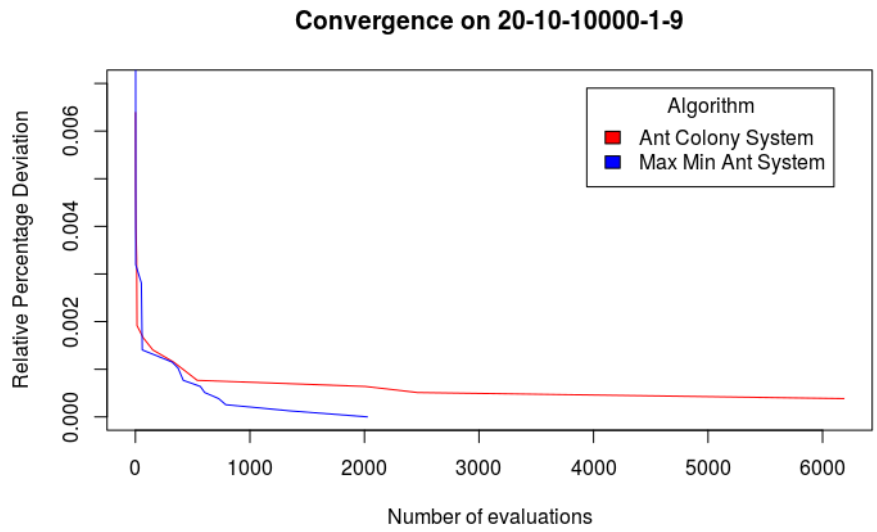
RPD's comparision

algorithm	min	max	mean	sd
MMAS	0.005	0.766	0.303	0.219
ACS	0.051	1.034	0.430	0.292
MMAS + LS	0	0.755	0.284	0.213

- Max Min seems to be always better
- However, not statistically significant (wilcoxon test, significant level equal to 0.05)
- Only few improvements with the local search on Max Min

Convergences





- Expiements with a budget of 20000 evaluations
- ACS seems to converge sooner
- However, at the very beginning, it could be better

Conclusion

- Good results, high quality solution reached by both algorithms
- Max Min seems to reach higher quality solutions
- ACS seems to be more performent at the very beginning, but more tests are required
- A pheromones re-initialization step could be interesting for ACS
- More sophisticated local searches can be implemented

References


Simone Faro and Elisa Pappalardo. Ant-CSP: An Ant Colony Optimization Algorithm for the Closest String Problem, pages 370-381. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.