Simplified Ant Colony System applied to the Quadratic Assignment Problem

Barton, Alan

https://doi.org/10.4224/5763149

National Research Council Canada
Conseil national de recherches Canada

Institute for Information Technology
Institut de technologie de l'information

NRC · CNRC

*A Simplified Ant Colony System Applied to the Quadratic Assignment Problem **

Barton, A.
March 2005

* published as NRC/ERB-1123. March 30, 2005. 4 Pages. NRC 47446.

Canada

National Research Council Canada
Conseil national de recherches Canada

Institute for Information Technology
Institut de technologie de l'information

ERB-1123

NRC-CNRC

A Simplified Ant Colony System Applied to the Quadratic Assignment Problem

Barton, A.
March 2005

Canada

NRC 47446

A Simplified Ant Colony System applied to the Quadratic Assignment Problem*

Alan J. Barton†
Integrated Reasoning Group
Institute for Information Technology
National Research Council Canada
Ottawa, Canada, K1A 0R6
alan.barton@nrc-cnrc.gc.ca

ABSTRACT
An attempt is made to solve the Quadratic Assignment Problem through the development of a simplified ant colony system in C. Experiments with the implementation are performed and reported.

Keywords
NP-hard, swarm intelligence, stigmergy, emergent behaviour

1. INTRODUCTION
The *Quadratic Assignment Problem* (QAP) is a challenge in combinatorial optimization, as it is computationally non-trivial to solve even modest sized problems (e.g. $n = 20$) [3]. Approaches for attempting to solve QAP include brute-force search, tabu search, and many others [2]. An implementation of a simple form of the Ant Colony System (ACS) (p.46[1]), (based upon the Ant System) as it applies to QAP is investigated.

2. QUADRATIC ASSIGNMENT PROBLEM
The QAP was first introduced by Koopmans and Beckman in 1957 (p.56[1]) and entails assigning n activities to n locations (or vice versa). The left portion of Fig-1 gives one possible visual representation (graph) of the distance matrix (depicted in (1)), of distances between locations. Such distance measurements may be based on the Euclidean, Manhattan, or other metrics. Whereas, the right portion of Fig-1 demonstrates a graph for the flow matrix in (2); representing flow of material, people, objects, etc. from one activity to another. This NP-hard problem, of assigning activities to locations in the best way, involves searching through an assignment space of all possible permutations, of size $n!$.

*Second Assignment for the course COMP5900Z entitled *Swarm Intelligence* at Carleton University

†Master of Computer Science (in progress)

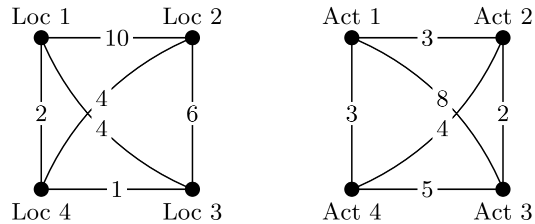

Figure 1: Left: Distances between locations, Right: Flow between activities

$$D = \begin{bmatrix} 0 & 10 & 4 & 2 \\ 10 & 0 & 6 & 4 \\ 4 & 6 & 0 & 1 \\ 2 & 4 & 1 & 0 \end{bmatrix} \quad (1)$$

$$F = \begin{bmatrix} 0 & 3 & 8 & 3 \\ 3 & 0 & 2 & 4 \\ 8 & 2 & 0 & 5 \\ 3 & 4 & 5 & 0 \end{bmatrix} \quad (2)$$

3. ANT SYSTEM SOLUTION
The Ant System (AS) solution to QAP was proposed in 1994 by Maniezzo et al (p.57[1]). The solution involves[1]:

1. a graph representation, similar to the Travelling Salesman Problem (TSP), where the nodes of the graph are locations, and the ants' goal is to visit all the locations and match an activity to it, forming a bijective mapping from activities to locations.

2. ants memorizing locations that they have visited, so that they never visit them again.

3. ants memorizing the activities that have been mapped to the visited locations.

4. devising a heuristic that allows solutions to be constructed directly.

A solution may be constructed from Fig-1 by constructing a vector comprised of the potentially best distances, called distance potentials, from (1), and potentially best flows, called flow potentials, from (2) by summing each row yielding (3) and (4), respectively.

$$\overline{d} = \begin{bmatrix} 16 \\ 20 \\ 11 \\ 7 \end{bmatrix} \quad (3)$$

$$\overline{f} = \begin{bmatrix} 14 \\ 9 \\ 15 \\ 12 \end{bmatrix} \quad (4)$$

An abstract form of the implementation of the algorithm is listed as Alg-1.

ALGORITHM 1. AS-QAP(α, β, ρ, m, t_{max}, τ_0, Q, seed)

```
assign_initial_pheromone_levels()
compute_distance_potentials()
compute_flow_potentials()
compute_coupling_matrix()
place_ants_on_locations()
for(t ← 1) to (t_max) do
    for(k ← 0) to (num_ants−1) do
        build_solution_for_ant(k)
    endfor
    for(k ← 0) to (num_ants−1) do
        compute_cost_of_solution_for_ant(k)
    endfor
    for(k ← 0) to (num_ants−1) do
        if(ants[k].cost_of_solution <
           BEST_SOLUTION.cost_of_solution)
        then
           BEST_SOLUTION ← ants[k]
           BEST_SOLUTION_t ← t
        endif
    endfor
    update_pheromone_trails()
    write_experimental_results(t)
endfor
```

— End of Algorithm —

4. EXPERIMENTAL SETUP

A C language based implementation of AS-QAP was made. 10 experiments were performed using experimental parameters listed in Table-1. The experiments focused on two problem data sets, *i)* the $n = 4$ case as given in [1] and *ii)* the $n = 12$ case as given by Nugent(12) [3].

5. RESULTS

For $n = 4$, it can be seen in Fig-2 that pheromone levels have a layered effect of approximately every 0.01 intervals on the z-axis over 150 ticks. One conjecture, is that once

Table 1: Experimental Parameters investigated using a C-based implementation of AS-QAP for two example problems ($n = 4, 12$)

ID	α	β	ρ	m	t_{max}	τ_0	Q	seed
Exp-0	1	1	0.9	4	150	$1.0 \cdot e^{-6}$	10	999
Exp-1	1	1	0.9	12	500	$1.0 \cdot e^{-6}$	10	999
Exp-2	1	1	0.9	12	5,000	$1.0 \cdot e^{-6}$	10	999
Exp-3	1	1	0.95	12	500	$1.0 \cdot e^{-6}$	10	999
Exp-4	1	1	0.99999	12	500	$1.0 \cdot e^{-6}$	10	999
Exp-5	1	1	0.9	12	50,000	$1.0 \cdot e^{-6}$	10	999
Exp-6	1	1	0.9	12	500	$1.0 \cdot e^{-6}$	10	−999
Exp-7	1	1	0.9	12	500	$1.0 \cdot e^{-6}$	10	0
Exp-8	2	1	0.9	12	500	$1.0 \cdot e^{-6}$	10	0
Exp-9	1	2	0.9	12	500	$1.0 \cdot e^{-6}$	10	0

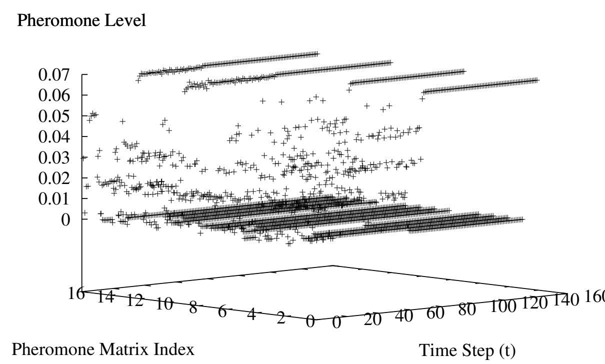

Figure 2: Exp-0: Pheromone levels changing over time for each $\tau_{i,j}$

a pheromone level is established that is slightly higher than the rest, then other ants continue to follow and more readily deposit pheromone on those activity/location couplings. In terms of the computed solution for QAP, where $n = 4$, the best result is found after 2 iterations, with a value of 631; the bottom line in Fig-3. It is interesting to notice the compression in the variance of the ant solutions after tick 80 to a value of approximately 652. Indicating that the algorithm is incapable of finding further solution candidates after this tick.

For $n = 12$, it can be seen in Fig-4 that a similar layered effect to Fig-2 is exhibited. It is unclear why this observed effect has occurred. One conjecture is that the number of ants m will lead to m layers, but that is not supported by the facts (Fig-2 and Fig-4). Another conjecture is that because the ants are all randomly walking and depositing pheromone, the equations for pheromone depositing lead directly to this effect. The equation would need to be more closely examined in order to support or dispute this claim.

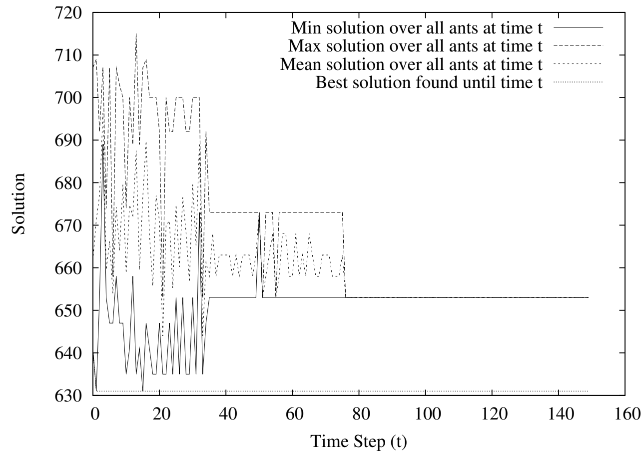

Figure 3: Exp-0: Solution costs changing over time. (Best solution cost $= 631$)

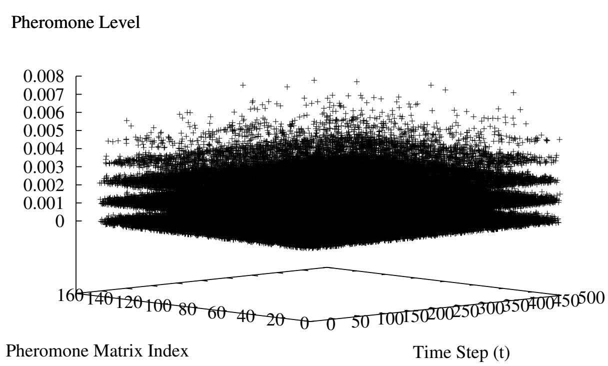

Figure 4: Exp-1: Pheromone levels changing over time for each $\tau_{i,j}$

All experiments were investigated as per the methodology implicit in Fig-5. All of the results portray similar large variance and approximately stable mean values (results not shown). This may indicate that either *i)* the ants are following pheromone too closely, and not properly exploring the search space, or *ii)* the fitness landscape is very difficult, in that a slight change in a permutation may lead to a large change in the fitness (solution) value, making search very difficult. Fig-6 shows the results for each of the Nugent(12) experiments (9 in total) that were performed. The best overall solution $(8,698)$ was discovered by Exp-5 after $28,296$ iterations, while the second best solution $(8,702)$ was discovered by Exp-1 and Exp-2 (both) after 448 iterations, indicating that the 10-fold increase in t_{max} (Table-1) did not enhance the ant solution discovery process.

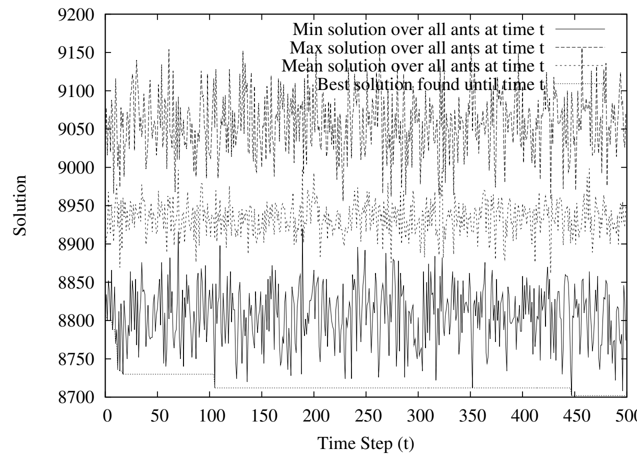

Figure 5: Exp-1: Value of minimum, mean and maximum per tick solutions constructed by the ants. Final best solution $= 8702$

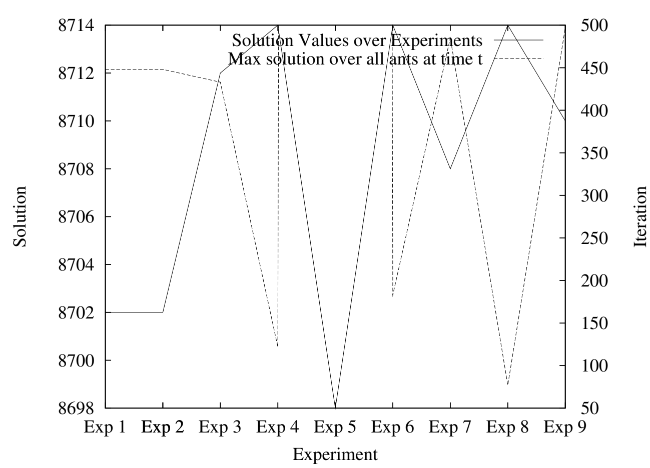

Figure 6: Absolute best solution discovered by each ant experiment for $n = 12$. Exp-5, with $28,296$ iterations, found the best solution value of $8,698$.

6. CONCLUSIONS

An ant based solution for the Quadratic Assignment Problem has been implemented and investigated. Good results are exhibited, with high variance, for $n = 4$ leading to the conjecture that a correct implementation has been performed. More detailed analysis of the results of the experiments (179MB of data) are necessary in future investigations. Including the possibility of making ρ a function of *i)* time and *ii)* the relative change of the mean (or best) solution.

7. REFERENCES

[1] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence From Natural to Artificial Systems*. Number ISBN 0-19-513159-2 in Sante Fe Institute Studies in the Sciences of Complexity. Oxford University Press, 198 Madison Avenue, New York, New York 10016, 1999.

[2] R. Burkard, E. Çela, S. Karische, and F. Rendl. QAPLIB - A Quadratic Assignment Problem LIBrary. `http://www.opt.math.tu-graz.ac.at/qaplib/`. Graz University of Technology, Austria., March 2005.

[3] R. E. Burkard, S. E. Karisch, and F. Rendl. QAPLIB - a Quadratic Assignment Problem Library. *Journal of Global Optimization*, 10:391–403, 1997.