World of Computer Science and Information Technology Journal (WCSIT)
ISSN: 2221-0741
Vol. 3, No. 5, 100-104, 2013

Adapting the Ant Colony Optimization Algorithm to the Printed Circuit Board Drilling Problem

Taisir Eldos, Aws Kanan, Abdullah Aljumah
Department Of Computer Engineering
College of Computer Engineering and Sciences
Salman in Abdulaziz University
Al Kharj, Saudi Arabia

Abstract — Printed Circuit Board (PCB) manufacturing depends on the holes drilling time, which is a function of the number of holes and the order in which they are drilled. A typical PCB may have hundreds of holes and optimizing the time to complete the drilling plays a role in the production rate. At an early stage of the manufacturing process, a numerically controlled drill has to move its bit over the holes one by one and must complete the job in minimal time. The order by which the holes are visited is of great significance in this case. Solving the TSP leads to minimizing the time to drill the holes on a PCB. Finding an optimal solution to the TSP may be prohibitively large as the number of possibilities to evaluate in an exact search is (n-1)!/2 for n-hole PCB. There exist too many algorithms to solve the TSP in an engineering sense; semi-optimal solution, with good quality and cost tradeoff. Starting with Greedy Algorithm which delivers a fast solution at the risk of being low in quality, to the evolutionary algorithms like Genetic algorithms, Simulated Annealing Algorithms, Ant Colony, Swarm Particle Optimization, and others which promise better solutions at the price of more search time. We propose an Ant Colony Optimization (ACO) algorithm with problem-specific heuristics like making use of the dispersed locales, to guide the search for the next move. Hence, making smarter balance between the exploration and exploitation leading to better quality for the same cost or less cost for the same quality. This will also offer a better way of problem partitioning which leads to better parallelization when more processing power is to be used to deliver the solution even faster.

Keywords - Ant Colony; Optimization Algorithm; Printed Circuits Board Drilling;Traveling Salesman.

I. Introduction

Ant Colony Optimization (ACO) algorithm is a probabilistic technique for solving computational problems which can be reduced to finding good paths through graphs. Proposed by Marco Dorigo in 1992, the first algorithm was meant to search for an optimal path in a graph, based on the behavior of ants seeking a path between their colony and a source of food. It has since diversified to solve a wider class of numerical problems, and as a result. In nature, ants start wandering randomly, and upon finding food return to their colony while laying down pheromone trails. If other ants find such a path, they are likely to follow the trail, returning and reinforcing it if they eventually find food. Over time, however, the pheromone trail starts to evaporate, thus reducing its attractive strength. The more time it takes an ant to travel down the path and back again, the more time the pheromones have to evaporate. A short path, by comparison, gets marched over more frequently, and thus the pheromone density becomes higher on shorter paths than longer ones. Pheromone evaporation also has the advantage of avoiding the convergence to a locally optimal solution. If there were no evaporation at all, the paths chosen by the first ants would tend to be excessively attractive to the following ones. In that case, the exploration of the solution space would be constrained. Thus, when one ant finds a short path from the colony to a food source, other ants are more likely to follow that path, and positive feedback eventually leads to all the ants' following a single path. The idea of the ant colony algorithm is to mimic this behavior with "simulated ants" walking around the graph representing the problem to solve. Generally, this algorithm is convergent; capable of finding a global optimum in finite time.

The ACO has been successfully applied to many problems; scheduling problems, assignment problems, data mining, classification, multiple knapsack problem, traveling salesman problem, and many others. However, some problems have not received enough attention and while they may have attractive attributes for this search technique. In our work, we will focus on the Printed Circuits Boards Drilling problem as a special case of the Travelling Salesman problem, taking advantage of the dispersed locales property.

II. Related Work

Among the early works of ant colony optimization, an artificial ant capable of solving the travelling salesman problem, both symmetric and asymmetric instances was proposed in [1]. The method is an example, like simulated annealing, neural networks and evolutionary computation, of the successful use of a natural metaphor to design an optimization algorithm. The first runtime analysis of a simple ACO algorithm that transfers many rigorous results with respect to the runtime of a simple evolutionary algorithm is presented in [2], along with and examining the choice of the evaporation factor. A sensibility analysis to help tune the parameters of an ant colony model where ants leave a fuzzy trace of pheromone to mark their track and its neighborhood, where tests were conducted on classical problems is presented in [3]. A case study demonstrating the capability of the Ant Colony Optimization (ACO) algorithm to solve the Travelling Salesman problem is presented in [4], in order to find the best solution in terms of the shortest distance. Many tough problems have been solved by the ant colony optimization technique with great success; [5] presents an edge detection technique that is based on ACO, by establishing a pheromone matrix that represents the edge information at each pixel based on the routes formed by the ants dispatched on the image. The movement of the ants is guided by the local variation in the image's intensity values, taking advantage of the improvements introduced in ant colony system. Experimental results show the success of the technique in extracting edges from a digital image. [6] develops a coarse-grain parallel ant colony algorithm to solve the problem develops optimization model for bus transit network based on road network and zonal OD. The model aims at achieving minimum transfers and maximum passenger flow per unit length with line length and non-linear rate as constraints. It uses a heuristic pheromone distribution rule, by which ants' path searching activities are adjusted according to the objective value. [7] reports the analysis of using a lower pheromone trail bound and a dynamic updating rule for the heuristic parameters based on entropy to improve the efficiency of this algorithm in solving Traveling Salesman Problems (TSPs), with extremely large problem space. The experiments show that the proposed algorithm has superior search performance over traditional ant colony algorithms. [8] addresses the problem by developing a general framework of grid scheduling using dynamic information and an ant colony optimization algorithm to improve the decision of scheduling. The performance of various dispatching rules such as First Come First Served (FCFS), Earliest Due Date (EDD), Earliest Release Date (ERD), and an Ant Colony Optimization (ACO) are compared. Moreover, the benefit of using an Ant Colony Optimization for performance improvement of the grid Scheduling is also discussed. It is found that the scheduling system using an Ant Colony Optimization algorithm can efficiently and effectively allocate jobs to proper resources. [9] proposes the Omicron ACO (OA), a novel population-based ACO alternative originally designed as an analytical tool. To experimentally prove OA advantages, this work compares the behavior between the OA and the MMAS as a function of time in two well-known TSP problems. A simple study of the behavior of OA as a function of its parameters shows its robustness.

The ant colony algorithm performance has been boosted by involving other techniques in its internal workings; for example [10] shows an ant colony and simulated annealing algorithms used to find the core of a graph, such that the total travel cost time required for the demand points to reach the closest vertex on this path is minimized. While, [11] develops a new fuzzy-logic based ACO algorithm, taking into consideration the uncertainties that can be found in both the heuristic and the pheromone factors. Where during the solution iterations the calculations are performed taking into consideration the fuzzy levels of the involved parameters. A stochastic-based technique is proposed to enable the artificial ant to choose the best oncoming step based on the values of the probabilities and their corresponding fuzzy levels. The proposed algorithm gives the optimal solution in a form of an optimal value and its corresponding fuzzy level, using benchmark Quadratic Assignment Problem (QAP) and Travelling Salesman Problem (TSP). An ant colony optimization technique for continuous domains is presented in [12], to provide improvements in computing time and robustness when compared to other optimization algorithms. The developed Modified Continuous Ant Colony (MCACO) algorithm was run for numerous classic test cases for both single and multi-objective problems. The results demonstrate that the method is robust, stable, and that the number of objective function evaluations is comparable to other optimization algorithms. [13] proposes an algorithm that incorporates key features of the tabu-search method in the development of a relatively simple but robust global ant colony optimization algorithm. Numerical results are reported to validate and demonstrate the feasibility and effectiveness of the proposed algorithm in solving electromagnetic (EM) design problems.

III. Ant Based Search

In ACO, artificial ants build solutions to the problem by traversing a construction graph, as sets denoted by cij, towards a set of all possible solution components denoted by C. A pheromone trail value τij is associated with each component cij, to allow the probability distribution of different components of the solution to be modeled. The ants move from vertex to vertex along the edges of the construction graph exploiting information provided by the pheromone values to incrementally build solutions. Ants deposit pheromone on the components, whose amount $\Delta\tau$ is a function of both the iteration and the solution quality.

Typically, a set of m artificial ants construct solutions from elements of a finite set of available solution components $C=\{c_{ij}\}$, $i=1,\ldots,n$, $j=1,\ldots,|D_i|$. The solution construction starts with an empty partial solution $s^p=\emptyset$. Then, at each construction step, the current partial solution s^p is extended by adding a feasible solution component from the set of feasible neighbors $N(s^p)\subseteq C$. This process can be regarded as a path on the construction graph.

Choosing the solution component from $N(s^p)$ is is carried out probabilistically at each construction step, and the best known rule is the one of ant system:

$$p(c_{ij}|s^p) = \frac{\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}}{\sum_{c_{ij} \in N(s^p)} \tau_{il}^{\alpha} \cdot \eta_{il}^{\beta}}, \forall c_{ij} \in N(s^p) \quad (1)$$

where τ_{ij} and η_{ij} are respectively the pheromone value and the heuristic value associated with the component c_{ij}. α and β are positive real parameters to impose relative importance of pheromone against heuristic information.

Pheromone update is meant to increase the pheromone values associated with good solutions, and to decrease those that are associated with bad ones. Usually, this is achieved by decreasing all the pheromone values through a process called pheromone evaporation, and increasing the pheromone levels associated with a chosen set of good solutions *Supd*:

$$\tau_{ij} \leftarrow (1-\rho) \cdot \tau_{ij} + \rho \cdot \sum\nolimits_{s \in S_{upd} | c_{ij} \in s} F(s) \quad (2)$$

where *Supd* is the set of solutions that are used for the update, $\rho \in (0,1)$ is a parameter called evaporation rate, and $F:S \rightarrow R+0$ is a function such that;

$$f(s) < f(s') \Rightarrow F(s) \geq F(s'), \forall s \neq s' \in S \quad (3)$$

F is commonly called the fitness function.

Pheromone evaporation implements a useful form of forgetting, favoring the exploration of new areas in the search space. Instantiations of the update rule given above are obtained by different specifications of *Supd*; typically a subset of $Siter \cup \{sbs\}$, where *Siter* is the set of solutions that were constructed in the current iteration, and *sbs* is the best-so-far solution, that is, the best solution found since the first algorithm iteration.

Usually, the update rule is *Supd←Siter*, although *Supd←arg maxs∈Siter F(s)*, is more often used in practive.

IV. MAIN ACO ALGORITHMS

Several special cases of the ACO metaheuristic have been proposed in the literature.

Ant System (AS)

This variant is mostly concerned with updating the pheromone values by all the ants that have completed the tour. Solution components c_{ij} are the edges of the graph, and the pheromone update for τ_{ij}, the pheromone associated with the edge joining nodes *i* and *j*, is performed as follows:

$$\tau_{ij} \leftarrow (1-\rho) \cdot \tau_{ij} + \rho \cdot \sum\nolimits_{k=1}^{m} \Delta\tau_{ij}^{k} \quad (4)$$

where $\rho \in (0,1)$ is the evaporation rate, m is the number of ants, and $\Delta\tau^{k}_{ij}$ is the quantity of pheromone laid on edge *(i,j)* by the *k*-th ant:

$$\Delta\tau_{ij}^{k} = \frac{1}{L_k} \ if\ ant\ k\ has\ edge(i,j) in\ tour, else\ 0 \quad (5)$$

where L_k is the tour length of the *k*-th ant.

When constructing the solutions, the ants traverse the construction graph and make a probabilistic decision at each vertex. The transition probability $p(c_{ij}|s^p{}_k)$ of the *k*-th ant moving from city i to city j is given by:

$$p(c_{ij}|s^p) = \frac{\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}}{\sum_{c_{ij} \in N(s^p)} \tau_{il}^{\alpha} \cdot \eta_{il}^{\beta}}, \text{if } j \in N(s_k^p), \text{else } 0 \quad (6)$$

where $N(s^p{}_k)$ is the set of components that do not belong yet to the partial solution $s^p{}_k$ of ant *k*, parameters α and β to control relative importance of pheromone against heuristic information $\eta_{ij}=1/d_{ij}$, where d*ij* is the length of component c*ij* (i.e., of edge *(i,j)*).

Ant Colony System (ACS)

A major improvement over the original ant system, through the decision rule during the construction process. Here, ants use the so-called pseudorandom proportional rule: the probability for an ant to move from city i to city j depends on a random variable q uniformly distributed over [0,1] ,and a parameter q0 ; if q≤q0 , then, among the feasible components, the component that maximizes the product $\tau_{il}\eta^{\beta}{}_{il}$ is chosen; otherwise, the same equation as in AS is used.

This greedy rule favors exploitation of the pheromone information, and is counterbalanced by the introduction of the local pheromone update for diversification. The local pheromone update is performed by all ants after each construction step. Each ant applies it only to the last edge traversed:

$$\tau_{ij}=(1-\varphi)\cdot\tau_{ij}+\varphi\cdot\tau_0 \quad (7)$$

where $\varphi \in (0,1]$ is the pheromone decay coefficient, and $\tau 0$ is the initial value of the pheromone.

The main goal of the local update is to diversify the search performed by subsequent ants during one iteration; decreasing the pheromone concentration on the edges as they are traversed during one iteration encourages subsequent ants to choose other edges and hence to produce different solutions. This makes less likely that several ants produce identical solutions during one iteration. Additionally, because of the local pheromone update in ACS, the minimum values of the pheromone are limited.

At the end the construction, an offline pheromone update is performed; performed only by the best ant, that is, only edges that were visited by the best ant are updated, according to the equation:

$$\tau_{ij}\leftarrow(1-\rho)\cdot\tau_{ij}+\rho\cdot\Delta\tau^{best}{}_{ij} \quad (8)$$

where $\Delta\tau^{best}{}_{ij} = 1/L^{best}$ if the best ant used edge (i,j) in its tour, $\Delta\tau^{best}{}_{ij} = 0$ otherwise (L^{best} can be set to either the length of the best tour found in the current iteration -- iteration-best, Lib -- or the best solution found since the start of the algorithm -- best-so-far, *Lbs*).

MAX-MIN Ant System (MMAS)

MMAS is another improvement, it differs from AS in that (i) only the best ant adds pheromone trails, and (ii) the minimum and maximum values of the pheromone are explicitly limited. The pheromone update equation takes the following form:

$$\tau_{ij} \leftarrow (1-\rho)\cdot\tau_{ij} + \Delta\tau^{best}_{ij} \quad (9)$$

where $\Delta\tau^{best}_{ij} = 1/ L^{best}$ if the best ant used edge (i,j) in its tour, $\Delta\tau^{best}_{ij} = 0$ otherwise, where Lbest is the length of the tour of the best ant. As in ACS, L^{best} may be set (subject to the algorithm designer decision) either to Lib or to L^{bs} ,or to a combination of both.

The pheromone values are constrained between τ_{min} and τ_{max} by verifying, after they have been updated by the ants, that all pheromone values are within the imposed limits.

$$\tau_{ij} \text{ is set to } \tau_{max} \text{ if } \tau_{ij}>\tau_{max} \text{ and to } \tau_{min} \text{ if } \tau_{ij}<\tau_{min} \quad (10)$$

The minimum and maximum values are experimentally or analytically selected.

V. PCB Drilling Problem

PCBs contain few hundreds of holes of different diameters and depths, typically portioned for the sake of drilling time, each set of holes need to be drilled in the shortest time possible. We will use the MAX-MIN Ant System with local search variants called 2-opt and 3-opt, in which 2 or 3 edges swap is injected during the search towards local minimization.

Table 1 shows the performance of the algorithm on three instances with relatively small number of holes as a test drive.

Figure 1 shows the Max Min Ant System algorithm progress on the PCB442 benchmark; tour length as a function of the processing time (in msec). Clearly, the first tens of iterations have the biggest share in the tour enhancement.

Table 1, Performance: tour improvement for 3 instances (Optima and Best tours)

		After 500 iterations		After 1000 iterations	
Size	**Optimal**	**Best Tour**	**Error**	**Best Tour**	**Error**
52	7542	8763	15.8 %	7562	0.26 %
127	118282	146117	23.5 %	123608	4.5 %
150	6528	8546	30.9 %	6697	2.5 %

Table 2 shows the effect of employing more ants in the system to run for a fixed amount of time. More ants represents a computational burden, but it pays off in terms of the solution quality, although not quite linearly.

Table 3 shows the effect of using local search within the global exploration. We could improve the solution by 13% using the 3-opt within the Max-Min variant of the Ant System.

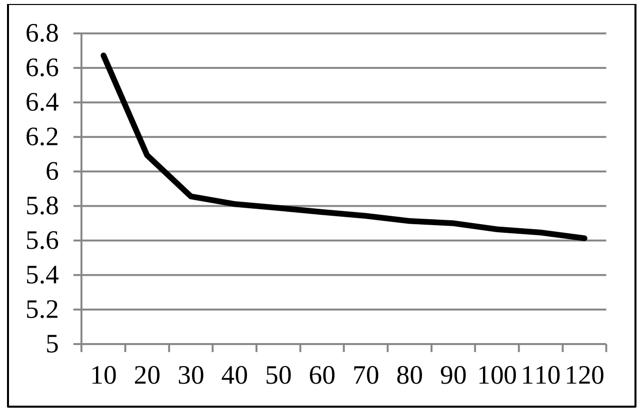

Figure 1. Search Progress: Tour Length versus Processing Time (1000s iterations)

Table 2: Performance versus Number of Ants

No. of Ants	Best Tour Length
5	59476
10	58428
15	58172
20	58432
25	57284
30	57396

Table 3: Performance versus 4 options related to local search (120 sec , 25 ants, alpha =1, beta =2, rho = 0.5)

Search	Best Tour Length
No Local	56631
2 - opt Local	50927
2.5 - opt Local	50921
3 - opt Local	50778

VI. Conclusion

Although the TSP and its derivatives have been solved by many evolutionary algorithms, the ACO algorithms seem to be a promising alternative in solving the PCB drilling problem in practically reasonable quality with practically reasonable time. The implementation shows that the problem is amenable to parallelization due to the dispersed locals, and hence may offer better performance when more computational resources is available.

Future Work

We plan to compare the performance of the Ant Colony Optimization (ACO) algorithm with competitive ones like Chemical Reaction Optimization (CRO) and Gravitational Search Algorithm (GSA) on several PCB benchmarks. We are analyzing few benchmarks to design few daemon actions towards better performance in certain identified locales.

Acknowledgment

This project was funded by a grant from the deanship of scientific research in Salman bin Abdulaziz University, Al

Kharj, Saudi Arabia (No. 1/T/1432) during the academic year 2012/2013.

REFERENCES

[1] Marco Dorigo, Luca Maria Gambardella, “Ant colonies for the travelling salesman problem,” Eslevier, BioSystems, Vol. (43), pp 73–81, 1997.

[2] Frank Neumann Dagstuhl, “Runtime Analysis of a Simple Ant Colony Optimization Algorithm,” Seminar Proceedings: Theory of Evolutionary Algorithms.

[3] Louis Gacogne, Sandra Sandri, “A study on ant colony systems with fuzzy pheromone dispersion,” Proceedings of IPMU'08, pp. 812, 2008.

[4] Helmi Md Rais, Zulaiha Ali Othman, Abdul Razak Hamdan, “Improved Dynamic Ant Colony System (DACS) on symmetric Traveling Salesman Problem,” International Conference on Intelligent and advanced systems, pp43-48, 2007.

[5] Anna Veronica Baterina, Carlos Oppus, “Image Edge Detection Using Ant Colony Optimization,” International Journal of Circuits, Systems and Signal Processing, Issue 2, Vol. (4), 2010.

[6] Bin Yu, Zhongzhen Yang, Chuntian Cheng, Chong Liu, “Optimizing Bus Transient Netwrok with Parallel Ant Colony Algorithm,” Proceedings of the Eastern Asia Society for Transportation Studies, Vol. 5, pp. 374 - 389, 2005.

[7] Kuo-Sheng Hung, Shun-Feng Su, Zne-Jung Lee, “Improving Ant Colony Optimization Algorithms for Solving Traveling Salesman Problems,” Journal of Advanced Computational Intelligence and Intelligent Informatics, Vol.11 No.4, 2007.

[8] Siriluck Lorpunmanee, Mohd Noor Sap, Abdul Hanan Abdullah, and Chai Chompoo-inwai, “An Ant Colony Optimization for Dynamic Job Scheduling in Grid Environment, “World Academy of Science, Engineering & Technology, 2007.

[9] Osvaldo G´omez, “Omicron ACO: A New Ant Colony Optimization Algorithm, “CLEI Electronic Journal, Vol. (8), No. (1), 2005.

[10] Mehdi Zaferanieh, Tarbiat Moallem, Tovhid town, Sabzevar, “Ant Colony and Simulated Annealing Algorithms for Finding the Core of a Graph,” World Applied Sciences Journal, Vol. (10), pp. 1335-1341, 2009.

[11] Ahmed Rabie Ginidi Ginidi, Ahmed M. A. M. Kamel, Hassen Taher Dorrah, “Development of New Fuzzy Logic-based Ant Colony Optimization Algorithm for Combinatorial Problems, “Proceedings of the 14th Int'l Middle East Power Systems Conference, Cairo University, Egypt, 2010.

[12] A. Aidov, G. S. Dulikravich, “Modified Continuous Ant Colony Algorithm,” 2nd International Congress of Serbian Society of Mechanics, Serbia, 2009.

[13] S. L. Ho, Shiyou Yang, Guangzheng Ni, Josè Márcio Machado, “A Modified Ant Colony Optimization Algorithm Modeled on Tabu-Search Methods,” IEEE Transactions on Magnetics, Vol. (42), No. (4), 2006.

AUTHORS PROFILE

Taisir Eldos

Born in Karak, Jordan 1958. He received BS in Electronic Engineering from Menofia University, Egypt 1981, MS and PhD in Computer Engineering from University of Alabama in Huntsville, Alabama USA in 1992 and 1996 respectively. Associate professor and Chair, department of computer engineering at Jordan University of Science and Technology (1996 till now), King Saud University (2008 till 2010), and Salman bin Abdulaziz University (2010 till 2013). Research interest include parallel processing/computing, optimizations algorithms and soft computing.

Aws Kanan

Born in Kufranjah, Jordan 1980. He received BS and MS in Computer Engineering from Jordan University of Science and Technology, Jordan in 2003 and 2006 respectively. Lecturer, department of computer engineering at University of Jordan, King Saud University, and Salman bin Abdulaziz University. Research interest include reconfigurable computing, computer networks, and optimizations algorithms.

Abdullah Aljumah

Received PhD in Electronic Engineering from University of Wales, Cardiff, UK in the year 1999. His main area of research is Artificial Intelligence, Digital Design, and Data mining. He is currently working as an Assistant Professor as well as dean of the College of Computer Engineering and Sciences, Salman Bin Abdulaziz University, Alkharj, Saudi Arabia. He is also a consultant for several Government Organizations and a member of councils of various boards and commissions.

Dr. Aljumah has published a number of research papers in reputed conferences and journals.