Multistage Ant System Optimization Algorithm

Azhar Waleed Hammad* & Faiz Faig Showkat*

Received on: 25/1/2010
Accepted on: 2/6/2011

Abstract

In this paper we introduce a modified ant system optimization algorithm, we call it the multistage ant system optimization (MS-ASO). This modified algorithm have many advantageous like releasing ants from local optima, enabling ants to accept temporary bad solution in order to find better one later, improving the diversification by forcing (not encouraging) ant to search new parts of the solution space, and also pass information about the global best solution between the predetermined number of iterations/stages. We test our modified algorithm on some benchmark problems of the traveling sales man problem to see how good it is, the result we get was encouraging that is we succeed in reducing the number of calculations (iterations) and we also find new optimum solution (i.e. routes) for some benchmark problems which are better (i.e. shortest) than published routes.

Keywords: traveling seals man problem, ant system optimization.

خوارزميه المراحل المتعدده لنظام النمل للامثليه

الخلاصة

نقدم في هذا البحث خوارزمية معدلة لنظام النمل للامثلية اطلقنا عليها تسمية خوارزمية المراحل المتعددة لنظام الامثلية, تمتاز هذه الخوارزمية بعدة فوائد مثل امكانية تحرير النمل عندما ينحصر في حل امثل موقعي , السماح للنمل بقبول حل سيء بشكل مؤقت من اجل التوصل الى حل افضل لاحقا, تحسين قابلية النمل في التنويع عن طريق اجبار (وليس تشجيع) النمل على الذهاب الى مناطق بحث جديده في فضاء الحل, تمرير البيانات الخاصة بافضل حل تم التوصل اليه بين المراحل / والمحاولات من اجل الاستفاده من نتائج بحث النمل السابقة , تم اختبار الخوارزمية المقترحة على اربعة مشاكل قياسية لمسأله رجل المبيعات والنتائج كانت مشجعة للغاية حيث تم التوصل الى حلول جديدة افضل (اي اقصر) من الحلول المثلى المنشوره كما تمكنا من تقليص الحسابات الكلية اللازمة لبلوغ هذه الحلول .

Introduction

Ant System Optimization (ASO) was inspired from the real ant's behavior, where a very interesting aspect of the behavior of several ants is their ability to find shortest paths between the ants nest and the food sources Dorgio[1],

* Science College, University of Al-Nahrain /Baghdad

this is done with the help of depositing some ants to a chemical material called pheromone, so if there is no pheromone trails ant move essentially at random, but in the presence of pheromone they have a tendency to fallow the trail. Experiments show that ants probabistically prefer paths that are marked by high pheromone concentration, the stronger pheromone trail in a path then this path will have the higher desirability and because ants follow that path they will in turn deposit more pheromone on the path and they will reinforce the paths, this mechanism allows the ants to discover the shortest path(s), now this shortest path(s) get another enforcement by noting that the pheromone evaporates after some time, in this way the less promising paths progressively loss pheromone because less and less ants will use these paths, for more information about the real ants behavior and experiments done about ants one referred to Dorgio [2].

Analysis of Ant Algorithm

Although we have today many variants of ant colony optimization metaheuristic algorithms (Ant System, Ant Colony System, Min-Max ant algorithm,…, etc) but they all share the main steps which we list down in brief, Dorgio[1].

Step (1), Initialization or setup values that concerns pheromone value, control variables evaporation rates,etc.

Step (2), Creating a route at each iteration by releasing ant(s) and allow it to pass through all states (cities, stations, drilling points, etc) using a transition rule, the general formula for the transition rule is listed bellow,

$$\mathbf{P_{ij}^{(n)} = \frac{\left[ph_{ij}^{(n-1)}\right]^a \left[y_{ij}\right]^b}{\sum_{k=1}^{M} \left[ph_{ik}^{(n-1)}\right]^a \left[y_{ik}\right]^b}} \quad \dots (1)$$

$\mathbf{k \in N(i)}$
$\mathbf{i = 1,2,..., N}$
$\mathbf{j = 1,2,..., N}$
$\mathbf{i \neq j}$

where

$P_{ij}^{(n)}$ Is the probability an artificial ant stand at point (i) will move to point (j) using the route (ij).

$ph_{ij}^{(n-1)}$ Is the net pheromone value along the route (ij) at the end of iteration (n-1), with ph_{ij}^{0} represent the initial pheromone value.

y_{ij} Is an index of how good is the solution (i.e. food) along the rout (ij), it is also called heuristic information (or function), and depend very much on the nature of the problem in hand, for example in traveling sales man problem, and car scheduling it is equal to ($\mathbf{1/d_{ij}}$) where $\mathbf{d_{ij}}$ is the distance (or time, cost,…etc) between point (i) & (j).

$N(i)$ Are the neighborhood points of (i) that are not visited yet by any ant.

M Is the number of points the ant can visit.

a,b Are control variables which determine the relative influence of pheromone trails and the heuristic function.

Step (3), Before beginning the (nth) iteration we have (n-1) routes each one is found during the (n-1) iterations we

made, the best one between these route we call the global best solution, while the solution obtained at the end of each iteration we call it iteration solution. At the end of each iteration we make a comparison between the iteration solution and global best solution, if it happen that the iteration solution is better than the global best solution we find then we consider the iteration solution as the (new) global best solution, otherwise we neglect it and keep the (old) global best solution and consider it as the best (optimum) solution we found yet.

Step (4), At the end of each iteration we change the pheromone value at each route (ij) so that ant(s) in next iterations can make use of the available information about the (optimum) solution from previous iterations, this is known as stigmergy concept Dorgio[2], the change of pheromone value carry out in three steps in the following order :

A- Reducing the pheromone value by an amount (ℓ) i.e. evaporation rate.

B- Increasing the pheromone of the route of the last iteration by a quantity proportional to the goodness of the solution found by ant in that iteration.

C- Increasing the pheromone of the route that construct the global best solution by an amount proportional to the global best solution value.

The new pheromone value at the beginning of iteration (n) can found from the equation bellow

$$\mathbf{ph_{ij}^{(n)} = ph_{ij}^{(n-1)}(1-\ell) + T^{(n-1)} + T^{gb} * \sigma} \quad \dots (2)$$

where

ℓ Is the evaporation rate.

$T^{(n-1)}$ is the iteration solution, for example in traveling sales man problem it can be found as the reciprocal of the total distance traveled by ant (or time, cost,…,etc).

T^{gb} As $T^{(n-1)}$ but obtained from global best solution states..

σ Is the number of elite ants, Dorgio[1], where elite ants is an improvement introduced to enhance the performance of ASO.

We can list the following weakness point of the above algorithms

A. The ASO algorithms are designed to make moves only from a solution to a better (minimum) and refuse any other movement even if that movement will lead to a better solution after some iterations.

B. Because of (A) above the ASO are know to be rapidly converge to local optima and stuck in it, as a matter of fact ants has no ability to escape from local optima and the only way know release ants from local optima is to complete the iteration of an experiment Markle [3], and then begin a new experiment hoping that ant make use of re-initialization of pheromone trails and find new minimum route (i.e. new optima) before the effect of adding pheromone to the global best solution found and evaporation take place and bring the ants again to the previous optima and stack again

C. In general all metaheuristic algorithms should balance between intensification which mean improving the solution in hand and diversification which mean searching new parts of the solution space to find better solution. In case of ASO diversification take place only when a new experiment begin and as iterations goes on only intensification take place due to the effect of adding pheromone and evaporation.

Introducing the multistage ant system algorithm

The classical ASO algorithm consist of carrying out a predetermined number of independent experiments, each experiment consist also of a predetermined number of iterations, the global best solution found at the end considered to be the best (approximate optimum) solution of the problem being handled by the ASO, Markle[3].

In our modified algorithm (i.e. MS-ASO) we still work with a predetermined number of independent experiments but each experiment is further divided into stages (usually 5-9 stages) and each stage consist of predetermined number of iterations, now for odd stage (1, 3,…) ants work as usual and search for the shortest routes, but in even stages (2, 4,…) we force ants to search for longest route (instead of shortest), to do so we make the following modifications (for even stages only)

A-Changing the heuristic function that appear in transition rule (eq(1)), so it give higher probability for longest routes instead of shortest routes, for example in traveling sales man problem and vehicle routing problems the usual heuristic function is($1/d_{ij}$), where dij represent the distance between states (i) and (j), in even stages we can take the heuristic function equal to (dij)

B-In even stages at the end of the iterations the pheromone update procedure take place and pheromone added to the iteration solution route but the longer route will have more pheromone.

C-The elite strategy is also applied but the global best route here considered the longest route we found.

D-The termination criteria for even stages is different from the usual odd stages because we don't want to spend time and effort to search for (optimum) worst solution. Our stop criteria for even stage consist of fixing a predetermined number of iteration, usually taken to be(10-20)% of the number of odd stages iterations, if we don't have new longest route during the iterations then the global best long route we have will be considered again, but during the calculations if any new long route emerge we stop the iterations, taking this new long route as global best long route and begin a new odd stage, we stop the iteration since the new long route can be considered as an indication that ants now can begin the search for shortest path in next

odd stage from a new position of the solution space.

At the beginning of each experiment we carry first stage(1) searching for a new better shortest route using the initial values, and when the predetermined number of iteration for stage (1) is over then we left at this point with a pheromone trail structure and global best solution, after that we begin the second stage (i.e. stage 2) we begin the iterations using the available value of pheromone trail at each route, of course since we search for (new) longest route at this stage then the routes that belongs to the global best (shortest) route will lose pheromone due to evaporation and some routes that at minimum value of pheromone trail will gain some pheromone. The iteration of stage (2) continued until the predetermined number of iterations achieved or a new longest route emerge then we stop iterations of stage (2) and go to stage(3) looking for new shortest route and again we use the available values of pheromone trails without initialization, we continue until the last stage finished which should be odd. Fig(1-a) represent the usual ant colony algorithm and Fig(1-b) represent the suggested (new) algorithm the effect of new algorithm take place in steps 1, 2, and 4.

The main advantageous of our modified algorithm are listed below.

A-Ants are forced to search new parts of solution space when we execute even stages and begin their next search of shortest route from that new part.

B-Better diversification achieved and now become dependent on pheromone trail.

C-Ant now can be released from local optima without initialization of pheromone trail, this is achieved because the next odd stage begin with different values of pheromone trail i.e. not the same pheromone trails value that lead ant to stuck in local optima .

Testing multistage algorithm using benchmark problems

To see how good our modified algorithm (MS-ASO) we test it on (4) benchmark traveling sales man problems, all the information about these problems can be found on the website: http://elib.zib.de including the optimum tour length and the sequence of stations (or states). Two of these problems(eil51 and st70) are also used as a test problem when the Max-Min Ant System was introduced by Stuzle [4].

For these problems we use evaporation rate (ℓ) equal to 0.25, $\alpha = 1, 1 \le \beta \le 10$, number of experiment=5, number of stages per experiment=7, and other variables and parameters are shown in table (1).

The new routes we found for above problems are:

1. For eil51 the new optimum route is: 1-32-11-38-5-49-9-50-16-2-22-27-51-46-12-47-18-4-17-37-44-15-45-33-39-10-30-34-21-29-20-35-36-3-28-31-26-8-48-6-23-7-43-24-14-25-13-41-19-42-40.
2. For att48 the new optimum route is: 1-9-40-15-12-11-23-13-25-14-34-41-3-22-16-8-38-31-44-18-7-28-6-37-19-27-17-43-30-36-46-33-20-47-21-32-39-48-5-29-2-42-10-24-45-35-26-4.

3. For st70 the new route is: 1-36-23-16-47-37-58-50-10-52-60-12-21-34-33-54-62-48-67-11-56-65-64-51-5-53-66-63-22-59-38-69-31-13-29-70-35-57-15-24-19-7-2-4-18-42-6-41-43-17-9-40-61-39-45-25-46-27-68-44-30-20-14-3-32-8-28-26-49-55.

Now if we compare our work with that of Stuzle[4], when they introduce the Max-Min ASO, then our modified model can be found better in two things:

A. We didn't use any local optimizer and still able to achieve the optimum tour length or better while Stuzle[4] in their work they mention that they required 1000000 trial with local optimizer to achieve the published optimum tour.
B. The total number of iteration in our modified model is very much less than that of Suzle [4], they require 25 experiments (max 5 in our model) and 10000 iteration per experiment (max 5000 in our model) with local optimizer to achieve the optimum routes.

Conclusion and further work In this paper we introduced a modified Ant System optimization algorithm, we call it the Multistage Ant System optimization algorithm, the main advantageous of this new algorithm are:

A- Releasing ant from local optima without re-initialization of pheromone trails as usually done
B- Force the ant (not encouraging) to search new parts of the solution space which lead to a better diversification
C- Enable the ant to look ahead, i.e. accept bad solutions temporary in order to achieve or find better one later
D- Less number of iterations required
E- Don't required any local optimizer to achieve the known optimum solutions, which means that there is a good opportunity to enhance the known optimum routes further by applying suitable local optimizer

We test our modified algorithm (the multistage ant system) on (4) traveling sales man problem benchmark, we were able to improve the solution of three benchmark problems as shown in table (1) below. For further work we suggest the following main directions

A. Testing the MS-ASO on other types of the traveling sales man problem like asymmetric traveling sales man problems or weighted traveling sales man problems .
B. Testing the MS-ASO on other problems like vehicle routing.
C. Adding local optimizer to see its effect for further improvement on the MS-ASO solution.

References

[1] Dorgio M. and Stuzle T., "Ant colony Optimization", A bradford book, MIT press, England, 2004 .
[2] Dorgio M., and Gamberdella L. M. "Ant Algorithms for Discrete Optimization "; Artificial life , Vol. 5 , No. 2, pp 137-172, 1999 .

[3] Merkle D. and Middendorf M., "ant colony optimization with the relative pheromone evaluation method ", 3rd European workshop on evolutionary methods for AT planning, Ireland, LNCS 2279, 2002, PP325-333 .

[4] Stuzle T., and Hoos H., "improvement on the ant System: Introducing the Max – Min Ant system", 3rd International Conference on Artificial network Genetic algorithm, 1997.

Table (1): the data of the test problems with published optimum routes and our results

Problem Name	states	Optima tour length (published)	Number of iteration		Our tour length
			Even stage	Odd stage	
att48	48	33523	600	150	32200.02
eil51	51	426	400	50	419.3
st70	70	675	1000	200	673.19
kroa100	100	21282	5000	750	21317.5

Input:

1. No. of iterations(=N)
2. x and y coordinates for each city.
3. Pheromone initialization value.
4. No. elite (daemon) ants.
5. Evaporation rate =0.25.
6. $\alpha = 1, and\ 1 \le \beta \le 10.$

Output:

1. Global shortest route obtained by ant.

1. Construct new short route.
   A. If not visited all cities
      a. Find probabilities using $P_{ij}^{(n)} = \dfrac{\left[ph_{ij}^{(n-1)}\right]^a \left[y_{ij}\right]^b}{\sum_{k=1}^{M} \left[ph_{ik}^{(n-1)}\right]^a \left[y_{ik}\right]^b}$
      b. Generates random number and find next city.
   B. If all cities visited
      a. Find the rout length (d).

b. Evaporate pheromone for all using $ph_{ij}^{(n)} = ph_{ij}^{(n-1)}(1-\ell)$

c. Increase pheromone of last route using $ph_{ij}^{(n)} = ph_{ij}^{(n-1)} + \frac{1}{d}$

d. Increase global best route pheromone using $ph_{ij}^{(n)} = ph_{ij}^{(n-1)} + T^{gb} * \sigma$

If iteration route < global best route then
Replace global by best iteration route.
Else
If max no. of iteration achieved then
Print global best shortest route
Exit
Else
Goto step 1.

Figure (1-a) the pseudo code of ant colony optimization algorithm

Input:
1. No. of iterations(=N)
2. x and y coordinates for each city.
3. Pheromone initialization value.
4. No. elite (daemon) ants.
5. Evaporation rate =0.25.
6. $\alpha = 1, and\ 1 \le \beta \le 10.$
7. Number of trials for even stage and odd stages per iteration

Output:
1. Global shortest route obtained by ant.

1. Find number of odd and even stage by divide N by 2, odd value of iteration represent search for shortest routes while even represent search for longest routes.
Stage number =I
Do while I < N
2. If I is odd then
3. Calculation for shortest route
A. If not visited all cities
a. Find probabilities using $P_{ij}^{(n)} = \frac{\left[ph_{ij}^{(n-1)}\right]^a \left[y_{ij}\right]^b}{\sum_{k=1}^{M}\left[ph_{ik}^{(n-1)}\right]^a \left[y_{ik}\right]^b}$
b. Generates random number and find next city.
B. If all cities visited
a. Find the rout length (d).
b. Evaporate pheromone for all using $ph_{ij}^{(n)} = ph_{ij}^{(n-1)}(1-\ell)$

c. Increase pheromone of last route using $ph_{ij}^{(n)} = ph_{ij}^{(n-1)} + \frac{1}{d}$

d. Increase global best route pheromone using $ph_{ij}^{(n)} = ph_{ij}^{(n-1)} + T^{gb} * \sigma$

C. If iteration route < global best route then
Replace global by best iteration route.
Else
a. If max no. of iteration achieved then
Increment stage.
b. If max no. of odd stage achieved then
Print global best shortest route
Exit
Else
Goto step 2.
Else
Goto step 3.
Else
4. If I is even then
5. Calculation for longest route
A. If not visited all cities
a. Find probabilities using $P_{ij}^{(n)} = \frac{\left[ph_{ij}^{(n-1)}\right]^a \left[y_{ij}\right]^b}{\sum_{k=1}^{M} \left[ph_{ik}^{(n-1)}\right]^a \left[y_{ik}\right]^b}$
b. Generates random number and find next city.
B. If all cities visited
a. Find the rout length (d).
b. Evaporate pheromone for all using $ph_{ij}^{(n)} = ph_{ij}^{(n-1)}(1-\ell)$
c. Increase pheromone of last route using $ph_{ij}^{(n)} = ph_{ij}^{(n-1)} + d$
d. Increase global best route pheromone using $ph_{ij}^{(n)} = ph_{ij}^{(n-1)} + T^{gb} * \sigma$
C.. If iteration route > global route then
Goto step 3.C.b
Else
If max no. of iteration achieved then
Increment stage.
Goto step 3.C.b.
Else
Goto step 5.

Figure (1-b) the pseudo code of the multistage ant system optimization algorithm