Extended Ant Colony Optimization Algorithm (EACO) for Efficient Design of Networks and Improved Reliability

Mohd Ashraf and Rajesh Mishra

School of Information and Communication Technology
Gautam Buddha University, India
ashraf.saifee@gmail.com, rmishra@gbu.ac.in

Abstract. The problem of efficient network design is nothing but the NP hard problem which consisting of possible links subset selection or network topology to lower network cost subjected to the reliability constraint. Thus, in this paper we are presenting the new improved method of ant colony optimization in order to overcome such network design problem. This new algorithm is based on existing ant colony optimization algorithm. This new algorithm is having aim to optimize network reliability with least cost. This new proposed algorithm we called as extended ant colony optimizations (EACO) in which two new methods are presented, those two methods are used to optimize the search process for neighborhood and re-initialization process. Here we presented the practical approach with different network topologies in order to show the efficiency of proposed algorithm. The results of proposed method are compared with previous existing algorithms such as tabu search algorithm (TSA), genetic algorithm (GA), and ACO. From the simulation results, the proposed approach is better reliability as compared to existing algorithms.

Keywords: ACO, Network Reliability, Optimization, Heuristic, Topology.

1 Introduction

The set of links (or arcs) and nodes (or switches) a communication set of connections can be illustrate where all nodes are linked by links. The first one is backbone network and the next one is local access network (LAN). The classic communication network structure is collected of two levels the backbone network is dedicated for delivery information from source to destination (end to end) using its switch nodes. The LAN network is naturally regional system which access hosts or local servers allows users. This manuscript is focused only on distributed network. The least cost devices availability in market results in tremendous improvement in communication networks. The design of network topology is nothing but the network planning is responsible for building the feasible topology by considering the network constraints to satisfy. The distributed network is having great support to improve its reliability as compared to centralized networks. The network reliability is majorly depends on

K. Singh, A.K. Awasthi, and R. Mishra (Eds.): QSHINE 2013, LNICST 115, pp. 939–950, 2013.

network nodes, links and network topology designed. The reliability of fully connected network is more as compared to ring network.

There are many research presents over the network design problem subjected to the network reliability. The problem of network reliability is defined as two terminal network reliability problem as well as overall reliability problem. The efficient network topology design problem in which links are selected those are either minimizes the cost or maximizes the reliability and formulated as combinatorial problem. Such kind of problem is called as NP hard problem which is tough task to solve. There were many researchers presented methods to overcome such problems. Jan et al. [3] developed an algorithm using the decomposition approach based on brand and bound to minimize link cost of communication network subjected to reliability constraint. Aggarwal et al. [4] employed greedy heuristic approach to maximize reliability given a cost constraint for networks with different reliability of links and nodes. Pierre et al. [5] also used simulated annealing to find the optimal design for packet switching networks where delay and capacity were considered, but reliability was not. For the network design, Kumar et al. [6] developed a genetic algorithm (GA) considering diameter, average distance and communication network reliability then applied it to four test problems of up to nine nodes. Deeter and Smith [7] presented a GA approach for minimum cost network design problem with alternative link reliabilities and all-terminal network reliability constraint. Furthermore, Glover et al. [8] used tabu search algorithm (TSA) to choose topologies of network when considering cost and capacity but not reliability. Other work of TSA, Beltran and Skorin-Kapov [9] used TSA to design reliable networks for searching the least cost spanning two-tree where the two-tree objective was a coarse surrogate for reliability. Next presented algorithm was ACO [10, 11] which was applied over combinatorial optimization problem successfully such as vehicle routing problem, travelling salesman problem (TSP) etc. However still to the date it's not applied network topology design problem with an objective of reliability optimization and least cost.

Hence in this paper we are discussing the new approach of ACO called as extended ACO in order to address the network topology design problem by satisfying the constraint of maximum reliability and minimum cost. Following section 2 will discuss the problem statement, problem formulation and will discuss how to calculate reliability. Section 3 will discuss the existing and section 4 discussing proposed ACO algorithms. Section 5 discussed the computational result

2 Problem Statement and Formulation

Before the formulation of problem, we will have to assume following notations to be used.

For two-terminal reliability and all-terminal reliability network design problems, there are a set of N nodes with specified topologies, which can be originate from real networks or interpreted as Euclidean distance between coordinate on a plane. It only

L	Set of Possible link
l_{ij}	Option of each link
d_{ij}	Distant between node I and node j
N	Set of given link
n	Number of node
$p(l_k)$	Reliability of link option
$c(l_k)$	Unit cost of link option
x	Architecture of network
C(x)	Total cost of network
R(x)	Reliability of network
R_0	Minimum network reliability

Fig. 1. Notations

represents some costs of connection between two nodes anyway reveal that the distance is not an offered space, of connection type. The network nodes are implicit completely dependable or assumed not to fail under any conditions. There are a set of L links which connected all nodes in N. In this problem fully connected network. It is also unspecified that there is only one link per a location. Then, all links is failed separately and restore link is necessary if any link fail. The search space of runner solutions is related to the number of the possible links, which can be found by:

$$|L| = \frac{|N||N-1|}{2} \tag{1}$$

A link can maybe have extra than two states. Thus a runner solution, x, is related to the state of the possible links. Therefore, this problem concern with selection a state or connected level of links l_{ij}= k where k is the level that those links connected nodes ni and nj. The mathematical formulation for the problem when minimize cost subjected to a smallest amount network dependability constraint is:

$$\textbf{Minimize } C(x) = \sum_{i-1}^{n-1} \sum_{j=i+1}^{n} c_{ij}.l_{ij}.d_{ij} \tag{2}$$

Subject to R(x)≥ R0

The cost of a specific architecture, x, is given by C(x) and the reliability of x is given by R(x). The problem is to find x which minimize the connection cost subjected to R(x) > = R0 by considering the following assumptions:

(1) All the nodes in network location is given.
(2) The cost Cij and the operation probability P_{ij} of each link (i, j) are fixed.
(3) Every link is bi-directional.
(4) No redundant link is allowed in the network.

2.1 Reliability Calculation

The problem of a network is one of research areas related to the economic network calculating or estimating the reliability of design. There are two main approach for finding the reliability which are the exact calculation through analytic methods it is tricky to find the correct dependability since these methods generally drop competence when network approaches a fully related state. And the estimation calculation through for all-terminal network reliability problem, There are also the upper and lower bound expressions for network reliability still, they drop the useful. Furthermore many bound procedures and superior efficiency surrogates in all-terminal design process. Which is related in this examine, simulation depend on the statement that all links have the equal using a backtrack procedure determination the arrangement reliability was calculated.

Due to the computational obedient size, a algorithm [12] is used to correctly calculate the global reliability R(x). The outline of the backtracking algorithm is given as follows:

Step # 1: [Initialize] Label all the node of communication network from 1 to N and all the link from 1 to M, where N is the number of node and M is the number of link between the nodes.
Step # 2: Represent the network with incidence matrix I.
Step #3: Generate all possible combination of link $^{M}C_{N-1}$ with N-1 link out of M link.
Step # 4: Repeat step #5 to step# 7 until the link combination list is empty.
Step # 5: Create the sub graph by taking the entire node (N) and add the N-1 link from reading link combination list.
Step # 6: Generate an adjacency matrix A corresponding to sub-graph.
Step# 7: Call Graph_connectivity (g (V, E), s), apply to the sub graph.
    For all node v do
        /* Check the status of all node in Sub graph*/
    If (STATUS[v]=3)
        then Store the sub graph as Spanning tree and Go to step# 4.
    Else
        Reject this combination of link (Sub graph). Go to step# 4.
    End of step # 4 loop
Step# 8: Display all spanning tree generated in step #7 (a).
Step # 9: R(x) = c/t.

2.1.1 Procedure Graph_Connectivity (g (V, E), s)

The graph_connectivity is a sub-algorithm which is used to check the connectivity of a sub graph. This algorithm visit all the node of sub graph and store the visited status in data structure array, STATUS[ ]. If the graph_connectivity ( ) visited all the node of the graph, Graph is connected otherwise graph is disconnected.

The general idea behind this algorithm beginning at a start node A is as follows. First we examine the starting node A. Then we examine each node v along a path P which begins at A; that is, we process a neighbor of A, then a neighbor of a neighbor of A, and so on. After coming to a "dead end" that is, to the end of path P, we backtrack on P until we can continue along another path P and so on. A field STATUS is used to tell us the current status of a node.

During the execution of our algorithms, each node v of G will be in one of three states, called the status of v, as follows:

STATUS [vi] ←1: (Ready state) The initial state of the node N.
STATUS [vi] ←2: (Waiting state) The node is in stack, waiting to be processed.
STATUS [vi] ←3: (Processed state.) The Node v has been processed.
Step #1: Initialize all the node to ready state STATUS[v]←1
Step #2: call Push(STACK, s) ;
/* push() used to insert the vertex on the top of the stack where STACK [1,-- n] be an array implementation of stack, s is the starting vertex/*
Set STATUS[s] ←2 ;
Step # 3: While stack is not empty
Step # 4: call Pop(STACK, v)
/* Remove the top node of stack an become visited node N */
set STATUS [v] ←3; /* visited node*/
Step #5: F**or** each neighbor of processed node v
If (STATUS [next node]= 1) /*ready State of the node*/
Then
call PUSH(STACK, v) ;
/* insert the adjacent node of N to the top of the stack*/
Set STATUS[v] ←2 ; /*waiting State*/
If (STATUS [next node]= 2)
Call Pop(STACK, top) ; /*delete the current node from the stack*/
Set STATUS [v] ←3;
call PUSH(STACK, v)
/*Insert the Adjacent node of v which have STATUS[v] ←1 */
If (STATUS[next node]= 3)
ignore the vertex.
END for
END while
Step # 6: **END** graph connectivity

When the need of a network's reliability simulation has been arisen, two issues become important. One is the biased estimator. Others are the difference of inference. Every referenced technique is a balanced estimator where the variance of the method describes above is:

$$Var(R(x)) = \frac{R(x)(1-R(x))}{t} \tag{3}$$

To get more accurate consistency estimation, t should have larger value.

3 Conventional ACO

3.1 Ant Colony Basic Principle

The study was further continued with positive feedback distributed computation and the use of a constructive greedy heuristic is the uniqueness of an artificial ant colony. Positive feedback accounts for rapid discovery of good solutions, distributed computation avoids premature convergence and the greedy heuristic helps to find acceptable solutions in the early stages of the search process. The authors apply this approach to the classical TSP, asymmetric TSP, quadratic assignment problem (QAP) and job-shop scheduling for demonstrate the AS approach. The AS shows very good results in each applied area. Just recently, Dorigo and Gambardella have worked on extended versions of the AS paradigm. ACO is one of the extensions and has been applied to the symmetric and asymmetric TSP with excellent results. Other combinatorial optimization problems such as the vehicle routing problem has been successfully applied to the Ant System ACO is an algorithm which was inspired by the behavior of real ants. Ethnologists have deliberated how blind animals such as ants capable of finding the shortest path from food sources to the nest without using visual cues. They are also able to adapt changes in the Environment.

3.2 ACO for Network Design Problem

In this section we will discuss, how to apply ACO over network design problem. For an application of ACO algorithm to design a network topology, it is convenient to represent the network by a graph G = (N, E) where N is the set of nodes and E is the set of links. Ant's support uses the indirect form of communication mediated by pheromone they drop on the links of the graph G while building solutions. For example, a two-terminal network has four nodes and five links as showing in figure 2 below. It is possible to choose four levels for each link as shown in this network can be modeled as the routes between nest and food source for ACO as shown in.

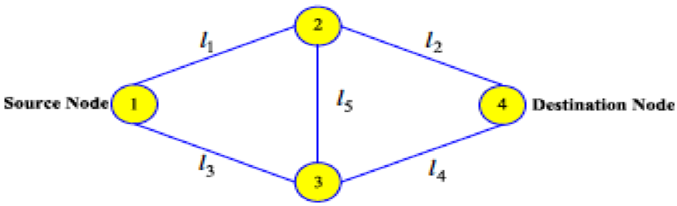

Fig. 2. Two terminals Network

This model reveals that the topology of network can be constructed from the selected connection level of each link, which seems to be the ant's route between nest and food source. In general, the procedure of ACO algorithm can be described as follows: m ants are initially positioned at the nest. Each ant will choose a possible route as a solution. In fact, each ant builds a feasible solution (called a tour) by repeatedly applying a stochastic greedy search called the state transition rule. Once all ants have terminated their tours, the following steps are performed. The amount of pheromone is modified by applying the global updating rule.

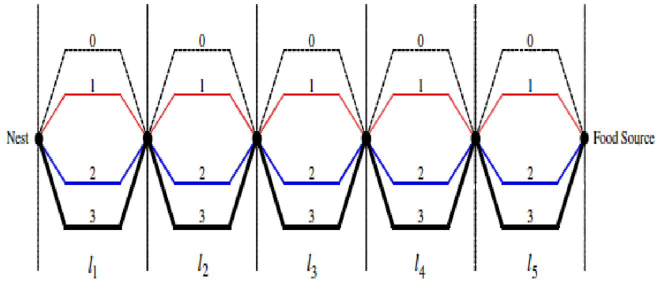

Fig. 3. Network mode as the routes between nest and food source

The pheromone updating rules are designed so that they tend to give more pheromone to edges, which should be visited by ants. A flowchart of a conventional ACO algorithm is shown.

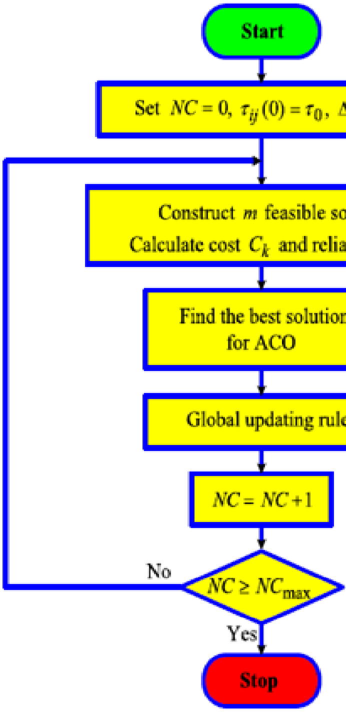

Fig. 4. Existing ACO Algorithm

3.3 Conventional Ant Colony Optimization Algorithm

Below is the description of Ant colony Optimization Algorithm

Step# A1: Initialization

```
Set NC=0  // NC: Cycle counter
For every combination( i, j)
   Set an initial value τij(0) = τ0 and Δτij=0
End
```

Step #A2: Construct feasible solution

```
For k-=1 to m  // m: number of ants
   For i=1 to n   // n: number of links
      Choose a level of connection with transition probability by given
   End
   Calculate cost Ck  //  Ck : cost for each ant
   Calculate reliability Rk  // Rk  : reliability for each ant
End
Update the best solution
```

Step #A3: Global Updating rule

```
For every combination(i,j)
    For k=1 to m
        Find Δτ^k_{i.j}
    End
    Update Δ τij
End
Update the trail value
Update the transition probability
```

Step # A4: Next Search

```
Set NC=NC+1
   For every combination (i,j)
      Δτij=0
   End
```

Step# A5: Termination

```
If (NC< NCmax)
   Then
      Go to Step A2
      Print the best feasible Solution
      Stop
   End
End
```

4 Proposed Extended Ant Colony Optimization Algorithm

One of the strong limitations of existing ACO is that all ants are taking same place. The algorithm may be intent in a local optimal point if this situation occurs. To improve the stagnation problem of conventional ACO algorithms, two improvement procedures are applied in order to improve the ant colony optimization method for better solutions or escape from this solution in order to guarantee ants diversity. This approach is called extended ant colony optimization (EACO). The additional procedures are a specific improvement algorithm (called neighbourhood search) and re-initialization. The neighbourhood search algorithm is shown in Step E3 in EACO's algorithm, and it is proceeded to change in turn each connection level of chosen link by another connection level. For each link, connection levels are indexed in ascending order in accordance with their reliability.

4.1 Extended Ant Colony Optimization Algorithm

```
Step#B1: Initialization
          Set NC=0  // NC: Cycle counter
          For every combination( i, j)
              Set an initial value τij (0) =τ0  and Δτij=0
          End
Step#B2: Construct feasible solution
           For k-=1 to m  // m: number of ants
              For i=1 to n   // n: number of links
                  Choose a level of connection with transition probability by given
              End
                   Calculate cost C_k  //  C_k : cost for each ant
                   Calculate reliability R_k  // R_k  : reliability for each ant
           End
               Update the best solution
   Step #B3: Apply the neighborhood search
               For k=1 to m
                   For i=1 to (2*n)
                       If i= odd
                           Change the chosen level of link i with level p by level p+1
                       Else
                           Change the chosen level of link i with level p-1
                   End
                           Calculate reliability R_k
                       If   ( R_k ≥ R_0)
                            Except for exchanging
                            Record the obtained solution
                       Else
                             Do not except for exchanging
                   End
                             Calculate the cost C_k
                   End
               Update the best solution
   Step #B4: Next Search
              Set NC=NC+1
                 For every combination (i,j)
                     Δτij=0
                 End
   Step #B5: Re-initialization
              If the best solution has not been improved for a long time
               Then
```

Set an initial value τij(0)= τ0

End

Step #B6: Global updating rule

For every combination(i,j)

For k=1 to m

$Find \vartriangle \tau_{i.j}^{k}$

End

Update Δ τ_{ij}

End

Update the trail value

Update the transition probability

Step #B7: Termination

If (NC< NCmax)

Then

Go to **Step#B2**

Print the best feasible Solution

Stop

End

End

4.2 Flow Chart for Extended Ant Colony Optimization

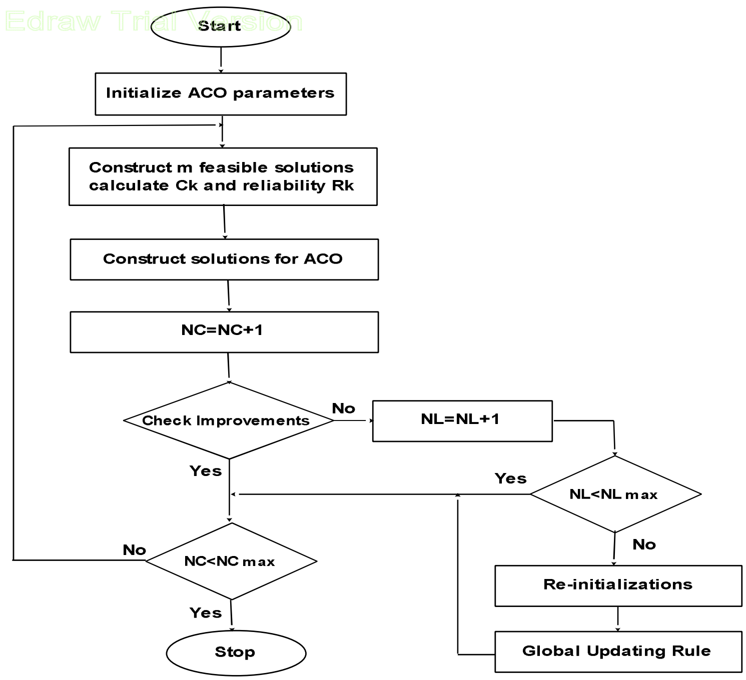

Fig. 5. Flow chart for proposed extended Ant colony optimization

5 Computation Result

The effectiveness of the proposed EACO algorithm has been evaluated with different network topology designs and compares its performance with existing ACO, TS, and GA approaches. Each studied system was run 30 times with differential random initial solutions. In order to evaluate the performance of each technique following graphs showings the cost required and reliability computation for each approach in case of solving Schaffer function and sphere function:

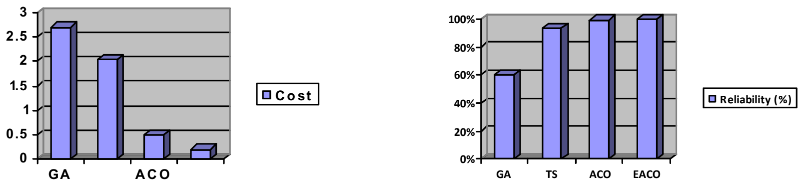

Fig. 6. The results obtained by GA, TS, ACO and IACO for solving Schaffer function

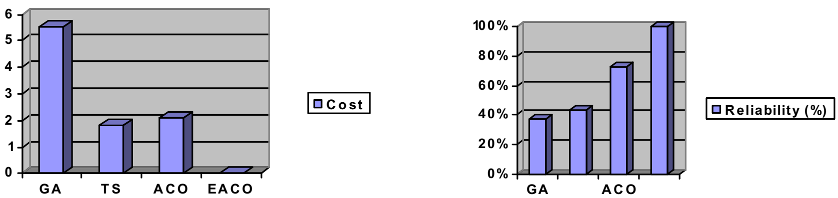

Fig. 7. The results obtained by GA, TSA, ACO and IACO for solving Sphere function

6 Conclusions

Thus in this paper we presented the EACO for optimal reliability for network design with least cost. The proposed EACO method has been useful to solve the topology network design problem considering both economics and reliability. The shows superior features such as high-quality solution, stable convergence characteristic and good computation efficiency EACO algorithm. Convergence characteristic and computation efficiency compared with GA, TSA and ACO methods Above results which we perform with sample test function showing that proposed algorithm having better improvements in reliability and having least cost. For future work we will apply this algorithm over graph theory based networks and evaluate its results with existing cases.

References

1. Aho, A.V., Hopcroft, J.E., Ullman, J.D.: Design and analysis of Computer Algorithms. Addison – Wesley, Massachusetts (1974)
2. Gerla, M., Frank, H., Eckl, J.: A cut saturation algorithm for topological design of packet witched communication networks. Proc. NTC, 1074–1085 (1974)
3. Rong-Hong, J., Fung-Jen, H., Sheng-Tzong, C.: Topology optimization of a communication network subject to a reliability constraint. IEEE Trans. Reliab. 42(1), 63–70 (1993)
4. Aggarwal, K.K., Chopra, Y.C., Bajwa, J.: Reliability evaluation by network decomposition. IEEE Trans. Reliab. R-31, 355–358 (1982)
5. Pierre, S., Hyppolite, M.A., Bourjolly, J.M., Dioume, O.: Topology design of computer communication network using simulated annealing. Eng Appl. Artif. Intell. 8, 61–69 (1995)
6. Kumar, A., Pathak, R.M., Guptaand, Y.P., Parsaei, H.: A genetic algorithm for distributed system topology design. Comput. Ind. Eng. 28, 659–670 (1995)
7. Deeter Darren, L., Smith Alice, E.: Economic design of reliable networks. IEEE Trans. Econ. Reliab. Eng. (July 1998)
8. Glover, F., Lee, M., Ryan, J.: Least-cost network topology design for a new service: an application of a tabu search. Ann. Operat. Res. 33, 351–362 (1991)
9. Beltran, H.F., Skorin-Kapov, D.: On minimum cost isolated failure immune networks. Telecommun. Syst. 3, 183–200 (1994)
10. Colorni, A., Dorigo, M., Maniezzo, V.: Distributed optimization by ant colonies. In: Proceedings of the European conference on artificial life, pp. 134–142 (1991)
11. Dorigo, M., Maniezzo, V., Colorni, A.: Ant system: optimization by a colony of cooperative agents. IEEE Trans.Syst. Man. Cybernet B: Cybernet 26(1), 29–41 (1996)
12. Ashraf, M., Rajesh, M.: Computing global reliability for wireless sensorNetworks subject to spanning tree enumeration approach. In: International Journal of Computer Science and Engineering (IJCSE), vol. 1(1), pp. 1–14 (Auguest 2012)