FoF-R Ant-based Survivable Routing Using Distributed Resilience Matrix

William Liu[1] , Harsha Sirisena[1] and Krzysztof Pawlikowski [2]
[1]Electrical and Computer Engineering
[2]Computer Science and Software Engineering
University of Canterbury, Christchurch, New Zealand
Email: william.liu@elec.canterbury.ac.nz, {harsha.sirisena, krys.pawlikowski}@canterbury.ac.nz

***Abstract*— Fast recovery from failures and overall high utilization of network capacity are two primary goals of network survivability design. Shared backup path protection has been shown to be efficient in terms of capacity utilization, due to the sharing capability among protection paths. However, the resulting Integer Linear Programming (ILP) formulation of the problem is known to be NP-hard. This paper tackles the survivable routing problem using a new distributed matrix-based structure, termed Resilience Matrix (RM), for capturing the local bandwidth usage information. Additionally, a heuristic ant-based routing algorithm, named Friend-or-Foe Resilient (FoF-R), is proposed for finding the optimal protection cycle (i.e., two node-disjoint paths between a source-destination node pair) and for exploring the sharing ability among protection paths using a headroom-dependent attraction and repulsion functions. Simulation results based on the OMNeT++ tool show that the FoF-R scheme with the distributed RM structure is a promising approach to solve the survivable routing problem and it gives a good trade off between solution's optimality and the time needed for finding a solution.**

Keywords- Survivable routing; protection cycle; heuristic; ILP; resilience matrix; ant colony optimization; attraction/repulsion

I. INTRODUCTION

Survivability is one of the most important design issues of future multi-service Next Generation Networks (NGNs). Survivable network design pre-plans the topology as well as the spare capacity allocation (SCA) of network's links in case of failures. As stated in [1], and [2], the shared backup path protection (SBPP) scheme is a promising approach to reduce the consumption of spare capacity, at the cost of a slightly longer recovery time. The amount of bandwidth sharing that can be achieved depends on how detailed is the bandwidth usage information available at the source nodes. This additional information is necessary for the source nodes to identify the dependencies between the working and protection capacity associated with each pair of links, from which the link metric for routing the path-pairs can be derived. The study in [3] proposed to use the spare provision matrix (SPM) structure, which allows concise modeling of such dependencies in a matrix form.

Based on the graceful SPM structure, some studies have proposed Integer Linear Programming (ILP)-based solutions to tackle the survivable routing problem. For example, in Sharing with Complete Information (SCI) [4], a routing algorithm calculates the cost function of a given pair of paths using the knowledge of all the existing working and protection paths. Thus, access to full per-flow information about the network is necessary. However, this approach could be constrained by the significant overhead due to dissemination of information about link-states and resulted in poor scalability. Therefore, other schemes were proposed to reduce the amount of overhead at the expense of a smaller degree of resource sharing. In the Shared with Partial Information (SPI) scheme, introduced in [4], the routing process needs to know the aggregated amount of working and protection capacity along each link for the path-pair selection, in order to reduce the overhead. A more advanced scheme called Distributed Partial Information Management with Sufficient and Aggressive cost estimation and Minimum bandwidth allocation (DPIM-SAM) introduced in [5], offers a better capability of link-state information dissemination than SCI and a higher degree of spare capacity sharing than SPI. The basic idea of DPIM-SAM is that each node estimates a local value for the maximum entry along each row of SPM it would have observed, instead of considering the whole SPM as it is done in SCI. In [6], an improvement of link-state information dissemination process, called Sharing with Reduced Information (SRI), was introduced, to take advantage of the Singular Value Decomposition (SVD) technique performed at each node for increasing the precision of SPM reconstruction. The study showed much improvement of the precision of SPM reconstruction.

In this paper, instead of using the estimation method or the complex SVD transformation to reconstruct the SPM in a distributed way, we propose a novel resilience matrix (RM) structure to decompose the exact SPM into the distributed RM matrices of individual nodes. It significantly mitigates problems related to the difficulty of precise estimation of sharable spare capacity.

An important integral part of any implementation of the RM structure is signaling, as it directly affects how the maintained information is updated and exchanged. Inspired by the principles of ant colony optimization [7-10], we propose the Friend-or-Foe Resilient (FoF-R) algorithm for identifying a set of protection cycles between every node pair, to exchange the bandwidth usage information and to update parts of the RMs at each node. Consequently, the protection cycles for each source-destination pair are always available and updated before the new request arrives and the best cycle in terms of the joint working and protection capacity can be easily selected.

The remainder of this paper is organized as follows. Section II introduces the novel RM structure that captures the network bandwidth usage information. The FoF-R routing algorithm is described in Section III. Section IV presents the results of our numerical comparison between FoF-R, SBPP, p-cycle and Dedicated Protection (DP) solutions. The conclusions are drawn in Section V.

II. PROPOSED RESILIENCE MATRIX STRUCTURE

A network is represented by an undirected graph with N nodes, L links and R traffic flows. The working path (WP) and protection path (PP) of flow r are represented by two 1xL binary row vectors, respectively. The l-th element in the vector equals to "1" if and only if the corresponding path uses link l. The path link incidence matrices for WPs and PPs are the stacks of row vectors, forming two RxL Matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$, respectively**.** $\boldsymbol{D}$ is defined as a diagonal RxR matrix and $D_{r,r}$ representing the bandwidth demand of the rth flow. Therefore, the SPM in [3] is defined as $\boldsymbol{G}=\boldsymbol{Q}^{T}\cdot\boldsymbol{D}\cdot\boldsymbol{P}$, which is a LxK matrix and each element g_{lk} in $\boldsymbol{G}=\{g_{lk}\}_{LxK}$ is the minimum spare capacity required on link l when link k fails. Note that $K=L$ when protecting all single link failures. The minimum spare capacity required on each link in case of any single failure is denoted by the column vector $\boldsymbol{s}=\{s_l\}_{Lx1}$, where $\boldsymbol{s}$=max $\boldsymbol{G}$ denotes a vector with maximum elements of the corresponding rows of $\boldsymbol{G}$**.** An example of the SPM structure for a five-node network is shown in Figure 1 below, here the bandwidth matrix $\boldsymbol{D}=\boldsymbol{I}$, where $\boldsymbol{I}$ is the identity matrix of size R.

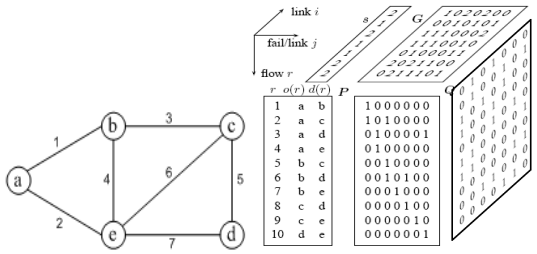


Fig. 1. SPM structure example for a five-node network [3]

In above SCI scenario, the maximum extent of spare resource sharing is achieved by assuming that each node must have per-flow information acquired from the matrix $\boldsymbol{P}$ and $\boldsymbol{Q}$. In addition, the lth column in $\boldsymbol{G}$ is advertised by each node by flooding the network. This yields a complexity O (L^2), leaving SCI impractical to implement in a distributed control environment. To tackle this problem, we propose a novel structure named Resilience Matrix (RM), which bears the same complete dependencies to obtain the $\boldsymbol{s}$ vector through the simplified additive operations, instead of the multiplicative operations as above. Thus, its resulted complexity equals O (L).

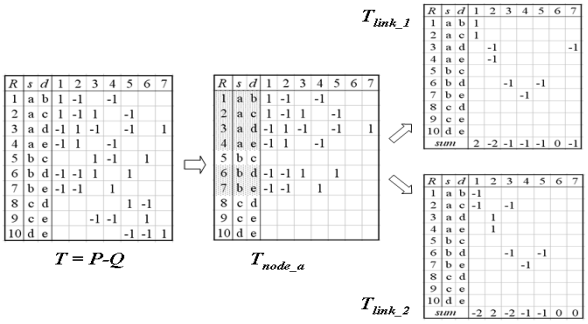
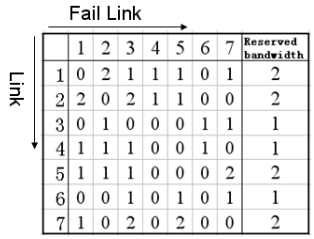


Fig. 2. Resilience Matrix (RM)

Let $\boldsymbol{T}=\boldsymbol{D}\cdot(\boldsymbol{P}-\boldsymbol{Q})$ denote the complete resilience matrix RM. Each "-1" represents a protection capacity, while "1" means a working capacity, see Figure 2. In each row, we can identify the capacity used and the links traversed by the WP (if "1") and PP (if "-1") for each flow. In each column, the sum of "1" entries represents the total amount of working capacity used by traffic flows in that link and the "-1"s represent the capacity reserved as protection capacity in each link by other WPs. For example, at node a, there are two outgoing links: link 1 and link 2. The 1st and 2nd columns of RM record the flows which traverse them and the usage status. If the status is "protection", represented by "-1", then from a horizontal view of that row, we can determine the corresponding links, denoted by "1"s, traversed by its WP. With these features, we can decompose the complete $\boldsymbol{T}$ into matrices of individual nodes. For example, the local RM in node a, denoted as $\boldsymbol{T}_{node\_a}$ only needs to record the bandwidth usage information related to link 1 and link 2. The $\boldsymbol{T}_{node\_a}$ can be further decomposed into two RM matrices, one for each outgoing link i.e., $\boldsymbol{T}_{link\_1}$ and $\boldsymbol{T}_{link\_2}$. The mapping rules between each node's RM and its outgoing links' RM is as below:

- If flow r traverses the link l in its WP, then $\boldsymbol{T}_{link\_l}[r, l]$=1
- If flow r traverses the link l in its protection path, then $\boldsymbol{T}_{link\_l}[r, k...]$=-1, where $k...$ indicates all links traversed by the WP of flow r

For example, flow 1 uses link 1 in its WP, so $\boldsymbol{T}_{link\_1}[1,1]$=1 here, while for the flow 3, link 1 is used in its PP and links 2 and 7 are in its WP, so $\boldsymbol{T}_{link\_1}[3,2]$ =-1 and $\boldsymbol{T}_{link\_1}[3,7]$ =-1. After mappings, a column summation is conducted on the $\boldsymbol{T}_{link\_1}$ to obtain a row vector $\boldsymbol{T}_{link\_1}[k]$ = [2, -2, -1, -1, -1, 0, 1]. The negative numbers represent the minimum spare capacity required on link 1 when link k fails. We can see that link 1 is occupied with 2 working capacity and at least 2 protection capacity i.e., as "-2" is reserved for any single failure. This is exactly information derived from the 1st row of vectors $\boldsymbol{s}$ in SPM, as seen in Figure 3.



Link \ Fail Link	1	2	3	4	5	6	7	Reserved bandwidth
1	0	2	1	1	1	0	1	2
2	2	0	2	1	1	0	0	2
3	0	1	0	0	0	1	1	1
4	1	1	1	0	0	1	0	1
5	1	1	1	0	0	0	2	2
6	0	0	1	0	1	0	1	1
7	1	0	2	0	2	0	0	2

Fig. 3 Spare Provision Matrix (SPM) and column vector $\boldsymbol{s}$

Therefore, it would be beneficial to transmit to the source nodes only these two derived values (2, -2) from $\boldsymbol{T}_{link\_1}$, informing about the working and spare capacity required in link 1, as it is sufficient for calculating the link metric needed for making the decisions on working and protection path-pair routing. We note that, in the distributed RM structure, the essential information on the links traversed by the corresponding WP need to be known to its protection links. This information exchange needs to be implemented by a special distributed signaling mechanism.

In the following section, we introduce the FoF-R ant-based routing algorithm to implement the above distributed signaling scheme as well as to jointly optimize the WPs and PPs routing problem. We will describe how the heuristic FoF-R algorithm

can find optimal protection cycles and also explore the sharing potentials among PPs in the network.

III. FRIEND-OR-FOE RESILIENT ANT-BASED ROUTING

The ant-based routing algorithms developed so far have concentrated on the coordination behavior between agents, and there has been little work put on exploring it for the survivable routing and SCA problems. Competitive behavior for disjoint path finding and load balancing was introduced in [11] and [12]. The former introduces a collaboration and competition strategy, known as Multi-type Ants Colony Optimization System, by using ants with different types of pheromones: ants cooperate with others if pheromone is of the same type as their own, or they compete if pheromone is of a different type. In [12], a distributed routing algorithm based on the cross-entropy method is proposed. These studies only concentrated on disjoint path finding and load balancing, and do not target the survivable routing and spare capacity allocation combinational optimization problem. The survivable routing for shared path protection application could benefit most from a certain level of repulsive behavior between agents. Therefore, we propose the Friend-or-Foe Resilient (FoF-R) ant-based routing algorithm, which is armed with an attraction and repulsion relationship function. Compared with the traditional ant algorithm, the FoF-R algorithm requires an ant agent with an ability to identify the relationship to previously laid pheromone, so as to decide what action to take: "friend" (i.e., attraction) or "foe" (i.e., repulsion). If "friend" is recognized, the idea is vivid: friends are attractive to each other to traverse the same PP to maximize the bandwidth sharing. On the other hand, the recognition of "foe" makes the ants detest each other, which means they will follow a disjoint route and thereby avoid overloading the PP. A concept of Bandwidth Headroom Function (BHF) is introduced here to aid modeling of the Friend or Foe relationships.

Let a flow r be specified by its source-destination node pair (s, d) and bandwidth b_r. Let W_l and P_l represent the sets of WPs and PPs that are traversing link l, respectively. Let C_l be the total bandwidth, B_l^W be the total bandwidth used by all WPs and B_l^P be the amount of sharable bandwidth reserved by PPs in link l. They can be calculated from $\boldsymbol{T}_{\boldsymbol{link\_l}}$ as follows:

$$B_l^W = \sum_{r \in W_l} b_r \text{ and } B_l^P = \max\{b_r\}, r \in P_l \text{ in } \boldsymbol{T}_{\boldsymbol{link\_l}} \qquad (1)$$

Thus, the Bandwidth Headroom Function (BHF) of link l, h_l, which indicates the ratio of the residual bandwidth to the total capacity of link l, is defined as:

$$h_l = 1 - \frac{(B_l^W + B_l^P)}{C_l} \qquad (2)$$

The relationship function for link l is

$$R_l(h_l) = -a \cdot h_l + \frac{b}{h_l^2} \qquad (3)$$

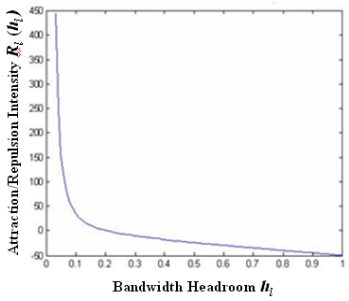


Fig. 4. An Example of Relationship Function R_l (h_l)

Figure 4 shows an example of R_l (h_l) function, which can model a trust degree from unbounded repulsion to a linear attraction as h_l increases. The parameter a, b are positive constants, with a representing the degree of attraction and b the degree of repulsion. The function R_l (h_l) is attractive (i.e., - a h_l dominates) for large h_l or repulsive (i.e., b/h_l^2 dominates) for small h_l, which is consistent with inter-individual attraction and repulsion phenomenon in biological swarms [13], [14]. We can adjust the (a, b) pair to specify the different trust degree in this relationship function.

FoF-R algorithm can implement SCI scenario in a distributed environment. The information about the sets of W_l and P_l is delivered to the local node whenever an ant passes by. We use Ant_W to denote FoF-R ant with WP search status, and Ant_P as FoF-R ant with PP search status. The subscript m and n represent the different connection requirements of the ants. Thus, the joint WP and PP optimization heuristic rules of FoF-R algorithm are described as follows:

- Heuristic Rule 1: If R_l (h_l) exceeds the bandwidth utilization threshold of a link l, there will be an unbounded repulsion to discourage any new ant from using this link. Otherwise, there will be a general relationship function with the adjustable attraction/ repulsion factors a and b
- Heuristic Rule 2: For Ant_W_m and Ant_W_n, $m \neq n$, there is a medium weight on repulsion force to encourage the same (s, d) pair ants to find different protection cycles for path diversity and load balancing
- Heuristic Rule 3: For Ant_W_m and Ant_P_n, $m=n$, there is a strong repulsion force to avoid using same path for WP and PP, so as to find disjoint paths (i.e., to form a protection cycle) for the same connection request
- Heuristic Rule 4: For Ant_P_m and Ant_P_n, $m \neq n$, if their WPs have common components, there is a medium weight on repulsion to discourage the ants from using the same PP, so as to avoid a common link's failure impacting on multiple PPs. Otherwise, there is a strong attraction force to encourage the sharing among PPs

To implement the above heuristic rules, every FoF-R ant need to carry a memory stack $M_{s \to d \to s}(k)$ of data, where the k refers to the k-th visited node in its cyclic journey. Let k be any network node, and j be an entry in the routing table of node k, to indicate a possible destination. Let N_k be set of neighboring nodes outgoing from node k and P_{di} be the probability for the ant jumps from node k to node i, $i \in N_k$, while traversing to the destination d and $d \neq k$. Then, for each

of the N entries in the routing table of node k, there will be n_k values of P_{di} with $\sum_{i \in N_K} P_{di} = 1$, $d = 1\ldots, N$.

The FoF-R ant-based distributed routing algorithm can be described as follows:

START
{

Routing Table and T_{link_l} Database Initialization: for each node k, the routing table is initialized with a uniform distribution P_{di} and the T_{link_l} is a RxL matrix for each link l outgoing from node k:

$$P_{di} = \frac{1}{n_k}, \ \forall i \in N_k \text{ and } T_{link\_l}[R][L] = \{0\}$$

DO (in parallel)
{

STEP 1: In regular time intervals, each node s launches a FoF-R ant to a randomly chosen destination d, its mission status is set as Ant_W

DO (in parallel, as each Ant_W)
{

STEP 2: Ant_W pushes in its stack the node k, previous link l identifiers traversed and the time between its launching from s to its arriving to k. Ant_W selects the next node to visit in the following way:

It draws between i nodes, $i \in N_k$, where each node i has a P_{di} probability in the routing table to be selected.

IF A cycle is found
FoF-R ant pops from its stack all data of cycle nodes to avoid infinite loops
END IF

} **WHILE** jumping node $i \neq d$

STEP 3: FoF-R ant changes its mission status to Ant_P when FoF-R ant arrives at destination d

DO (in parallel, as each Ant_P)
{

In its return journey, FoF-R ant delivers and maps the traversed links information into T_{link_l}, which is the RM for each outgoing link from node k following the mapping rules. We can obtain the h_l to calculate $R_l(h_l)$ and then P_{dl} according to the heuristic rules 1-4, so as to select the next hop to jump

} **WHILE** ($k \neq s$)

STEP 4: The source node will evaluate the degree of goodness associated with the capacity usage of each protection cycle collected by FoF-R ant by using the following additive cost function:

$$L(s,d) = \sum_{l \in cycle(s,d)} R_l(h_l)$$

STEP 5: Update the routing table of each node visited in the best cycle, i.e., increasing the pheromone probabilities of links that cycle used and decrementing, by normalization, the other links' pheromones.

STEP 6: All the protection cycles are constantly updated, thus the source node can easily select the best quality cycle during each connection setup.

} **END**

The internal structure of each FoF-R network node implemented in OMNeT++ is illustrated in Figure 5.

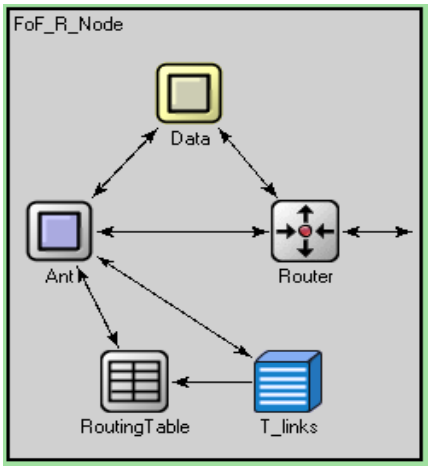


Figure 5. The FoF-R node structure

- **Router** is responsible for traffics queuing modeling and switching the FoF-R ant message and data traffic to other routers.
- **Data** is the submodule that generates and receives the data traffic and collects related statistic values.
- **Ant** is handling of FoF-R ant-based routing implementation e.g., ant generator and sink. It processes the FoF-R ant message such as loading and delivering the traversed nodes, links information and elapsed time into local node when an ant passing,.
- **RoutingTable** holds the routing information of the node. In our study, the routing information has the next hop probability values, which can be calculated from the **T_links** database for each outgoing links in the node.
- **T_links** is the Resilience Matrix (RM) database for recording the traversed links on the working and protection paths for each traffic flow, used to calculate next hop selection probability i.e., to explore the sharing ability among protection paths.

The above heuristic rules and ant movement behavior encourages the ants to find the protection cycle consisting of links with larger BHF, so as to minimize the total bandwidth usage in the network. The detailed local and global pheromone table updating formula and attraction/repulsion parameters configuration can be found in [15].

IV. SIMULATION RESULTS

The FoF-R ant routing algorithm and distributed RM structure presented in this paper have been investigated by simulation on a PC with Intel(R) Celeron(R) 1.70GHz, 504MB of RAM, using OMNeT++ discrete event simulator [16]. The ILP formulations for other three protection schemes i.e., SBPP, p-cycle [17], [18] and Dedicated Protection (DP) were solved using AMPL/CPLEX 11.1 [19], [20].

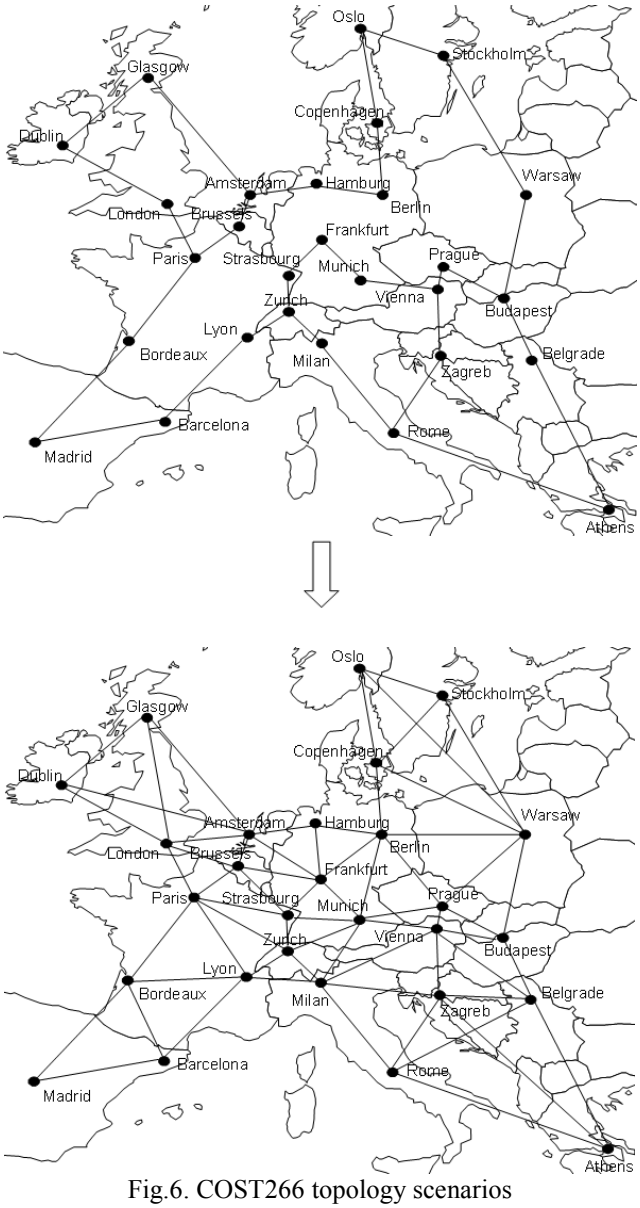


Fig.6. COST266 topology scenarios

The basic reference network considered in this paper is the pan-European fiber-optic network adopted from the IST project LION and COST Action 266 [21]. We studied 12 versions of COST266 topologies. Figure 6 depicts two limit cases considered: the most sparse topology with 31 links (upper case) and the most dense topology with 64 links (lower case). Without loss of generality, we assume symmetrical traffic flows, i.e., *R* flows between any pair of nodes and each flow has one unit bandwidth demand for easy comparison. We compared FoF-R, SBPP, p-cycle and DP solutions by considering the total amount of working and protection capacity and the algorithm's execution time.

ILP-based schemes give the optimal solution. In our OMNeT++ simulation of the FoF-R scheme, we used sequential version of the independent replications method [22] for analysis of the data collected during simulation. In this method, simulations were repeated a number of times by using different independent traffic request sequences, until a relative error less than 1%, at 95% confidence level, was achieved. Then the mean value of the total capacity measure was computed. The final mean values and relative errors obtained for the FoF-R algorithm are shown in Table 1 below.

Links	Mean	Relative Error(%)	No. required of replications
31	8264	0.97	18
34	5630	0.94	25
37	5411	0.84	25
40	5077	0.99	30
43	4845	0.98	32
46	4711	0.93	36
49	4559	0.83	46
52	4276	0.96	50
55	4076	0.98	63
58	4011	0.83	63
61	3826	0.92	72
64	3628	0.90	74

Table 1. FoF-R simulation: statistical error analysis

Links	Average node degree *d*	SBPP	FOF-R	P-Cycle	DP
31	2.214	8264	8264	8372	13056
34	2.429	5584	5630	5987	8340
37	2.643	5362	5411	5717	7728
40	2.857	5029	5077	5316	7146
43	3.071	4787	4845	5063	6808
46	3.286	4641	4711	4814	6482
49	3.500	4482	4559	4676	6292
52	3.714	4184	4276	4349	5934
55	3.929	3980	4076	4076	5715
58	4.143	3886	4011	3885	5468
61	4.357	3696	3826	3655	5178
64	4.571	3481	3628	3448	5066

Table 2. Total capacity of four protection schemes

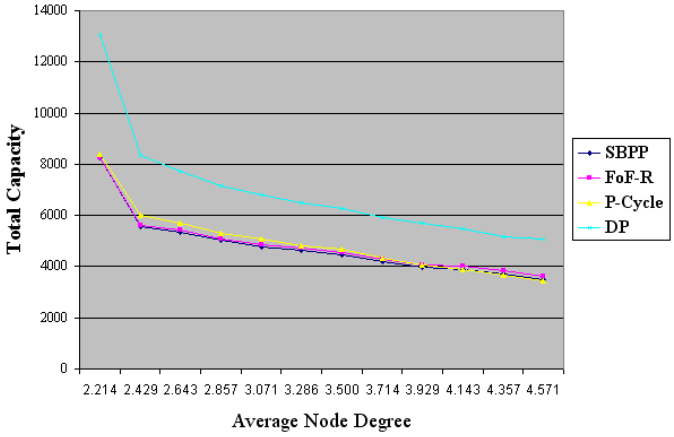


Fig. 7. Total capacity usage

Table 2 and Figure 7 show the total capacity used by the four different protection schemes versus the average nodal degree *d*. As the average nodal degree increases gradually from

2.214 to 4.571, the total capacity required for establishing restorable connections for all the requests decreases for all four protection schemes. The best total capacity allocation comes from the SBPP scheme in sparse networks i.e., when the average node degree is between 2.214 to 3.929, while p-cycle performs best in dense networks in which *d* is greater than 4.143. The FoF-R scheme uses 2% more capacity than SBPP in sparse networks and that of 5% more capacity than the p-cycle scheme in dense networks. DP provides no spare capacity sharing ability. Consequently, it gives the highest total capacity usage, generally exceeding the working capacity. It also suggests that the total capacity usage is dependent on the topological connectivity. All these four schemes tend to require less capacity when the network is become denser. There are two possible reasons for the algorithm's sensitivity to network connectivity. First, as the network connectivity increases, both the predetermined working and protection path-pairs become shorter, this leads to a decrease in both working and protection capacity. Second, the potential for capacity sharing among PPs is likely to increase as the network connectivity increases, which leads to a decrease in protection capacity.

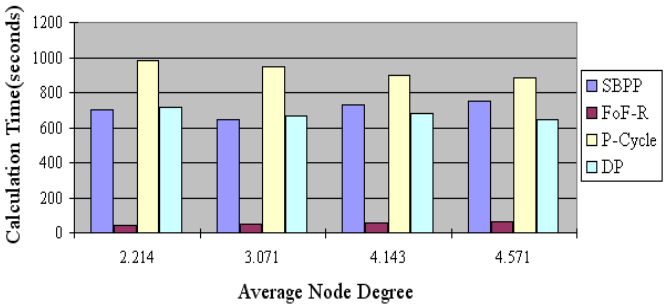


Fig. 8. Algorithm calculation time

Figure 8 shows that the calculation times for different protections schemes. FOF-R is significantly different from the other three schemes. We can see that SBPP, p-cycle and DP find an optimal solution in hundreds of seconds with p-cycle always taking a longer time than SPBB, while FoF-R needs only in tens of seconds to find the near optimal solutions. Thereby, it provides a good tradeoff between algorithm computation speed and capacity efficiency, especially in low-connectivity networks.

V. CONCLUSIONS AND FUTURE WORK

We have addressed the problem of survivable routing with shared path protection. The complete information on bandwidth usage can be fully captured by the distributed RM structure. We also proposed a novel FoF-R algorithm to find the protection cycles and to explore the sharing potential between protection paths by using a Bandwidth Headroom Function (BHF), an attraction/repulsion relationship function. By using the proactive ant mobile agents to continuously investigate the network bandwidth usage, and updating the protection cycle tables and RMs at each node in the distributed control environment, our FoF-R algorithm shows good performance in trading off computational speed against capacity efficiency.

Work on improving the effectiveness of the relationship function is underway. A full comparison of our solution with ILP and other heuristic algorithms is planned, together with exhaustive examination of the FoF-R algorithm in various topologies and traffic scenarios. In addition, extension of the FoF-R routing algorithm to handle differentiated resilience requirements in a multi-service NGN environment is being considered.

REFERENCES

[1] J. Tapolcai, P. Laborczi, P. -H. Ho, T. Cinkler, A. Recski, and H. T. Mouftah, "Algorithms for Asymmetrically Weighted Pair of Disjointed Paths in Survivable Networks", Proc. Third International Workshop on DRCN 2001, Budapest, Hungry, Oct. 2001, pp. 239-249.

[2] Pin-Han Ho, "State-of-the-Art Progresses in Developing Survivable Routing Strategies in the Optical Internet", IEEE Communications Surveys and Tutorials, Vol. 6, No. 4, 2004.

[3] Y. Liu, D. Tipper, and P. Siripongwutikorn, "Approximating Optimal Spare Capacity Allocation by Successive Survivable Routing," Proc .IEEE INFOCOM '01, pp. 699-708, 2001

[4] M. Kodialam and T.V. Lakshman, "Dynamic Routing of Locally Restorable Bandwidth Guaranteed Tunnels Using Aggregated Link Usage Information," Proc. IEEE INFOCOM '01, pp. 376-385, 2001.

[5] C. Qiao and D. Xu, "Distributed Partial Information Management (DPIM) Schemes for Survivable Networks—Part I," Proc. IEEE INFOCOM '02, pp. 302-311, June 2002.

[6] Tapolcai, J.; Pin-Han Ho; Haque, A, "TROP: A Novel Approximate Link-State Dissemination Framework For Dynamic Survivable Routing in MPLS Networks" , IEEE Trans. on Parallel and Distributed Systems, Volume 19, Issue 3, March 2008 Page(s):311 – 322

[7] M. Dorigo, M. Birattari, and T. Stützle, "Ant Colony Optimization". IEEE Computational Intelligence Magazine, 1(4):28-39, 2006.

[8] A. Colorni, M. Dorigo, and V. Maniezzo, "Distributed optimization by ant colonies," Proc. European Conference on Artificial Life(ECAL'91), Elsevier Publishing, Amsterdam, 1991.

[9] G.Di Caro and M. Dorigo, "AntNet: Distributed Stigmergetic Control for Communications Networks," Journal of Artificial Intelligence Research, 9:317–365, Dec 1998.

[10] G. Di Caro and M. Dorigo, "Two ant colony algorithms for best-effort routing in datagram networks, " Proc. the Tenth IASTED International Conference(PDCS'98), IASTED/ACTA Press.1998.

[11] A. Nowe., K. Verbeeck and P.Vrancx, "Multi-Type Ant Colony System: the Edge disjoint path problem," Proc the ANTS 2004 Workshop, LNCS vol 3172, p202-213, 2004.

[12] O. Wittner and B. E. Helvik, " Cross Entropy Guided Ant-like Agents Finding Dependable Primary/Backup Path Patterns in Networks," Proc. CEC2002, Honolulu, Hawaii, May 12-17th 2002.

[13] V. Gazi and M.K. Passino, "Stability analysis of swarms," IEEE Transcations on Automatic Control, 48, 692–697, 2003.

[14] H. J. Reif and H. Wang, "Social potential fields," Robotics and Autonoumous Systems, 27, 171–194, 1999.

[15] W. Liu, H. Sirisena and K. Pawlikowski, "FoF-R Ant: Ant-Based Survivable Routing Scheme for Shared Path Protection," Proc. Australasian Telecommunication Networs and Applications Conf. (ATNAC2008), Adelade, December 2008.

[16] OMNeT++ website: http://www.omnetpp.org/

[17] W. D. Grover and D. Stamatelakis, "Cycle-oriented distributed preconfiguration: Ring like speed and mesh-like capacity for self-planning network Restoration", Proc. IEEE ICC, vol. 1, pp. 737-543, June 1998.

[18] J. Doucette, D. He, W. D. Grover and O. Yang, "Algorithmic approaches for efficient enumeration of candidate p-cycles and capacitated p-cycle network design,", Proc. Fourth International Workshop on Design of Reliable Communication Networks (DRCN 2003), pp. 212-220, Oct. 2003.

[19] R. Fourer, D. M. Gay, and B. W. Kernighan, AMPL: A Modeling Language for Mathematical Programming. San Francisco, CA,1993.

[20] CPLEX User Manual v11.1, ILOG, Inc., 2008.

[21] "LION and COST 266," Reference Networks, part of the European Information Soc. Technologies (IST) Fifth Framework program, 2003

[22] K. Pawlikowski, G. Ewing and D. McNickle. "Performance Evaluation of Industrial Processes in Computer Network Environments". Proc. ECEC'98(1998 European Conf. on Concurrent Eng., Erlangen, Germany, April 1998), Int. Society for Computer Simulation, 1998, 160-4