p^2-Cycles: p-Cycles with Parasitic Protection Links

Long Long Ahmed E. Kamal

Dept. of Electrical and Computer Eng., Iowa State University, Ames, IA 50011, U.S.A.

E-mail: {longlong, kamal}@iastate.edu

***Abstract*—The p-cycle and its Failure Independent Path Protection (FIPP) extension are known to be efficient and agile protection strategies. The p-cycle is preconfigured such that if there is a failure, only the switches at two end nodes need to be reconfigured. In this paper, we extend the p-cycle by allowing cycles to have attached links, called Parasitic Protection Links (PPL), in order to protect paths whose source and destination nodes are not only located on the cycle but also connected by the PPL to the cycle. A p-cycle with PPL is named p^2-cycle.**

We address the unicast service protection problems against single-link failures by using p^2-cycle in mesh networks and the problem is formulated as an Integer Linear Program (ILP). The numerical results show that the p^2-cycle scheme provides better capacity efficiency than the FIPP p-cycle scheme in all the traffic scenarios considered and consumes $2.7\% - 14.8\%$ extra total cost over the optimum, provided by Shared Backup Path Protection (SBPP) approach. Moreover, we study the failure recovery performance by comparing it to FIPP and SBPP in terms of average number of reconfigurations (NOR). The results achieved by the p^2-cycle are much better than that of SBPP and gets close to FIPP as the number of traffic demands increases. In conclusion, the p^2-cycle protection scheme provides greater overall performance over existent protection schemes, especially when the number of traffic demands is large.

I. Introduction

Network survivability, defined as the network's ability to continue functioning correctly in the presence of the failures of network components [1], is an important requirement for WDM optical networks due to their ultra-high capacity. A single failure can disrupt millions of applications and users. Ring-based resilience schemes were prevalent due to the simple manageability and fast recovery mechanism, in which traffic restoration process can be completed within 50-60 ms, but require 100% capacity redundancy [2]. As mesh-based networks emerged, more capacity efficient protection schemes were proposed, and are mainly composed of three categories: link-based, segment-based and path-based [3]. Among them, a path protection scheme, namely, Shared Backup Path Protection (SBPP), has been proven to be the most capacity efficient protection scheme and can achieve optimal solutions [3].

Preconfigured protection cycles, referred to as p-cycles, absorbs the merits of both ring-based and mesh-based protection schemes and achieves the speed of ring and capacity of mesh [5], [6]. Since the concept of the p-cycle was first introduced in [5], a large number of works in literature have studied the p-cycle design problem with unicast traffic against a single-link failure. Authors in [5], [6], [7] solved the problem in two steps by first routing the connections using routing algorithms and then selecting the best candidates from the enumeration of all the cycles to protect the established connections. However, the optimality of the solution was relaxed by dividing the problem into two subproblems. Accordingly, the authors solved the problem optimally by minimizing the total cost of the primary and protection capacities jointly in [8] and [9]. Besides link protection, p-cycles are also extended to protect segments and paths [10], [11], in which [11] proposed a Failure Independent Path-Protecting (FIPP) p-cycle which achieves the best capacity efficiency among all p-cycle schemes.

Regardless of the protection schemes, there is always a trade-off between the capacity efficiency and failure recovery speed. Considering that the p-cycle has a good combination of both, we attempt to extend the FIPP p-cycle scheme to a new paradigm in which a p-cycle can have a number of protection links attached to the cycle, which we call "Parasite Protection Links (PPL)", and such a p-cycle with PPL is called a p^2-cycle. A p^2-cycle can protect not only the connections whose end nodes are located on the cycle, but also those whose end nodes are connected to the cycle through the PPL.

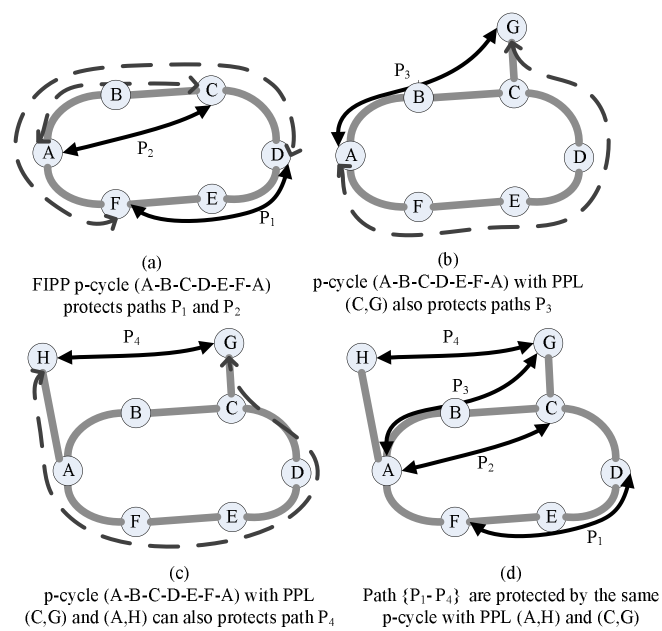


Fig. 1. An example of a p-cycle with PPL

An example shown in Figure 1 illustrates the concept of the p^2-cycle. In Fig. 1(a), a p-cycle $(A-B-C-D-E-F-A)$ is used to protect two bidirectional paths, P_1 and P_2, where

This research was supported in part by grant CNS-0626741 from the National Science Foundation.

path P_1 traverses on-cycle span (D, E) and (E, F) and is protected by on-cycle segment $(F-A-B-C-D)$ and path P_2 is a straddling path and protected by on-cycle segment $(A-B-C)$. Working paths are denoted by solid lines and protection paths are represented by dashed lines. Assuming we have another working path P_3 (shown in Fig.1(b)) traversing on-cycle span (A, B) and non-cycle span (B, G), the original p-cycle cannot protect it, since the end node G is not on the cycle. We then extend the p-cycle to have a PPL (C, G) and hence the p-cycle is able to provide a on-cycle path $(A-F-E-D-C-G)$ to protect P_3. The idea can also be applied to the path whose both end nodes are not on the cycle (as shown in Fig.1(c)), since two PPLs (A, H) and (C, G) can be found and attached to the cycle, and we can find a protection path $(H-A-F-E-D-C-G)$ on the extended p-cycle for P_4. Therefore, instead of two paths, the extended p-cycles can protect four paths (shown in Fig.1(d)). Thus, extending a p-cycle to have PPLs enhances the flexibility of protection and thus may decrease spare capacity redundancy and reduce overall capacity cost.

The rest of the paper is organized as follows: In Section II, we analyze the p^2-cycle protection scheme in depth and formally state the problem addressed in this paper. We then formulate the unicast protection problem with the p^2-cycle by using Integer Linear Programing (ILP) in Section III. Numerical results of multiple criteria will be presented in Section IV. Finally, we conclude the paper in Section V.

II. Preliminaries

A. Protection Mechanism

The protection ability of a p^2-cycle is an enhancement to that of the p-cycle by adding attached spans to the cycle, which enables the cycle to provide protection to the connections whose end nodes are one hop away from the cycle. All the nodes on the cycle still remain preconfigured. Given a unicast session, the primary path and the corresponding protection path, which may consist of an on-cycle segment and PPL(s), will be determined regardless of the location of the failure, and therefore the p^2-cycle protection scheme is also failure independent. Upon a failure on the primary path, the failure is detected by the end nodes of the failed span and the corresponding signals will transmitted to the source and destination nodes of the path. The distinction between a p^2-cycle and a FIPP p-cycle here is that the source or the destination may not be on the cycle. Therefore, in order to reroute the traffic onto the backup path, the source and destination nodes need to reconfigure their switches, as do the end nodes of the protection segment on the cycle.

Let us review Figure.1 again. In Fig.1(b), if a failure happens to span (A, B) or (B, G), then node A, G and C will reconfigure their switches such that the traffic can be routed on the backup path, as denoted by the dashed line. Accordingly, if any span on path (H, G) fails (shown in Fig.1(c)), then both end nodes H and G, along with two end nodes of the on-cycle protection segment, A and C, will be reconfigured to reroute the traffic. Although the whole failure recovery process consists of three phases - failure detection, signal broadcasting and node reconfigurations - node reconfigurations usually take the most time during the process, since each reconfiguration takes 10 - 20s ms [13] depending on the technology used. More node reconfigurations on the protection path will result in longer traffic restoration. It is apparent that each connection protected by an FIPP p-cycle carried out two reconfigurations upon a failure, one at the source and the other at the destination, and rest of the nodes are preconfigured on the cycle. However, for a path protected by a p^2-cycle, the number of configurations can be two, three or four depending on whether the end nodes are on the cycle or not.

B. Problem Statement

In this paper, we address the unicast services protection in WDM networks against single-link failure scenarios. In order to study the performance of the p^2-cycle scheme, we address the joint capacity placement (JCP) problem in which working paths and protection cycles are provisioned jointly in order to achieve the minimum total cost. A number of assumptions are given as follow:

1) Each unicast session is bidirectional with a unitary traffic rate (one wavelength) and the traffic in both directions have to be routed through the same paths and protected by the same p^2-cycle.
2) Each p^2-cycle is also bidirectional and has unitary capacity on both on-cycle spans and PPLs.
3) Each span has enough wavelengths and each network node is equipped with wavelength converter.

Therefore, we now state the JCP problem formally: Given a bidirectional unicast traffic matrix D where $D = d_l(s_l, t_l),\ (0 \le l < M)$ and a weighted undirected graph $G = (V, E)$ in which each span $e \in E$ has a cost c_e, provision and protect all the unicast sessions with minimal total cost.

III. ILP Formulation

We formulate the JCP problem as an Integer Linear Program (ILP). Since the number of cycles increases exponentially with network sizes, we do not enumerate all the cycles in a given network in our ILP formulation. Instead, the flow variables will form the cycles in the solution.

The variables used in the ILP formulation are defined in Table I:

- *Objective:*

$$\text{Minimize:} \quad \sum_{(m,n)\in E} \left(\sum_{l} p^l_{mn} + \sum_{p} (e^p_{mn} + b^p_{mn})\right)$$

- *Subject to:*

TABLE I
LIST OF SYMBOLS

Symbol	Meaning
P :	The maximum number of p^2-cycles in the solution where p denotes the pth p^2-cycle $(0 \le p < P)$
p^l_{mn} :	A binary, indicates whether the primary path of session l traverses span $(m, n) \in E$
q^l_{mn} :	A binary, indicates whether the protection flow of session l traverses span $(m, n) \in E$
Kp^l_n :	A binary, indicates whether primary path of session l passes node n
Kq^l_n :	A binary, indicates whether protection path of session l passes node n
e^p_{mn} :	A binary, equals 1 if p traverses span $(m, n) \in E$
z^p_n :	a binary, equals 1 if p passes node $n \in V$
b^p_{mn} :	A binary, equals 1 if span (m, n) is a parasite protection link of p
$B^{p,l}_{mn}$:	A binary, equals 1 if PPL (m, n) is used by p to protect session l
x^p_l :	A binary, equals 1 if p protects session l
$X^p_{l_1 l_2}$:	A binary, indicates whether session l_1 and l_2 share the protection of p^2-cycle p
$\phi^{l_1 l_2}$:	A binary, indicates whether session l_1 and l_2 share protection of any p^2-cycle
$\gamma^{l_1 l_2}_{mn}$:	A binary, equals 1 if the primary paths of both session l_1 and l_2 use span (m, n)
$\gamma^{l_1 l_2}$:	A binary, equals 1 if the primary paths of session l_1 and l_2 use at least one common span
μ^p_u :	A binary, equals 1 if node v is the master node of p
$\alpha^p_{mn,v}$:	A binary, equals 1 if span (m, n) is used to reach node v from the master node of p
$\beta^p_{n,v}$:	A binary, equals 1 if node n is traversed by the flow from the master node to node v through the cycle of p
$\bigtriangledown$:	A small positive constant (0.0001)

1) Flow Conservation Constraints:

$$\sum_{n:(s_l,n)\in E} p^l_{s_l,n} = \sum_{m:(m,t_l)\in E} p^l_{m,t_l} = 1, \forall l, p; \quad (1)$$

$$\sum_{n:(u,n),(n,u)\in E} p^l_{u,n} = 2Kp^l_n, \quad (2)$$
$$\forall n \in V \backslash \{s_l, t_l\}, \quad \forall l, p;$$

$$\sum_{n:(s_l,n)\in E} q^l_{s_l,n} = \sum_{m:(m,t_l)\in E} q^l_{m,t_l} = 1, \forall l, q; \quad (3)$$

$$\sum_{n:(u,n),(n,u)\in E} q^l_{u,n} = 2Kq^l_n, \quad (4)$$
$$n \in V \backslash \{s_l, t_l\}, \quad \forall l, q;$$

2) Protection Constraints:

$$\sum_p x^p_l = 1, \qquad \forall l; \quad (5)$$

$$e^p_{mn} \ge x^p_l \wedge q^l_{mn}, \quad (6)$$
$$\forall p, l, \forall (m,n) \in E \backslash \{(s_l, v), (v, t_l)\}, \forall v \in V;$$

$$B^{p,l}_{mn} = x^p_l \wedge q^l_{mn} \wedge (1 - e^p_{mn}), \quad (7)$$
$$\forall p, l, \forall (m,n) \in E$$

$$b^p_{mn} \ge \bigtriangledown(\sum_l B^{p,l}_{mn})), \quad (8)$$
$$\forall p, \forall (m,n) \in E;$$

3) Link Disjoint Constraints:

$$p^l_{mn} + q^l_{mn} \le 1, \quad \forall l, \ \forall (m,n) \in E; \quad (9)$$

4) Protection Capacity Sharing:
$\forall p, l_1, l_2, \ \forall (m,n) \in E,$

$$X^p_{l_1 l_2} = x^p_{l_1} \wedge x^p_{l_2}; \quad (10)$$

$$\phi^{l_1,l_2} \ge \bigtriangledown(\sum_p X^p_{l_1 l_2}); \quad (11)$$

$$\gamma^{l_1,l_2}_{mn} = p^{l_1}_{mn} \wedge p^{l_2}_{mn}; \quad (12)$$

$$\gamma^{l_1,l_2} \ge \bigtriangledown(\sum_{(m,n)\in E} \gamma^{l_1,l_2}_{mn}); \quad (13)$$

$$q^{l_1}_{mn} + q^{l_2}_{mn} \le 2 - (\phi^{l_1,l_2} \wedge \gamma^{l_1,l_2}); \quad (14)$$

5) Cycle Constraints:

$$\sum_{n:(m,n)\in E} e^p_{mn} = 2z^p_n, \ \forall n \in V, \quad \forall p; \quad (15)$$

$$|z^p_m - z^p_n| \ge b^p_{mn}, \quad \forall (m,n) \in E, \quad \forall p; \quad (16)$$

6) Cycle Uniqueness:

$$\sum_{u \in V} \mu^p_u = 1, \quad \forall p; \quad (17)$$

$$\sum_{n:(m,n)\in E} \alpha^p_{mn,v} = 2\beta^p_{m,v} - \mu^p_m, \quad (18)$$
$$\forall p, \quad \forall m, n, v \in V, m \ne v;$$

$$\sum_{m:(m,v)\in E} \alpha^p_{mv,v} = 1 - \mu^p_v, \quad (19)$$
$$\forall p, \quad \forall v \in V;$$

$$e^p_{mn} \ge \alpha^p_{mn,v} - \bigtriangledown \beta^p_{m,v}, \quad (20)$$
$$\forall p, \forall (m,n) \in E, \forall v \in V;$$

Since the network $G = (V, E)$ is undirected, if there is a span between m and n, the span is denoted by (m, n) where $m < n$. The objective function includes the total cost used by the primary paths and all the p^2-cycles in which a p^2-cycle is composed of the on-cycle spans e and parasitic protection spans b. In the flow conservation constraints, equations (1)-(4) ensure that each session l has a primary and a protection path in which each path has only one span connected to the source and destination but each intermediate node is traversed twice.

In the protection constraints, equation (5) ensures that each session is protected exactly once by a p^2-cycle. If a span (m, n) is used by a protection flow of session l, which is protected by p, then (m, n) should be an on-cycle span of p except that m or n is the source or the destination. The conjunction in equation (6) expresses the constraint. A conjunction expression can be simply represented by two linear equations. If span (m, n) is used by a protection flow, it should be a part of p^2-cycle p. But if it is not on the cycle, it must a PPL of p, which is described by equations (7). A PPL of p can be used to protect multiple connections. As long as there exists at least one connection using span (m, n) as a PPL of p, then (m, n) should be counted as a PPL of p. The equation (8) ensures this constraint. Moreover, the working and backup paths of any session should be link-disjoint to survive any single-link failure. This is ensured by equation (9).

In the protection capacity sharing constraints, equations (10) and (11) ensure $\phi^{l_1 l_2} = 1$ if session l_1 and l_2 share the protection of any p and equation (12) and (13) make sure that $\gamma^{l_1 l_2} = 1$ if the primary paths of session l_1 and l_2 are not link disjoint. In this case, the protection flow of l_1 and l_2 cannot traverse the same span, which is described by equation (14). The cycle constraints make sure that each node on the cycle is passed twice, as described in Eq. (15). And if span (m, n) is a PPL of p, then one of the two nodes must be on the cycle while the other should not, which is depicted by Eq. (16) and it can also be easily replaced by two equations.

However, equation (15) is not enough to guarantee that there is only one cycle with index p, since multiple cycles can be formed with the same index p while still complying with constraint (15). Some work has been done to address this issue. The method proposed in [8] is simple and the number of introduced variables is linear to the size of the network. However, it can only apply to unidirectional cycles. Hence, we use the approach proposed in [12]. This approach picks a node on each cycle randomly and defines it as the master node such that there must exist a flow from the master node to every other on-cycle node through the cycle. Equation (17) ensures that there is only one unique master node for each p^2-cycle. Equations (18) and (19) ensures the flow conservation between the master node and each on-cycle nodes, in which Eq. (18) guarantees that the flow uses one span connected the master node but two connected to any intermediate node passed by the flow and Eq. (19) ensures that only one span connecting to the destination node is used by the flow on the cycle. Finally, equation (20) guarantees that if a node v is on the cycle p, then all the spans traversed by the flow from the master nodes to node v should be on-cycle spans of p. Therefore, the uniqueness of one cycle for each index p is ensured. The number of variables introduced to guarantee a single cycle for each p is $P|E||V| + P(|V| + 1)|V|$.

In summary, the total number of variables used in the ILP formulation is dominated by $M^2(|E| + P) + P|V|(|E| + |V|)$ and the total number of constraints can be denoted by $O(M^2(|E| + P) + P|V|(|E| + |V|))$.

IV. Numerical Results

In this section, we investigate the performances of the proposed p^2-cycle protection scheme and compare it with two other path protection schemes, the SBPP and FIPP p-cycle, in terms of two criteria: total capacity cost and average number of reconfigurations (NOR). In order to compare the performances of these protection schemes, we obtain the optimal solutions of the JCP problem for each problem by formulating it as an Integer Linear Program (ILP) and solved by a commercial software - ILOG CPLEX 10.1.0 on a Linux server with four Xeon 2.4GHz CPU and 4 GB of RAM. The ILP for SBPP is obtained from [4]. Since the ILP for FIPP p-cycle proposed in [11] does not address JCP problem but only spare capacity assignment, we use the ILP proposed in this paper without PPLs.

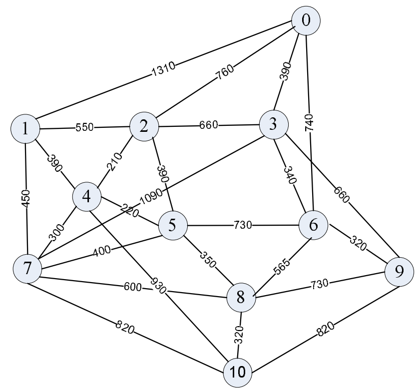


Fig. 2. COST239 network (11 nodes, 26 spans)

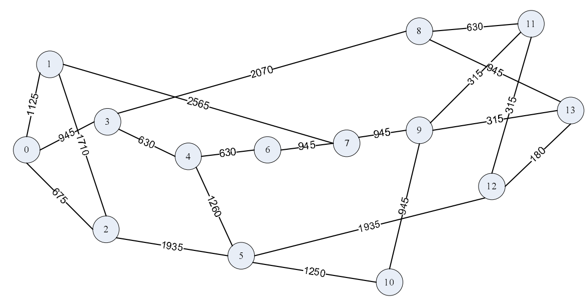


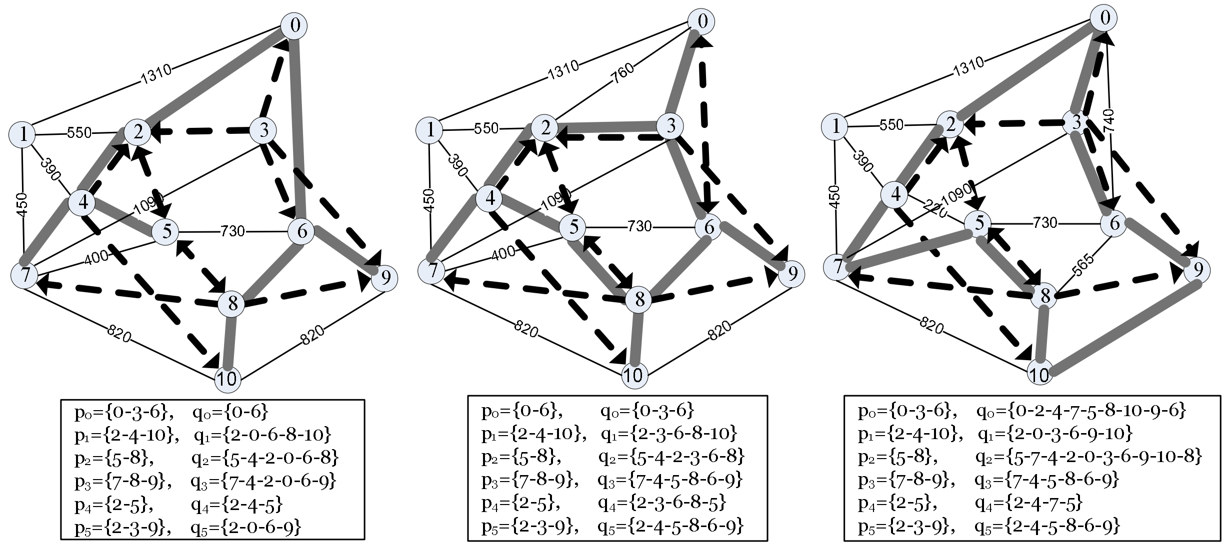
Fig. 3. NSFNET network (14 nodes, 21 spans)

The experiments are conducted on two practical networks, the pan-European COST 239 and NSFNET, shown in Fig.2 and 3. Both networks have similar numbers of nodes, but COST239 has a larger average node degree (4.72) than NSFNET (2.7). Each span in the two networks has a cost, which is the actual distance between the two end nodes in kilometers. We assume that the networks have wavelength conversion capability and unlimited wavelengths on each span.

A. Total Capacity Cost

We first study the capacity performance of the three schemes in COST239 network. Given six unicast traffic requests, such that each session has bidirectional demand with unitary traffic rate (one wavelength), we obtain the solutions of employing each protection scheme, respectively, as shown in Fig.4. The source and destination of each session is depicted in the pair of braces and each session is indexed from 0 to 5, counted from the left to the right. Dashed lines with arrows represent the primary paths and thick gray solid lines without arrows represent the assigned protection capacity. The description in the box lists the routes of each primary path, denoted by p, and the corresponding protection path, denoted by q for each session.

The optimal solution obtained by employing the SBPP scheme is presented in Fig.4(a). We need to note that one wavelength assigned on a gray span is shared by multiple



(a) solution of SBPP protection scheme with total cost 8695.0 and running time 3336.34 sec

(b) solution of *pl*-cycle protection scheme with total cost 8945.0 and running time 814.38 sec

(c) solution of FIPP *p*-cycle protection scheme with total cost 9470.0 and running time 111.11 sec

Fig. 4. Comparison the total cost of SBPP, p^2-cycle and FIPP p-cycle in COST239 network given six bidirectional unicast requests with unitary traffic rate $\{(0,6),\ (2,10),\ (5,8),\ (7,9),\ (2,5),\ (2,9)\}$

sessions. For instance, the spare capacity on span $(6,8)$ is shared by sessions 1 and 2 and $(0,6)$ is shared by sessions 0,1,2,3 and 5. This feature makes SBPP the most capacity efficient scheme. However, it is the most complex design problems. The same traffic demands are provisioned by using the p^2-cycle and the FIPP p-cycle and the solutions are shown in Fig.4(b) and (c). Each scheme provisioned one cycle, but the cycle established by the p^2-cycle scheme, denoted by $(2-3-6-8-5-4-2)$, is smaller and has four attached PPL, which enables p^2-cycle to save some spare capacity, compared to the solution achieved by FIPP. In the given example, the FIPP scheme uses 8.9% more capacity over the optimal solution whereas the p^2-cycle reduce this number to 2.9%, which is very close to the optimal solution.

We also studied the average performance of each scheme in NSFNET network. We provisioned from 2 to 7 sessions, and for each traffic scenario, we ran 50 independent cases and then took the average value of the total cost. The end nodes of each session were randomly chosen for each case, but the three schemes use exactly the same traffic demands in each case in order to make a fair comparison.

The results of the comparison are presented in Table II. The first column denotes the number of traffic requests. In the third and fourth column, the extra cost over the optimum is calculated as $(cost - optimum)/optimum$, where optimum is achieved by SBPP in the same row. We can observe that the p^2-cycle always achieves better results than FIPP in each scenario. As the number of sessions increase, both schemes achieve better results compared to the optimal solution. Cycle-based protection is inefficient when the traffic demands are low since there are not enough connections to share the protection. As the session size increases, a cycle likely protects multiple connections and becomes more capacity efficient.

TABLE II
COMPARISON OF AVERAGE TOTAL CAPACITY COST IN NSFNET

Sessions	SBPP	p^2-cycle (extra cost(%))	FIPP (extra cost(%))
2	10734.8	12335.4 (14.9)	13310.1 (24)
3	14774.7	16060.5 (8.7)	17515.8 (18.6)
4	19146.1	19759.5 (3.2)	21185.1 (10.6)
5	21818.3	22624.2 (3.7)	24122.7 (10.6)
6	25539.2	26395.2 (3.4)	27514.8 (7.7)
7	29525.2	30327.5 (2.7)	31518.5 (6.8)

When the number of sessions reaches 7, the p^2-cycle becomes extremely efficient and only uses 2.7% extra cost over optimal solution. Hence, we can predict that as the number of sessions further increases, the results achieved by the p^2-cycle will be extremely close to the optimal solutions.

B. Average Number of Reconfigurations

We also compare the traffic recovery performance of p^2-cycles to the other two protection schemes in terms of the average number of reconfigurations (NOR) per connection. Based on the analysis in Section II.A, it is pretty easy to obtain the NOR for each connection protected by FIPP p-cycle and p^2-cycle schemes, respectively, given the primary and protection paths for each connection. However, this is not as straightforward for SBPP scheme. One wavelength reserved for protection on a span can be used for multiple sessions. If we divide the final survivable topology into working and protection structures, only the nodes with more than 2 nodal degrees in the protection structure are potential reconfiguration nodes, since when the node degree is equal to or less than 2, the route of backup traffic passing through this node is always fixed. A potential node requires reconfiguration upon a network failure if this node is connected to a span on which a

TABLE III
COMPARISON OF AVERAGE NUMBER OF RECONFIGURATIONS PER CONNECTION IN NSFNET

No. of Sessions	SBPP	p^2-cycle	FIPP
2	2.48	2.47	2
3	2.77	2.55	2
4	2.84	2.42	2
5	3.01	2.43	2
6	3.09	2.34	2
7	3.19	2.31	2

spare wavelength will be used by one of the failed connections and this wavelength is not a dedicated protection resource, which means that this node cannot be preconfigured and has to reconfigure its switch to enable the rerouted traffic to use that spare wavelength.

The results of the average NOR for each connection for each scheme are presented in Table III. The results are obtained by taking the average value over 50 independent cases in each traffic scenario based on the solutions obtained in NSFNET network in Section IV.A.

Clearly, FIPP achieves the best solution since it always takes two nodes to reconfigure. On the other hand, the average NOR of SBPP increases as the number of connections increases, since the primary and protection structures get more complex and more connections may share spare capacity such that more nodes become potential reconfiguration nodes. However, it is the contrary for the p^2-cycle scheme, in which the average NOR actually decreases. One of the reasons is that a larger number of connections usually results in a larger size of cycle in order to utilize the spare capacity more efficiently. Therefore, more nodes will be covered by the cycle such that less number of connections will use PPL as a part of protection path. Hence, more sessions actually carry out two reconfigurations if the primary path fails.

As we can see, the p^2-cycle definitely outperforms SBPP in every scenario. The advantage becomes more significant as the number of sessions increase, especially when the session number is equal to 7, where the average number of reconfigurations is equal to 2.31, which is only 15% more than the optimal number of 2 that is achieved by the FIPP p-cycle, compared to 3.19, obtained by employing SBPP. Therefore, the p^2-cycle is a much faster protection scheme than SBPP and provides an enhancement of capacity efficiency over the FIPP p-cycle with a small increase in the recovery time.

V. CONCLUSIONS

In this paper, we proposed a new p-cycle based protection scheme in mesh network, named p^2-cycle, by augmenting the FIPP p-cycle with attached parasitic protection links (PPL) in order to enhance the protection ability by protecting the paths whose end nodes are not located on the cycle but only one hope away from the cycle. The reasons of considering only one hop are to limit the length of backup path and control the largest NOR for each connection to be no greater than four. Note that this hop constraint is not necessary if we can guarantee that the lengths of backup paths are limited and the NOR for each connection is within a acceptable range. However, adding these constraints to the ILP will make the problem too complex to solve.

In addition, we assume both on-cycle spans and PPLs have unitary capacity for each p^2-cycle. However, we can also loose the constraint for PPLs to further enhance the protection ability. Considering a session with unitary traffic rate protected by a p^2-cycle, if the primary path is a straddling path and the end nodes do not lie on the cycle, doubling the capacity reserved on the PPLs allows the p^2-cycle to provide two protection paths for the session, each of which uses one unit capacity on the same PPLs but distinct link-disjoint on-cycle segment.

Based on the numerical results, the p^2-cycle protection scheme outperforms the FIPP p-cycle scheme in terms of the total capacity cost and achieves a cost close to the optimal solution, provided by SBPP, when the number of sessions is large. Meanwhile, the p^2-cycle has much better recovery performance than SBPP and gets closer to an FIPP p-cycle as the number of sessions increases. Considering the trade-off between capacity efficiency and restoration speed, the p^2-cycle protection scheme provides greater overall performance over existent protection schemes, especially when the number of traffic demands is large.

REFERENCES

[1] D. Zhou and S. Subramanian, "Survivability in optical networks," *IEEE Networks,* 2000.

[2] P. Arijs, B. V. Caenegem, P. Demeester, and P. Lagasse, "Design of ring and mesh based wdm transport networks," *Optical Networks Magazine,* vol. 1, no. 3, pp. 27-41, 2000.

[3] S. Ramamurthy and B. Mukherjee, "Survivable WDM mesh networks. part I-protection," in *Proceedings of IEEE INFOCOM,* vol. 2, pp. 744-751, 1999.

[4] S. Ramamurthy, L. Sahasrabuddhe, and B. Mukherjee, "Survivable WDM mesh network," *Journal of Lightwave Technology,* vol. 21, no. 4, pp. 870-883, 2003.

[5] W.D Grover and D. Stamatelakis, "Cycle-oriented distributed preconfiguration: Ring-like speed with mesh-like capacity for self-planning network restoration," in *Proc. IEEE ICC' 98,* 1998, pp. 537-543.

[6] W.D. Grover and D. Stamatelakis, "Bridging the ring-mesh dichotomy with p-cycles," in *Proc. of DRCN Workshop,* 2000.

[7] D.A Schupke, C.G. Gruber, and A. Autenrieth, "Optimal configuration of p-cycles in WDM networks," in *Proc. of IEEE ICC,* 2002.

[8] Bin Wu, Kwan L. yeung, and Shizhong Xu, "ILP Formulations for p-Cycle Design without Candidate Cycle Enumeration," accepted in *IEEE/ACM Transactions on Networking,* 2008.

[9] W.D. Grover and J.E. Doucette, "Advances in optical network design with p-cycles: Joint optimization and preselection of candidate p-cycles," in *Proc. of IEE LEOS Summer Topical Meeting,* 2002.

[10] G. X. Shen and W. D. Grover, "Extending the p-cycle concept to path segment protection for span and node failure recovery," *IEEE Journal on Selected Areas in Communications,* vol. 21, no. 8, pp. 1306-1319, Oct. 2003.

[11] Adil Kodian and W. D. Grover, "Failure-Independent Path-Protecting p-Cycles: Efficient and Simple Fully Preconnected Optical-Path Protection," in *IEEE Journal of Lightwave Technology,* Vol. 23, 2005, pp. 3241-3259.

[12] D. A. Schupke, "An ILP for optimal p-cycle selection without cycle enumeration," *8th Conference on Optical Network Design and Modelling, ONDM,* 2004.

[13] S. Thorpe, G. Edwards, and D. Stevenson, "Using Just-in-Time to Enable Optical Networking for Grids", *Proc. IEEE Broadnets Workshop*, Oct. 2004.