GMPLS-Based Hybrid 1+N Link Protection over p-Cycles: Design and Performance

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

***Abstract*—In [1], the author introduced a strategy to use network coding on p-Cycles in order to provide 1+N protection for straddling connections and links against single link failures in mesh optical networks. In this paper we extend the work in [1] and introduce a GMPLS-based implementation of this strategy for link protection, that is also used to protect on-cycle links. We refer to this scheme as hybrid 1+N protection, since some data units are transmitted without being combined with other data units. The strategy uses a combination of GMPLS standard Label Switched Paths (LSP) for protecting on-cycle links, and modified LSPs, which we refer to as Label Switched Cycles (LSC), for protecting straddling links. The strategy does not have to explicitly detect failures, but rather detects the absence of data units. Destinations receive backup copies of packets within no more than the p-Cycle propagation delay for on-cycle link failures, and no more than the sum of the propagation delays of the p-Cycle and the longest straddling link for straddling link failures. The implementation details of this strategy are presented, and the cost of implementation, in terms of link usage is evaluated and is shown to be modest. This scheme can therefore provide protection at higher layers, at a speed that is comparable to that achieved at the physical layer, but at a much reduced cost, while being flexible, and requiring less involvement from the management and control planes.**

I. Introduction

A large number of techniques for network survivability have been introduced in the literature, and these can be classified as *Predesigned Protection* and *Dynamic Restoration* techniques [2]. In predesigned protection, bandwidth is reserved in advance so that when a failure takes place, backup paths which are pre-provisioned, are used to reroute the traffic affected by the failure. These techniques include the proactive 1+1 protection, the reactive 1:1 protection, 1:N protection, and its M:N generalization. Under dynamic restoration, capacity is not reserved in advance, but when a link fails, spare capacity is discovered, and is used to reroute the traffic affected by the failure. Protection techniques provide fast recovery from failures, but require significant amounts of resources. On the other hand, restoration techniques are more optimal in terms of resource usage for survivability, but are much slower than protection techniques. A new protection approach, called the p-Cycles has been introduced in [3], to mimic protection techniques of BLSR SONET ring networks, and they provide 1:N protection to links with the same transport capacity, e.g., DS-3. p-Cycles provide protection to on-cycle links, as well as to straddling links, i.e., links not on the cycle, but with their two end nodes on the cycle itself. Therefore, p-Cycles provide a higher degree of protection than the BLSR. Since the protection capacity can be used to protect multiple links, the p-Cycle belongs to the 1:N protection class. The endpoints of the failure are responsible for detecting the failure, which is done by the management plane, and for rerouting the traffic on the p-Cycle, which is a function of the control plane.

Recently, the author introduced another new concept for protection, namely, 1+N protection in [1]. The technique is based on using a bidirectional p-Cycle to protect a number of link disjoint connections which are straddling from the cycle, and using network coding [4] to transmit modulo 2 sums of the connections signals on the cycle. A failure of any link on a working path can be recovered from by using a decoding operation of the signals transmitted on the p-Cycle. This strategy was introduced to provide 100% protection against single link failures. The 1+N protection can be implemented at a number of layers, and using a number of protocols.

In this paper we extend the 1+N protection scheme to allow p-Cycles to protect on-cycle links, and hence provide 100% protection against single link failures to both on-cycle, and straddling links. We call this extension *Hybrid 1+N* protection since copies of some data units, which are used to provide backup copies, are transmitted without being linearly combined with other data units. Moreover, the same bandwidth resources reserved for protection using network coding against the failure of straddling links are also used to provide protection against the failure of on-cycle links, but without using network coding. This technique is best implemented using the Generalized Multiprotocol Label Switching (GMPLS) protocol [5]. Nodes do not have to detect failures, and do not have to reroute their data units once a failure takes place. Instead, the 1+N technique is used for protecting straddling links, and for protecting on-cycle links, opportunistic transmission of backup copies takes place, so that if there is a failure, the backup copy will reach the destination.

The paper is organized as follows. Section II provides a brief background on the 1+N protection introduced in [1], while Section III introduces the Hybrid 1+N protection and its implementation using GMPLS. In Section IV we study the cost of this strategy, and in Section V we conclude the paper.

II. Background

In this section we provide a brief description of the 1+N protection scheme developed in [1]. This technique is based

This research was supported in part by grants CNS-0626741 and CNS-0721453 from the National Science Foundation, and a gift from Cisco Systems.

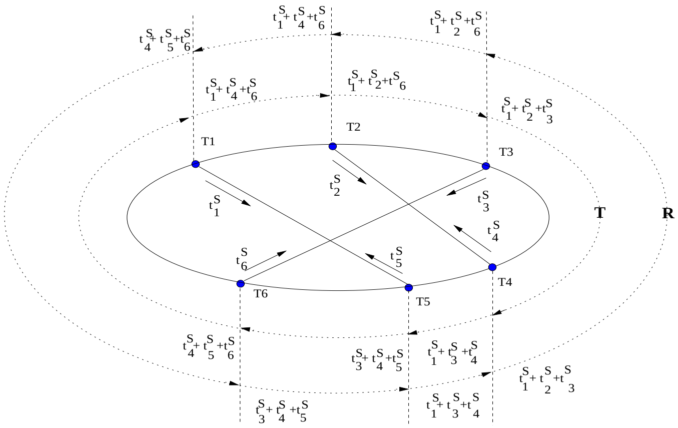


Fig. 1. An example to illustrate 1+N protection.

on the use of the new technique of network coding [4] over p-Cycles [3]. Network coding refers to performing linear coding operations on traffic carried by the network at intermediate network nodes. A node that receives information from all, or some of its input links, encodes this information, and sends the information to all, or some of its output links. This approach can result in enhancing the network capacity, hence facilitating the service of sessions which cannot be otherwise accommodated, especially under multicasting.

The author in [1] used addition operations over $\mathbb{GF}(2)$ field, i.e., **modulo two** or XOR operations, to introduce the 1+N protection scheme. In this scheme, a p-Cycle is provisioned to protect a number of connections, or paths. Paths protected by the same p-Cycle must be link disjoint with each other, and with the p-Cycle. It is assumed that the capacities of all connections are the same, which is also equal to the capacity of the p-Cycle. End nodes of the connections are denoted by T_i, and they are in the set $\mathcal{T}$. Transmissions are in terms of fixed size data units, and all transmissions occur in rounds. If the two end nodes of a connection are T_i and T_j, then T_i sends data units $t_i^{\mathbf{S}}(n)$ to node T_j in round n. Similarly, node T_j sends data units $t_j^{\mathbf{S}}(n)$ to node T_i in round n. An example is shown in Figure 1, where the connection pairs are (T_1, T_5), (T_2, T_4) and (T_3, T_6).

The p-Cycle, which is used to provide backup copies of signals, carries data units in two direction, the clockwise direction, $\mathbf{T}$, and the counter-clockwise direction, $\mathbf{R}$. In each direction, data is transmitted in rounds, such that there are a rounds on the cycle simultaneously, where

$$a = \lceil \tau / (data\ unit\ size\ in\ bits)/B \rceil \tag{1}$$

and τ is the round trip propagation delay around the p-Cycle. Each round is identified by two fields:

1) The round number field, n, which is sequentially updated by a special node called the monitor node.
2) A bit map field, with one bit for each node using the cycle, which is used to indicate if the data unit transmitted on the p-cycle belongs to this round, n, or to round $n - a$. If node T_i which has a connection to node T_j receives a combined data unit with a round number, n, it complements its map bit. If the map bit of node T_j matches that of node T_i, then data units put on the cycle by T_j belong to round n. Otherwise, they belong to round $n - a$.

Each node, T_i, which communicates with node T_j, will execute two steps:

Step I: It will add the following data units to round n on $\mathbf{T}$:

1) A new $t_i^{\mathbf{S}}(n)$, which will add this data unit to $\mathbf{T}$, and
2) Either $t_j^{\mathbf{S}}(n)$ or $t_j^{\mathbf{S}}(n - a)$, depending on the bit map of node T_j. Such data units are received by T_i on the working path, and their addition to $\mathbf{T}$ will remove the data units added by T_j to $\mathbf{T}$ in step I.1.

Step II: It will add the following data units to $\mathbf{R}$:

1) A new $t_j^{\mathbf{S}}(n)$ which is received on the working path, and
2) Either $t_i^{\mathbf{S}}(n)$ or $t_i^{\mathbf{S}}(n-a)$, also depending on the bit map of node T_j. This will also remove the data unit added by T_j in step II.1.

Node T_i, in addition to receiving $t_j^{\mathbf{S}}(n)$ on the working path, can receive another copy by adding:

- The signal received on $\mathbf{T}$,
- The signal received on $\mathbf{R}$, and
- The $t_i^{\mathbf{S}}$ data unit, generated by T_i, and received by T_j which it added to $\mathbf{R}$.

For example, in Figure 1, node T_5 adds the signals received on $\mathbf{T}$ and $\mathbf{R}$ in addition to t_5^S to obtain t_1^S.

III. Hybrid 1+N Protection

This section introduces the Hybrid 1+N protection. The operational assumptions are introduced first, and the basics of this technique are described. The required GMPLS support, and the scheme implementation in GMPLS are then explained.

A. Operational Assumptions

We assume the following:

- The network is represented by a graph, $G(V, E)$, where V is the set of nodes, and E is a set of undirected edges. A node is a GMPLS Label Switched Router (LSR). An edge consists of a number of channels. GMPLS may be used to establish LSPs at one or more interfaces of the LSR. The interfaces that this technique apply to include, but are not limited to, Packet Switched capable, Ethernet capable (including Fast, GigE and 10GigE, and ATM. Following the terminology in [3], an edge will be referred to as a span, and each of the channels on a span will be referred to as a link. The failure of a span will result in the failure of all links on the span.
- A bidirectional p-Cycle embedded in G with a certain bandwidth, B, is used to protect all bidirectional on-cycle and straddling links. The protected links must have the same transport capacity B[1].

[1] This is different from the p-Cycle approach, where the straddling link may have twice the capacity of the protection cycle.

- The p-Cycle is terminated, processed, and retransmitted at each node (LSR) on the cycle.
- It is assumed that data units are fixed in size, and are equal to the Maximum Transmission Unit (MTU). Smaller data units can therefore fit within this MTU[2].
- The scheme presented in this paper is designed to protect against a single link failure. That is, when a span fails, then a p-Cycle protecting a link on this span will be able to provide protection against this failure.
- When a span carrying active links fails, the tail node of each active link will not receive any data units.

B. Basics of the Hybrid 1+N Protection Scheme

In this section, we describe the basics of Hybrid 1+N protection scheme. All addition operations (+) in this paper are in the $\mathbb{GF}(2)$ field, i.e., addition is **modulo two**, or XOR.

A p-Cycle will be provisioned to protect on-cycle and straddling links. Nodes are in the set $\mathcal{T}$. A node $T_i \in \mathcal{T}$ which is at the end of a straddling link is connected to node $S(T_i)$ which is at the other end of the straddling link. In this case, in round n, node T_i sends data units $t_i^{\mathbf{S}}(n)$ to node $S(T_i)$.

We also define $\mathbf{C}(T_i)$ and $\overline{\mathbf{C}}(T_i)$ as the next node in the clockwise and counterclockwise directions on the p-Cycle from node T_i, respectively. We denote the data units sent in round n on the on-cycle working links by node T_i to nodes $\mathbf{C}(T_i)$ and $\overline{\mathbf{C}}(T_i)$ by $t_i^{\mathbf{C}}(n)$ and $t_i^{\overline{\mathbf{C}}}(n)$, respectively. If nodes T_i and T_j are connected by a straddling link of the p-Cycle, then T_i sends data units $t_i^{\mathbf{S}}$ to T_j, and T_j sends data units $t_j^{\mathbf{S}}$ to T_i.

A node on the p-Cycle can have one of two roles:

Type 1: An end node of both an on-cycle link, and a straddling link, or

Type 2: An end node of an on-cycle link only.

One of the Type 1 nodes will act as a *Monitor* node, and it will be the node to start rounds on the p-Cycle in both directions, $\mathbf{T}$ and $\mathbf{R}$, which will be used in exactly the same way described in Section II. Let T_x be the node to start the rounds: It will start round n on $\mathbf{T}$ by transmitting $t_x^{\mathbf{S}}(n)$, and will start round n on $\mathbf{R}$ without transmissions. Node $\overline{\mathbf{C}}(T_x)$ will be the first node to transmit on $\mathbf{R}$ in round n.

A Type 1 node will do two things:

1) It will behave similar to an on-cycle node in the 1+N protection scheme described in Section II. The data units combined by the Type 1 nodes and transmitted on the p-Cycle are used to protect *straddling links*, are called *Straddling Links Protection* (SLP) data units.
2) If a Type 1 T_i node does not receive a data unit on the $\mathbf{T}$ cycle, it assumes that the link on the $\mathbf{T}$ cycle between $\overline{\mathbf{C}}(T_i)$ and T_i has failed, and sends the $t_i^{\overline{\mathbf{C}}}$ downstream on the $\mathbf{T}$ cycle (i.e., in a direction opposite to that of the working link) so that it can be received by node $\overline{\mathbf{C}}(T_i)$. Also, if the node does not receive a data unit on the $\mathbf{R}$ cycle, it assumes that the link on $\mathbf{R}$ cycle between $\mathbf{C}(T_i)$ and T_i has failed and sends $t_i^{\mathbf{C}}$ downstream on the $\mathbf{R}$ cycle, so that it can be received by $\mathbf{C}(T_i)$. In the above two cases, node T_i also receives the data units from $\overline{\mathbf{C}}(T_i)$ and $\mathbf{C}(T_i)$ on $\mathbf{R}$ and $\mathbf{T}$, respectively.
The data units which are used to protect *on-cycle links* are called *On-Cycle Links Protection* (OLP) data units.

A Type 2 node will only perform Step 2 performed by Type 1 nodes only, and will transmit OLP data units only.

Two more mechanisms are needed to guarantee that the above will work:

1) At any of the nodes on the cycle, SLP data units have priority in transmission on the cycle over OLP data units.
2) At the monitor node, SLP data units for round n are not generated unless SLP data units for round $n - a$ are received, where a is the propagation delay of the p-Cycle in terms of SLP data units given in equation (1) above.

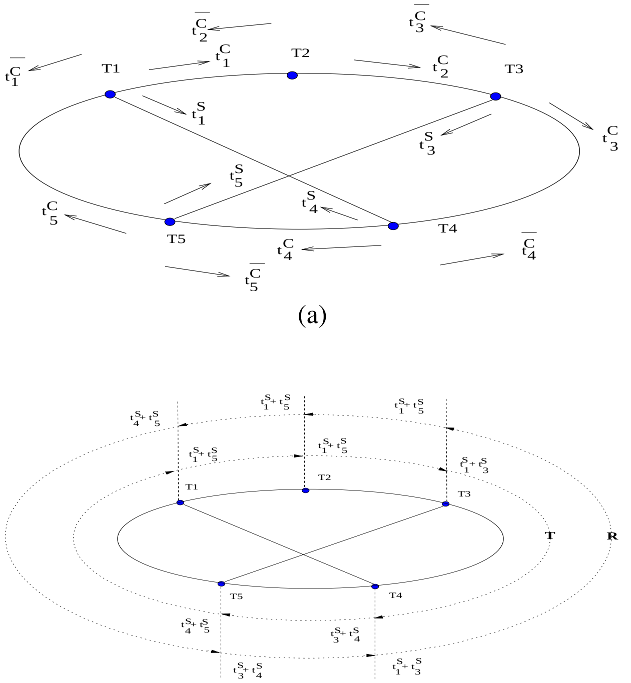
We show an example in Figure 2 of a p-Cycle protecting five nodes, T_1 through T_5, where node T_2 is a Type 2 node, while all other nodes are of Type 1. In the absence of failures, the data units transmitted on the working links are shown in Figure 2.(a), while the linear combinations carried on the $\mathbf{T}$ and $\mathbf{R}$ cycles are shown in Figure 2.(b). When a straddling link fails, e.g., between T_1 and T_4 shown in Figure 2.(b), the combinations received at T_1 and T_4 can be used to recover t_4^S and t_1^S, respectively. However, when an on-cycle link fails, e.g., between nodes T_2 and T_3, the $\mathbf{T}$ cycle is used to carry $t_3^{\overline{\mathbf{C}}}$ to T_2 in the clockwise direction. Similarly, the $\mathbf{R}$ cycle is used to carry $t_2^{\mathbf{C}}$ data units to T_3, and in the counterclockwise direction.

C. GMPLS Support

The following support is needed from GMPLS to implement the Hybrid 1+N scheme:

1) When an LSP is established under GMPLS, it can request protection, and it may identify the protection as either *primary* or *secondary* protection. Primary protection resources are reserved for the protection LSP. However, secondary protection resources may be used by other LSPs until they are needed, and then the secondary protection LSP preempts such LSPs. A p-Cycle used for transmitting SLP data units needs to be implemented as a primary protection LSP. The transmission of OLP data units can be on secondary protection LSPs, which share the same resources with the primary protection LSPs.
2) We also need to define a new type of label, which is used by the primary protection LSP, and is identified by the following Type-Length-Value (TLV) fields:
   - **Type:** 1+N SLP label.
   - **Length:** N + R + M bits.
   - **Value:** the first N bits are used for the generalized label, the next R bits are used for the *round number*, and the final M bits are used for the *bit map* which indicates whether the data units belong to the same round, or a previous round (see Section II).

[2] A shorter size for data units may be used, and longer data units will then have to fragmented.



(a)

(b)

Fig. 2. An example of a p-Cycle used to protect 2 straddling and 5 on-cycle links: (a) the working links; (b) the protection circuits used to protect straddling links

D. GMPLS Implementation

We now present the implementation of the above strategy using the GMPLS protocol. Under GMPLS, an LSP is a unidirectional path that is established between an ingress and an egress LSR on one of a number interfaces The path is fixed and traverses a number of intermediate LSRs. GMPLS allows the establishment of bidirectional paths, which is used to establish p-Cycles, as will be explained below. An LSP which is explicitly routed is known as an LSP tunnel. LSP tunnels with or without resource reservation can be established using the Resource Reservation Protocol with Traffic Engineering extension (RSVP-TE) [7]. All LSPs which are created under this implementation will have to be routed explicitly through the Explicit Route Object [6].

We define a new type of LSP, a Label Switched Cycle (LSC), which is very similar to an LSP, except that it starts and ends at the same LSR. LSCs are used to implement p-Cycles, and they are established in both directions.

There are three phases in this technique, *initialization*, *failure free operation*, and *operation in the case of failure*. We now describe the three phases and the steps involved in implementing the Hybrid 1+N Protection scheme in GMPLS:

Initialization:

1) A node on the p-Cycle, e.g., T_x, is chosen as the *Monitor* node, whose function is to establish the $\mathbf{T}$ and $\mathbf{R}$ cycles, and to start the different rounds of SLP data units.
2) The *Monitor* node establishes a bidirectional LSC. The two directions correspond to the $\mathbf{T}$ cycle, and the $\mathbf{R}$ cycle, and they both have the same reserved bandwidth of B, which is equal to the protected capacity. This bandwidth, however, is also used by the LSPs which are used to protect on-cycle links, as will be explained below. The LSC is established as a 1+N SLP *primary* protection LSP, as defined above. It will start and end at the same LSR, the monitor node, but at two different interfaces, which must be of the same type. In GMPLS terminology, the *Monitor* node is both the *LSP Initiator* and *Terminator*. The generalized label request, sent in the Generalized Label Request object [6], includes encoding and switching types which correspond to the interface, e.g., Packet Encoding type, and Packet Switching. It also includes the client layer of the LSP, e.g., the IP Ethertype is used if the client layer is the IP layer. The bandwidth B is usually reserved in a per protocol specific manner (see [5] for details).
3) Each node, T_i, on the p-Cycle, including the *Monitor* node, establishes two unidirectional *secondary* protection LSPs, which are explicitly routed using the same route of the LSC:
   a) One LSP is in the direction of the $\mathbf{R}$ cycle, and it is initiated by node T_i, and terminates at node $\mathbf{C}(T_i)$. This LSP is used to protect against the failure of the link between nodes T_i and $\mathbf{C}(T_i)$ by delivering data units $t_i^{\mathbf{C}}$ from T_i to $\mathbf{C}(T_i)$. This LSP is identified as a *secondary* protection LSP.
   b) The other LSP is in the direction of the $\mathbf{T}$ cycle, and it is initiated by node T_i and terminates at node $\overline{\mathbf{C}}(T_i)$. This LSP is used to protect against the failure of the link between nodes T_i and $\overline{\mathbf{C}}(T_i)$ by delivering data units $t_i^{\overline{\mathbf{C}}}$ from T_i to $\overline{\mathbf{C}}(T_i)$. This LSP is also identified as a *secondary* protection LSP.

   If there are P nodes on the p-Cycle, then there are $2P$ such LSPs, with P LSPs in each of the two directions. Such LSPs do not have any reserved bandwidth of their own, but share the bandwidth reserved by the LSC. Each such LSP is only used when the link it protects fails.
4) The *Monitor* node keeps running counters of the LSC rounds on $\mathbf{T}$ and $\mathbf{R}$ cycles, $Count_{\mathbf{T}}$ and $Count_{\mathbf{R}}$, respectively. Those counters indicate the next round numbers, and they are both initialized to zero.
5) The *Monitor* node starts transmitting on the LSC, and on both $\mathbf{T}$ and $\mathbf{R}$ cycles, using the round numbers contained in $Count_{\mathbf{T}}$ and $Count_{\mathbf{R}}$, respectively. These counters are then incremented. In each of the $\mathbf{T}$ cycles, the *Monitor* node also initializes all bits in the round map to 1, except for the bit that corresponds to the Monitor node, where it is reset to 0. In round 0, it also appends the $t_x^{\mathbf{S}}(0)$ data unit to the end of the label. In each of the $\mathbf{R}$ cycles, the *Monitor* node also initializes all round map bits to 1. It attaches an empty data unit (all zeroes) at the end of the label in round 0. The

Monitor node repeats this operation (while updating the counters), until the counters reach a, as defined in equation (1) above. If it receives lower numbered rounds on both cycles, it buffers them until the counters reach the value of a.

Failure Free Operation:
In this case, only the LSC cycle will be used.

At the Monitor Node:

As indicated above, suppose the *Monitor* node is node T_x, and suppose it communicates with node T_y using a straddling link.

1) Because of step 5 in the *Initialization* phase above, all rounds will be self clocked.
2) When the nth round of cycle $\mathbf{T}$ of LSC arrives:
   - If $Count_{\mathbf{T}} < n + a$, the round data is buffered at the *Monitor* node until the $Count_{\mathbf{T}}$ reaches $n + a$.
   - If $Count_{\mathbf{T}} \geq n + a$, the node changes the round number to $Count_{\mathbf{T}}$, increments the counter, and adds (XORs) the new $t_x^{\mathbf{S}}(Count_{\mathbf{T}})$ data unit, and the $t_y^{\mathbf{S}}(n)$ which it received on the straddling link to the trailing data unit field. It also complements the bit map for node T_x.
3) When the nth round of cycle $\mathbf{R}$ of LSC arrives:
   a) If $Count_{\mathbf{R}} < n + a$, the round data is buffered by the *Monitor* until $Count_{\mathbf{R}}$ reaches $n + a$.
   b) If $Count_{\mathbf{R}} \geq n + a$, the node changes the round number to $Count_{\mathbf{R}}$, increments the counter, adds $t_x^{\mathbf{S}}(n)$, in order to cancel the same data unit added by T_y, and also adds the $t_y^{\mathbf{S}}(n)$ data unit it received on the straddling link to the data unit field. The monitor node, T_x complements its map bit.

At a Type 1 Node:

Let the node be T_i and let it have a straddling link to node T_j, on which it will receive the $t_j^{\mathbf{S}}$ data units.

1) When the nth round of cycle $\mathbf{T}$ of the LSC arrives at node T_i, it complements its map bit. Then, it adds $t_i^{\mathbf{S}}(n)$ to the trailing data unit field. If the map bit for T_j is the same as the map bit for T_i, then it also adds $t_j^{\mathbf{S}}(n)$ to the trailing data field. Otherwise, it adds $t_j^{\mathbf{S}}(n - a)$.
2) When the nth round of cycle $\mathbf{R}$ of the LSC arrives at node T_i, it complements its map bit. Then, it adds $t_j^{\mathbf{S}}(n)$ to the trailing data unit field. If the map bit for T_j is the same as the map bit for T_i, then it also adds $t_i^{\mathbf{S}}(n)$ to the trailing data field. Otherwise, it adds $t_i^{\mathbf{S}}(n - a)$.

At a Type 2 Node:

Under failure free operation, normally the LSC cycle will be carrying data units all the time, and Type 2 nodes will not act on the data units carried by the LSC cycles. However, in case the LSC units are delayed[3], then the Type 2 nodes will assume on-cycle link failures, and act in an opportunistic manner. Therefore, if node T_i observes absence of data on its cycle $\mathbf{T}$ incoming link, it sends $t_i^{\overline{\mathbf{C}}}$ on its clockwise secondary protection LSP using the same bandwidth of the LSC. However, if it observes absence of data on its cycle $\mathbf{R}$ incoming link, it sends $t_i^{\mathbf{C}}$ on its counter-clockwise secondary protection LSP, also using the bandwidth allocated to the LSC. These data units will be relayed by all nodes until they reach the receivers, $\overline{\mathbf{C}}(i)$ and $\mathbf{C}(i)$, respectively. Since this is a failure free operation mode, these data units will be duplicates and will be ignored by the receivers.

Operation in the Case of Failures:
We distinguish between two types of failures, a straddling link failure, and an on-cycle failure.

Case I: A Straddling Link Failure:
Suppose the straddling link between nodes T_i and T_j fails. Nodes T_i and T_j must be Type 1 nodes. In this case, the two end nodes of the link will detect the failure by observing the absence of data units, and will start using the information transmitted on the LSC to recover $t_j^{\mathbf{S}}(n)$ and $t_i^{\mathbf{S}}(n)$, respectively, where n is the round in which the failure was detected. Data units transmitted in higher numbered rounds will be recovered similarly. Node T_i will do the following:

1) Invert the node T_i map bit in round n of cycle $\mathbf{T}$.
2) Invert the node T_i map bit in round n of cycle $\mathbf{R}$.
3) Add the data received in round n on cycle $\mathbf{T}$, to the data received in round n on cycle $\mathbf{R}$. Call the sum A. Then, perform one of the following two steps:
   a) If the map bit in $\mathbf{R}$ for T_j is the same as the new map bit for T_i, add $t_i^{\mathbf{S}}(n)$ to A;
   b) Otherwise, add $t_i^{\mathbf{S}}(n - a)$ to A.

   A should now contain either $t_j^{\mathbf{S}}(n)$ or $t_j^{\mathbf{S}}(n-a)$, depending on whether the map bit in $\mathbf{T}$ for T_j is the same as that for T_i or not, respectively. The rest of the operation to update the $\mathbf{T}$ and $\mathbf{R}$ data is similar to that in the Failure Free Operation mode described above.

Node T_j behaves similarly to recover the $t_i^{\mathbf{S}}(n)$ data units.

Case II: An On-Cycle Link Failure:
Suppose the on-cycle link between nodes T_i and T_j fails, where $T_j = \mathbf{C}(T_i)$, and $T_i = \overline{\mathbf{C}}(T_j)$. Such nodes may be either Type 1 or Type 2 nodes. In this case, node T_i observes absence of SLP protection data units on its cycle $\mathbf{R}$ incoming link, and it sends $t_i^{\mathbf{C}}$ on its counter-clockwise secondary protection LSP using the same bandwidth of the LSC. Also, node T_j will not receive SLP data units on the incoming link of the $\mathbf{R}$ cycle, and will send its $t_j^{\overline{\mathbf{C}}}$ data units on its clockwise secondary protection LSP using the bandwidth allocated to the LSC. These data units will be relayed by all nodes until they reach the receivers, T_j and T_i, respectively.

E. Properties of the Hybrid 1+N Strategy

The above strategy has the following properties:

1) It provides 100% protection against single link failures.
2) It recovers from on-cycle link failures within the delay around the p-Cycle, τ.

[3] It is assumed that the operation is synchronized, and such delays will not take place. However, this provision is included in cases where packet multiplexing is employed, and data units on protection LSPs are delayed.

3) It recovers from straddling link failures within the delay around the p-Cycle and the longest straddling link, $\tau + l$, where τ and l are the propagation delays of the p-cycle and the longest straddling link, respectively.
4) It does not require any cooperation from either of the management or the control planes during normal operation, while the complexity of the forwarding plane is not affected.
5) It also does not require reconfiguring any switches.
6) It can be implemented at higher layers, hence lending flexibility to protection.

Because of properties 2 and 3, p-Cycles used for protection may have to be limited in length in order to provide upper bound guarantees on the outage time.

IV. Performance of the Hybrid 1+N Protection

In this section, we provide some performance results to illustrate the cost of the proposed scheme. The cost we use in this section is in terms of the number of links to provide protection for all links in the network. We compare the cost of the hybrid 1+N protection scheme to that of the 1+1 protection. The cost is based on optimal solutions which are evaluated using ILP formulations similar to those in [8] for the case of 1+1 protection [4], and in [9] for the case of hybrid 1+N protection. We assume that there is no upper bound on the number of links per span. Such a restriction can be included in the formulations, but it was chosen not to include it in order to obtain the most optimal solution for 1+1 protection.

The experiments considered a number of networks where the number of nodes assumed two values, 8 and 14 nodes. We allowed the graph density for each network to assume one of four values, namely, 1, 1.5, 2 and 2.5. The graphs were generated randomly, but we made sure that all graphs were at least 2-connected. For each network, 8 different random graphs were generated, and we took the average of the results.

In Table I, we show the cost of the protection circuits. For the Hybrid 1+N protection, the protection cost is shown, while the number of links which are protected as straddling links is shown between parentheses. It should be noted that the ILP formulation for the Hybrid 1+N protection case attempts to maximize the number of links which are protected as straddling links in order to allow nodes to always receive two copies of the data units at the same time. However, this does not come at the cost of increasing the cost of protection.

Under 1+1 protection, the worst case cost of protection circuits is always when the nodal degree is 2, i.e., the network has a ring topology. There is exactly one way of choosing the protection path, namely, the entire ring topology excluding the protected link. However, under Hybrid 1+N protection, the problem reduces to p-Cycle protection, where all the protected links are on-cycle links, and the cycle corresponds to the entire graph. This results in the largest percentage of protection circuits, 100%. Note that this in this case, for the Hybrid 1+N protection, there are no 1+N protected links, and it is 1:N protection. As the number of edges increases, and consequently the nodal degrees, the cost of 1+1 protection remains high, which is always around 200% of the cost of working links. Under Hybrid 1+N protection, the ratio of the protection circuits to the working circuits decreases. Notice also that as the number of edges increases, the number of links which are 1+N protected, i.e., straddling links, also increases.

TABLE I
Comparison between 1+1 and hybrid 1+N protection

$\|V\|$	$\|E\|$	1+1 Protection protection cost	Hybrid 1+N Protection protection cost (# straddling links)
8	8	56	8 (0)
	12	29.6	8.75 (3.63)
	16	32	8 (8)
	20	39.75	8 (11.88)
14	14	182	14 (0)
	21	64.63	16.38 (6.25)
	28	56	19.5 (18.5)
	35	70	14.88 (23.88)

V. Conclusions

This paper introduced a hybrid 1+N protection strategy, and a GMPLS-based implementation of this strategy. The strategy protects on-cycle and straddling links on the p-Cycle. To protect straddling links, network coding is used to linearly combine data units, and carry them on the p-Cycle. A copy of the data units transmitted on a straddling link can be extracted by the receivers from these linear combinations. At the same time, if an on-cycle link fails, end nodes of the link transmit the data units usually carried by the link, on the cycle, and in the opposite direction. The strategy does not have to explicitly detect failures, but rather have the nodes behave opportunistically and react to the absence of data units. The strategy is implemented using a combination of GMPLS standard LSPs for protecting on-cycle links, and modified LSPs for protecting straddling links. The implementation details of this strategy were presented, and the cost of implementation, in terms of link usage was evaluated, and was shown to be modest compared to the cost of implementing 1+1 protection.

References

[1] A. E. Kamal, "1+n protection in optical mesh networks using network coding on p-cycles," in *in the proceedings of the IEEE Globecom*, 2006.

[2] D. Zhou and S. Subramaniam, "Survivability in optical networks," *IEEE Network*, vol. 14, pp. 16–23, Nov./Dec. 2000.

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

[4] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. on Info. Theory*, vol. 46, pp. 1204–1216, July 2000.

[5] L. Berger *et al.*, "Generalized multi-protocol label switching (gmpls) signaling functional description." RFC 3471, Jan. 2003.

[6] A. Farrel and I. Bryskin, *GMPLS: Architecture and Applications.* Morgan Kaufman, 2003.

[7] L. Berger *et al.*, "Generalized multi-protocol label switching (gmpls) signaling resource reservation protocol-traffic engineering (rsvp-te) extensions." RFC 3473, Jan. 2003.

[8] C. Mauz, "Unified ilp formulation of protection in mesh networks," in *7th Intl. Conf. on Telecomm. (ConTEL)*, pp. 737–741, 2003.

[9] W. He, J. Fang, and A. Somani, "A p-cycle based survivable design for dynamic traffic in wdm networks," in *Proceedings of Globecom 2005.*

[4] A polynomial time algorithm like Bhandari's algorithm may also be used.