Approaches to p-cycle network design with controlled optical path lengths in the restored network state

Diane Prisca Onguetou1,2,* and Wayne D. Grover1,2,3

1TRLabs, #401 9426–51 Avenue NW, Edmonton, Alberta T6E 5A6, Canada
2Department of Electrical and Computer Engineering, University of Alberta, 2nd Floor, 9107–116 Street NW, Edmonton, Alberta T6G 2E, Canada
3E-mail: grover@trlabs.ca
*Corresponding author: donguetou@trlabs.ca

Received November 2, 2007; revised April 14, 2008; accepted May 14, 2008; published June 24, 2008 (Doc. ID 89333)

In a transparent optical network it is desirable to have design control over the length of normal working paths and over the end-to-end length of paths in any restored network state. An obvious approach with p-cycles is to limit the maximum allowable circumference of candidate cycles considered in the network design. But this is somewhat inefficient and does not directly control the end-to-end length of paths in a restored state; it only controls the maximum length of protection path-segments that might be substituted into a working path on failure. Another basic strategy is now considered. It consists of systematically matching shorter working paths with longer protection path-segments through p-cycles, and vice versa, with direct consideration of the end-to-end length of paths in the restored network state during the design. This complementary matching notion is studied through an integer linear programming (ILP) model to minimize cost while intelligently associating longer working paths with shorter protection path-segments and vice versa. The basic ILP is adapted in one case to minimize the average restored state path lengths; in another to achieve the least possible longest path length; and, finally, to also constrain all restored path lengths under a fixed limit. Each variation can also be subject to a requirement of using only the theoretically minimal spare capacity or, through bi-criteria methods, a minimal amount of additional spare capacity for the corresponding objective on path lengths. Taken overall the work provides the means to design an entire transparent survivable island that respects the transparent reach limits of a given ultra-long-haul technology. A heuristic combination of ILP and genetic algorithm methods is also developed to solve some of the larger problems and is shown to perform well.

© 2008 Optical Society of America
OCIS codes: 060.4257, 060.4261, 060.4254, 060.4256, 060.4251, 060.1155.

1. Introduction

In optical transport networks lightpath signals undergo degradation in the optical domain due to factors primarily associated with the length [and hence loss and other accumulated impairments affecting the signal-to-noise ratio (SNR)] of optical paths. To ensure a signal with adequate postdetection bit error rate (BER) levels ($<10^{-12}$ at least), an optical signal eventually may require regeneration along its path. However, regeneration is costly because it involves electro-optical conversions. So it is generally desirable to try to avoid, or minimize, the need to regenerate signals en route during the network design phase.

A networking approach related to this issue is the concept of a “translucent” network. A translucent network consists of optically “transparent” domains or islands interconnected with electro-optical switches having regeneration capabilities. The idea is not to require any intermediate signal regeneration in the same island. This functionality is, instead, confined at domain boundaries. This constrains, in a specific domain, optical lightpaths to be under the maximum length that the signal can travel before the SNR degenerates unacceptably. A typical maximum transparent distance of currently sold ultra-long-haul (ULH) dense WDM systems is reported in [1] to be about 4000 km.
Routing lightpaths over the shortest routes is an obvious way to reduce the length of working paths in normal operation. Defining length-limited protection paths can take more consideration, however, depending on the self-healing architectures implemented in the survivable network. In the case of 1+1 automatic protection switching (APS) [2], for example, the longer half of the smallest circumference-size cycle joining the two end-nodes of a given demand-pair would usually be considered as the restoration path, and the shorter side used for the working route. The backup is then the next-shortest route fully disjoint from the primary one, and it is simple to see if a length limitation is met or not in the restored state (although for 1+1 APS the protection path length may be high due to the full path disjointedness requirement, especially in sparse networks). In more capacity-efficient path-protecting architectures, such as shared backup path protection (SBPP) [2] or demand-wise shared protection (DSP) [3], it is still simple to judge if a restored state path length is under the limit or not, because protection paths completely replace the failed working paths, end-to-end.

But controlling end-to-end optical path lengths in the restored network state is not as simple with span-protecting architectures because the end-to-end path length will depend, in the event of span failure, on both the length of the protection path-segment and the lengths of the non-failed portions of the affected working path. Nonetheless, span-protecting architectures, and in particular p-cycles, offer other advantages such as speed, locality of action, minimal database state dependencies, and the property of pre-cross-connectedness, which may be key to use in a transparent optical network [4]. There is, therefore, motivation to consider methods that can efficiently control the restored network end-to-end path lengths in the design process for p-cycles.

1.A. Overview and Contribution
This study explores p-cycle network design with the purpose of reducing the length of end-to-end lightpaths in the restored state, when a span failure occurs. The basic strategy to be explored consists of systematically matching longer working paths crossing a failed span with the shorter path-segments within the p-cycles available to protect that span. Ideally the aim will be to thereby control the end-to-end length of all paths in the restored network state, as part of a new p-cycle minimum spare capacity design model. The main technical difference over prior studies is that now in the design problem we will choose p-cycles and produce the related failure protection preplanning information with the original lengths of working paths taken into consideration. Attempts to reduce or limit optical path lengths in the restored state have previously been based on the more indirect approach of limiting the circumference of cycles [5] or the length of segments used within cycles [6], without taking information about the original lengths of working paths into account as we do now. Under the matching strategy, we will not have to limit candidate cycle lengths with the advantage that we can still use the set of all possible p-cycle candidates in design, leading to the most capacity-efficient solutions. And we will have globally optimized control of the end-to-end optical lightpaths in the restored state, not just control of the length of a maximum protection path-segment.

Figure 1 illustrates the idea of matching longer working lightpaths with shorter protection path-segments through available p-cycles. Figure 1(a) shows two p-cycles that protect against failures on the span Zurich to Prague, in the COST239 pan-European network. The longer cycle is in a so-called “on-cycle” relationship to the protected span and thus provides one protection path-segment for it in the event of failure. The same span is in a “straddling” relationship to the other shorter p-cycle, which offers two protection path-segments to it. Three working paths, shown in Fig. 1(b), cross the span from Zurich to Prague in normal operation, i.e., Zurich–Prague, Paris–Zurich–Prague, and Copenhagen–Prague–Zurich–Milan. In total there are enough protection path-segments to fully protect working paths against failure of the span in question, but whether all affected working paths wind up with suitably limited end-to-end optical path lengths will depend on the allocation of the available protection path-segments.

To illustrate the basic concept, the end-to-end path in the restored state from Copenhagen to Milan is clearly very long in the arbitrary assignment shown in Fig. 1(c). For illustration we have associated a failed channel on the longest working path with the longest protection segment possible through the two available p-cycles shown for the given failure. Although perhaps not a preferred mapping, there would be many
situations where it would not technically matter if this mapping arose. For instance, one could observe that in the bi-directional line switched ring (BLSR), we—in effect—always accept the worst mapping, i.e., an entire loopback around the ring. But once operational, this does not actually matter as long as the ring size obeys length design limits.

However, if we are trying to adhere to some maximum transparent domain length limit that is as large as possible, we can obviously see that a better association of working paths to protecting \( p \)-cycles is possible than in Fig. 1(c). An example is the mapping in Fig. 1(d), which greatly reduces the maximum length of the end-to-end path in question in the restored state. [To simplify the illustration in Figs. 1(c) and 1(d) only the working path for Copenhagen–Milan is shown in its restored state.] As long as the preferred mapping for the path illustrated can be obtained without requiring other affected paths to exceed length limits, then that is the idea we are pursuing as preferable. In the design problems we consider we will not always be limited to just making the best associations within an existing otherwise ordinary set of minimum-capacity \( p \)-cycles. We will do that as a first instance but then go on to also choose the \( p \)-cycles that best suit the overall strategy, ideally with little or no increase in overall spare capacity.

Matching longer working paths with shorter path-segments through available \( p \)-cycles in the event of failure is a simple principle to state. Although simple in con-

Fig. 1. Notion of matching long working paths with short path-segments in the available \( p \)-cycles. (a) Zurich–Prague span protecting \( p \)-cycles. (b) Three working paths share the span connecting Zurich to Prague. (c) A blind assignment of \( p \)-cycles to working paths, which allocates the longer path to the longer cycle, when the span from Zurich to Prague fails. (d) Systematic mapping of shorter paths to the longer restoration path-segments in the case of Zurich–Prague span failure, e.g., traffic from Copenhagen to Milan.
cept, we venture that the reason this approach has not been developed yet for p-cycles is that the basic design problem for p-cycles (as with other span-protecting principles) is in a framework that is not required to resolve path information, only to protect working channels (counts per span). Technically there is considerable added complexity to bring the end-to-end routing of every working path into visibility in the design problem.

1.B. Background and Related Work

The method of p-cycles is a now widely studied pre-cross-connected span-protecting architecture for network survivability. A conventional p-cycle protected network consists of a set of cyclical structures of spare capacity, which provides 100% restorability against single span failures. Where wavelength conversion is assumed at all nodes, the length of protection lightpaths is not of primary concern because regeneration happens at every node. Thus in the most commonly used basic design model for minimum spare capacity we just need to ensure that there is enough p-cycle capacity available to protect all the failed working channels on any span failure. The routing and length of prefailure working paths does not even appear in the problem. That information is in effect just boiled down into the number of working channels on each span. (More on the basic logical capacity design p-cycle networks can be found in [7,8].)

In a transparent optical network the basic design problem is of greater complexity because we also must not exceed specific optical path length limits. In earlier work on this problem optical restored state path lengths were indirectly limited or reduced by using only circumference-limited p-cycle candidates, instead of considering all possible cycles as eligible in the design [5,8]. This works if there is some upper limit on cycle circumference size that can be stated independently of any working path length such that the combined maximum lengths will still be adequate. But that condition may cause many paths to be shorter than needed or the diameter of a transparent domain to be smaller than it otherwise might be with the present approach. It is also applicable in the case where working paths (and p-cycles) may both be optically transparent but there is regeneration specifically (and only) located at the entry points of working paths into p-cycles. The latter architecture is one of the strategies considered in [9].

An improvement over just limiting candidate cycle sizes is found in [6], which recognizes that with respect to any given protected span, a given p-cycle may offer a “long side” and a “short side,” and we may preferably choose the short side or exclude any use of the long side if it is too long. But in [6] we still always do so without regard to the affected working paths’ original length. This approach can minimize or strictly cap the maximum of the length of protection path-segments without directly constraining the circumference of candidate p-cycles used in the design. Although the work in [6] distinguished between the long and short sides of a p-cycle with respect to prospective span failures, the method involved simply not using any protection path-segment that in its own right exceeded a length limit. There was no consideration or direct control over the combined length of working paths plus protection path-segments they might use. A somewhat related work [10] aimed at showing the flexibility possible in p-cycle network design in terms of possible operator objectives of using only small p-cycles or using the fewest number of p-cycles in total, and so on. That work, however, as with the other references above, did not introduce any methods for direct control of end-to-end optical path lengths in the restored network state.

From here we proceed as follows: Section 2 develops the concept of matching longer working paths with shorter protection path-segments through available p-cycles in an integer linear programming (ILP) p-cycle minimum spare capacity network design model and shows how variants of that model are created for various strategies for implementing the basic idea. For larger or more difficult problem instances, Section 3 then introduces a genetic algorithm (GA) to come up with a reduced but highly effective set of candidate cycles (a kind of preselection strategy such as in [11]), on which the fully detailed ILP problem model can then be solved. Section 4 studies the improvements in restored path optical lengths resulting from implementation of the “matching” concept in p-cycle protected networks. Section 5 is a concluding discussion and thoughts on future work.
2. ILP Models for Complementary Matching of Working Paths and Protection Segments

This section introduces mathematical design models that aim to match, in various ways, the longer working paths crossing a span with the shorter path-segments through p-cycles protecting the span in question. Subsection 2.A.1 first recalls the conventional p-cycle minimum spare capacity network design model for reference. Subsection 2.A.2 provides ILPs that perform an optimal assignment of protection path-segments to working paths within the otherwise ordinary set of p-cycles produced by an instance of the conventional model for minimum spare capacity in a way that minimizes the average lengths of all restored path states. Subsection 2.A.3 considers a variant that minimizes the maximum optical path length in the restored state. Subsection 2.A.4 introduces bi-criterion objectives to optimize the intelligent matching effect while retaining minimal spare capacity or more generally used to explore any feasible trade-off between spare capacities and minimized optical path lengths. The last ILP is further adapted in Subsection 2.A.5 to produce designs having an explicit maximum length limit on paths in the restored state. The following definitions serve for all variants of the ILP model for those problems.

**Sets:**
- $\mathcal{N}$ is the set of nodes in the network and is indexed by $k$.
- $\mathcal{S}$ is the set of spans in the network and is indexed by $i$ for failing spans and $j$ for surviving spans or spans in general.
- $\mathcal{D}$ is the set of demand-pairs and is indexed by $r$. Units of a given demand-pair $r$ have the same origin and destination nodes. The present models assume that all unit-capacity paths in the demand bundle have the same working route.
- $\mathcal{P}$ is the set of eligible candidate cycles indexed by $p$. Without changing the ILP model itself, $\mathcal{P}$ may be populated for different purposes with all distinct simple cycles of the graph or (in the GA-ILP to follow) some kind of smaller “elite” subset of candidate cycles selected by a preprocessing method.
- $\mathcal{B} = \{\text{left, right}\}$ is a set indexed by $b$ to differentiate the two sides of each p-cycle under span failure conditions. Given a failed span straddling a specific p-cycle, the shorter protection path-segment is considered here as the left side of the p-cycle, while the longer one is considered as its right side. Each on-cycle span has uniquely a left side, i.e., there is no right side.

**Parameters:**
- $C_j$ is the cost of each capacity unit on span $j$.
- $d_r^i$ is the number of units for demand-pair $r$.
- $\delta_j^i \in \{0, 1\}$ encodes spans on the working route of demand-pair $r$. $\delta_j^i = 1$ if the working route of demand-pair $r$ crosses span $j$, and $\delta_j^i = 0$ otherwise.
- $x_j^i \in \{0, 1, 2\}$ encodes the number of protection route-segments that one unit-sized copy of p-cycle $p$ provides to span $i$. $x_j^i = 2$ if span $i$ straddles the p-cycle $p$, $x_j^i = 1$ if span $i$ is on the cycle, and $x_j^i = 0$ otherwise.
- $s_j^i \in \{0, 1\}$ encodes the spans crossed by p-cycle $p$. $s_j^i = 1$ if span $j$ is on the cycle (i.e., $x_j^i = 1$), and $s_j^i = 0$ otherwise.
- $r_{j,b}^i \in \{0, 1\}$ encodes spans in protection path-segments provided by p-cycle $p$ to a failed span $i$. $r_{j,b}^i = 1$ if side $b$ of the p-cycle $p$ crosses span $j$ in the event of span $i$ failure, and $r_{j,b}^i = 0$ otherwise. $r_{j,b}^i \text{ "right"}$ is always equal to 0 for on-cycle failed spans, since p-cycle $p$ does not have a right side for the spans crossed en route (i.e., $s_j^i = 1$). The pre-computation of input parameters $r_{j,b}^i$ requires information about span end-nodes, which is part of the traditional preprocessing network data file.
- $L_j^i$ is the optical length of the end-to-end paths in the restored state of demand-pair $r$, which results from the usage of side $b$ of the p-cycle $p$ in the event of span $i$ failure.
- $w_i$ is the number of working channels (capacity units) to be protected on span $i$ (arising from whatever routing of the demand matrix is employed.)

**Decision Variables:**
- $s_j$ is the total number of spare channels needed on span $j$ in the design; integer.
- $n^i$ is the number of unit-channel copies of candidate p-cycle $p$ used in the design; integer.
- $n_{j,b}^i$ is the number of restoration path-segments from p-cycle $p$ that are used on its side $b$ for demand-pair $r$ when span $i$ fails; integer.
- $\kappa_{j,b}^i \in \{0, 1\}$ records the usage of restoration path-segments in the models that
seek the least maximum optical path length under minimum capacity costs or constrains the path under fixed maximums. \( \kappa_{r,i}^{p,b} = 1 \) when at least one unit-sized copy of side \( b \) of \( p \)-cycle \( p \) is committed for restoring traffic from demand-pair \( r \) in the failure of span \( i \) (i.e., \( \kappa_{i}^{p,b} = 1 \) if \( n_{r,i}^{p,b} > 0 \), and \( \kappa_{i}^{p,b} = 0 \) otherwise.

- \( I^{\text{node}} \) is a constant giving the length-equivalent insertion loss of each node en route to any path. For instance, with typical optically transparent node equipment, the optical multiplexer–demultiplexer and transparent cross-connect core may produce an insertion loss per node equivalent to \( \sim 80 \) km of fiber length [12].

- \( M^e \) is a suitable large constant that serves as a surrogate for infinity.

2.A. ILP Formulations

2.A.1. Conventional \( p \)-Cycle Minimum Spare Capacity Design Model

The basic \( p \)-cycle minimum spare capacity design model is given as a starting point in Eqs. (1)–(3). The aim is to minimize total spare capacity while ensuring 100% restorability against single span failures through the \( p \)-cycles that are built in the spare capacity:

\[
\text{Minimize } \sum_{j \in S} C_j \cdot s_j, \quad (1)
\]

subject to

\[
w_i \leq \sum_{p \in P} x_i^p \cdot n^p, \quad \forall i \in S, \quad (2)
\]

\[
s_j = \sum_{p \in P} s_j^p \cdot n^p, \quad \forall j \in S. \quad (3)
\]

2.A.2. Minimizing the Total Length of Restored Lightpaths in a Conventionally Designed \( p \)-Cycle Network

A first new mathematical model is defined from Eqs. (4)–(8) for an intelligent assignment of protection path-segments to failure-affected working paths in an otherwise conventional 100% span-restorable \( p \)-cycle network design optimized for minimum spare capacity. Thus, the scope here is not to change the spare capacity, the \( p \)-cycles present, or the routing of working paths; we keep all those “as is.” This model will do the best that can be done with the matching principle in any given \( p \)-cycle network design to minimize the total (and hence average) length of optical paths over all possible paths in restored path states of the network.

The objective Eq. (4) minimizes the total length of end-to-end optical paths in the restored state. The total is run over all affected paths arising from all single span failure states. Equation (5) allocates a restoration path-segment to every working path crossing a failed span. Equation (6) certifies that protection path-segments assigned to working routes do not go through the right side for on-cycle span failures, since an on-cycle span cannot expect more than one protection segment. Equation (7) constrains the number of path-segments handling span failures in a given \( p \)-cycle to not exceed the number of unit copies of every cycle that is available in the design. Equation (8) finally records statistics on the resulting restored state path lengths:

\[
\text{Minimize } \sum_{r \in D, i \in S, p \in P, b \in B} L_{r,i}^{p,b} \cdot n_{r,i}^{p,b}, \quad (4)
\]

subject to

\[
\sum_{p \in P, b \in B} n_{r,i}^{p,b} = d^r, \quad \forall r \in D, \quad \forall i \in S: \delta_i = 1, \quad (5)
\]

\[
n_{r,i}^{p,b} \leq \left( \sum_{j \in S} n_{j,i}^{p,b} \right) M^e, \quad \forall r \in D, \quad \forall i \in S, \quad \forall p \in P, \quad \forall b \in B, \quad (6)
\]

\[
n^p \geq \sum_{r \in D} n_{r,i}^{p,b}, \quad \forall i \in S, \quad \forall p \in P, \quad \forall b \in B, \quad (7)
\]
\[ L_{p,j}^{r,b} = \sum_{j \in S: \ j \neq i \text{ and } (\delta_j = 1 \text{ or } \delta_j^{b,j} = 1)} (I_{mode} + C_j) - I_{mode}, \quad \forall r \in D, \forall i \in S, \forall p \in P, \forall b \in B. \]

2.A.3. Minimizing the Maximum Length of Restored Lightpaths

The effect of the objective function in Eq. (4) is to provide a general improvement to the length of paths in the restored state. This may help from an economic point of view in that shorter paths on average use less equipment overall. In [13] Gunkel et al. discuss some cost models to price signal processing (optical amplification, optical-to-electrical-to-optical (o-e-o) conversion, etc.) along a lightpath in the WDM layer. A different viewpoint would be to say that from an optical reach performance standpoint, minimizing the single longest path that arises anywhere in the network for any given restored state would be a more important objective. This gives rise to a “minimax” type of optimization problem. To effect this objective, Eqs. (9) and (10) replace the objective function Eq. (4), and Eq. (11) states that the minimax arises out of used protection path-segments only. The other constraints, i.e., Eqs. (5)–(8), would remain the same.

\[ \text{Minimize } X, \]

subject to Eqs. (5)–(8),

\[ X \geq L_{p,j}^{r,b} \cdot n_{p,j}^{r,b}, \quad \forall r \in D, \forall i \in S, \forall p \in P, \forall b \in B; \]

\[ n_{p,j}^{r,b} \leq k_{p,j}^{r,b} \cdot M, \quad \forall r \in D, \forall i \in S, \forall p \in P, \forall b \in B. \]

2.A.4. Bi-criterion Network Design Model

The ILP presented in Subsection 2.A.2 optimizes the matching of restoration path segments with working paths in any given or existing p-cycle protected network. Knowing that the p-cycle minimum capacity design model might have many different solutions for the same capacity costs, we can obtain a refined model that will select a solution of equal or similar cost to the optimal capacity solution, but which potentially can further minimize the restored state path lengths.

This involves combining the ILPs of Subsections 2.A.1 and 2.A.2 and introducing the bi-criterion objective function in Eq. (12). The aim is still to minimize spare capacity as the primary goal, but now to do so with concern for reducing optical path lengths as well. In one mode of use the parameter \( \alpha \) is chosen to be small enough not to upset the principal objective of staying at minimal spare capacity, while biasing the design towards selection of cycles that will reduce lengths of restored state paths. (We mainly use the model in this mode.) The other mode is where \( \alpha \) is made large enough, and varied, to produce the set of Pareto-optimal trade-off points between ever lower average path lengths and increasing total spare capacity:

\[ \text{Minimize } \sum_{j \in S} C_j \cdot s_j + \alpha \sum_{r \in D, j \in S, p \in P, b \in B} I_{p,j}^{r,b} \cdot n_{p,j}^{r,b}. \]

The bi-criterion model has the same constraints as above, i.e., Eqs. (3) and (5)–(8). Equation (2) is, however, no longer required because Eq. (5) is sufficient to guarantee full restorability to every demand-pair crossing a failed span. In the same way, Eq. (7) now also calculates the number of unit-sized copies of selected p-cycles as well as constraining the number of protection path-segments assigned under what is available, as in Subsection 2.A.2.

Similar improvements can be obtained by merging the Subsection 2.A.3 design model, instead of the Subsection 2.A.2 model, with the conventional ILP given in Subsection 2.A.1. The bi-criterion objective will then consist of limiting the maximum length of restoration paths while also minimizing the total required spare capacity. The corresponding objective function is given in Eq. (13). The constraint Eq. (10) is added to the new bi-criterion model accordingly:
2.A.5. Asserting an Absolute Transparent Path Length Limit

Equations (11) and (14) can also be added if one desires to strictly disallow paths over some fixed length limit "\(L_{\text{max}}\)." Note that this also freely allows any path length up to the limit, whereas minimax forces all paths to be lower than the lowest possible longest path length. One can then also either use the bi-criterion objective function given in Eq. (12) or focus on minimizing spare capacity only [Eq. (1)]:

\[
L_{r,i}^{p,b} \leq L_{\text{max}} \cdot k_{r,i}^{p,b}, \quad \forall \ r \in D, \ \forall \ i \in S, \ \forall \ p \in P, \ \forall \ b \in B. \tag{14}
\]

The difference between this model and the one discussed in Subsection 2.A.3 is that here the lengths of all optical lightpaths are constrained under the absolute limit \(L_{\text{max}}\) and we allow that doing so may involve an increase of the total spare capacity. This is in contrast to the ILP in Subsection 2.A.3, where transparent distance was minimized without extra capacity over minimum cost requirements.

Table 1 numbers and summarizes all these design models. All ILP models were implemented in AMPL 10.1 and solved using CPLEX 10.1.0, with a mixed integer program gap for optimality (MIPGAP) of \(10^{-4}\) on a four-processor Sun UltraSparc III running at 900 MHz with 16 Gbits of RAM. Preprocessing for initial routing, eligible \(p\)-cycles, and all the other input parameters for the ILPs were done on a four-processor Pentium 3 GHz with 512 Mbits of RAM, running Windows 2000. Corresponding running times as well as full completion aspects of related problem instances are later discussed in Section 4.

3. Preselection of Candidate Cycles Using Genetic Algorithms

Before getting to the results, we have one more method to describe. This is a GA-based approach to "preselection" of reduced-size sets of candidate cycles for use in cases where the third, fourth, and fifth ILP models become too large for a practical solution using the complete set of candidate cycles of the graph. The latter three models in Table 1 are especially challenging as they require variables and constraints for each path-segment that may be employed in each possible cycle for each single span failure. Thus, in practice our approach is to manage the size of the problem instances by restraining the number of eligible cycles. From prior experience with preselection approaches in general, however, we know that for the best solution quality (in terms of minimum spare capacity solutions), the preselection criteria cannot be as simple as just taking the \(n\) shortest or longest cycles, or even purely the \(n\) cycles with top-

### Table 1. Summary of the ILP Mathematical Models Introduced in This Paper

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>ILP Formulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model 1</td>
<td>Conventional reference ILP (p)-cycle minimum capacity design.</td>
<td>Reference values and actual cycles for use in model 2—Eqs. (1)–(3).</td>
</tr>
<tr>
<td>Model 2</td>
<td>Minimum average restored state path length model, given a 100% span restorable network design as input.</td>
<td>Assuming an existing set of (p)-cycles and working paths—Eqs. (4)–(8).</td>
</tr>
<tr>
<td>Model 3</td>
<td>Minimax restored state path length model.</td>
<td>Eqs. (5)–(11). The bi-criterion objective given by Eq. (13) may also replace Eq. (9).</td>
</tr>
<tr>
<td>Model 4</td>
<td>(p)-cycle minimum capacity planning under overall optical path length minimization.</td>
<td>Bi-criterion minimization of both capacity and path lengths—Eqs. (12), (3), and (5)–(8): i.e., models 1 and 2.</td>
</tr>
<tr>
<td>Model 5</td>
<td>ILP design model constraining optical paths under a specific fixed limit.</td>
<td>Model 4 plus absolute length limit—model 4 and Eqs. (11) and (14).</td>
</tr>
</tbody>
</table>
ranked “a priori efficiency” (AE) such as in [8,11], for example. Accordingly, for the present effort we have experimented with a new approach to preselection of reduced size candidate cycle sets based on a GA. The reason that the GA approach to follow is a significant advance within the domain of preselection-based methods is that, by its nature, it is able to identify combinations of candidate cycles that collectively may offer important advantages to the final ILP instance solving a given problem with the reduced space of candidate cycles. Previously, all work on preselection-based approaches evaluated and ranked or selected cycles on the basis only of individual attributes or figures of merit such as AE, for example.

To state it first in a summary form, the basic idea of the GA-ILP is as follows: some ILP problem is to be solved, but is too large to solve with the complete set of candidate structures. The same ILP (or potentially a surrogate cousin ILP) can, however, be used as the fitness function for a smaller subset of candidate structures that comprise an individual. The hypothesis is that the normal steps of a GA-like iteration will evolve a still suitably small overall population, the union of which embodies a preselected subset of candidate structures on which a final, only somewhat larger instance of the original ILP problem, can very likely be solved to the same solution quality that could be achieved had the problem been solvable in the first place with the entire set of candidate structures. With this conceptual overview in mind, let us provide details of the method as used here.

3.A. Encoding
We initially partition the set \( P \) of all possible distinct simple cycles of the graph (i.e., all possible candidate \( p \)-cycles) into \( n \) subsets, each containing a nominally equal number of eligible cycles. These subsets make up our initial population of which every individual consists of a possible set of \( \left\lfloor \frac{|P|}{n} \right\rfloor \) candidate cycles. Intuitively, we must not make \( n \) so high that a feasible solution for the \( p \)-cycle network design problem is not possible for each of the above individuals. However, if this happens, the solver quickly reports it and we can work with fewer slightly larger candidate subsets.

Generally the idea is to have \( n \) as high as possible, so there are the most individuals in the population, each easily evaluated (as below) without resulting in infeasibility. In the specific implementation here, we identify, prior to the GA iterations, the \( p \)-cycles that are able to provide protection path-segments to each span. Next we partition \( P \) in a way that the \( p \)-cycles of each subset cover all spans through the network, so they provide solutions for \( p \)-cycle network design. Making \( n \) less than (or equal to) the number of \( p \)-cycles protecting the span that is covered by the least number of cycles is therefore suitable to guarantee the existence of feasible solutions to the \( p \)-cycle network design problem for each individual of the initial population.

3.B. Evaluation
We use the regular \( p \)-cycle ILP minimum spare capacity design model (which is fast and easy to solve) as the fitness function for the GA and the index numbers of the cycles constituting each individual defines its genome, to follow the GA framework. In other words, the evaluation process consists of running the regular version of the \( p \)-cycle design problem (i.e., model 1 above, with no optical length control) using each of the individuals as the candidate cycle set in turn. Every individual is then assigned a weight that is the spare capacity cost that its optimal solution involves; if the problem was not feasible for certain individuals, the weight is set to a very high number.

This ILP fitness function is not relatively complex to evaluate and is also by definition of a controllable size, whereas the direct complete ILP design problem (i.e., with optical path length considerations) could often be too large. Instances of the above-defined fitness function run in a couple of seconds to a few minutes even for bigger networks—e.g., for 3531 eligible cycles in the COST239 long-haul European network we get the optimal solution in less than a minute under the experimental conditions described earlier in Section 2.

3.C. Crossover
Thus, the individuals of this GA are candidate cycle sets and their fitness function is the objective function value of the corresponding conventional ILP \( p \)-cycle minimum spare capacity network design model. For crossover, we select the \( n/2 \) best pairs of individuals: the smaller the sum of the weights of two parents is, the better that par-
ent pair is for reproduction. Every parent pair selected for breeding produces two children by crossing the first half of one parent’s genome (set of cycle index numbers) with the second half of the genome of the second parent, and vice versa.

A problem-specific aspect of the GA method developed here is that following crossover, all actual solution \( p \)-cycles of individuals not selected for breeding are kept and used as the raw material for mutation. For example, an individual may consist of 118 candidate cycles and its spare capacity ILP solution may be based on a specific dozen or fewer solution \( p \)-cycles (typically). If the spare capacity value of the individual is not low enough to rank it for reproduction, it will not continue into the next generation as an individual, but its specific solution cycles will be kept and used to strengthen and maintain genetic diversity in the offspring.

3.D. Mutation
Mutation follows crossover. In our case the operation is that given a “child” in an offspring \( t+1 \), we substitute some (randomly chosen) cycles from the child just mentioned above for solution cycles from any of the individuals that were not selected for crossover operations in the previous generation \( t \). The mutation process is therefore concerned with at most half of the offspring because the \( n/2 \) best parent pairs of individuals were selected for breeding, i.e., at least one half of the \( n \) individuals were subject to crossover and no more than half will stay for mutation.

This may, of course, be deleterious to the individual, as is the nature of GA, but the philosophy is that these cycles may embody some meritorious design elements given that they arose as solution \( p \)-cycles albeit from individuals that did not go on to reproduce. Additionally we note that due to mutation by random substitution of cycles to a few individuals we may have, at the GA completion time, slightly different fitness values and solutions for two experiments under the same conditions. In the results that follow, the specific mutation policy consisted of inserting the solution cycles from one of the individuals not selected for breeding into a randomly chosen one of the child individuals of the same iteration and removing the corresponding number of other cycle indexes from that child.

3.E. Terminating Conditions
The regular \( p \)-cycle ILP minimum spare capacity design model is used to evaluate the fitness of the new individuals and the generational process (i.e., selection, crossover, and mutation) is repeated until all the individuals of a given generation have nearly the same fitness. Specifically, in the experiments here, GA iterations were stopped when every individual in the population had the same ILP objective function value for total spare capacity within the MIPGAP being employed. The set of eligible cycles for the final ILP design problem with restored state length control is then the union of all unique cycles of the individuals of the most recent population. Note that the union may lead to a small upward creep in the size of the set of candidate cycles in comparison with the size of the individuals, given that individuals of the last generation are not necessarily identical although having the same fitness, but this is not of practical concern in this method as used here. Some other problem-specific details of assembling the final candidate cycle set to solve instances of the fourth and fifth ILPs are given when describing the experimental results below.

4. Case Study and Experimental Results
We now report results of tests using the ILP models and methods above in the well-known pan-European COST239 network in Fig. 2. Subsection 4.A first characterizes working path lengths in the conventional network design under normal conditions, and Subsection 4.B discusses how the new matching strategy applied to the prior reference design performs in the event of single span failures. The ILP-based GA is next implemented to preselect a reduced number of candidate cycles for the third, fourth, and fifth ILP models, which directly control optical path lengths in the restored network state. Subsection 4.C then discusses the benefits of using the GA preselected sets in comparison with equivalent sized sets of shortest circumference cycles, and Subsection 4.D considers length optimization aspects.

*COST239* consists of 11 nodes and 26 spans, with lengths indicated in Fig. 2(a). The average nodal degree is 4.73. In this network there are 3531 distinct simple cycles of which 394 are also Hamiltonian cycles. For problems not using the GA method here, the complete set of 3531 candidate cycles are present in \( P \). The demand matrix applied to the *COST239* network contains 55 node pairs of which the number of lightpaths between each is randomly generated from a uniform distribution on the interval \([1...20]\). Each demand bundle is routed via a single shortest distance route for the non-failure conditions. The resulting working capacities per span are shown in Fig. 2(b). The total working capacity cost is 137,170 channel km.

The histogram in Fig. 3 shows the statistics of the optical lengths of the working lightpaths corresponding to the initial shortest distance routing. Here, the optical length of a path is the sum of its physical length in kilometers, with an additional 80 km insertion loss incurred at every node transited en route (i.e., \( I_{\text{node}} = 80 \) km). This is the insertion loss estimate used for node equipment given to us in some of our cooperative work with the industry [12]. Within the ILP symbology defined for this work, the optical length of a path is computed as shown in Eq. (15):

![Fig. 2. COST239 long-haul European network. (a) Length of span on edges (kilometers). (b) Working capacities atop spans.](image)

![Fig. 3. Optical transmission lengths of normal working paths alone in the COST239 network.](image)
4.B. Results for Matching Longer Working Paths with Shorter Protection Segments through p-Cycles Available in the Conventional Minimum Spare Capacity Design

Here, the p-cycle minimum spare capacity design model ILP 1 is first solved in the COST239 network, considering all possible simple cycles as eligible. This ILP reaches full termination in about 45 s under the experimental conditions indicated in Section 2. The total spare capacity cost is 85,640 channel km, and the redundancy (ratio of total distance-weighted spare capacity to total working) of this reference solution is 62.43%.

Following completion of the conventional p-cycle design model the matching strategy is applied to working paths and the specific p-cycles involved in the reference solution, using either the second or the third ILP design model formulated in Subsections 2.A.2 and 2.A.3. Table 2 records, in its first and second data columns, the completion times and statistics on resulting path lengths in the restored states. Both problem instances reach full termination in less than 2 min, including completion times of the reference network.

4.B.1. Minimizing the Average Length versus Minimax on the Length of Lightpaths in the Restored Network State

Table 2 shows that the second model provides a general improvement of restoration lightpaths with an average length of about 3869 km and 48% paths over 4000 km (which we take as a reference criterion for what could be considered excessively long paths). With the third model the average length path is higher (4162 km) and 54% of restored state paths are over 4000 km. The third ILP, however, achieves a lower maximum path length (i.e., 6745 km). Figure 4 shows, in greater detail, a comparative view of the path length statistics when matching protection path-segments with working lightpaths using the second (minimum total length) and the third (minimax) ILP models.

The first set of data in Fig. 4 is from the third design model, which achieves the minimax solution (i.e., the solution having the shortest possible longest length path). In this case the longest path is kept under 7000 km, but many other paths tend to be increased in length to accommodate this and statistically squeezed-up to just below that maximum value. In comparison, the other histogram in Fig. 4 is concerned with

<table>
<thead>
<tr>
<th>Table 2. Improving Restoration Path Lengths in an Existing COST239 Network Design</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metrics</td>
</tr>
<tr>
<td>Spare capacity (channel km)</td>
</tr>
<tr>
<td>Redundancy (%)</td>
</tr>
<tr>
<td>Minimum length (km)</td>
</tr>
<tr>
<td>Average length (km)</td>
</tr>
<tr>
<td>Maximum length (km)</td>
</tr>
<tr>
<td>&gt;4000 km</td>
</tr>
<tr>
<td>Running times</td>
</tr>
</tbody>
</table>
the second model, which minimizes the average path length in the restored network state. Here we encounter a few protection paths that are longer than (say) 7000 km, but considered all together many fewer paths tend to be as long individually as with the third ILP. (Recall, however, that so far both the second and third models are solved only within the existing design from the first model.) Henceforth, we consider only model variants that minimize the average path length in the restored state. This will restrict our scope of subsequent considerations of both cases where the context involves trying to manage restored path lengths in an already existing set of working paths and p-cycles or models where we have full design control. Some justification for being more interested in the minimum average path length case, as opposed to the minimax case, is given in the closing discussion. But clearly the methods that are developed here could similarly be worked through for the minimax case, should the reader wish.

4.B.2. Effectiveness of the Matching Strategy on Restoration Path Lengths within an Existing p-Cycle Network

Table 3 compares statistics on the length of restored state optical paths in the COST239 minimum spare capacity network design for three different ILP policies for associating protection segments available only from pre-existing solution cycles with

![Graph showing frequency distribution of path lengths in the restored network state.](image)

**Table 3. Restored State Route Lengths from Different Assignments of Protection Segments from p-Cycles to Working Paths in the COST239 Conventional Network Design**

<table>
<thead>
<tr>
<th>Mapping (Worst to Best)</th>
<th>Restored State Optical Path Lengths (km)</th>
<th>Restored State Paths over 4000 km</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Minimum</td>
<td>Average</td>
</tr>
<tr>
<td>Longest path—longest p-cycle segment</td>
<td>850</td>
<td>4625</td>
</tr>
<tr>
<td>Random assignment</td>
<td>760</td>
<td>4230</td>
</tr>
<tr>
<td>Longest path—shortest p-cycle segment</td>
<td>780</td>
<td>3869</td>
</tr>
</tbody>
</table>
the working path length (i.e., this is model 2 where we are just testing different possible policies for assignment of available protection segments to working paths for each span failure). Corresponding table entries, going from the worst to the best assignment, show that the new strategy (of which results are given in the last row) is of some benefit, even just as a way of deciding which path-segments from which p-cycles to use for each failure in an already existing p-cycle network design. More generally, the concept of matching shortest segments through p-cycles with the longest working paths does provide the best average length—i.e., 3869 km versus 4230 or 4625 km. It provides the fewest number of restored state paths of “long” length as well (again using 4000 km as a criterion), i.e., 48% over 4000 km (versus 58% or 69%). Not surprisingly, to achieve this the minimum and maximum path lengths may increase slightly, but inconsequentially. Also not too surprisingly, we see that when working within the confines of an already existing set of p-cycles, the matching principle is demonstrated to be effective, although the extent of path length reduction is somewhat limited.

4.C. Applying the Preselection Genetic Algorithm

Now we consider the ILP design models where there is freedom to select the solution p-cycles themselves, as well as the association of protection segments with working paths, with the aim of reducing restored state path lengths. As alluded to previously, to approach the COST239 network design problem with the direct path length control ILP models including full design of the p-cycles themselves, we adopt a candidate preselection strategy to reduce the set of eligible cycles in the problem instances here (i.e., these are the cases where Table 2 shows that we exceeded the 16 Gbit memory limit while attempting to solve problem instances for the third, fourth, and fifth ILP design models, using the total set of 3531 possible cycles as candidates). The GA method was used for these larger-size problem instances to preselect fewer candidate cycles.

As an initial exploration of performance with the GA-ILP methodology we tested it on the basic minimum spare capacity design problem, to which the exact solution is well known. Figure 5 shows the excess spare capacity requirements over this minimum as it depends on the number of individuals that were considered using the GA in comparison to the reference approach just preselecting the corresponding number of shorter-ranked cycles as candidates. As an example of how to understand the data in Fig. 5, using $n=30$ (for example) builds up populations in which each individual consists of 117 or 118 cycles. Figure 5 then gives the spare capacity required using any
individual of the last population. (Recall that the generational process reaches its
terminating conditions when all individuals of a given population provide the same fit-
tness within the MIPGAP being employed.) This spare capacity is compared with what
is required if instead the preselection policy was simply to use the 118 smallest
circumference-size cycles as candidates in the design. (In taking this as a comparison
benchmark for performance, we are assuming that the smallest circumference cycles
would be the most relevant simple preselection criterion for the problem of achieving
low restored state end-to-end path lengths.)

The results show that for the basic spare capacity design problem the GA approach
is well within 1% of optimality even with just 36 candidate cycles per individual. In
fact, rather unexpectedly, but understandably in hindsight, in this experiment the GA
individuals themselves were reaching strict optimality for the given problem (simply
minimum spare capacity). In this case no final ILP solution on the merged set of
evolved individuals is actually needed. We also note that in these tests the GA was
typically completed in about 7 to 12 iterations, meaning that 7 to 12 populations were
iteratively generated before the algorithm reached its terminating conditions. The
process not surprisingly ran in only a few minutes (for each iteration) due to the fact
that instances of the regular p-cycle ILP problem, which represents the fitness func-
tion in the algorithm, complete in less than a minute even when given the entire set
of candidate cycles. The full process of preselecting candidate cycles using the GA
method was generally completed in 20 to 30 min. This established that the GA-ILP
method is quite effective as a preselection strategy, at least when applied to the basic
spare capacity design problem. Comparatively Fig. 5 shows that simply choosing the
36 shortest cycles limits the final ILP solution to being not much better than about
86% above optimality while, even with as few as 36 cycles evolved by the GA, optim-
ality on this problem is achieved. Additionally, we could not get a feasible p-cycle net-
work design with less than those 36 smallest circumference-size cycles while the GA is
guaranteed to work for less than 26 cycles per individual (this is in fact the number of
spans within the network).

4.D. Network Design Models with Direct Optical Path Length Control Solved on the
Genetic Algorithm Preselected Cycle Sets

Based on the tests of the GA-ILP method above, on the basic spare capacity problem,
we went ahead and used it as a basis for providing candidate cycle sets for instances
of the fourth and fifth design models. The initial intent with the GA-ILP method was
that when addressing more complex models, such as models 4 or 5 in this work, one
then uses the corresponding fully detailed ILP model as the fitness function to evalu-
ate each individual. As this technique is still under development to nonetheless obtain
good solutions for the present study of restored state path length design strategies, we
compromised and used the following composition for the preselected sets to solve the
fourth and fifth models on COST239 for the present. Specifically, to obtain good solu-
tion quality for instances of either of the latter models we combined the set of 201 can-
didates found from \( n = 30 \) GA evolution (merged at the end), which are highly fit for
the pure spare capacity problem with an additional 199 shortest-circumference cycles
from the set of all possible cycles. Empirically, this mixture works well because it con-
tains many good combinations for achieving minimum capacity and the raw material
to also address path length control and in addition produces run times under 15 min
(whereas these problems with the full set of candidate cycles cannot even be held in
memory). Results are summarized in Table 4 and Fig. 6.

The fourth ILP design model which minimizes the total length of optical paths in
bi-criterion weighting against capacity cost used \( a = 10^{-5} \), which was found to be suit-
able to effect path length control as a combined objective without upsetting the main
objective value of minimum spare capacity. The second data column in Table 4 sum-
maries statistics on the resultant path lengths in this case. We see that the average
path length is very slightly lower than in a comparative instance with the second ILP
and that the slight reduction in average (restored state) path length is apparently
obtained through some significant reduction in the shortest path(s) and a slight reduc-
tion in the longest path(s) as well. But the overall effect in terms of path shortening is
not great. The result tends to suggest that minimizing average path length at mini-
imum spare capacity with the detailed model 4 may not yield much improvement over
simply designing for minimum capacity and adjusting path lengths within those
structures. This could be true in general if path lengths are simply strongly dictated
or dominated by properties of any minimum capacity design instance.

In interpreting the results in Table 4, however, we need to keep in mind that there
can be many different equivalent minimum spare capacity designs and only one was
used here (essentially at random) for the corresponding model 2 design result in col-
umn 1. There could be many other model 2 outcomes, starting with different initial
minimum-capacity designs that do not do as well as the fourth model does here. As a
check on this, we ran one extra model 1 solution, which is also at minimum capacity,
and followed it by the application of model 2, with the result being an average of
3935 km and a significantly greater maximum path length of 8035 km. Therefore, one
can argue that a benefit of the fourth model is that it at least assuredly will always
produce a solution with average path length statistics, which are the best possible
under minimum spare capacity, whereas any given succession of a model 1 solution
followed by a model 2 accommodation does not have this guarantee. Additionally, it
should be kept in mind that the fourth model can also be used with high \( \alpha \) values,
which allow one to enter the domain of trading minimal addition spare capacity for
further reduced path length average statistics, and this is not an option with a combi-
nation of the first and second models.

But perhaps the most direct way to control the path length is to simply stipulate a
tolerable maximum and, of course, that brings us to the fifth ILP model. With it we
have means to simply assert a maximum optical path length limit in the restored
state and still achieve absolute minimum spare capacity, if that remains technically

Table 4. Statistics on Restored State Path Lengths Considering the GA
Preselection of 400 Candidate Cycles

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Model 2 (for reference from Table 2)</th>
<th>Model 4 (at minimum spare capacity)</th>
<th>Model 5 (4000 km limit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Redundancy (%)</td>
<td>62.43</td>
<td>62.43</td>
<td>74.16</td>
</tr>
<tr>
<td>Minimum (km)</td>
<td>780</td>
<td>590</td>
<td>680</td>
</tr>
<tr>
<td>Average (km)</td>
<td>3869</td>
<td>3866</td>
<td>2905</td>
</tr>
<tr>
<td>Maximum (km)</td>
<td>7705</td>
<td>7650</td>
<td>4000</td>
</tr>
<tr>
<td>Over length (&gt;4000 km)</td>
<td>142 (48%)</td>
<td>147 (50%)</td>
<td>0 (0%)</td>
</tr>
<tr>
<td>Running times</td>
<td>&lt;2 min</td>
<td>&lt;15 min</td>
<td>&lt;15 min</td>
</tr>
</tbody>
</table>

Fig. 6. Improvements on transparency versus capacity penalty.
feasible for the given length limit. Model 5 can, however, also be allowed to increase the spare capacity requirements over the absolute minimum spare capacity depending on what optical reach limit we assert. For instance, the third column of Table 4 now shows us that constraining optical path lengths under 4000 km does in fact require the design redundancy to increase to 74.16% as shown in Table 4. This is ∼19% extra spare capacity over what is required in the strictly minimum-cost network designs made with the second and the fourth ILP models. It is interesting to note, however, that with the addition of this ∼19% extra spare capacity, we can provide a design where every working and restored state path is under 4000 km. As a design tool this is perhaps the most general methodology for how to design a completely protected transparent island using, for example, 4000 km ULH technology.

Let us further consider the generalized trade-off of spare capacity for optical path length control in COST239, using the fifth model, if we are willing to trade capacity to achieve any given limit on maximum reach. This is the data portrayed in Fig. 6, in which the fifth ILP model has been solved repeatedly for several fixed length limits using in each case the set of 400 candidate cycles as above (i.e., 201 from GA, 199 shortest) and the resulting excess capacity to satisfy the design goal is plotted in comparison to the same ILP model 5 being solved for a set of the same overall size (400) smallest eligible cycles.

As expected, Fig. 6 shows that with suitably large optical reach limits we can again achieve absolute minimum spare capacity design. It is also to be expected that the problem becomes infeasible when we try to assert very low reach limits. In Fig. 6 the minimal tolerable path length limit at which the ILP is feasible was found to be 3035 km for both sets of preselected candidate cycles. We also note that the capacity requirements are nearly the same under short reach limits for the two sets of candidate cycles, but the combined GA–shortest cycles set performs much better as the reach limits increase (typically over 4250 km). This is explainable by appreciating that with the GA contribution to the candidate set there are many more efficient prospects present, such as Hamiltonian cycles, but which also evidently can be employed with suitable regard to path length issues. In contrast the set of even twice as many shortest cycle candidates cannot match the protection capacity efficiency.

Comparing maximum optical path lengths resulting from the second and third ILP models (the first and second columns of Table 2) with results shown in Fig. 6, one can conclude that allowing the model to change its selection of p-cycles to accommodate fixed path length limits can achieve the same redundancy as minimum cost design, but with a lower maximum path length. In the results here the shortest reach limit at which strictly minimal spare capacity could still be realized (when considering the GA set of candidate cycles) was 6250 km in Fig. 6, which is comparable with the 7705 km or 6745 longest paths under the second ILP and the minimax case (model 3), respectively, in Table 2, which also retains minimal spare capacity.

5. Concluding Discussion

We have studied several variants on the idea of systematically matching longer working paths with shorter protection path-segments found in p-cycles chosen for the network design. The motivation is considerations of optical transparency limits in p-cycle protected networks. Specifically, we have introduced ILP models to perform the above assignment in an existing conventional p-cycle minimum capacity design model, to minimize both capacity requirements and optical path lengths in a one-step optimization ILP design model, and to constrain when desired optical paths under certain absolute length limits. This matching of longer paths with shorter path-segments through available cycles is the first attempt to control by design the length of end-to-end optical paths in the restored state. An ILP-based GA was proposed as well, to deal with the problem of the computational complexity for ILP instances involving all possible candidate cycles. The GA approach filters candidate cycles to reduce the number of variables in the three more advanced p-cycle ILP network design models. It should be possible to further develop and combine the GA preselection method in this study with any p-cycle planning proposal in which enumeration of all candidates is not possible or practical.

In closing, let us consider some practical implications of the work. One question is, if we enhance the off-line design problem in the way outlined, does this complicate the
on-line operation of the network? We think not because once the preplanning information is generated (at design time) it can be simply stored in each node as a predecided policy for which path-segments to associate with which working channels in the event of failure. The wisdom of associated longer working paths with shorter segments in locally accessible p-cycles is thus implicit and automatic, once embedded in the network, and is not an operational complexity in real time. Moreover, the associations between working paths and available p-cycle segments could also be developed adaptively in the network itself. As long as nodes have information on the length of each working path transiting them and know the lengths of p-cycles available at their site (in the left and right senses as above), then it is a simple in situ problem for each node to work out the best local mapping of failed channels to affected paths ahead of any actual failure. Such information could be disseminated through optical supervisory channel– (OSC–) borne signaling before failures, generalized multi-protocol label switching (GMPLS) messaging, or by means of signal overheads themselves.

Another consideration given the variations that are possible with the ILPs is to ask which is preferable—to have a lower average path length or the shortest minimax path length? Or, still further, to simply have all paths below a set limit, even if the latter may require extra capacity? In practice, we think the choice would depend on the network circumstances. If many of the path lengths were already over the distance requiring regeneration, then minimizing the average path length corresponds to the least total consumption of channel kilometer and regenerator costs. On the other hand, if one was already close to having every path below a transparent limit to start with in the restored state, then seeking to squeeze the longest path to its minimum could enable an entirely transparent solution, although this is a solution in which, possibly, numerous other paths are individually longer than they would have to have been otherwise. But the work also showed that one can just assert an upper length limit on all paths based on technological capabilities and solve for the p-cycles and path assignments and/or added capacity to satisfy that requirement. This last option (model 5) seems to be the most general and straightforward engineering approach. Although it may require some extra spare capacity, it would be the most direct method to design an entire transparent island based on a given ULH technology capability, such as, say, a 4000 km reach. In this case an entire transparent island can be defined that fully routes and protects all demands inside itself in a fully transparent manner.

A possible line of future work is the extension of models that, in this paper, are formulated under the assumption of a single working route for serving each demand-pair, to initial routings allowing the usage of multiple length-equivalent shortest working routes per demand. The extension could also cover joint (i.e., both working and spare) capacity placement. However, this adds still further dimensionality to represent multiple eligible working routes per demand-pair and would increase the problem complexity even more. In this case, the GA-ILP preselection strategy may be a key enabler to obtain solutions in such further work, especially in highly connected networks. Indeed, we plan continued research on the GA-ILP type of approach to either very large or very detailed p-cycle-related design problems. This may involve simplifications or relaxations of the GA-level ILP version used for fitness evaluation and/or use of multiple sub-populations combined for the final ILP to address various desired attributes in multi-objective problems. The GA approach may be especially practical in use on problems of a scale where the candidate cycle set cannot even be fully enumerated.

Acknowledgments
The authors thank Aden Grue, from the TRLabs Network Systems Group, for proof-reading and discussions. The anonymous reviewers also provided numerous thoughtful inputs.

References