Joint Optimization of Working and p-cycle Protection Capacity

Thomas Stidsen[*] Tommy Thomadsen[†]

Informatics and Mathematical Modelling
Technical University of Denmark
DK-2800 Kongens Lyngby, Denmark

May 21, 2004

Abstract

Recently, p-cycles were suggested as a way of protecting communication networks from link failures. p-cycle protection is considered a fast protection method similar to rings and 1+1 protection, but achieves capacity efficiency comparable to meshed protection methods.

We consider the problem of jointly routing communication channels and protecting links using p-cycles. An integer linear programming model is presented and a column generation algorithm for solving the problem is implemented. This algorithm enables an experimental study of the efficiency of p-cycles. The results show, that p-cycles are significantly more efficient than rings and comparable to any meshed protection method. The results also show that substantial savings can be achieved when routing and protection is performed jointly as compared to when it is not.

Based on the integer linear programming model, we discuss how protection costs can be taken into account in routing methods. We also we discuss an alternative efficiency measure of the p-cycles, which takes into account the interplay with existing p-cycles.

Keywords: p-cycles, protection, capacity planning, Rings, Column-Generation.

[*]Email: tks@imm.dtu.dk
[†]Email: tt@imm.dtu.dk

1 Introduction

Reliable communication networks are important in the society today because of the increasing dependency on communication. To increase reliability protection methods may be applied, i.e. methods to reroute the communication in case of failure. For this reason, there has been a widespread research in different protection methods and many different types have been suggested. In general there is a tradeoff between achieving a fast fault recovery and minimizing the capacity usage of the links in the network. Historically, the requirement that communication has to be restored in less than 50 ms has divided the different methods into two groups, the fast protection methods which can recover from failures in 50 ms and the capacity efficient methods which cannot. Some examples of protection methods are:

- **Fast protection methods:** Pre-configured methods, e.g. 1+1 protection and ring protection.
- **Capacity efficient protection methods:** Meshed methods, e.g. Shared Backup Path Protection and complete rerouting.

Common for all fast protection methods are that they use pre-configured capacity to protect the network. This enables rerouting to be performed by end nodes of the failed link only. Since only end nodes of the failed link perform any rerouting, the protection is fast but the capacity efficiency is limited.

Recently a new fast protection method, called p-cycle protection (Pre-configured Protection Cycle), has been suggested [8]. It is claimed that the p-cycle protection method is also capacity efficient, leading to the claim that p-cycles provide “ring-like speed with mesh-like capacity”.

The main contribution in this article is the implementation of a column generation algorithm which is used to obtain close to optimal solutions (usually within 1% from optimum) for the *joint* routing and protection problem, where protection is performed using p-cycles.

The remainder of the article is organized as follows. In Section 2 the general problem of routing and p-cycle protection is described. Previous work on optimization of p-cycle protection is briefly reviewed in Section 3. In Section 4 the column generation algorithm for joint routing and p-cycle allocation is described. The developed algorithm is tested on 6 networks and in Section 5 the results are presented and discussed. Finally concluding remarks are given in Section 6.

2 The p-cycle Protection

Consider a given network consisting of a set of nodes and a set of bi-directional links. A set of connection demands specifies the number of communication channels which needs to be established between different nodes. The *routing* problem is then the problem of routing the channels along paths through the network such that all demands are fulfilled.

The established connections may be interrupted through link failures. The *protection* problem is to ensure that enough capacity on the remaining links exists such that the channels may be rerouted in case of a link failure. We denote the capacity used for routing the *working capacity* and the capacity used for protection the *protection capacity*. The *total capacity* is then the sum of the working capacity and the protection capacity for all links in the network. The joint routing and protection optimization problem is then to minimize the total capacity requirement, by choosing the best paths and the best protection methods. Many different routing and protection methods have been suggested, for a general introduction to routing and protection, see e.g. [6].

A p-cycle is a protection method which use a cycle of pre-configured capacity (i.e. capacity allocated prior to link failures) to protect links. The same amount of pre-configured capacity is required on all links of the cycle. The capacity is pre-configured such that in case of a link failure, the only nodes that need to do rerouting are the end nodes of the failed link. Thus no signaling is required. The p-cycle protect two types of links, *on-cycle* links, see Figure 1 and *straddling* links, see Figure 2. In the figures, the thick solid lines indicate the pre-configured capacity of the p-cycle. The failed links, in Figure 1 link EF and in Figure 2 link BF, are marked with a cross.

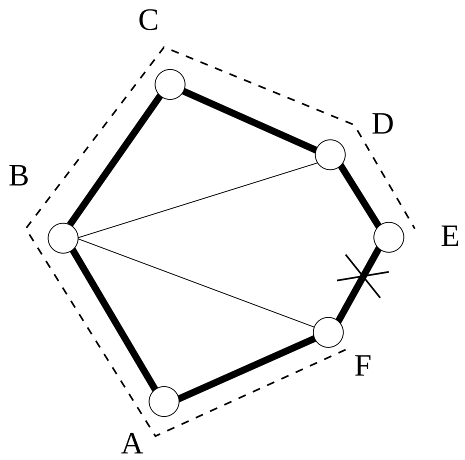

Figure 1: On-cycle link protection

On-cycle protection uses the fact that there is always one other way around the cycle, in case of a single link failure. In case a link on the cycle fails, the communication channels which use the failed link may be restored by routing them the reverse way around the cycle, illustrated with the dashed line from node E through D C B A to node F. Naturally the maximal number of channels which can be protected for the on-cycle links corresponds to the pre-configured capacity of the p-cycle. This is essentially standard ring protection.

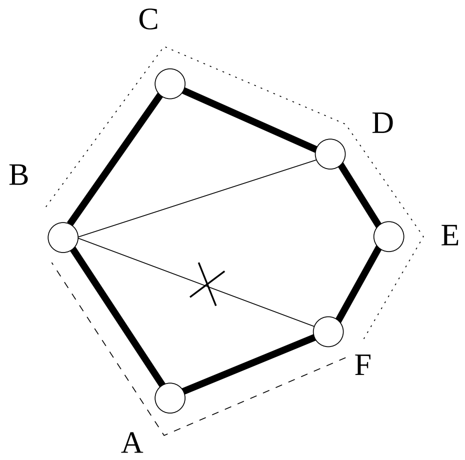

Figure 2: Straddling link protection

In Figure 2 it is shown how a link, which is a chord of the cycle, can be protected. This link is in the p-cycle articles called a *straddling* link and we will use this term. Because the end-nodes are on the cycle but the link is *not* on the cycle, the cycle has two routes between the end-nodes (B and F) of the failed link, illustrated with the dashed line and the dotted line in Figure 2. The p-cycle can hence protect twice the pre-configured capacity of the p-cycle. The straddling link protection is what differentiates p-cycles from standard rings. For a more comprehensive description of p-cycles we refer to chapter 10 in [6].

A link may be protected by several p-cycles, i.e. if a link fails, the channels using that link may be protected by rerouting them along several different p-cycles. Furthermore we are restricting the possible p-cycles to simple p-cycles where a node may only have two incident links on the cycle. In [6] cases of non-simple p-cycles are studied.

In this article we study the problem of jointly planning routing and p-cycle protection. Given a network and a connection demand, the total capacity of the network is minimized.

3 Previous Work on p-cycle Planning

p-cycles were first suggested in [8] and it was claimed that p-cycles provide "ring-like speed with mesh-like capacity". Since then, a number of articles have been published regarding different aspects of p-cycles. In this section we briefly review those which are most relevant in connection with the joint routing and p-cycle protection planning problem considered here.

In [12] theoretical arguments are given for the efficiency of p-cycles. Bounding-type arguments are given for the claim that p-cycles are the most efficient type of pre-configured (capacity) pattern. The argumentation is consistent, but is based on fully connected networks, which seems far from the rather sparse telecommunication networks.

In [8] a Mixed Integer Program (MIP) model for planning p-cycle protection is given. A prerouted network is assumed, i.e. the protection capacity requirement is minimized given the working capacity. The main problem with this approach is that it requires enumeration of all possible p-cycles. Because the number of p-cycles grows exponentially, only networks of moderate size may be solved to optimality. By pre-selecting "promising" p-cycles the size of the networks which can be handled may be increased albeit sacrificing the optimality guarantee, see also [4, 15]. In [4] two measures for evaluating p-cycles are suggested. In Section 4.4.2 we study these measures in more detail.

The problem of *joint* routing and p-cycle protection is studied in [7, 10]. In [7] a number of paths and p-cycles are pre-selected, making optimization of networks of medium size possible, again sacrificing the optimality guarantee. [10] apply column generation to implicitly represent all paths and p-cycles. The generation of p-cycles is not guaranteed to generate the best p-cycles, hence optimality is not guaranteed, see section 4.4.1.

A different approach is taken in [11]. Here a complex MIP model, which does *not* require enumeartion of all possible p-cycles, is formulated. The number of binary variables of the formulation is $O(|N| \cdot |L| \cdot |C|)$, where $|C|$ is the number of p-cycles which are actually used. While this is certainly an improvement compared to an exponential number variables in the MIP formulation from [8], the size of the formulation still grows significantly making optimal solution methods difficult for networks of medium size. Instead an elaborate method for stepwise optimization of gradually refined models is suggested. This enables heuristic optimization of networks with up to 25 nodes.

4 Solution Methodology

As discussed in Section 3, the MIP model suggested in [8] requires enumeration of all possible p-cycles to achieve an optimal solution. In this section, we describe how the use of a column generation algorithm allows us to solve a relaxation of the MIP model through *implicit* enumeration of the p-cycles. This enables solution of the LP-relaxed MIP model to optimality generating only a fraction of the possible

p-cycles.

In the following sections we will in Section 4.1 describe the MIP model for joint routing and p-cycle protection. In Section 4.2 the column generation algorithm which is needed to solve the relaxed MIP model is described. The column generation algorithm requires the solution of two sub-problems: The path generation problem, described in Section 4.3, and the p-cycle generation problem, described in Section 4.4. Finally, in Section 4.5 we describe how to use the generated paths and p-cycles to find near optimal solutions to the original MIP model.

4.1 The Joint Routing and Protection Planning Problem

Consider a network $G(N, L)$, a set of connection demands D, indexed by $kl \in D$, between the nodes k and l requiring d_{kl} communication channels, a set of paths P, indexed by $p \in P_{kl}$, for each demand pair kl and a set of p-cycles R, indexed by $r \in R$. Let c_{ij} be the unit cost of allocating one unit of capacity on link $ij \in L$. The unit cost of a path $c_p^{kl} = \sum_{ij \in P_{kl}} c_{ij}$ is calculated as the sum of the unit costs for each link in the path, and the unit cost of a p-cycle $c_r = \sum_{ij \in r} c_{ij}$ is the sum of the unit cost of the links of the cycle. The constants $PATH_{p,ij}^{kl} \in \{0, 1\}$ with value 1 if path $p \in P$ for demand $kl \in D$ use link $ij \in L$ and 0 otherwise defines the paths and the constants $PCYC_{r,ij} \in \{0, 1, 2\}$ with value 1 if link ij of p-cycle $r \in R$ is on-cycle, 2 if link ij is straddling and 0 otherwise defines the protection offered by the p-cycle. The variables $v_p^{kl} \in Z^+$ define the number of channels of demand $kl \in D$ on path $p \in P_{kl}$ and the variables $u_r \in Z^+$, define the pre-configured capacity of p-cycle $r \in R$. Then a MIP model for the Joint Routing and p-cycle Protection problem, henceforth called the **JRPP** model can be formulated:

minimize:

$$\overbrace{\sum_{r \in R} c_r \cdot u_r}^{protection\ cost} + \overbrace{\sum_{kl \in D} \sum_{p \in P_{kl}} c_p^{kl} \cdot v_p^{kl}}^{routing\ cost} \tag{1}$$

subject to:

$$(\xi_{kl}) \qquad \sum_{p \in P_{kl}} v_p^{kl} \geq d_{kl} \quad \forall\ kl \in D \tag{2}$$

$$(\pi_{ij}) \qquad \sum_{r \in R} PCYC_{r,ij} \cdot u_r - \sum_{kl \in D} \sum_{p \in P_{kl}} PATH_{p,ij}^{kl} \cdot v_p^{kl} \geq 0 \quad \forall\ ij \in L \tag{3}$$

$$v_p^{kl} \in Z^+ \qquad \forall kl \in D,\ p \in P_{kl} \tag{4}$$

$$u_r \in Z^+ \qquad \forall r \in R \tag{5}$$

The objective function (1) calculates the combined routing and protection cost. The constraints (2) ensure that all demands are satisfied by routing the required channels along one or more of the available paths. The constraints (3) ensure that each link is protected against failure by allocation of enough protection capacity along p-cycles which offers protection to the link. Notice the difference between on-cycle link protection and straddling link protection is included in the $PCYC_{r,ij}$ constant. The dual variables of constraints (2) are ξ_{kl} and the dual variables of constraints (3) are π_{ij}.

The JRPP model is a generalization of the MIP model suggested in [8], which arises when P_{kl} contain exactly one path, the shortest, for each demand kl. The same model as the above is used in [7, 10]. The main problem with the JRPP model is that the number of paths and p-cycles grows exponentially with the number of nodes (and links) in the network. In order to ensure optimality, all paths and p-cycles must be considered explicitly or implicitly. To avoid explicit representation of paths and p-cycles, column generation is applied. The Relaxed JRPP model (**R-JRPP**) is created by relaxing the integer domain constraints (4) and (5) of the variables v_p^{kl} and u_r, i.e. $v_p^{kl}, u_r \in R^+$. The R-JRPP model is an LP model, which can then be solved using column generation where the paths and p-cycles are taken into account implicitly. The column generation algorithm solves a R-JRPP model of reduced size where only a small subset of paths P and p-cycles R are included. We denote this the **R-JRPP(P,R)** model. Paths and p-cycles are then generated when needed.

4.2 Column Generation Algorithm

The idea of a column generation algorithm is to only generate the variables when needed, i.e. when the *reduced cost* of a variable is negative. For each iteration of the column generation algorithm the paths (one for each demand) with the minimal reduced cost is found and the p-cycle with the minimal reduced cost is found. If the reduced cost of a path or a p-cycle is negative, they are called *improving*. If no improving paths or p-cycles are found, the algorithm terminates and the R-JRPP model has been solved to optimality using only a subset of possible paths and p-cycles. The column generation algorithm is given in pseudo-code in Figure 3.

Initially the column generation algorithm is started with a set of shortest paths, one for each demand, and a set of *dummy* p-cycles one for each link ij. A dummy p-cycle is a (non-existent) p-cycle which has the ability of protecting just one link and which is so expensive that it will never show up in the optimal solution. Then the R-JRPP(P,R) model is solved based on the current set of paths P and the current set of p-cycles R. Based on the dual variables (prices) from equation (2) (ξ_{kl}) and equation (3) (π_{ij}), improving paths and p-cycles are found. This process continues until no improving paths or p-cycles are found.

P = Shortest path for each demand nodepair kl
R = one dummy p-cycle for each link ij
do
  Solve the R-JRPP(P,R) problem
  Solve routing subproblems searching for improving paths
  if improving paths found **then**
    Add improving paths to P
  Solve p-cycle subproblem searching for an improving p-cycle
  if improving p-cycle found **then**
    Add improving p-cycle to R
while improving path or improving p-cycle is found

Figure 3: *The joint routing and p-cycle protection column generation algorithm*

4.3 Subproblem I: Path Generation

The path generation problem is the problem of generating (improving) paths with negative reduced cost. The reduced cost of a variable $\widehat{c_{p,kl}}$ can be calculated based on the dual variables ξ_{kl} and π_{ij} from equation (2) and equation (3) in the R-JRPP(P,R) model and the link cost c_{ij} as follows.

$$\widehat{c_{p,kl}} = \sum_{ij \in p} c_{ij} - \xi_{kl} + \sum_{ij \in p} \pi_{ij} \tag{6}$$

Each reduced cost contains three terms, the sum of the link costs c_{ij}, a reward term ξ_{kl} for providing an additional path to route the demand kl and a sum of the link protection costs π_{ij}. ξ_{kl} appear in the reduced cost for all kl-paths. Therefore the path with the lowest reduced cost for a demand kl can be found as the shortest path in a network with link costs.

$$\overline{c_{ij}} = c_{ij} + \pi_{ij} \tag{7}$$

By duality $\pi_{ij} \geq 0$ and by assumption $c_{ij} \geq 0$, this means that $\overline{c_{ij}} \geq 0$. Thus we can apply the Floyd-Warshall algorithm [3] and the shortest paths for all demands kl can be calculated in $O(|N|^3)$. For all nodepairs kl, if a path exists with $\widehat{c_{p,kl}} < 0$ it is an improving path and it is included into the set of paths P in the R-JRPP(P,R) model.

When the algorithm terminates the price $\widehat{c_{ij}}$ is the price for using that link, including both routing costs and protection costs. Often joint routing and p-cycle protection is un-realistic because, as is argued in [11], introduction of new p-cycles in a network is a strategic decision, whereas routing is an operational decision. We suggest that after the strategic choice of p-cycles, based on a *forecast* of the demand, routing is performed as shortest path routing based on $\overline{c_{ij}}$ prices. This is not optimal because new p-cycles might be needed, but an estimation of the protection costs is utilized in the routing.

4.4 Subproblem II: p-cycle Generation

The second subproblem is the p-cycle generation problem, i.e. the problem of generating p-cycles with negative reduced costs. The reduced costs of the p-cycles depends only on the π_{ij} dual values. The reduced costs may hence be calculated as given below.

$$\widehat{c_r} = \sum_{ij \in r} c_{ij} - \sum_{ij \ straddling \ r} 2 \cdot \pi_{ij} - \sum_{ij \in r} \pi_{ij} = \sum_{ij \in r} c_{ij} - \sum_{ij \ straddling \ r \ or \ ij \in r} 2 \cdot \pi_{ij} + \sum_{ij \in r} \pi_{ij} \tag{8}$$

The last equality sign follows immediately by including both straddling and on-cycle links into the second sum and afterwards correcting by adding the third sum.

The p-cycle generation problem is an NP-hard optimization problem [10, 13] which we have previously termed the Quadratic Selective Travelling Salesman problem. This problem is described in detail in [13] and we refer to this article for an in-depth treatment. In [9] a polyhedral study of the problem is carried out.

The following MIP model of the P-Cycle Generation problem, henceforth called the **PCG** model, use three types of variables: The variables $y_i \in \{0, 1\}$ represents the nodes which are part of the p-cycle, 1 for being included 0 otherwise. The variables $x_{ij} \in \{0, 1\}$ for $ij \in L$ represents the links where the protection capacity is pre-configured, 1 for being included 0 otherwise. Finally the variables z_{ij} represents all nodepairs $ij \in L$ which are included in the p-cycle, 1 for the nodepair being included 0 otherwise.

minimize:

$$\sum_{ij \in L} (c_{ij} + \pi_{ij}) x_{ij} - \sum_{ij \in L} 2\pi_{ij} z_{ij} \tag{9}$$

subject to:

$$\sum_{j \in V} x_{ij} = 2y_i \qquad \forall\, i \in N \tag{10}$$

$$z_{ij} \leq y_i \qquad \forall\, ij \in L \tag{11}$$

$$z_{ij} \leq y_j \qquad \forall\, ij \in L \tag{12}$$

$$z_{ij} \geq y_i + y_j - 1 \qquad \forall\, ij \in L \tag{13}$$

$$\sum_{i \in S, j \notin S, ij \in L} x_{ij} \geq 2(y_k + y_l - 1)$$

$$\forall\, S \subset N, 3 \leq |S| \leq |N| - 3\ , k \in S, l \notin S \tag{14}$$

$$x_{ij}, y_i, z_{ij} \in \{0, 1\} \tag{15}$$

The objective equation (9) calculates the reduced cost as described above in equation (8). All nodes which are in the p-cycle are required to have to 2 incident links, which is ensured by equation (10). Therefore only non-simple p-cycles are feasible. For each $ij \in L$ where i and j are included in a p-cycle, i.e. $y_i = 1$ and $y_j = 1$, the variable $z_{ij} = 1$. This is ensured by equation (11), (12) and (13). Sub-tour elimination constraints are added in order to ensure that connected rings are constructed (14). Finally the domain constraints (15) ensure integer variables.

The PCG model is solved using the branch-and-cut algorithm described in [13]. Solving the PCG model is *the* bottleneck of the column generation algorithm. This is validated by computational tests, see Table 2 in Section 5.1. To speed up the column generation algorithm, heuristic generation of improving p-cycles could be applied. Inspiration for this could be sought in algorithms for the TSP problem and for the quadratic knapsack problem. Ultimately however, to ensure optimality of the column generation algorithm, guarantee of non-negative reduced costs is required, i.e. the optimal solution of the PCG model is required.

4.4.1 Reduced cost of rings

For comparison we modify the algorithm to deal with ring protection. We use the same column generation algorithm, with almost the same JRPP model. The only difference is the removal of straddling protection from the $PCYC_{r,ij}$ constant (i.e. $PCYC_{r,ij} = 1$ if link ij of p-cycle $r \in R$ is on-cycle and 0 otherwise). The path generation problem, see Section 4.3, remains the same, but the MIP model for the ring generation problem is slightly different from the PCG model. The objective is to find the ring with the most negative reduced cost, hence equation (9) is changed to equation (16) below, where the reward for the straddling links has been removed.

$$\widehat{c_r^{ring}} = \sum_{ij \in L} (c_{ij} - \pi_{ij}) x_{ij} \tag{16}$$

Only the objective function is changed to find improving rings instead of improving p-cycles. However, at the same time the z_{ij} variables and the constraints in equation (11), (12) and (13) become obsolete. This indicates that ring generation is easier than p-cycle generation, and in fact rings can be generated in polynomial time. Applying the Bellman-Ford algorithm [3], rings with negative reduced costs, negative cycles, can be found in $O(|L| \cdot |N|^2)$ time.

The Bellman-Ford algorithm can also be applied to heuristically generate p-cycles, which is done in [10]. Since the Bellman-Ford algorithm is a polynomial algorithm, a significant speed improvement compared to the branch-and-cut algorithm we apply, can be achieved. However, the Bellman-Ford algorithm does not take straddling protection into account. In [10], the reduced costs of p-cycles are estimated based on the reduced costs of rings which are found using the Bellman-Ford algorithm. Afterwards contribution to the reduced costs from the straddling links are added. These will always improve the reduced costs because $\pi_{ij} \geq 0$. Based on the fact that on average long p-cycles contain more straddling links than short p-cycles and hence on average are more efficient, the network on which the Bellman-Ford algorithm is applied is modified. The costs c_{ij} of all the weights in the network is reduced by a fixed amount, before applying the Bellman-Ford algorithm again. This adjustment makes long p-cycles more attractive to the Bellman-Ford algorithm. The problem with this approach is that it is only on average that long p-cycles are more efficient than short p-cycles, i.e. a short p-cycle may be the p-cycle with the most negative reduced cost. Hence optimal solution of the relaxed R-JRPP model is not guaranteed in [10]. As noted in [10] and as we demonstrate in Section 5.4, straddling link protection is an essential feature of p-cycles and thus important to take into account when generating p-cycles.

4.4.2 p-cycle Efficiency

As mentioned in Section 3 a way to reduce the problem of the large number of possible p-cycles is to pre-select a fraction of promising p-cycles. In [4] two different measures of the p-cycles efficiency for p-cycle pre-selection is suggested: "A Priori p-cycle Efficiency" $AE(r)$, see equation (17) and "Demand-weighted p-cycle Efficiency" $EW(r)$, see equation (18).

$$AE(r) = \frac{\sum_{ij} PCYC_{r,ij}}{\sum_{ij \in r} c_{ij}} \tag{17}$$

$$EW(r) = \frac{\sum_{ij} CAP_{ij} \cdot PCYC_{r,ij}}{\sum_{ij \in r} c_{ij}} \tag{18}$$

The efficiency measure $AE(r)$ counts the number of protected links, divided by the cost of the p-cycle. In $EW(r)$ the offered protection capacity is weighted with the working capacity which needs to be protected for each link, CAP_{ij}. Hence this measure assumes that the demands are already routed.

To compare $AE(r)$ and $EW(r)$ measures with the reduced cost ($\widehat{c_r}$) from equation (8), the reduced costs of the p-cycles is divided by the cost $\sum_{ij \in r} c_{ij}$ (assuming $\sum_{ij \in r} c_{ij} > 0$) of the p-cycle and equation (19) is obtained.

$$\widehat{c_r'} = 1 - \frac{\sum_{ij \in r} \pi_{ij} \cdot PCYC_{r,ij}}{\sum_{ij \in r} c_{ij}} \tag{19}$$

Given a p-cycle r, the sign of $\widehat{c_r'}$ is the same as $\widehat{c_r}$, because we assume $c_{ij} \geq 0$, i.e. $\widehat{c_r} < 0 \Rightarrow \widehat{c_r'} < 0$. However, the division may have changed the order of the p-cycles with negative reduced costs, hence the best p-cycle according to equation (8) is not necessarily the best p-cycle according to equation (19). The division effectively makes the shorter p-cycles more attractive. If we ignore the constant term, change the sign of the fraction we obtain a new measure which should be maximized.

$$\widehat{c_r''} = \frac{\sum_{ij} \pi_{ij} \cdot PCYC_{r,ij}}{\sum_{ij \in r} c_{ij}} \tag{20}$$

It is interesting to compare the optimal p-cycles according to the three different measures: $AE(r)$ (equation (17)), $EW(r)$ (equation (18)) and $\widehat{c_r''}$ (equation (20)). The optimal p-cycle according to the $AE(r)$ measure, is the p-cycle with the lowest average cost for link protection. The main problem is that it does not take into account the actual need for protection, i.e. the working capacity which needs to be protected. The $EW(r)$ measure weights the importance the protection of the links according to the working capacity CAP_{ij} of each link. The main problem with the $EW(r)$ measure is that it does not take the interplay of the different p-cycles into account, i.e. a link may not be very interesting to protect, even though CAP_{ij} is high, because the link might already be cheaply covered by other efficient p-cycles. Given a network, a set of existing p-cycles and a demand, we conjecture that the measures defined in equation (8) and equation (20) are the best measures of future p-cycles to include into the network.

4.5 Getting Integer Solutions

The column generation algorithm obtains an optimal solution to the R-JRPP model, but it is *not* guaranteed to return an integer solution, i.e. the optimal solution to the JRPP model. In this article we have chosen the simple solution of solving the JRPP model using a standard MIP solver with the paths P and p-cycles R collected during the column generation algorithm. Hence it is really the JRPP(P,R) model which is solved and it is important to acknowledge that the MIP solver only returns the optimal solution given the available paths and p-cycles and *not* the optimal integer solution to the full JRPP model. But because we have an optimal lower bound from the column generation algorithm, the solution to the R-JRPP model, we can quantify the worst case optimality gap. As the results clearly illustrates in Section 5.3 this approach is fully sufficient to achieve close to optimal performance for all the networks tested. In order to get the real optimal solution a branch-and-price algorithm is needed [1, 14].

5 Results and Discussion

Our column generation algorithm and the integer heuristic is tested on 6 networks, see Table 1. The objective of the tests and discussions in this section is twofold: To examine the efficiency of the column generation algorithm and to compare the capabilities of p-cycles with ring protection and the meshed protection lower bound.

	Nodes	Links	Avg. Node. degree	Working capacity	Meshed Protection Lower Bound
Cost239[2]	11	26	4.73	86	11.6
Europe	13	21	3.23	158	90.0
USA[4]	28	45	3.21	1273	641.2
Italy[5]	33	68	4.12	1718	-
France[4]	43	71	3.3	3473	-
France 2[4]	43	71	3.3	4043	-

Table 1: The tested networks

The columns in Table 1 contains (in order): The number of nodes, the number of links, the average node degree, the working capacity i.e. capacity after shortest path routing of all demands, and the meshed protection lower bound. Note that complete rerouting is a lower bound for any meshed protection method. Thus, the meshed protection lower bound can be found by performing complete rerouting for all single link failures. To calculate the optimal routing for complete rerouting, a column generation algorithm was implemented. Because the number of variables rise as $O(|N|^2|L||P_{kl}|)$, complete rerouting could only be performed for the three smallest networks.

For all networks, one bi-directional channel is requested for all nodepairs, except for the network France 2 where the same (sparse) demand pattern as in [4] was used. Otherwise, the networks France and France 2 are identical. In the tests we assume unit costs for the links, i.e. $c_{ij} = 1$, as have been done previously in [4].

Based on the test networks, in Section 5.1 results regarding computational efficiency of the algorithm are given. In Section 5.2 the protection capacity of p-cycles is compared with the protection capacity of rings, using shortest path routing and joint routing, and with the meshed protection lower bound. In Section 5.3 the integer solutions are compared with the bound. Finally in Section 5.4 the importance of straddling link protection offered by the p-cycle method is studied.

5.1 Computational Efficiency

In Table 2 data regarding the running time of the column generation algorithm (and the MIP solver) is given. For each network, the total running time, the percentage of the time spent on the master problem (including time spent on initialization and path generation), the percentage of the time spent on generating p-cycles, the percentage of the time spent on obtaining integer solutions, the number of p-cycles generated, the number of p-cycles used in the integer solutions and the time spent on generating one p-cycle on average. The CPLEX 9.0 solver is used both to solve the R-JRPP model in the column generation algorithm, in the branch-and-cut algorithm for the PCG model and to obtain the integer solutions of the JRPP(P,R) model. The MIP solver generates the integer solutions as described in Section 4.5, but if a provably optimal solution is not fund after 30 seconds, the MIP solver is terminated and the best feasible solution found is returned. The computer used was a 1200 MHz SUN Fire 3800 machine.

	Total Time (sec.)	JRPP Time (%)	PCG Time (%)	Integer Time (%)	#p-cycles Gen.	#p-cycles Used	Avg. PCG Time (sec.)
Cost239	0.4	25.0 %	75.0 %	0.0 %	8	3	0.04
Europe	1.0	10.0 %	90.0 %	0.0 %	10	5	0.09
USA	44.3	2.0 %	30.2 %	67.7 %	17	11	0.79
Italy	160.6	3.1 %	78.1 %	18.7 %	44	14	2.85
France	360.1	2.1 %	96.3 %	1.6 %	43	22	8.07
France 2	239.2	0.7 %	99.1 %	0.2 %	41	19	5.78

Table 2: Computational efficiency

The column generation algorithm terminates in less than 361 seconds for all the test networks. The main part of the running time is spent on generating p-cycles (and to some extend finding an integer solution). The number of generated p-cycles is low, always less than 50, eventhough the number of available p-cycles for e.g. the France network is at least 500000 [4]. Furthermore only about half of these are used in the integer solutions. The running time may be improved by pre-generating a number of p-cycles e.g. by using the pre-selection methods suggested in [4, 15].

5.2 Protection Capacity Efficiency

In this section we will compare the (integer) solutions for the JRPP model, the Shortest Path Routing p-cycle Protection **SPRPP** model, the Joint Routing and Ring

Protection **JRRP** model and the Shortest Path Routing Ring Protection **SPRRP** model. The shortest path protection models for p-cycle protection SPRPP and ring protection SPRRP only allows the demands to be satisfied using one path: The shortest. Hence the JRPP model is reduced to contain only the shortest path in the set of paths P_{kl} for each demand. The ring protection models are solved using the same column generation algorithm described in Section 4.2, with the modifications described in Section 4.4.1.

For each network in Table 3, the required working capacity is given in the first column and then the protection capacity and required extra capacity protection capacity (compared to working capacity) in % of the working capacity is given for the integer solutions for the four different models.

		p-cycle Protection				Ring Protection			
	Working	JRPP		SPRPP		JRRP		SPRRP	
	Abs.	Abs.	Rel.	Abs.	Rel.	Abs.	Rel.	Abs.	Rel.
Cost239	86	30	35%	37	43%	90	105%	94	109%
Europe	158	112	71%	147	93%	162	103%	182	115%
USA	1273	861	68%	1064	84%	1328	104%	1472	116%
Italy	1718	868	51%	1206	70%	1774	103%	1892	110%
France	3473	2255	65%	2904	84%	3732	107%	3954	114%
France 2	4043	3345	83%	3470	86%	4774	118%	4848	120%

Table 3: p-cycle and ring protection efficiency

From Table 3 it is clear that the joint routing and p-cycle protection method is the most efficient of the fast protection methods. This method requires between 35% and 83% extra capacity to protect the network. The corresponding ring protection method requires between 103% and 118% extra capacity. p-cycles are most capacity efficient for the networks with the highest node degree: Cost239 and Italy. The higher density of the networks enables better use of straddling links, which is shown in Section 5.4.

In [11] a number of good arguments against joint routing and protection are given. While we acknowledge these, we find it interesting that a saving of between 3% and 22% of the required protection capacity is possible for p-cycles. A possible explanation of the improved efficiency of joint routing and protection is offered in Section 5.4.

In a sense the comparison in Table 3 is not fair, because the percentages are given compared to no protection at all. In Table 4, the meshed protection lower bound is compared to p-cycle protection with and without joint routing. The first column contains the working capacity and the second column contains the meshed protection lower bound. Then follows the protection capacity, the extra capacity compared to the meshed protection lower bound and the extra capacity in percent of the working capacity for JRPP and SHRPP. These extra capacities are the worst case extra capacities necessary to obtain fast protection.

	Working	Meshed Protection	JRPP			SPRPP		
			Abs.	Extra	Extra %	Abs.	Extra	Extra %
Cost239	86	11.6	30	18.4	21%	37	25.4	30%
Europe	158	90.0	112	22.0	14%	147	57.0	36%
USA	1273	641.2	861	219.8	17%	1064	422.8	33%

Table 4: Meshed protection vs. p-cycle protection

5.3 Integer Solution Quality

Table 5 shows the gap between the lower bound found by the column generation algorithm and the integer solution found by the MIP solver.

	p-cycle Protection		Ring Protection	
	SPRPP	JRPP	SPRRP	JRRP
Cost239	4.65 %	2.33 %	2.50 %	5.45 %
Europe	0.59 %	0.00 %	0.99 %	0.75 %
USA	0.15 %	0.35 %	0.00 %	0.09 %
Italy	0.11 %	0.81 %	0.14 %	0.08 %
France	0.01 %	0.15 %	0.00 %	0.00 %
France 2	0.08 %	0.08 %	0.03 %	0.04 %

Table 5: Integer gap (%) to Column generation lower bound

As can be seen from Table 5 the solutions obtained are within 1% from optimum for all variants of the algorithms for all networks, except Cost239 where the integer solutions are up to 5.45% from the lower bound. Thus in general the algorithm produce close to optimal solutions.

5.4 Straddling Link Protection and Surplus Capacity

The difference between p-cycles and rings is in essence the possibility of protecting straddling links. In this section we investigate how much of the protection capacity is straddling protection compared to on-cycle protection. The p-cycles may be able to protect more working capacity in the link than is actually present. This is denoted *surplus capacity*. The surplus capacity for a link ij corresponds to the value of the left hand side of equation (3).

Table 6 compare the on-cycle protection capacity, the straddling protection capacity and the surplus capacity for JRPP and SRPP. All capacities are given in percent of the total protection capacity, i.e. on-cycle plus straddling protection capacity. It

	JRPP			SRPP		
	On-cycle	Straddling	Surplus	On-cycle	Straddling	Surplus
Cost239	29 %	71 %	17 %	31 %	69 %	29 %
Europe	52 %	48 %	25 %	52 %	48 %	44 %
USA	52 %	48 %	20 %	56 %	44 %	33 %
Italy	39 %	61 %	18 %	43 %	57 %	39 %
France	50 %	50 %	17 %	58 %	42 %	30 %
France 2	56 %	44 %	30 %	55 %	45 %	36 %

Table 6: Pre-configured capacity: On-cycle, straddling and surplus

can be seen in Table 6 that for the networks Cost239 and Italy more than 50% of pre-configured protection capacity is straddling protection. For the rest of the networks 40% to 50% is straddling protection. The higher amount of straddling capacity in the networks Cost239 and Italy is due to the higher density of these networks. This also explains the higher capacity efficiency of the p-cycle protection method for these networks in Table 3.

The effect of joint routing and protection only slightly increases the amount of straddling capacity. On the other hand, joint routing and protection significantly decreases the amount of surplus pre-configured protection capacity and this seems to be the main reason for the improved capacity efficiency of joint routing and protection.

6 Conclusion

In this article we have described an integer linear programming model for the problem of jointly routing and protecting a network using p-cycles. Based on the model, it is discussed how the protection cost can be taken into account in routing methods. Furthermore, a new measure of p-cycle efficiency is discussed, which takes the interplay of existing p-cycles into account.

A column generation algorithm is implemented to solve the problem. This enables an experimental study of the efficiency of p-cycles. Lower bounds for networks with up to 43 nodes and 71 links are found in 361 sec. Based on the columns found, integer solutions are found which are at most 5.45% above the lower bound and for most networks less than 1% above the lower bound. Analyzing the integer solutions, we observe that straddling protection delivers 42% to 71% of the offered protection capacity, most in the more dense networks where the p-cycle protection is most efficient. The capability of straddling protection is hence a valuable addition to standard ring protection. Joint routing reduce the required protection capacity with between 3% and 22%. The main reason for the improvement is the reduced surplus protection capacity.

The algorithm is also used to compare p-cycles with rings and a lower bound for meshed protection methods. The gap between the joint routing and protection for p-cycles and the meshed protection lower bound is between 14% and 21%. The capacity efficiency of joint routing and protection using p-cycles is thus closer to meshed networks than rings, which is in agreement with the claim "*ring-like speed with mesh-like capacity*" [8].

References

[1] C. Barnhart, E.L. Johnson, G.L. Nemhauser, M.W.P. Savelsbergh, and P.H. Vance. Branch-and-price: column generation for solving huge integer programs. *Operations Research*, 46(3):316–29, 1998.

[2] Peter Batchelor, Benedetto Daino, Peter Heinzmann, Dag R. Hjelme, Peter Leuthold, Robert Inkret, Giancarlo De Marchis, Hubert A. Jager, Francesco Matera, Michel Joindot, Branko Mikac, Anton Kuchar, Hans-Peter Nolting, Emile Le Coquil, Jan Spath, Franois Tillerot, Bart Van Caenegem, Nico Wauters, and Carl Weinert. Study on the implementation of optical transparent transport networks in the european environment-results of the research project cost 239. *Photonic Network Communication*, 2(1):15–32, 2000.

[3] Thomas H. Cormen, Leiserson Charles E., and Rivest Ronald L. *Introduction to Algorithms*. McGraw-Hill Book Company, 1998.

[4] J. Doucette, D. He, W. Grover, and Yang O. Algorithmic approaches for efficient enumeration of candidate p-cycles and capacitated p-cycle network design. In *Proceedings of the Fourth International Workshop on the Design of Reliable Communication Networks*, 2003.

[5] W. Grover, J. Doucette, M. Clouqueur, D. Leung, and D. Stamatelakis. New options and insights for survivable transport networks. *IEEE Communications Magazine*, 40(1):34–41, 2002.

[6] Wayne D. Grover. *Mesh-Based Survivable Networks: Options and Strategies for Optical, MPLS, SONET and ATM Networking*. Prentice Hall, 2003.

[7] W.D. Grover and J. Doucette. Advances in optical network design with p-cycles: joint optimization and pre-selection of candidate p-cycles. *All-Optical Networking: Existing and Emerging Architecture and Applications/Dynamic Enablers of Next-Generation Optical Communications Systems/Fast Optical Processing*

in Optical Transmission/VCSEL and Microcavity Lasers. 2002 IEEE/LEOS Summer Topical Meeting, pages WA2–49, 2002.

[8] W.D. Grover and D. Stamatelakis. Cycle-oriented distributed preconfiguration: ring-like speed with mesh-like capacity for self-planning network restoration. *Communications, 1998. ICC 98. Conference Record. 1998 IEEE International Conference on*, 1:537 –543 vol.1, 1998.

[9] V. Mak and T. Thomadsen. Polyhedral combinatorics of the cardinality constrained quadratic knapsack problem and the quadratic selective travelling salesman problem. *Submitted for publication*, 2004.

[10] D. Rajan and A. Atamturk. Survivable network design: routing of flows and slacks. *Telecommunications Network Design and Management. Sixth INFORMS Telecommunications Conference*, pages 65–81, 2003.

[11] D.A. Schupke. An ilp for optimal p-cycle selection without cycle enumeration. In *Eighth Working Conference on Optical Network Design and Modelling (ONDM)*, 2004.

[12] D. Stamatelakis and W.D. Grover. Theoretical underpinnings for the efficiency of restorable networks using preconfigured cycles (p-cycles). *Communications, IEEE Transactions on*, 48(8):1262 –1265, 2000.

[13] T. Thomadsen and T. Stidsen. A branch-and-cut algorithm for the quadratic selective tsp. *Submitted to Telecommunication Systems*, 2003.

[14] F. Vanderbeck and L.A. Wolsey. An exact algorithm for ip column generation. *Operations Research Letters*, 19:151–159, 1996.

[15] H. Zhang and O. Yang. Finding protection cycles in dwdm networks. In *Proceedings of the IEEE International Conference on Communications (ICC)*, volume 5. IEEE, 2002.