## ABSTRACT

JAIKUMAR, PRASHANT. Differential Capacity p-Cycles. (Under the direction of Associate Professor Rudra Dutta).

Survivability has become a central part of modern optical network design as the hundreds of wavelengths get multiplexed on fibers carrying data at Tbps speeds in DWDM networks. Provisioning for 100% restoration on failure using minimum amount of resources has become an important design problem. *p*-Cycles have emerged as a useful fault tolerance mechanism that operate at the speed of SONET rings, but also have low mesh-like spare capacity requirement.

In this thesis, a modified version of *p*-cycle, called differential capacity *p*-cycle, is proposed that improve spare capacity efficiency beyond what is provided by a set of traditional *p*-cycles. Different variants of differential capacity *p*-cycles are proposed, analogous to some of the traditional *p*-cycle variants. The designs of the various types of differential capacity *p*-cycles are formulated using integer linear programs, and the spare capacity usage of these new structures are compared with that of traditional *p*-cycles and their variants.

Differential Capacity p-Cycles

by
Prashant Jaikumar

A thesis submitted to the Graduate Faculty of
North Carolina State University
in partial fullfillment of the
requirements for the Degree of
Master of Science

Computer Science

Raleigh, North Carolina

2008

APPROVED BY:

Dr. David Thuente

Dr. Injong Rhee

Dr. Rudra Dutta
Chair of Advisory Committee

# DEDICATION

To my family

## BIOGRAPHY

Prashant Jaikumar was born in Chennai, India in 1985. He received his Bachelors of Engineering degree in Computer Science and Engineering from IIIT, Hyderabad, India, in 2006. Since August 2006, he has been a student of the computer science Department in North Carolina State University at Raleigh, North Carolina, USA.

## ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my advisor, Dr. Rudra Dutta, for his constant guidance and support. His insightful comments and research ideas were instrumental in shaping my thesis.

I would like to acknowledge Dr. Rhee and Dr. Thuente for agreeing to be on my thesis committee and for their valuable advice and recommendations.

Further, I would like to thank the graduate school and the Department of Computer Science for providing me with the opportunity to pursue a Masters degree in Computer Science.

I would also like to thank my friends in Raleigh for all the fun of the last couple of years.

# TABLE OF CONTENTS

## LIST OF FIGURES

# LIST OF TABLES

# Chapter 1

# Introduction

## 1.1 Motivation

Survivability is an important area of research in the optical networking domain. With recent advances in optics, hundreds of wavelengths can be packed in a fiber, resulting in bandwidth of the order of terabits per second. The growing usage of online video streaming and other high bandwidth applications had fueled the demand for these high bandwidth pipes. As the number of internet users and their bandwidth requirements grow exponentially, the amount of traffic is expected to keep increasing.

As the traffic carried by optical increases, it becomes very important to ensure that effective backup strategies are provisioned in case of failures. A downtime of only a few seconds can result in a loss of terabytes of data. So it is imperative that effective fault tolerance mechanisms are present. The amount of spare capacity used for protection is another crucial factor. As optic fibers as expensive, it is desirable to minimize the redundancy in the network.

Failures may be of two types: node failures and link failures. We can protect against node failures using node redundancy. At each switching station, backup nodes are installed that take over as soon as a fault in the primary is detected.

Protection against link failures, however, is not as straightforward. The most common cause of link failure failures are fiber cuts. Fibers buried underground can get cut by backhoes. Submarine cables sometimes get cut by ship anchors. Natural disasters also account for some fiber cuts. Link redundancy is not a viable unless the backup link is routed over a physically diverse route. This is because, if the primary and backup links are part of the same fiber bundle, a backhoe thats uproots the cable would destroy both the primary and backup links.

Also, the number of links is much larger than the number of nodes, so a 100% redundancy-based solution is often not feasible from a cost point of view. Ease of configuration is also an important design consideration. SONET rings require 100% capacity redundancy, but they offer the benefit of preconfiguration and fast restoration times of less than 50 ms. Other schemes likes oriented cycle double covers, enhanced rings and generalized loopback recovery have been proposed, but all of these require high redundancy [3]. On the other hand, mesh networks have a much lower redundancy of around x%, but the restoration time is much higher - of the order of hundreds of milliseconds. The protection actions are not preconfigured. The backup paths are created from the remaining spare capacity in the network after the failure occurs.

$p$-Cycles are a relatively new survivability mechanism that provide fast restoration speed, while also having high spare capacity efficiency. The concept of $p$-cycles has spawned a vast body of related work, applying the $p$-cycle concept to different aspects of survivability research.

## 1.2 Contributions

This thesis focusses on modifying $p$-cycles to increase their capacity efficiency. The concept of differential capacity $p$-cycles in introduced. Some variants of the differential capacity $p$-cycle are also proposed. ILP formulations for the different types of differential capacity $p$-cycles are provided. The difference between regular and differential capacity $p$-cycles are discussed from a theoretical perspective. The utility of the new types of $p$-cycles are validated using data from running the ILPs on random inputs.

## 1.3 Structure of Thesis

The next chapter gives an overview of *p*-cycles and its variants. The application of *p*-cycles to fault tolerance in different types of networks and traffic demands is discussed. The third chapter introduces the concept of differential capacity *p*-cycles and some of their variants. ILP formulations for these new types of *p*-cycles are provided. The implications of this new design are also discussed. Chapter four describes the results of performance evaluation of the spare capacity efficiency of new ILP models against the existing *p*-cycles variants. Heuristics for *p*-cycle preselection are also analyzed. Finally, chapter five concludes the thesis.

# Chapter 2

# Background

## 2.1 Introduction to survivability

Survivability is a foremost consideration in modern network design. Our dependence on communication network infrastructure keeps increasing as government services, business infrastructure, and other mission critical services like the power grid and transportation systems go online. This is a huge amount of data that is carried on optic fiber links across the country, and around the world. As the speed of optical links links increases, the importance of fast restoration of failures at the optical layer is of utmost importance.

The aim of the survivable design problem is to design a network in which all affected traffic flows are assigned a backup path when they are disrupted by a link failure. Most of the research has focussed on single link failures are optic fibers have very high reliability, and the probability of simultaneous failures is low. However, simultaneous failures do occur when groups of fiber links, called Shared Risk Link Groups (SRLG), which are physically routed over the same path are cut by the same agent, such as a back-hoe.

There are many different approaches to survivable network design. Protection and restoration are two ways of classifying survivability schemes. In protection, the backup paths are pre-determined before the failure occurs. Restoration strategies, on the other hand, try to find backup paths from the remaining spare capacity available in the system, after a failure is detected. The tradeoff between these two approaches is that protection is faster, leading to less loss of data, but restoration is more capacity efficient. Minimizing spare capacity usage is very important from a monetary point of view as fiber links are expensive. We will only consider protection based approaches in this thesis.

There are a couple of different approaches to protection: span protection and path protection. In span protection, the backup strategy is on a per link basis. That is, a protection mechanism is configured for every possible link failure. In path protection, the backup strategy is on a per flow basis. When a link breaks, all flows that traverse the failed link would need to to switch to their alternate end-to-end paths.

Path protecting schemes might be failure dependent or independent. In a failure dependent scheme, the backup path chosen depends on which link on the path failed. The backup path chosen by a failure independent scheme is the same regardless of where the failure occurred on the path. Failure independent schemes are simpler to configure and are faster as they don't have the overhead of signaling the location of the failure on the path. On, the other hand, failure dependent schemes are more capacity efficient.

The choice of a survivability scheme is dependent on a number of factors like the desired restoration time, amount of redundant capacity that one can afford and the nature of traffic patterns (static or dynamic).

## 2.2 Overview of $p$-cycles

$p$-Cycles are an important new development in the field of network survivability proposed by Grover and Stamatelakis [4]. $p$-Cycles have the best features of both ring and mesh networks - they operate at the speed of SONET rings, but at the same time they are almost as efficient as meshes. It has been shown that $p$-cycles can achieve the same theoretical lower bound on spare capacity as mesh networks [16].

$p$-Cycles are similar to rings in that they are preconfigured and precrossconnected. That is, the protection switching action is configured beforehand, for every possible link failure. The difference between $p$-cycles and rings is that $p$-cycles can protect working traffic on straddling links. Straddling links are links between nodes that are on the cycle, but the link itself does not lie on the cycle. Spare capacity does not need to be reserved on straddling links. This is the source of the capacity efficiency of $p$-cycles. Links that lie on the $p$-cycle are protected by one protection path that bypasses the failed link. Straddling links, on the

Table 2.1: Differences between $p$-cycles and SONET rings

| $p$-Cycle | SONET Ring |
|---|---|
| 1 or 2 protection paths | 1 protection path only |
| $<$ 100% redundancy - comparable to mesh | $>$ 100% redundancy |
| Can protect on-cycle and straddling links | Can protect traffic along ring only |
| Working paths routed independent of underlying protection | Working paths must be routed along the ring |

other hand, can be protected by two protection paths. Thus $p$-cycles can protect twice the amount of reserved spare capacity on straddling links.

Another advantage of $p$-cycles is that the routing of the working traffic is independent of the configuration of $p$-cycles. In a ring network, the working traffic is constrained to be routed along the rings. There is no such constraint in a $p$-cycle protected networks. Working traffic could be routed along a suitable path, such as the shortest path.

Traditionally, the number of units of spare capacity reserved for a $p$-cycle is referred to as copies of the $p$-cycle. A unit might be a wavelength, a SONET channel, or bandwidth of a certain granularity, depending on the context in which the $p$-cycle is being used.

Every link on a $p$-cycle carries the same amount of spare capacity. The amount of spare capacity reserved for the $p$-cycle is equal to the maximum of traffic on the on-cycle links and half of the traffic on straddling links protected by the $p$-cycle.

To sum up, the following are the main features of $p$-cycles:

- Restoration speeds as fast as rings.
- Capacity efficiency close to that of mesh networks.
- $p$-Cycles have preconfigured protection paths.
- Protection paths are cross-connected when the network is setup.
- Working traffic can be routed independent of the placement of protection capacity.

This unique combinations of features is what makes $p$-cycles so important. They bridge the divide between the ring and mesh worlds, providing high capacity efficiency without compromising on restoration speed and ease of configuration.

Table 2.2: Differences between $p$-cycles and mesh networks

| $p$-Cycle | Mesh network |
|---|---|
| 50ms restoration time - SONET like speed | $> 100ms$ restoration time |
| $<$ 100% redundancy | $>$ 100% redundancy |
| Protection paths are pre-cross-connected | Protection paths are constructed after failure occurs |
| The protection paths are pre-configure | Protection paths are created from the residual spare capacity after the failure occurs |
| Almost as capacity efficient as mesh networks | Redundancy of around 40% |

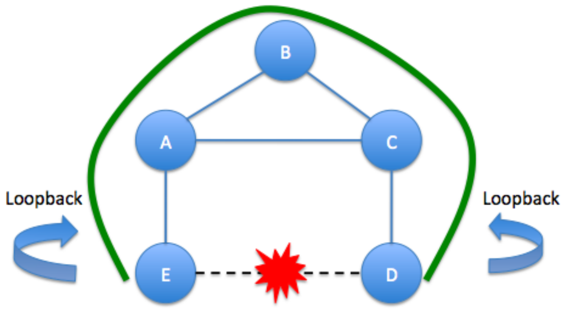


Figure 2.1: On-cycle link protection: 1 protection path.

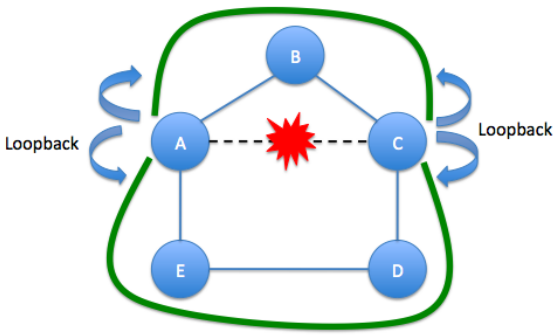


Figure 2.2: Straddling link protection: 2 protection paths.

## 2.3 Working of a $p$-cycle

The working of a $p$-cycle is illustrated in Figures 2.1 and 2.2. When an on-cycle link fails, the protection action is the same as in a ring. The nodes adjacent to the failed

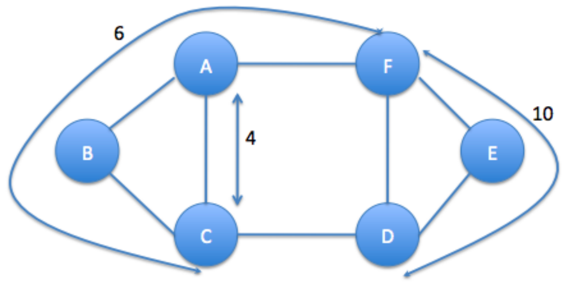


Figure 2.3: First routing of traffic demands.

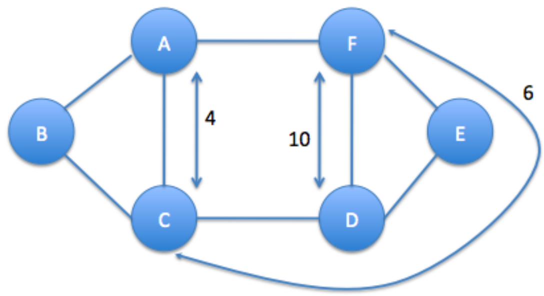


Figure 2.4: Working traffic with first routing of traffic demands.

link do a loopback to route traffic along the part of the $p$-cycle that is still intact as shown in Figure 2.1. For every on-cycle link failure, there is one protection path.

When a straddling link fails, there are two protection paths along the cycle. In Figure 2.2, when link AC goes down, the failure is detected at nodes A and C. A and C then switch the traffic that along link AC to the two protection paths ABC and AEDC. One unit of capacity reserved along the $p$-cycle can protect two units of traffic along a straddling link. The protection of the straddling link is essentially obtained for free because there is no spare capacity reserved on the straddling link. This gives rise to $p$-cycle's low spare capacity usage.

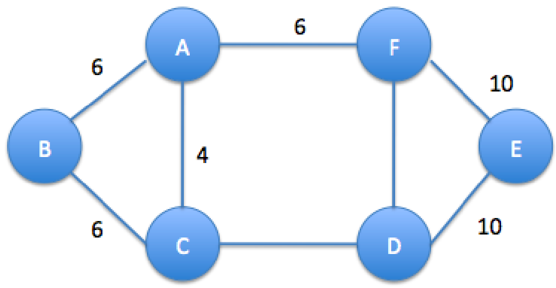


Figure 2.5: Optimal routing of traffic demands.

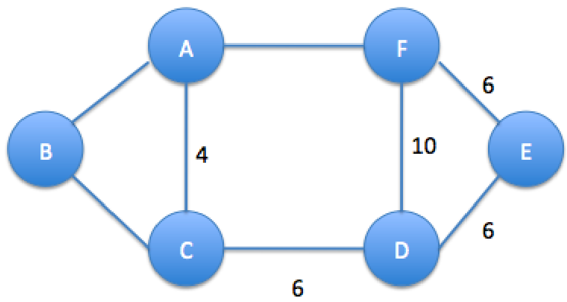


Figure 2.6: Working traffic with optimal routing of traffic demands.

## 2.4 Joint design

The $p$-cycle design described in the previous section does handle the allocation of protection resources separately from the routing of traffic demands. This decoupling does have the advantage of simplicity and flexibility as noted earlier. However, a joint design that optimizes the allocation of $p$-cycles in conjunction with the routing of spare capacity can result in a 20-25% decrease in spare capacity usage [3]. On the flip side, the ILP for the joint formulation is more computationally intensive.

The utility of a joint design is shown in figures 2.3-2.6. Figure 2.3 shows a default routing of traffic demands, that results in the span working traffic shown in Figure 2.5. The optimal $p$-cycle design for this scenario is 10 copies of $p$-cycle ABCDEF: a total of 60 units of spare capacity. If, however, the traffic was routed as shown in Figure 2.4, the resulting span working traffic is shown in figure 2.6. In this case only 6 copies of $p$-cycle ACDEF are needed: a total of only 30 units of spare capacity.

## 2.5 Flow $p$-cycles

Flow $p$-cycles are an extension of the $p$-cycle concept to support protection of flow segment [6]. In the span protecting $p$-cycle design problem, it is assumed that the working traffic flows have already been routed, an we are given a set of working capacities on each of the links after the traffic demands have been routed. The flow $p$-cycle formulation assigns sets of $p$-cycles to protect segments of each flow, such that every link of every flow is fully protected. Segments may span multiple links. If a segment lies completely on a $p$-cycle, it is said to be on-cycle. If a segment is partly on-cycle and partly straddling, it is still considered on-cycle because the same amount of spare capacity will be required on all links of the $p$-cycle. If none of the links of the segment lie on the cycle, it is considered to straddle the cycle. In this case, only half the amount of spare capacity needs to be reserved along the $p$-cycle as there are two protection paths available.

The ILP implicitly divides each flow into segments. Each segment is protected by one or more $p$-cycles. In effect, flow $p$-cycles are a superset of span protecting $p$-cycles. In the case where, every flow is split at every link, that is, every segment is of length 1, the flow $p$-cycle solution is identical to the span protecting $p$-cycle solution.

## 2.6 Failure Independent Path Protecting $p$-cycles

Failure Independent Path Protecting (FIPP) $p$-cycles are another flow protection scheme for $p$-cycles. The advantage of FIPP $p$-cycles is that they are failure independent [10]. That is, the protection switching action occurs at the end points of the flow, and the protection switching action is independent of where the failure occurred in the flow's route. Protection paths are fully preconfigured and precrossconnected. This means that the integrity of the protection path can be tested before the failure occurs. Since failure detection only has to be done at the end nodes, the need for extra signaling is eliminated. FIPP $p$-cycles are suitable for use in multi-hop optical networks.

The FIPP design stipulates that the end points of each flow must lie on a cycle. Each flow is protected end-to-end by a $p$-cycle. Further, a $p$-cycle cannot protect two intersecting flows.

## 2.7 Some other extensions of $p$-cycles

$p$-Cycles as described in the previous sections have are centralized by design. The $p$-cycle configuration is decided by a network administrator when the network is being provisioned. The Distributed Cycle Preconfiguration Protocol (DCPC) is a protocol for configuring $p$-cycles in a decentralized manner, without requiring the intervention of a network administrator [4]. In this self-organizing algorithm, the network nodes autonomously change the configuration of $p$-cycles as the working traffic varies. DCPC is implemented using a statelet broadcasting algorithm. It runs in the background and responds to global traffic variations by reconfiguring the existing $p$-cycles.

A cycle which visits every node in the network exactly once is called a Hamiltonian cycle. A Hamiltonian $p$-cycle has the property of having minimum number of on-cycle spans and maximum number of straddling spans [13]. In a homogeneous network (network in which all spans have equal working capacity), the optimal $p$-cycle design is a single hamiltonian $p$-cycle that covers all the nodes. In a non-homogeneous network, a hamiltonian $p$-cycle is not necessarily the optimal solution.

A $p$-cycle design consideration is the length of the $p$-cycles chosen in the design. The light passed through optic fibers attenuates as the length of fiber increases. So error rates increase as optical paths become longer. Hence regenerators are used at different intermediate nodes to ensure that good signal strength. Also, any forwarding or routing decision to be made forces the signal to be converted from the optical domain to electrical domain to retrieve and parse the packet. So, as the number of hops increase, the end-to-end delay also increases due to the added delay of Optical/Electronic/Optical (O/E/O) conversions. As a result of these factors, a practical $p$-cycle design must limit the length of protection paths used. It is important to note that limiting the length of the protection path is not equivalent to limiting the maximum length of the protecting $p$-cycle.

A hop limited $p$-cycle is one in which the protection path length is limited whereas a circumference limited $p$-cycle is one which the $p$-cycle itself is limited in length. The circumference limit for a $p$-cycle is equal to circumference - 1. The spare capacity savings in using an ILP that incorporates the maximum protection path length constraint over a circumference limited design was not significant enough to justify the increased complexity of the ILP formulation [9].

Recent work has shown that a more effective strategy is to systematically match 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 [12]. The advantage over using a circumference limit is that the full set of *p*-cycles can be used as input to the ILP. Further, this approach provides 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.

*p*-Cycles are designed only to protect against single link failures, since optic fibers are very reliable and the multiple simultaneous failures are unlikely. However, with exponential increase in the size of networks and traffic demands, the probability of multiple failures happening has increased. Hence designing *p*-cycles to handle multiple failures has become a relevant research topic. This has been dealt with in [18].

There has also been some work on protecting multiple service classes using *p*-cycles [8], where an ILP formulation is provided for protecting traffic that is segregated in different service classes.

Node protection has not been a focus of research in optical networks because nodes can be protected using redundancy. However, node protection gains importance at the network layer. The most common cause of failure at the IP level is router failure, so virtual *p*-cycles that provide node protection at the IP/MPLS layer are required [2].

IP layer protocols themselves provide fault tolerance. But the time to regain connectivity is usually long. But introducing *p*-cycles we can minimize the restoration time considerably [17].

Physical layer *p*-cycles provide integrated IP and optical layer protection for IP over WDM networks. Router failure at the IP/MPLS layer, and optical fiber cuts at the physical layer can both be handled by separate sets of *p*-cycles, but this is not a bandwidth efficient solution. [1] describes strategies for multi-layer survivability and bandwidth management.

M-cycles or monitoring cycles are centralized mechanism used to detect failed links on the network using minimum amount of extra capacity [19]. The idea is to cover the network with a set of monitoring cycles such that each link is covered by a different set of cycles. So, for every link failure a different set of cycles will fail. A centralized protection mechanism can deduce which link has failed when it knows the cycles that have failed. This scheme is much more bandwidth efficient than requiring a monitoring channel on every link. Grover et al, show that a regular *p*-cycle design can be augmented to support m-cycles with minimal additional spare capacity [5].

A Shared Risked Linked Group (SRLG) is a group of links that may fail simultaneously. This may be because the links share a common path at some point, or the fibers corresponding to the links are in the same bundle. Lu and Ruan describe a *p*-cycle based design to protect networks with SRLG constraints [11].

*p*-Cycles are most suitable for scenarios in which the traffic matrix is known and does not vary with time. This is because *p*-cycles are preconnected according to the expected traffic demands. Reconfiguring *p*-cycles for on a pre-connection request basis is not feasible as *p*-cycle computation is time consuming. So *p*-cycle based solutions for dynamic traffic usually resort to overprovisioning bandwidth. *p*-Cycle reconfiguration may be done occasionally as in the case of Adaptive PWCE to adapt to changing traffic patterns. He el al [7] propose an algorithm to determine an optimal set of *p*-cycles for a given network topology and describe routing strategies to accommodate dynamically arriving requests.

Protected Working Capacity Envelopes (PWCE) are another method for dealing with dynamic traffic demands [15]. The fundamental idea behind the PWCE is to divide a network into working and spare capacity such that all of the working capacity is protected by the spare. Thus, any traffic demand that can be routed over the working capacity is inherently protected . Although the PWCE is designed for one particular configuration of working capacity, any set of traffic demands that are contained within the allocated working capacity are protected. So a PWCE could potentially protect a large number of working path configurations.

A PWCE will support dynamically changing traffic demands as long as the total traffic lies within the envelope of working capacity. Variations in traffic demands can be accommodated if the traffic pattern is statistically stationery. This is typically the case in today's networks at high levels of aggregation. When there are long term variations in the traffic pattern, other techniques like the Adaptive Protective Working Capacity Envelope (APWCE) [14] need to be used.

# Chapter 3

# Differential Capacity p-Cycles

## 3.1 Span protecting differential capacity p-cycles

We propose a new survivability mechanism called differential capacity *p*-cycles that incrementally improves on regular *p*-cycles to increase their capacity efficiency. They exploit the fact that *p*-cycles reserve the same amount of spare capacity on all on-cycle links, though not all of the spare capacity reserved is used in most cases (this is explained in more detail below). They are the similar to regular *p*-cycles except that they may have varying amounts of spare capacity on different on-cycle links.

The main idea behind this thesis is that unused protection channels on a regular *p*-cycle can be eliminated resulting in a differential capacity *p*-cycle that has a lower amount of spare capacity reserved on the link with maximum on-cycle traffic.

The amount of spare capacity that needs to be present on an on-cycle link is equal to the maximum of working traffic on all other links (other than the current link), and the amount of straddling traffic whose protection path passes through this link. Note that the traffic on straddling links need not be split equally along the two paths. We will elaborate on this in Section 3.4.

When the spare capacity reserved on an on-cycle link is determined by working traffic, $w_{max}$, on an on-cycle link, every other on-cycle link will need to have $w_{max}$ spare capacity, which is the same as in the case of regular *p*-cycles. However, the amount of

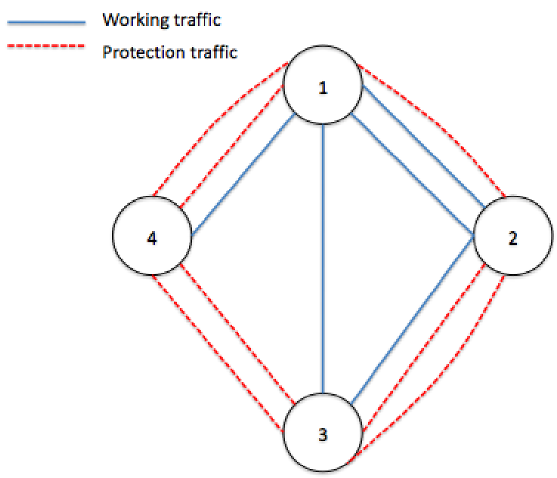


Figure 3.1: Differential capacity $p$-cycle.

capacity reserved on the $w_{max}$ link is $w_{max_2}$, which is less than $w_{max}$. Thus, $w_{max} - w_{max_2}$ units of capacity are saved on the $w_{max}$ link. This is also the total reduction in spare capacity requirement on the $p$-cycle.

For example, in Figure 3.1, link 1-2 has only 1 unit of spare capacity as 1 unit of working capacity needs to be protected by this link. All the other on-cycle links need to have 2 units of spare capacity in order to provide a backup path if link 1-2 fails. A regular $p$-cycle would have required 2 units of spare capacity on each of 1-2, 2-3, 3-4 and 4-1. Thus, there is a saving of 1 unit of spare capacity over the regular $p$-cycle.

## 3.2 Protection switching

The switching action at each node is preconfigured and precrossconnected as in regular $p$-cycles. Each node knows in advance which of the incoming and outgoing channels are used for working and protection traffic. There is a mapping between incoming (working, protection) channels and outgoing (working, protection respectively) channels. In addition, there is a mapping between working channels, and the corresponding protection channel that will be used in the event of failure of the working channel. Each node stores this fixed mapping. The mapping differs from regular $p$-cycles in that the number of protection channels is not the same on all links. At some nodes there might be no outgoing protection

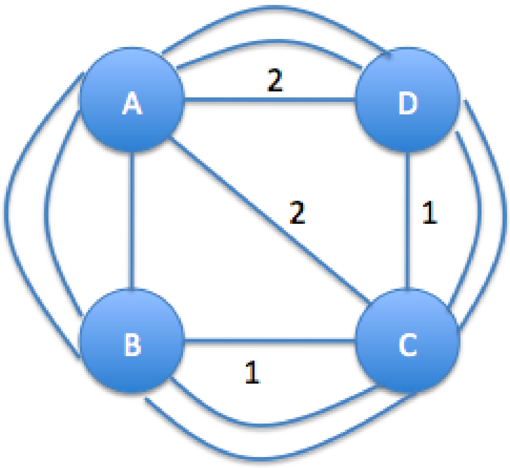


Figure 3.2: Protection wavelengths in a traditional $p$-cycle.

channel corresponding to an incoming protection channel and vice versa.

A node forwards data from its incoming protection channel to the outgoing protection channel, regardless of whether the channel is in use or not.

When a node detects a failure on its outgoing working channel, it immediately switches to the corresponding protection channel on the outgoing link in the opposite direction. When a failure on an incoming working channel is detected, the node starts looking for data on the corresponding protection channel.

In Figure 3.2, 2 protection wavelengths are required on each of the links AB, BC, CD, and AD as part of the $p$-cycle ABCD. The spare capacity usage in a differential $p$-cycle solution is shown in Figure 3.3. In this case, the protection switching is identical to the traditional case. But there is a small change in the preconfiguration at nodes A and D. A and D are configured the same as in the traditional case, except that they don't forward the second protection wavelength. That is, A does not forward the traffic coming on the second protection wavelength from B on to D. Also the second wavelength on link AD is not configured as a protection wavelength as in the traditional case. This is because, since link AD has the maximum traffic, the difference between the maximum and second maximum is part of traffic flows between A and D that do not pass through other nodes on the cycle. So, this extra traffic need not be protected on the link AD itself.

In summary, the amount of preconfiguration effort required is the same as in

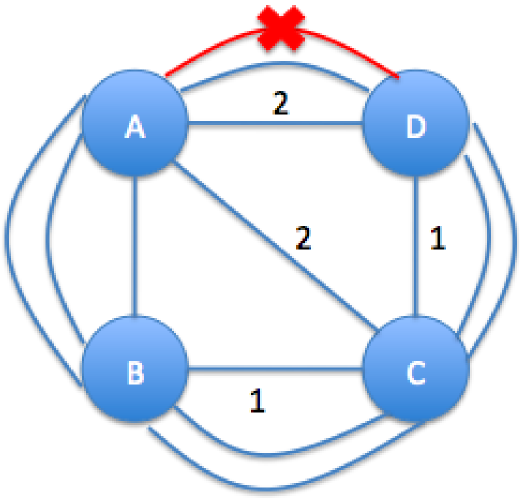


Figure 3.3: Protection wavelengths in a differential capacity $p$-cycle.

regular $p$-cycles. The only difference in preconfiguration is that the unused channels on the maximum links are a part of the configuration. So the rules to forward traffic to and from these channels are omitted. On the other hand, the protection switching action is identical to the regular $p$-cycle case. The benefit reduced spare capacity usage is achieved without compromising on the speed of restoration, or ease of configuration.

## 3.3 Fiber capacity requirement

The maximum number of wavelengths used on any link, calculated as sum of working and protection wavelengths, is usually lower in the case of DC $p$-cycles. Even when the objective of the ILP does not explicitly minimize the number of wavelengths, we observed that DC $p$-cycles require fewer wavelengths.

For instance, a set of regular $p$-cycles cannot be used to protect the network shown in Figure 3.4. This is because any $p$-cycle that protects link CD will require 7 units of protection capacity. This implies that a minimum of $7+7=14$ units of protection capacity will required on link CD. But since only 10 units are available on each fiber, a solution using regular $p$-cycles cannot be obtained.

On the other hand,a DC $p$-cycle ABCDEA can be used to protect all traffic demands in the network. All on-cycle links except for link link CD need to have 7 units of spare capacity. Since none of these links have more than 3 units of working capacity, the fiber capacity constraint is not violated. Link CD requires only 3 units of spare capacity if the working traffic on the straddling span is protected by 3 units of spare capacity in either direction along the DC $p$-cycle.

## 3.4 Straddling link protection

In Grover's $p$-cycle formulation, straddling traffic is divided equally along the two protection paths.

Claim: Dividing traffic equally along the two protection paths is optimal.

Proof:

- if $w^{straddle} \geq 2w_{max}^{oncycle}$

  Total spare capacity $C =$ k$max(w_1, w^{straddle} - w_1) + (n-k)max(w_1, w^{straddle} - w_1)$
  i.e. $C =$ n$max(w_1, w^{straddle} - w_1$
  $C$ is minimum when $w_1 = w^{straddle}/2$

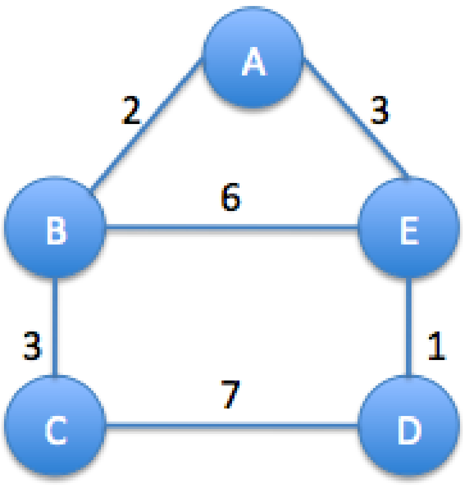


Figure 3.4: Differential capacity $p$-cycles can benefit from an uneven split of protection traffic for straddling spans.

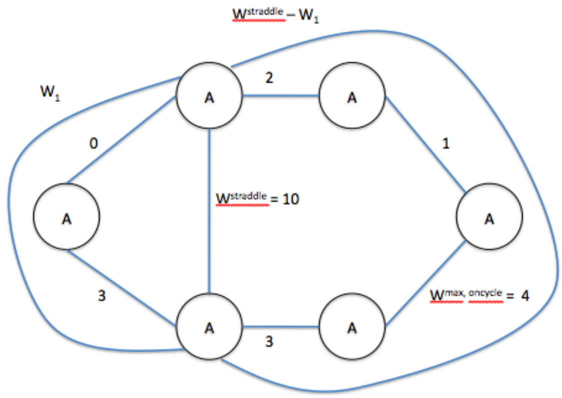


Figure 3.5: DC $p$-cycles can benefit from an uneven split of protection traffic for straddling spans.

- if $w^{straddle} < 2w_{max}^{oncycle}$

  The amount of spare capacity on each on-cycle span must be atleast $w_{max}^{oncycle}$ . When the traffic on the straddling span is split as $w^{straddle}/2$ along the 2 paths, the spare capacity reserved on each on-cycle span is still $w_{max}^{oncycle}$ as $w^{straddle} < 2w_{max}^{oncycle}$.

In the case of differential capacity $p$-cycles, splitting the traffic evenly may not be optimal as seen in Figure 3.5. In this case, splitting the traffic on the straddling link as 5 units along the 2 protection paths requires a total spare capacity of $5 \times 6 = 30$ units. On, the other hand sending 6 units along the longer path and 4 units along the shorter path, that is, $w_1 = 6$, results in a total spare capacity requirement of only $6 \times 2 + 4 \times 4 = 28$ units. Regular $p$-cycles cannot take advantage of this as they are constrained to have the same protection capacity on all links.

However, the case where an uneven split produces better results occurs only when a the working traffic protected by the $p$-cycle on a straddling link is greater than the protected traffic on any on-cycle link, that is, $w^{straddle} > w_{max}^{oncycle}$.

### 3.4.1 Routing straddling span traffic in differential capacity $p$-cycles

When there is a single straddling span, the routing of straddling traffic can be easily determined. The split depends on the relative magnitudes of the straddling traffic and maximum on-cycle traffic. This is dealt with in the appendix.

When there are multiple straddling spans in the cycle, the problem has a combinatorial flavor.It is conjectured to be NP-Complete.

## 3.5 ILP formulation

The model assumes that all links are bidirectional. That is, every link consists of a pair of fibers in opposite directions. Traffic demands are assumed to be bidirectional. Asymmetric demands can be accommodated if we take the maximum of the forward and reverse demands to be flowing in either direction. The network is assumed to either have full wavelength conversion, or O/E/O conversion at every node.

### 3.5.1 Sets

**S** Set of spans

**C** Set of *p*-cycles

**M** Set of routes between each source-destination pair

**D** Set of traffic demands

### 3.5.2 Parameters

$w_j$ Working traffic on link j

$f_j$ Capacity of link j

$x_{i,j}$ 1 if link j lies on *p*-cycle i, 2 if link j straddles *p*-cycle i, 0 otherwise

$t_{i,j}$ 1 if link j lies on *p*-cycle i, 0.5 if link j straddles *p*-cycle i, 0 otherwise

$p_{i,j}$ 1 if link j lies on *p*-cycle j, 0 otherwise

$q_{i,j}$ 1 if link j lies on or straddles *p*-cycle i, 0 otherwise

$\beta_{l,m,j}$ 1 if demand $l$ traverses link $j$ when routed over the $m^{th}$ route between the source and destination

$d^l$ Amount of traffic between source and destination of demand $l$

### 3.5.3 Variables

$s_j$ Spare capacity on span j

$s_{i,j}$ Spare capacity on span j that is used by $p$-cycle i

$r_{i,j}$ Number of units of working capacity on span j are protected by $p$-cycle i

### 3.5.4 Non-joint differential capacity $p$-cycle

**Objective**

$$\text{Minimize} \sum_{i \in C, j \in S} s_{i,j}$$

**Constraints**

$$s_{i,j} \geq r_{i,k} \qquad \forall i \in C, j,k \in S, k \neq j, p_{i,j} > 0 \tag{3.1}$$

$$w_j^{max\ oncycle} \geq w_j \qquad \forall i \in C, j,k \in S, p_{i,j} > 0 \tag{3.2}$$

$$s_{i,j} \geq \bar{\zeta}_{i,j,k} w_j^{max\ oncycle} + \zeta_{i,j,k}(r_{i,k} - w_j^{max\ oncycle}) \tag{3.3}$$
$$\forall i \in C, j,k \in S, k \neq j, p_{i,j} > 0$$

$$s_j = \sum_{i \in C} s_{i,j} \qquad \forall j \in S \tag{3.4}$$

$$w_j + s_j \leq f_j \qquad \forall j \in S \tag{3.5}$$

$$\sum_{i \in C} r_{i,j} q_{i,j} = w_j \qquad \forall i \in C \tag{3.6}$$

$$s_j, s_{i,j}, r_{i,j} \in \{0, 1, 2, 3, \cdots\} \tag{3.7}$$

### 3.5.5 Explanation of ILP

(3.1) The amount of spare capacity that needs to be reserved on a link of a $p$-cycle should be sufficient to provide a protection path when other on-cycle links fail.

(3.2), (3.3) Traffic on straddling links can be split on left and right protection paths in any proportion. Links on the left and right part of the cycles must have sufficient spare capacity to protect the portions of straddling traffic that are assigned to the respective sides.

(3.4) The total spare capacity reserved on a link is equal to the sum of spare capacities reserved on it to protect each $p$-cycle.

(3.5) The total capacity used on a link is the sum of working traffic and spare capacity on the link.

(3.6) Working traffic on every link is protected by some set of $p$-cycles.

## 3.6 Joint differential capacity $p$-cycle

The joint version of DC $p$-cycle problem chooses both working and backup paths in the ILP. The working path for each traffic demand is chosen from a set of candidate routes between each node pair. The joint version of the problem is more complicated, but it gives better solutions as the ILP has a larger search space.

The following additional equations are required for the joint formulation:

$$\sum_{m \in M} g_{d,m} = d^l \qquad \forall l \in D \tag{3.8}$$

$$w_j = \sum_{l \in D, m \in M} g_{l,m} * \beta_{l,m,j} \qquad \forall j \in S \tag{3.9}$$

$$w_j, s_j, s_{i,j}, r_{i,j}, g_{l,m} \in \{0, 1, 2, 3, \cdots\} \tag{3.10}$$

In the joint formulation, working traffic $w_j$ is a variable that is determined by the routing of traffic demands $d^l$. There is a fixed set of routes between each pair of nodes. Demands are constrained to be routed over these paths. Equation (3.8) asserts that all traffic demands must be routed over one or more paths. Equation (3.9) says that the working traffic on a link is equal to sum traffic demands that are routed over that link.

### 3.6.1 Joint differential capacity $p$-cycle: traffic demands must not be bifurcated

In some situations it might not be feasible to split traffic demands. This can be accommodated with the following additional constraint.

$$\sum_{m \in M} g_{d,m} = 1 \qquad \forall d \in D \tag{3.11}$$

## 3.7 Heuristics for $p$-cycle preselection

The aim of $p$-cycle preselection heuristics is to select a set of candidate cycles that are a subset of the set of all cycles in the network. These candidate cycles are given as input to the ILP. The size of the candidate set is typically much smaller than the total number of cycles in the network, so the running time of the ILP is a fraction of the time taken to do the computation on the complete set of cycles. A good heuristic will result in a solution that is close to the optimal solution obtained on the complete set of cycles.

We found that heuristics that have been proposed for traditional $p$-cycles perform very well when applied to differential capacity $p$-cycles. We evaluated Topological Score, Apriori Efficiency and Demand Weighted Efficiency as metrics for cycle preselection. Demand Weighed Efficiency was found to work best for traditional $p$-cycles. Interestingly, DWE turned out to be a better metric for differentiated $p$-cycles. The solutions produced by the heuristic were closer to optimal on an average in the case of differential capacity $p$-cycles. The results are presented in Section 4.4

## 3.8 Differential Capacity Flow $p$-Cycles

The differential capacity idea can be applied to flow $p$-cycles as well. When a flow straddles a cycle as shown in Figure 3.9, the backup traffic can be routed over the shorter path. In the figure, routing over path BAE, incurs a spare capacity cost of 4 units, while

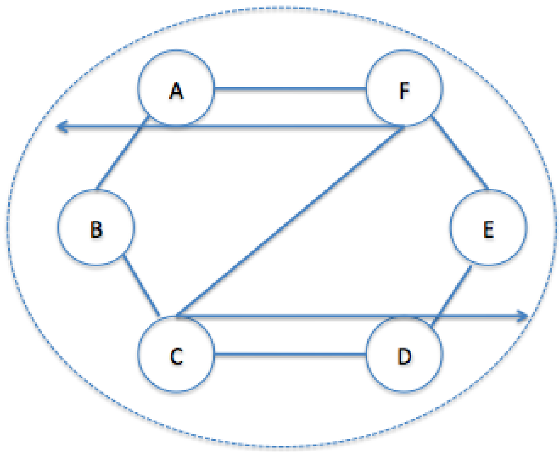


Figure 3.6: On-cycle flow with spare capacity required on all spans.

splitting the traffic evenly along the two paths requires 5 units of spare capacity. When a flow is on-cycle, as is the case in Figure 3.8, the backup traffic can be routed over path ABCDE. Similarly, when the flow is partly on-cycle and partly straddling, as shown in Figure 3.7, the backup traffic can be routed over the part of the cycle that does not carry working traffic. In this example, backup traffic is routed over ABCD. However, in the case shown in Figure 3.6, spare capacity will need to be provisioned all along the cycle, as spare capacity will be required on every on-cycle link in some failure scenario.

### 3.8.1 Protection switching

Protection switching in traditional flow $p$-cycles is more complicated than in the corresponding span protecting case. Each node has to keep track of all flows passing through it. If a flow segment is terminated at a node, it must have a preconfigured backup segment. It sends/receives traffic on the backup segment if it receives a failure notification from an upstream/downstream node or detects failure of an adjacent outgoing/incoming link. If a node does not terminate a segment, it is responsible for sending a failure notification the end nodes of the segment. Thus, preconfiguration as well as the protection switching action is more complicated in the case of flow $p$-cycles.

For example, when link AE fails, the failure is detected by nodes A and E. The

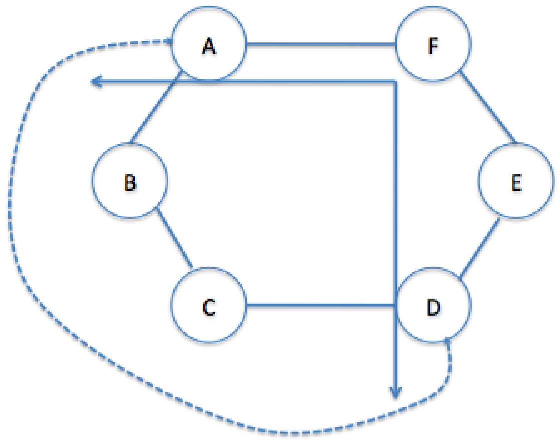


Figure 3.7: Partly on-cycle flow.

failure information is communicated to end nodes of all flows that pass through the failed link. In this case, flows AC and DE traverse link AE. So nodes A and C, D and E, are notified of the failure by the nodes A and E which detect the failure. Upon receiving the failure notification, protection switching actions occur, and flow AC is switched to path ABC, and flow DE is switched to path DAE.

Differential capacity flow *p*-cycles have the same preconfiguration and protection switching complexity as their traditional counterparts. As in the span protecting case, certain spare capacity wavelengths may not be required, so they are just omitted from the configuration. The difference here is that there might be ́differential capacityón multiple spans of the *p*-cycle. It is important to note, however, that this does not result in any added switching complexity as there are no new rules added. The only change is that some forwarding actions are eliminated.

### 3.8.2 ILP formulation

**Sets**

$C$ Set of *p*-cycles

$S$ Set of spans

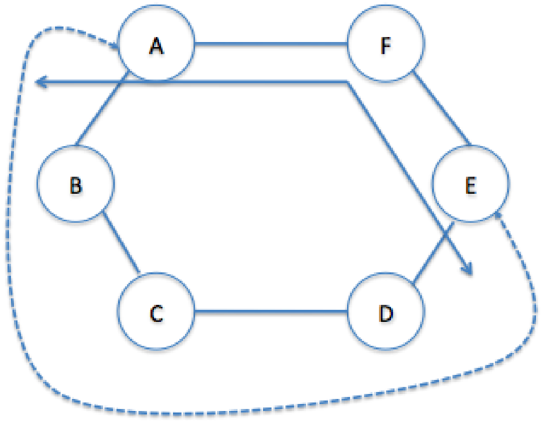


Figure 3.8: Fully on-cycle flow.

$D$ Set of traffic demands

**Parameters**

$d^r$ The magnitude of traffic demand $r$.

$x_j^r$ 1 if cycle $j$ can protect demand $r$, 0 otherwise

$o_k^r$ 1 if traffic demand $r$ is routed over span $k$, 0 otherwise

$q_{i,j}$ 1 if span cycle $i$ can protect span $j$ (either in an on-cycle or straddling relationship), 0 otherwise

$\zeta_{i,j}^r$ 1 if span $j$ is on the left of cycle $i$ with respect to traffic demand $r$, 0 otherwise

$\zeta_{i,j}^{\bar{r}}$ 1 if span $j$ is on the right of cycle $i$ with respect to traffic demand $r$, 0 otherwise

**Variables**

$s_{i,j}$ Amount of spare capacity required on span $j$ as part of cycle $i$

$s_{i,j,k}$ Amount of spare capacity required on span $j$ as part of cycle $i$ to protect against failure of span $k$

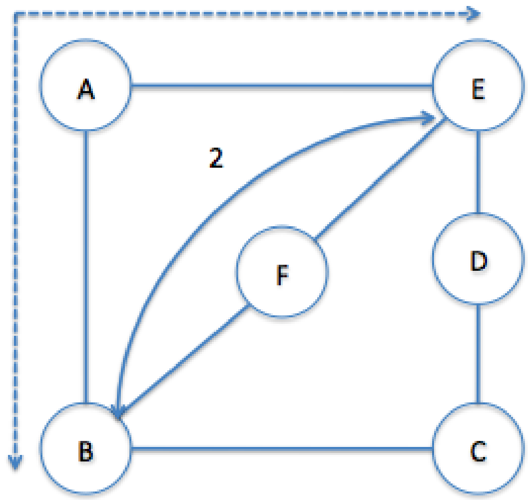


Figure 3.9: Straddling flow.

$r_i^r$ Portion of traffic demand $r$ that is routed protected by cycle $i$

$\eta_i^r$ Portion of traffic demand $r$ protected by cycle $i$, whose protection path is over the left part of cycle $i$. Here left, right refer to the left and right parts of the cycle formed by the intersection of the traffic demand with the cycle.

$\bar{\eta_i^r}$ Portion of traffic demand $r$ protected by cycle $i$, whose protection path is over the right part of cycle $i$

**Objective**

$$\text{minimize} \sum_{i \in C, j \in S} s_{i,j}$$

**Constraints**

$$\sum_{i \in C} \gamma_{i,j}^r \, r_j^r \geq d^r \, o_j^r \qquad \forall r \in D, j \in S \tag{3.12}$$

$$r_i^r \leq \eta_i^r + \bar{\eta_i^r} \qquad \forall r \in D, i \in C \tag{3.13}$$

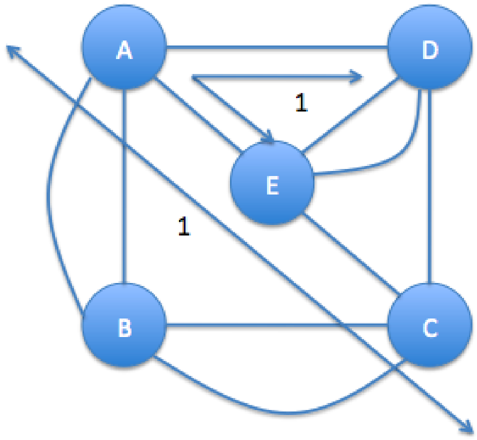


Figure 3.10: Protection switching in flow $p$-cycles.

$$\eta_i^r \geq \zeta_{i,j}^{\bar{r}}\, q_{i,j}\, o_j^r\, r_i^r \qquad \forall r \in D, i \in C, j \in S \tag{3.14}$$

$$\eta_i^{\bar{r}} \geq \zeta_{i,j}^{r}\, q_{i,j}\, o_j^r\, r_i^r \qquad \forall r \in D, i \in C, j \in S \tag{3.15}$$

$$s_{i,j,k} \geq \sum_{r \in D} ovr_k^r(\zeta_{i,j}^r\, \eta_i^r + zet\bar{a}_{i,j}^r\, \eta_i^{\bar{r}}) \qquad \forall i \in C, j, k \in S \tag{3.16}$$

$$s_{i,j} \geq s_{i,j,k} \qquad \forall i \in C, j, k \in S \tag{3.17}$$

### 3.8.3 Explanation of ILP

(3.18) The objective is to minimize the total spare capacity in the network

(3.12) Each span of along which a traffic demand is routed is protected by one or more $p$-cycles.

(3.13) The sum of capacity of the left and right protection paths of each traffic demand must be at least equal to the traffic protected (if the flow straddles the cycle, or all of the protection traffic is routed over one side of the cycle), or more (twice the magnitude of the protected flow, if it is an on-cycle relationship where protection capacity is required on all spans of the cycle).

(3.14) The protection capacity along the left part of a cycle must be sufficient to protect any working traffic, either straddling or on-cycle, along the right part of the cycle.

(3.15) The protection capacity along the right part of a cycle must be sufficient to protect any working traffic, either straddling or on-cycle, along the left part of the cycle.

(3.16) Spare capacity on a given span $j$ of a $p$-cycle $i$, required to protect span $k$, is equal to the sum of the portions of traffic demands protected by the $p$-cycle, that traverse span $k$.

(3.17) Spare capacity on a span of a $p$-cycle is equal to the maximum spare capacity on that span of the $p$-cycle required to protect any other span.

### 3.8.4 Limitations of model

Differential flow $p$-cycles can perform worse than regular flow $p$-cycles and even span protecting $p$-cycles, when they are formulated as shown above. This is because, in the differential flow $p$-cycle formulation we constrain a $p$-cycle to protect the *maximum* segment of a flow that crosses it. That is, a $p$-cycle is forced to protect the longest segment of a flow that intersects it. This makes it less general flow $p$-cycles, though it still has the advantage of differential capacity.

For example, in Figure 3.11, traffic demands AB, EF and BHDF need to be protected by the network shown. The following protection scheme could be used by a regular flow solution: flow AB and 1 unit of flow BDHC on span BH can be protected by a unit $p$-cycle ABH. flow EF and 1 unit of BDHC on span FD can be protected by a unit $p$-cycle DEF. 2 units on straddling span DH of flow BHDF and the remaining 1 units of spans BH and FD can be protected by 1 unit of capacity on $p$-cycle BCDFGH. This requires a total of $3 + 3 + 6 = 12$ units of spare capacity. The same amount of spare capacity will be required by a span protecting $p$-cycle solution by considering the working traffic on a link-by-link basis.

When a differential flow $p$-cycle formulation is used, however, a total of 14 units of spare capacity will be required. This is because 2 units of spare capacity will be required

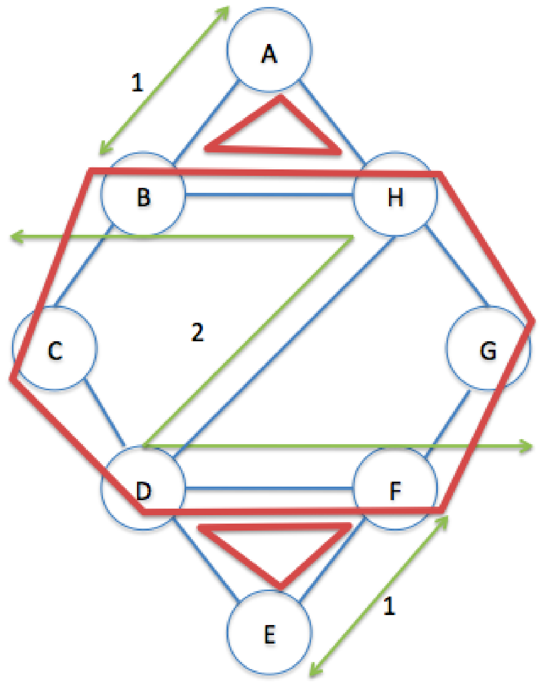


Figure 3.11: 8 node network with traffic flows.

on $p$-cycle BCDFGH to protect flow BDHC. In addition 1 unit of spare capacity will have to be allocated along $p$-cycles ABH and DEF to protect flows AB and EF respectively.

However, we observed that in the average case, the differential flow $p$-cycle formulation gives good capacity efficiency. It turns out that in most cases the benefit of the differential capacity and straddling flow properties outweigh the loss of overlooking a portion of the solution space. Data to support this conclusion is presented in the Results section.

### 3.8.5 Computation of parameters $\zeta^{r}_{i,j}$ and $\zeta^{\bar{r}}_{i,j}$

Algorithm 3.8.5, on the next page, describes how to compute the parameters $\zeta^{r}_{i,j}$ and $\zeta^{\bar{r}}_{i,j}$.

$i$ contains a list of vertices in the order they appear in the cycle. The first vertex of the cycle is chosen arbitrarily.

$r$ contains the a list of vertices that represents the path of an end-to-end flow.

**if** Flow $r$ intersects cycle $i$ at less than 2 vertices **then**

$\zeta^{r}_{i,j} = \zeta^{\bar{r}}_{i,j} = 0$

return

**end if**

Let $u$ = first occurrence of a vertex in $r$ that is also in $i$

Let $v$ = last occurrence of a vertex in $r$ that is also in $i$

$left$ = set of edges between $u$ and $v$ in list $i$.

$right$ = remaining edges of cycle $i$ that don't lie between $u$ and $v$

**if** Span $j$ is present in left **then**

$\zeta^{r}_{i,j} = 1$

$\zeta^{\bar{r}}_{i,j} = 0$

return

**else if** Span $j$ is present in right **then**

$\zeta^{r}_{i,j} = 0$

$\zeta^{\bar{r}}_{i,j} = 1$

return

**else**

$\zeta^{r}_{i,j} = \zeta^{\bar{r}}_{i,j} = 0$

return

**end if**

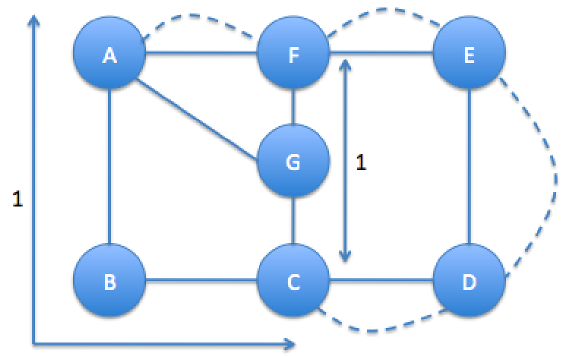


Figure 3.12: Differential capacity FIPP $p$-cycle example. Dashed arcs are protection channels.

## 3.9 Differential Capacity Failure Independent Path Protecting $p$-Cycles

Differential capacity FIPP $p$-cycles are the differential capacity version of the FIPP $p$-cycles proposed by Kodian and Grover [10]. FIPP $p$-cycles protect lightpaths in a failure independent end-to-end manner. As explained in Section 2.6, failure independent means that the protection switching action is independent of the location of the failure. Link failure is detected at the end points of the lightpath, and a switch to the preconfigured and precrossconnected backup path is initiated at the end-points. There is no signaling required to determine the link on the path that failed.

The differential capacity version of this problem introduces the variable capacity aspect to potentially reduce the amount of spare capacity that needs to be reserved.

Figure 3.12 shows an example of differential capacity FIPP $p$-cycle protection. In this example, flows ABC and FGC are protected by $p$-cycle ABCDEF. Note that these can be protected by the same $p$-cycle as they don't intersect. Flow ABC has protection path CDEFA. The protection path of flow FGC is CDEF. 1 unit of protection capacity is allocated on links CD, DE, EF, and AF. A regular FIPP $p$-cycle design would have required spare capacity to be allocated on links AB and BC as well, though this capacity would never have been utilized. So, in this case a differential capacity FIPP $p$-cycle design can 2 units of unused protection capacity.s

### 3.9.1 Protection switching

Preconfiguration and protection switching are very simple in FIPP *p*-cycles. Each flow is configured as a single end-to-end lightpath. The protection switching action is only taken at the end points of the flow. When a link failure occurs, the end nodes will be able to detect a loss of light. When this occurs they switch to the backup path. In terms of preconfiguration, every node must be configured to forward (at the optical layer) flows that pass through it. Also, a node must be configured to terminate flows for which it is an endpoint.

As in the case of span protecting and flow *p*-cycles, the only difference in the differential capacity version is that some unused wavelengths channels are eliminated from the configuration. The switching logic doesn't consider these channels to be a part of the *p*-cycle. The speed of protection switching and complexity of preconfiguration remain the same as in regular FIPP *p*-cycles.

### 3.9.2 ILP formulation

**Sets**

$C$ Set of *p*-cycles

$S$ Set of spans

$D$ Set of traffic demands

**Parameters**

$d^r$ The magnitude of traffic demand $r$.

$x_j^r$ 1 if cycle $j$ can protect demand $r$, 0 otherwise

$o_k^r$ 1 if traffic demand $r$ is routed over span $k$, 0 otherwise

$q_{i,j}$ 1 if span cycle $i$ can protect span $j$ (either in an on-cycle or straddling relationship), 0 otherwise

$\rho^{p,q}$ 1 if traffic demands $p$ and $q$ intersect, 0 otherwise

$\zeta^{r}_{i,j}$ 1 if span $j$ is on the left of cycle $i$ with respect to traffic demand $r$, 0 otherwise

$\zeta^{\bar{r}}_{i,j}$ 1 if span $j$ is on the right of cycle $i$ with respect to traffic demand $r$, 0 otherwise

$\Delta$ A large constant, greater than the largest traffic demand

**Variables**

$s_{i,j}$ Amount of spare capacity required on span $j$ as part of cycle $i$

$r^r_i$ Portion of traffic demand $r$ that is routed protected by cycle $i$

$\eta^r_i$ Portion of traffic demand $r$ protected by cycle $i$, whose protection path is over the left part of cycle $i$. Here left, right refer to the left and right parts of the cycle formed by the intersection of the traffic demand with the cycle.

$\eta^{\bar{r}}_i$ Portion of traffic demand $r$ protected by cycle $i$, whose protection path is over the right part of cycle $i$

$\gamma^r_i$ 1 if cycle $i$ protects demand $r$, 0 otherwise

**Objective**

$$\text{minimize} \quad \sum_{i \in C, j \in S} s_{i,j} \tag{3.18}$$

**Constraints**

$$\sum_{i \in C} x^r_i r^r_i \geq d^r \qquad \forall r \in D \tag{3.19}$$

$$r^r_i \leq \eta^r_i + \eta^{\bar{r}}_i \qquad \forall r \in D, i \in C \tag{3.20}$$

$$\eta^r_i \geq \zeta^{\bar{r}}_{i,j}\, q_{i,j}\, o^r_j\, r^r_i \qquad \forall r \in D, i \in C, j \in S \tag{3.21}$$

$$\eta^{\bar{r}}_i \geq \zeta^{r}_{i,j}\, q_{i,j}\, o^r_j\, r^r_i \qquad \forall r \in D, i \in C, j \in S \tag{3.22}$$

$$s_{i,j} \geq \zeta_{i,j}^{r} \, \eta_i^r + \zeta_{i,j}^{\bar{r}} \, \eta_i^{\bar{r}} \qquad \forall i \in C, j \in S, r \in D \tag{3.23}$$

$$\gamma_i^r \Delta \geq r_i^r \qquad \forall r \in D, i \in C \tag{3.24}$$

$$\gamma_i^r \leq \Delta r_i^r \qquad \forall r \in D, i \in C \tag{3.25}$$

$$\rho^{p,q} + \gamma_i^i + \gamma_i^q \leq 2 \qquad \forall i \in C, p, q \in D, p \neq q \tag{3.26}$$

### 3.9.3 Explanation of ILP

(3.18) The objective is to minimize the total spare capacity in the network

(3.19) Every traffic demand is protected end to end by one or more *p*-cycles.

(3.20) The sum of capacity of the left and right protection paths of each traffic demand must be at least equal to the traffic protected (if the flow straddles the cycle, or all of the protection traffic is routed over one side of the cycle), or more (twice the magnitude of the protected flow, if it is an on-cycle relationship where protection capacity is required on all spans of the cycle).

(3.21) The protection capacity along the left part of a cycle must be sufficient to protect any working traffic, either straddling or on-cycle, along the right part of the cycle.

(3.22) The protection capacity along the right part of a cycle must be sufficient to protect any working traffic, either straddling or on-cycle, along the left part of the cycle.

(3.23) The spare capacity reserved on a given span as part of a cycle must be sufficient to carry protection traffic for all traffic demands that are protected by the cycle.

(3.24),(3.25) These constraints ensure that $\gamma_i^r$ is 1 if and only if $r_i^r > 0$, otherwise it is 0.

(3.26) A cycle cannot protect two intersecting traffic demands

## 3.10 Comparison of capacity efficiencies of *p*-cycle variants

1. Flow *p*-cycles are atleast as good as regular span protecting *p*-cycles.
2. Flow *p*-cycles are atleast as good as FIPP *p*-cycles.
3. FIPP *p*-cycles may be better or worse than span protecting *p*-cycles.
4. Differential capacity span protecting *p*-cycles are atleast as good as regular *p*-cycles.
5. Differential capacity span protecting *p*-cycles can be better or worse than regular flow *p*-cycles.
6. Differential capacity span protecting *p*-cycles can be better or worse than regular FIPP *p*-cycles.
7. Differential capacity flow *p*-cycles are better than regular and differential capacity span protecting *p*-cycles, regular flow *p*-cycles in some cases, and worse in others.
8. Differential capacity FIPP *p*-cycles can be better than regular and differential capacity span protecting *p*-cycles, and regular flow *p*-cycles. They can never be better than differential capacity flow *p*-cycles. On the other hand, they can be worse than all the other types of *p*-cycles.
9. Differential capacity flow *p*-cycles are always atleast as good as regular FIPP *p*-cycles.

**1** The protection offered by a flow *p*-cycle is a superset of that of a regular *p*-cycle. A flow *p*-cycle design can be used to protect an end-to-end flow by splitting the flow into any arbitrary sequence of contiguous segments. Splitting the flow on a link-by-link basis, which is essentially what a span protecting *p*-cycle does, is a special case of the type of protection offered by a flow *p*-cycle design. Flow *p*-cycles outperform span protecting *p*-cycles when they protect straddling segments that span more than one link. In Figure 3.13, the flow ABCD spanning 3 links lies on cycle ABCDE. A flow *p*-cycle would be able to exploit this relationship, but a span protecting *p*-cycle would have to consider the 3 traffic carrying spans as on-cycle, where the *p*-cycle is the cycle formed the the straddling segment ABCD and the shorter path between A and C.

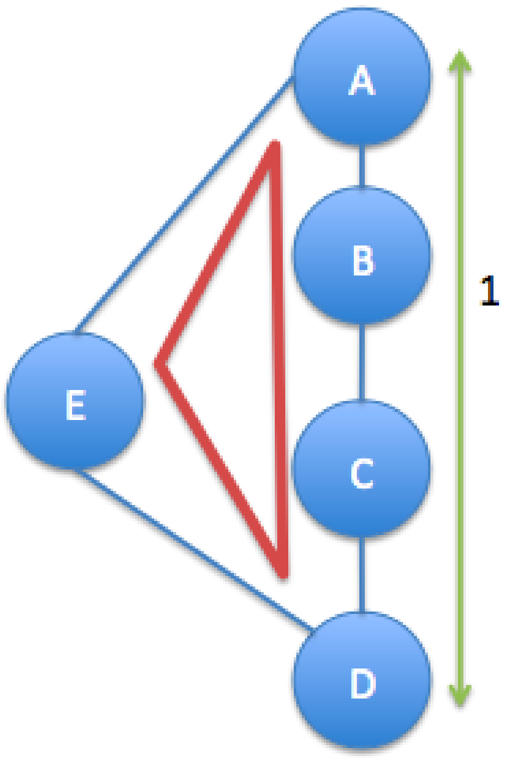


Figure 3.13: Straddling flow.

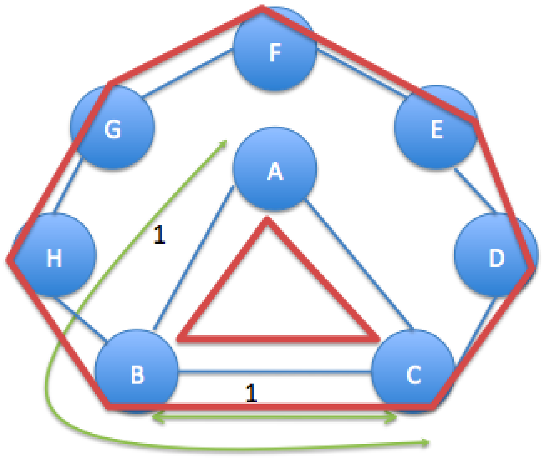


Figure 3.14: Straddling flow.

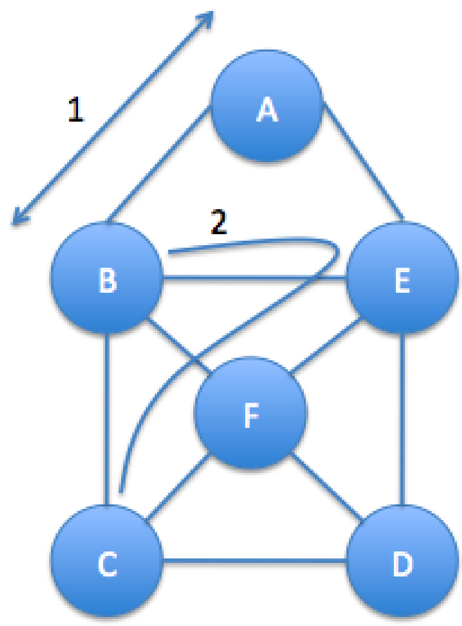


Figure 3.15: Straddling flow.

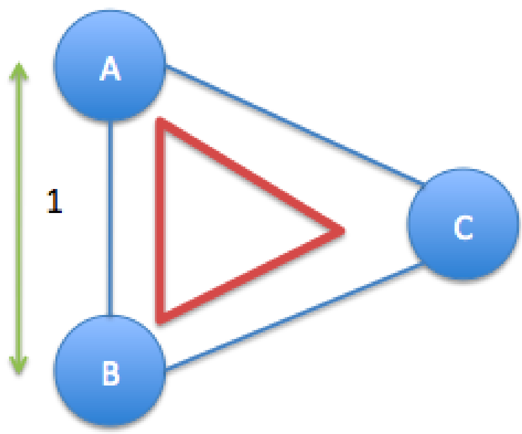


Figure 3.16: Traffic flows in a 3 node network.

**2** As in the previous case, the protection offered by a flow $p$-cycle is a superset of that of an FIPP $p$-cycle. This is because FIPP $p$-cycles are flow $p$-cycles with the added constraints that each flow must be protected end-to-end by a cycle, and that a cycle cannot protect two intersecting flows. Flow $p$-cycles can give more capacity efficient solutions as their search space is less constrained. In Figure 3.14, a (regular or differential) flow or span protecting $p$-cycle could protect flows ABC and BC using 2 copies of $p$-cycle ABC. A FIPP solution, however cannot protect both ABC and BC using cycle ABC as a $p$-cycle is not allowed to protect intersecting flows due to mutual capacity considerations. So On copy each of cycles ABC and BCDEFGH are required. This requires 10 units of spare capacity, as against 6 units in the previous case.

**3** FIPP $p$-cycles can be better than span protecting $p$-cycles when they are able to exploit straddling flow relationships. They can be less efficient than span protecting $p$-cycles due to the constraint of a $p$-cycle protecting only non-intersecting flows. As explained above, in Figure 3.14, a regular FIPP $p$-cycle solution will require 8 units of spare capacity. A differential FIPP solution will require 1 unit of spare capacity for differential $p$-cycle ABC, and 6 units of spare capacity for differential $p$-cycle BCDEFGH, or a total of 7 units of spare capacity. On the other hand, a regular span protecting $p$-cycle design requires only 6 units of spare capacity.

**4** Every candidate solution to the traditional $p$-cycle formulation is also a candidate solution for the DC $p$-cycle formulation. This is because the traditional $p$-cycle formulation requires a more constrained solution - it requires all on-cycle links to have the same amount of spare capacity, and straddling traffic has to be split evenly. Solutions that satisfy these constraints also satisfy the constraints of DC $p$-cycles, but there are solutions to the DC $p$-cycle formulation that don't satisfy the traditional $p$-cycle formulation as shown in Figure 3.4. It is these solutions that result in the reduction in spare capacity requirement over traditional $p$-cycles.

The traditional and DC $p$-cycle formulations have the same spare capacity requirement on networks with the same traffic demands on all links. This is because there is no difference in working traffic for the DC $p$-cycle to exploit.

**5, 6** Differential capacity span protecting $p$-cycles are more capacity efficient in cases where they can exploit the differential capacity property. But in the case of Figure 3.13, FIPP and flow $p$-cycles are require less spare capacity as they can exploit the straddling segment feature.

**7** Differential capacity flow $p$-cycles, as formulated in this thesis, can be inefficient when an optimal solution requires a $p$-cycle to protect different segments of the same flow with different amounts of spare capacity. When a differential capacity flow $p$-cycle protects a flow, it is constrained to allocate protection capacity for the maximal segment of the flow that intersects the cycle. For example, in Figure 3.15, an optimal solution would be to use 1 copy of $p$-cycle ABE to protect flow AB and 1 unit of segment BE of flow BEFC. Another 1 copy of $p$-cycle BCDE would protect 1 unit of segment BE and 2 units of segment EFC, which straddles the cycle. On the other hand, a differential capacity flow $p$-cycle formulation will have to protect 2 units of the maximal segment of flow BEFC, that, segment BEFC. So, 2 copies of $p$-cycle BCDE are required. In, addition, as in the previous case, 1 copy of $p$-cycle ABE is needed to protected flow AB.

**8** Differential capacity FIPP $p$-cycles are better than regular and differential capacity span protecting $p$-cycles when the $p$-cycle design is able to exploit the straddling flow relationship. As described previously, this can be seen in Figure 3.13.

Differential capacity FIPP $p$-cycles can be better than regular flow $p$-cycles when the benefit of differential capacity factor outweighs the cost of the additional constraints of an FIPP solution. For example, in Figure 3.16, when only flow ABC is present, a differential capacity FIPP $p$-cycle will need to reserve capacity only span BC. However, a regular FIPP $p$-cycle will require spare capacity along spans AB, BC, and AC.

Differential capacity FIPP $p$-cycles can never be better than differential capacity flow $p$-cycles. This is because differential capacity FIPP $p$-cycles, as formulated above, are essentially FIPP $p$-cycles without the restriction of end-to-end protection, and $p$-cycles protecting disjoint flows. So, every differential capacity FIPP solution is also a differential capacity $p$-cycle solution, but some differential capacity flow $p$-cycle solutions are not valid differential capacity FIPP solutions.

**9** Differential capacity flow $p$-cycles are always atleast as good as regular FIPP $p$-cycles. This is because, differential capacity flow $p$-cycles perform sub-optimally when a $p$-cycle is not allowed to protect different segments of a flow with different amounts of spare capacity, as described in Section 3.8. But FIPP $p$-cycles cannot take advantage of this as they are constrained to protect flows end-to-end. So, they can never do better than differential capacity flow $p$-cycles. On the other hand, differential capacity flow

$p$-cycles outperform regular FIPP $p$-cycles in cases where an optimal solution includes cycles protecting intersecting flows as shown in Figure 3.14.

# Chapter 4

# Results

## 4.1 Test Networks

The tests are carried out on the standard NSFNET and ARPA2, and some test networks shown in Figures 4.1-4.6. Details about the test networks are in Table 4.2.

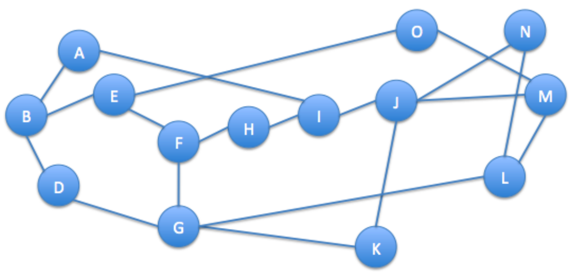


Figure 4.1: NSFNET network.

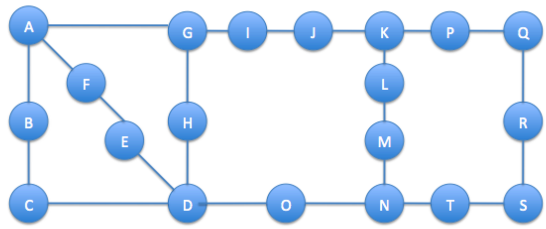


Figure 4.2: ARPA2 network.

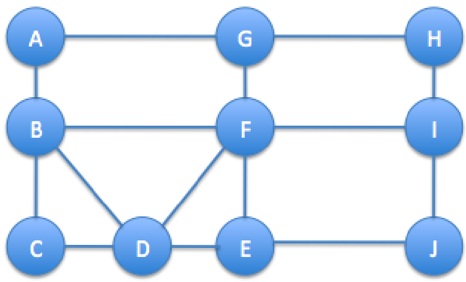


Figure 4.3: 10 node 17 span test network.

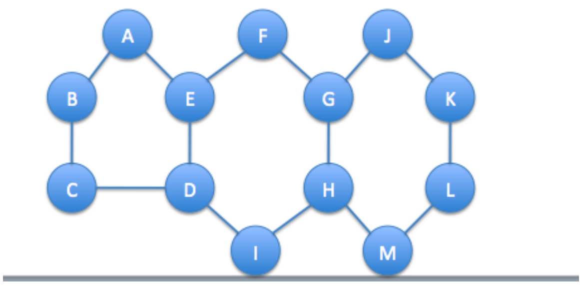


Figure 4.4: Rings test network.

Table 4.1: Test network

| Network | Nodes | Spans | Cycles |
|---|---|---|---|
| NSFNET | 14 | 21 | 139 |
| ARPA2 | 21 | 25 | 22 |
| 10n17s | 10 | 17 | 76 |
| Rings | 13 | 15 | 6 |
| OCT | 8 | 8 | 1 |
| K5 | 5 | 10 | 37 |

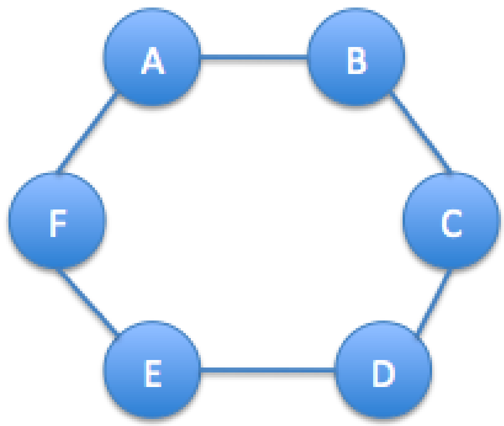


Figure 4.5: OCT test network.

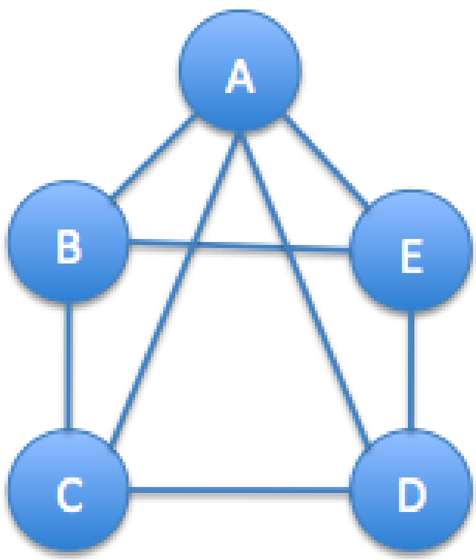


Figure 4.6: K5 test network.

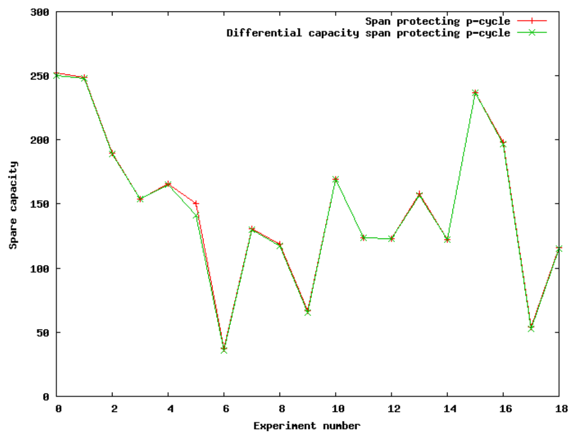


Figure 4.7: Differential capacity vs. traditional span protecting $p$-cycles on NSFNET.

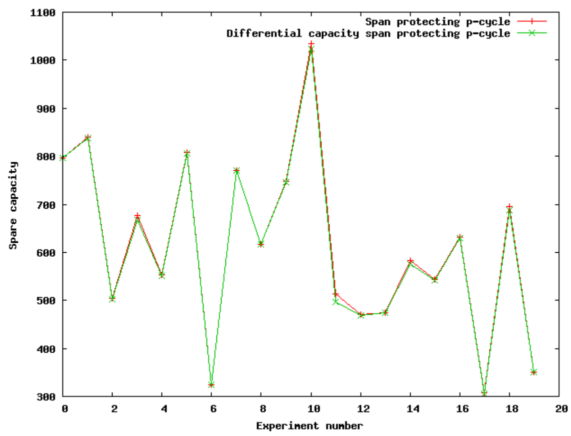


Figure 4.8: Differential capacity vs. traditional span protecting $p$-cycles on ARPA2.

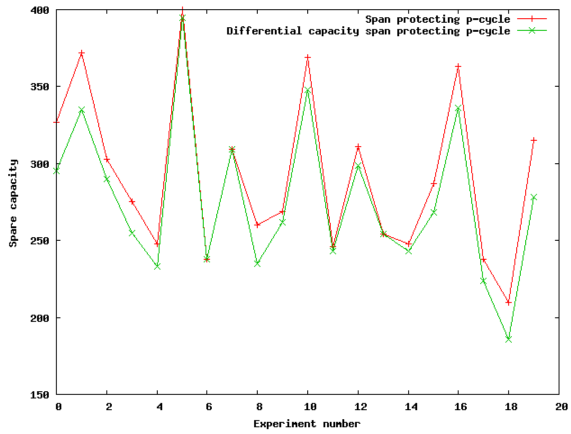


Figure 4.9: Differential capacity vs. traditional span protecting $p$-cycles on 10n17s.

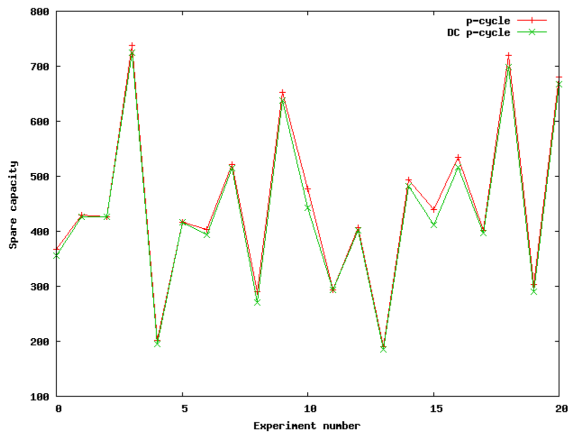


Figure 4.10: Differential capacity vs. traditional span protecting $p$-cycles on the ring test network.

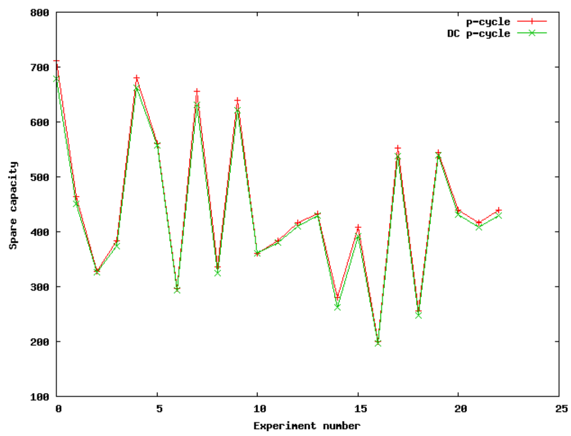


Figure 4.11: Differential capacity vs. traditional span protecting *p*-cycles on the OCT test network.

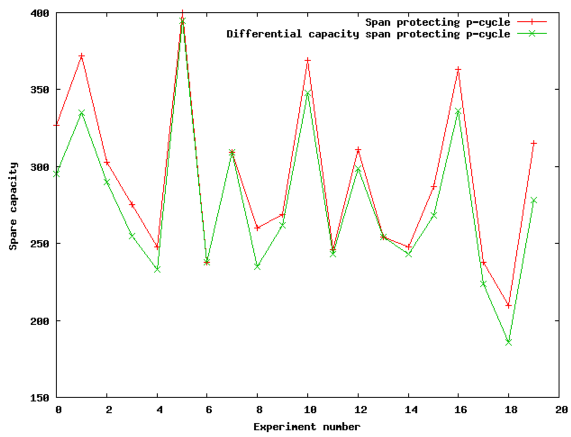


Figure 4.12: Differential capacity vs. traditional span protecting *p*-cycles on the K5 network.

Table 4.2: Test network

| Network | % spare capacity reduction | Average node degree | Number of cycles |
|---|---|---|---|
| NSFNET | 1.07 | 3.0 | 139 |
| ARPA2 | 0.53 | 2.38 | 22 |
| 10n17s | 5.34 | 3.4 | 76 |
| Rings | 2.6 | 2.3 | 6 |
| OCT | 2.28 | 2 | 1 |
| K5 | 1.67 | 4 | 37 |

## 4.2 Span protecting *p*-cycles

### 4.2.1 Comparison of capacity efficiency

The differential capacity span protecting *p*-cycles are always atleast as good as their traditional counterparts. Our test results in Figures 4.7-4.9 show that a small reduction in spare capacity when differential capacity span protecting *p*-cycles are used. When the largest traffic demand protected by *p*-cycle is an on-cycle demand, the differential capacity *p*-cycle spare capacity on this span alone. The amount of spare capacity saved is equal to the difference between the maximum and second maximum of of protected on-cycle working capacities. When the largest protected traffic demand is a straddling span, there is scope for capacity savings on multiple spans. However, this case occurs very infrequently.

This can be illustrated mathematically. Consider an $n$ node cycle in a network with $N$ nodes and $S$ spans. The total number of on-cycle links is $n$. Assuming a random network, the probability that a link exists between 2 nodes is $\frac{S}{\binom{N}{2}} = \frac{2 \cdot S}{N \cdot (N-1)}$. The number of straddling links is $\frac{2 \cdot S}{N \cdot (N-1)} \cdot \frac{n \cdot (n-3)}{2}$. Assuming random assignment of working traffic, the probability that the maximum working traffic is on a straddling link is equal to $\frac{\frac{2 \cdot S}{N \cdot (N-1)} \cdot \frac{n \cdot (n-3)}{2}}{\binom{n}{2}} = \frac{2S \cdot (n-3)}{N \cdot (N-1) \cdot (n-1)}$. For a 10 node cycle on NSFNET, this evaluates to about 18%. Further, a significant benefit will be seen only if the traffic on the straddling span is atleast twice that of the maximum on-cycle traffic. The probability of this occurring is clearly much lower.

Thus small gains can be obtained using differential capacity *p*-cycles with protecting switching requirements similar to traditional *p*-cycles.

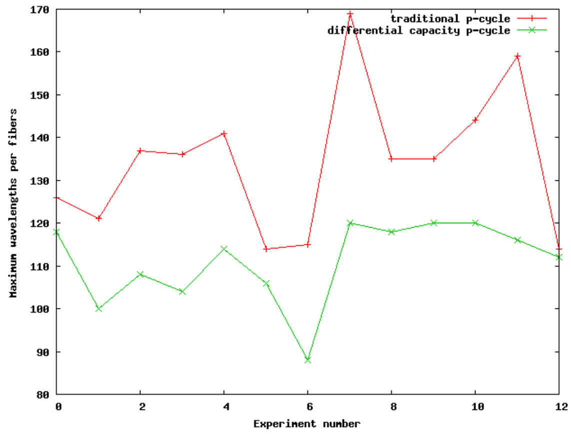


Figure 4.13: Maximum number of wavelengths per fiber using differential capacity vs. traditional span protecting *p*-cycles on 10n17s.

### 4.2.2 Maximum number of wavelengths per fiber

It is observed in Figure 4.13 that when using differential capacity *p*-cycles, the maximum number of wavelengths used per fiber on any span is lesser than or equal to the corresponding value when a traditional *p*-cycle formulation is used. This is important because, as explained in Section 3.3, the number of wavelengths available is limited, so a solution that uses fewer wavelengths is better. Secondly, if a fewer number of wavelengths are used, a optical switches with fewer ports can be used, thus reducing equipment cost.

## 4.3 Joint design

The joint version of the differential capacity span protecting *p*-cycle problem was implemented with with two alternate routes between each node pair. Figure 4.14 shows the result of the joint formulation as against a fixed routing on the 10n17s test network. The joint version gives better results as in the case of traditional *p*-cycles.

## 4.4 Heuristics for *p*-cycle preselection

Figures 4.15-4.17 show the increase in spare capacity when Topological Score (TS) and Demand Weighted Efficiency (DWE) [3] are used as metrics to rank *p*-cycles. We ob-

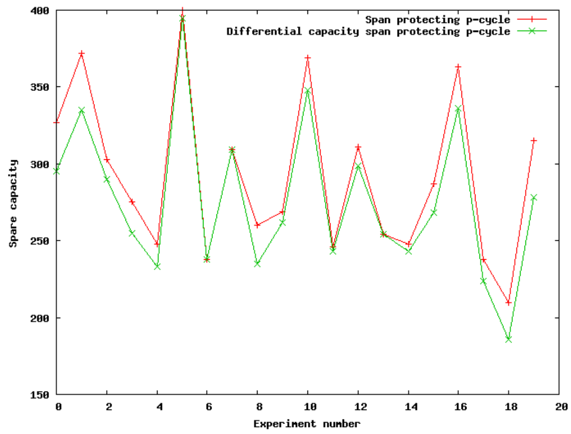


Figure 4.14: Differential capacity vs. traditional joint formulation of span protecting *p*-cycles on 10n17s.

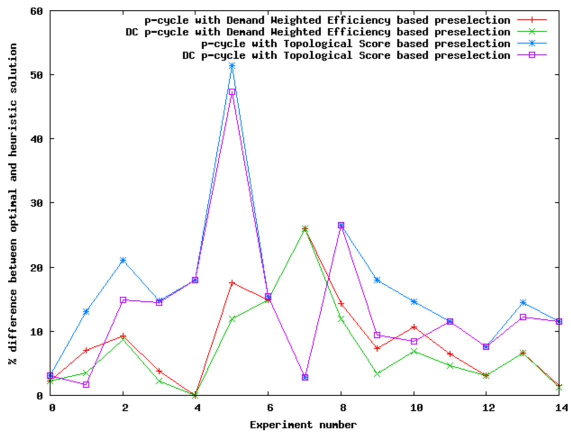


Figure 4.15: Topological Score and Demand Weighted Efficiency as preselection heuristics on NSFNET.

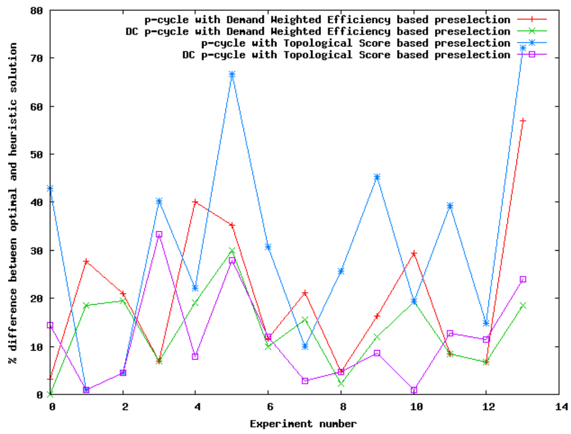


Figure 4.16: Topological Score and Demand Weighted Efficiency as preselection heuristics on Smallnet.

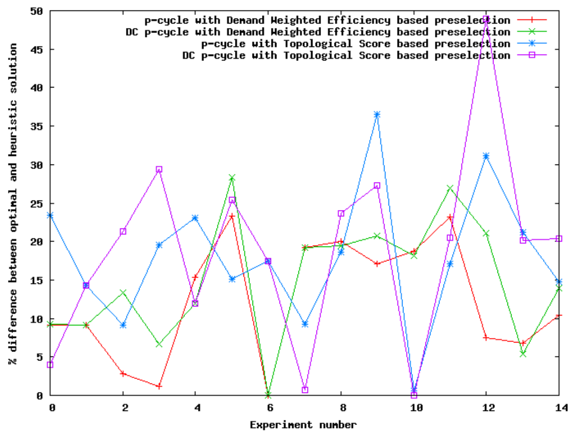


Figure 4.17: Topological Score and Demand Weighted Efficiency as preselection heuristics on 10n17s.

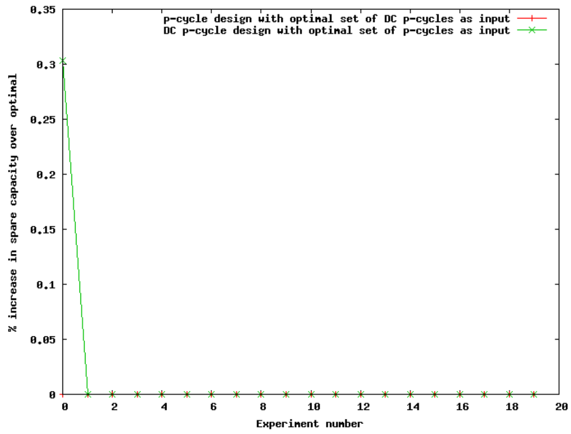


Figure 4.18: Percentage increase over optimal solution when optimal set of $p$-cycles for traditional $p$-cycle formulation are given as input to differential capacity $p$-cycle ILP and vice versa on ARPA2 network.

serve that TS and AE are good candidate cycle preselection metrics for differential capacity $p$-cycles as well.

For a set of random inputs, we gave the optimal set of $p$-cycles as the input set of cycles to the differential capacity $p$-cycle as input, and vice versa as shown in Figures 4.18-4.19. It can be seen that the difference between the resulting spare capacity and optimal solution on complete set of $p$-cycles is very small.

We can conclude that the optimal set of cycles for the traditional $p$-cycle formulation, is close to optimal for the differential capacity $p$-cycle formulation, and vice versa. Hence, a good heuristic for traditional $p$-cycles is likely to perform well when applied to differential capacity $p$-cycles.

## 4.5 Flow $p$-cycles

Figures 4.21-4.23 show the result of using differential capacity versus flow $p$-cycles on the 3 test networks. Larger reduction in spare capacity requirement are observed here than in the span protecting case, as there is more scope for savings. In this case, a spare capacity reduction is possible on multiple spans of the same $p$-cycles. Also, this reduction is not limited to the difference between maximum and second maximum on-cycle working capacities as in the span protecting case.

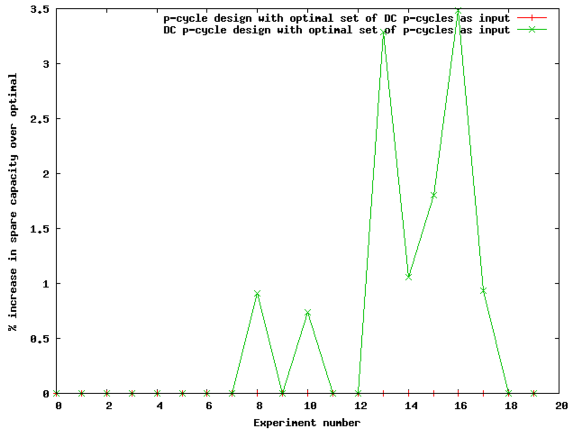


Figure 4.19: Percentage increase over optimal solution when optimal set of *p*-cycles for traditional *p*-cycle formulation are given as input to DC *p*-cycle ILP and vice versa on rings test network.

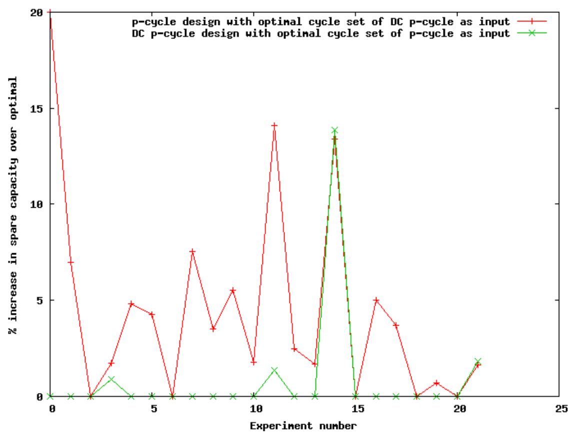


Figure 4.20: Percentage increase over optimal solution when optimal set of *p*-cycles for traditional *p*-cycle formulation are given as input to DC *p*-cycle ILP and vice versa on 10n17s network.

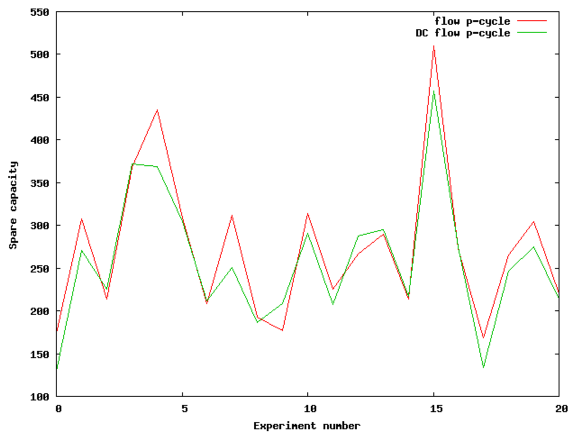


Figure 4.21: Differential capacity vs. traditional flow $p$-cycles on NSFNET.

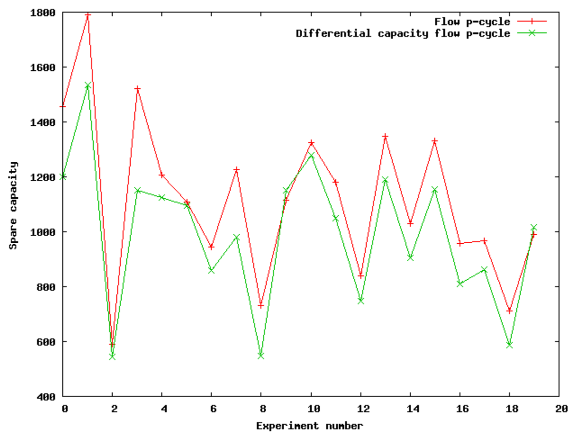


Figure 4.22: Differential capacity vs. traditional flow $p$-cycles on ARPA2.

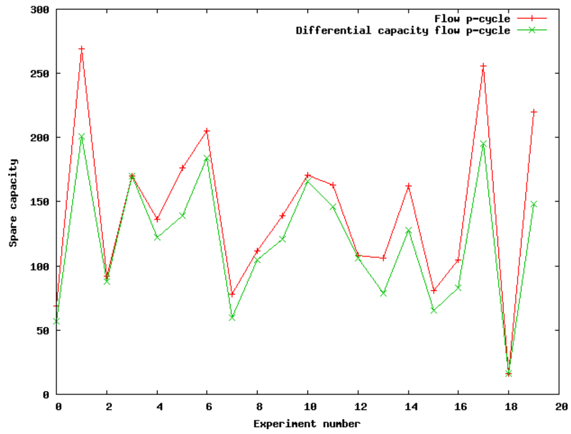


Figure 4.23: Differential capacity vs. traditional flow $p$-cycles on 10n17s.

## 4.6 FIPP $p$-cycles

Figures 4.24-4.25 show the result of using differential capacity versus flow $p$-cycles on the 3 test networks. In this case as well, there are is a larger spare capacity reduction than in the span protecting $p$-cycle case for the same reasons outlined in the previous section.

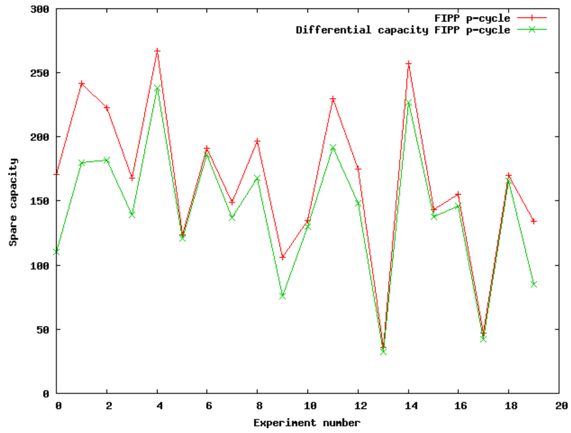


Figure 4.24: Differential capacity vs. traditional FIPP $p$-cycles on NSFNET.

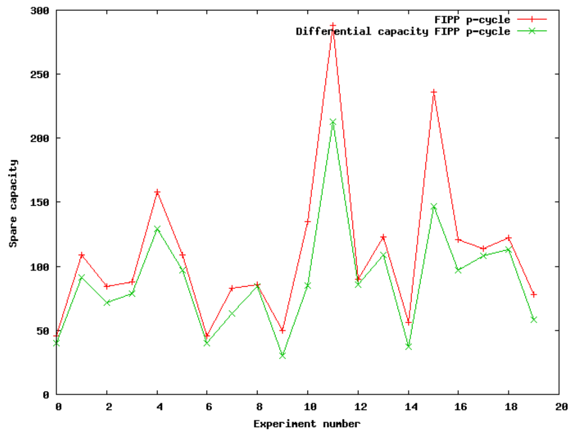


Figure 4.25: Differential capacity vs. traditional FIPP $p$-cycles on 10n17s.

# Bibliography

[1] J. Doucette, W.D. Grover, and P.A. Giese. Physical-layer p-cycles adapted for router-level node protection: a multi-layer design and operation strategy. *IEEE J. Sel. Areas Commun. (USA)*, 25(5):963 – 73, 2007/06/.

[2] John Doucette, Peter A. Giese, and Wayne D. Grover. Combined node and span protection strategies with node-encircling p-cycles. volume 2005, pages 213 – 221, Island of Ischia, Naples, Italy, 2005.

[3] Wayne D. Grover. *Mesh-based Survivable Networks*, chapter 10. Prentice Hall, 2003.

[4] Wayne D. Grover and Demetrios Stamatelakis. Cycle-oriented distributed preconfiguration: Ring-like speed with mesh-like capacity for self-planning network restoration. volume 1, pages 537 – 543, Atlanta, GA, USA, 1998.

[5] W.D. Grover and A. Grue. Self-fault isolation in transparent p-cycle networks: p-cycles as their own m-cycles. *IEEE Commun. Lett. (USA)*, 11(12):1004 – 6, 2007/12/.

[6] W.D. Grover and Gangxiang Shen. Extending the p-cycle concept to path-segment protection. volume vol.2, pages 1314 – 19, Anchorage, AK, USA, 2003//.

[7] Wensheng He, Jing Fang, and Arun K. Somani. A p-cycle based survivable design for dynamic traffic in wdm networks. volume 4, pages 1869 – 1873, 2005.

[8] A. Kodian and W.D. Grover. Multiple-quality of protection classes including dual-failure survivable services in p-cycle networks. volume Vol. 1, pages 231 – 40, Boston, MA, USA, 2005//.

[9] A. Kodian, A. Sack, and W.D. Grover. p-cycle network design with hop limits and circumference limits. pages 244 – 53, San Jose, CA, USA, 2004//.

[10] Adil Kodian and Wayne D. Grover. Failure-independent path-protecting p-cycles: Efficient and simple fully preconnected optical-path protection. *Journal of Lightwave Technology*, 23(10):3241 – 3259, 2005.

[11] C. Liu and L. Ruan. p-cycle design in survivable wdm networks with shared risk link groups (srlgs). *Photonic Netw. Commun. (USA)*, 11(3):301 – 11, 2006/05/.

[12] Diane Prisca Onguetou and Wayne D. Grover. Approaches to p-cycle network design with controlled optical path lengths in the restored network state. *Journal of Optical Networking*, 7(7):673 – 691, 2008.

[13] A. Sack and W.D. Grover. Hamiltonian p-cycles for fiber-level protection in semi-homogeneous homogeneous and optical networks. *IEEE Netw. (USA)*, 18(2):49 – 56, 2004/03/.

[14] Gangxiang Shen and Wayne D. Grover. Automatic lightpath service provisioning with an adaptive protected working capacity envelope based on p-cycles. volume 2005, pages 375 – 383, Island of Ischia, Naples, Italy, 2005.

[15] Gangxiang Shen and Wayne D. Grover. Design and performance of protected working capacity envelopes based on p-cycles for dynamic provisioning of survivable services. *Journal of Optical Networking*, 4(7):361 – 390, 2005.

[16] D. Stamatelakis and W.D. Grover. Theoretical underpinnings for the efficiency of restorable networks using preconfigured cycles. *IEEE Trans. Commun. (USA)*, 48(8):1262 – 5, 2000/08/.

[17] D. Stamatelakis and W.D. Grover. Ip layer restoration and network planning based on virtual protection cycles. *IEEE J. Sel. Areas Commun. (USA)*, 18(10):1938 – 49, 2000/10/.

[18] Hongxia Wang and H.T. Mouftah. P-cycles in multi-failure network survivability. volume 1, pages 381 – 384, Barcelona, Catalonia, Spain, 2005.

[19] Bin Wu and Kwan L. Yeung. Monitoring cycle design for fast link failure detection in all-optical networks. pages 2315 – 2319, Washington, DC, United States, 2007.