Scalable Fault Management Architecture for Dynamic Optical Networks: An Information-Theoretic Approach

by

Yonggang Wen

B.Eng., Electrical Engineering and Information Science
University of Science and Technology of China, 1999
M.Phil., Information Engineering
The Chinese University of Hong Kong, 2001

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2008

Author ....
Department of Electrical Engineering and Computer Science
February 29, 2008

Certified by ....
Vincent W.S. Chan
Joan and Irwin Jacobs Professor of Electrical Engineering and
Computer Science, and Aeronautics and Astronautics
Thesis Supervisor

Accepted by ....
Terry P. Orlando
Chairman, Department Committee on Graduate Students

Scalable Fault Management Architecture for Dynamic Optical Networks: An Information-Theoretic Approach

by

Yonggang Wen

Submitted to the Department of Electrical Engineering and Computer Science
on February 29, 2008, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

All-optical switching, in place of electronic switching, of high data-rate lightpaths at intermediate nodes is one of the key enabling technologies for economically scalable future data networks. This replacement of electronic switching with optical switching at intermediate nodes, however, presents new challenges for fault detection and localization in reconfigurable all-optical networks. Presently, fault detection and localization techniques, as implemented in SONET/G.709 networks, rely on electronic processing of parity checks at intermediate nodes. If similar techniques are adapted to all-optical reconfigurable networks, optical signals need to be tapped out at intermediate nodes for parity checks. This additional electronic processing would break the all-optical transparency paradigm and thus significantly diminish the cost advantages of all-optical networks.

In this thesis, we propose new fault-diagnosis approaches specifically tailored to all-optical networks, with an objective of keeping the diagnostic capital expenditure and the diagnostic operation effort low. Instead of the aforementioned passive monitoring paradigm based on parity checks, we propose a proactive lightpath probing paradigm: optical probing signals are sent along a set of lightpaths in the network, and network state (i.e., failure pattern) is then inferred from testing results of this set of end-to-end lightpath measurements. Moreover, we assume that a subset of network nodes (up to all the nodes) is equipped with diagnostic agents - including both transmitters/receivers for probe transmission/detection and software processes for probe management to perform fault detection and localization. The design objectives of this proposed proactive probing paradigm are two folded: i) to minimize the number of lightpath probes to keep the diagnostic operational effort low, and ii) to minimize the number of diagnostic hardware to keep the diagnostic capital expenditure low.

The network fault-diagnosis problem can be mathematically modeled with a group-testing-over-graphs framework. In particular, the network is abstracted as a graph in which the failure status of each node/link is modeled with a random variable (e.g., Bernoulli distribution). A probe over any path in the graph results in a value, defined as the probe syndrome, which is a function of all the random variables associated in that path. A network failure pattern is inferred through a set of probe syndromes resulting from a set of optimally chosen probes. This framework enriches the traditional group-testing problem by introducing a topological structure, and can be extended to model many other network-monitoring problems (e.g., packet delay, packet drop ratio, noise and etc) by choosing appropriate state variables.

Under the group-testing-over-graphs framework with a probabilistic failure model, we initiate an information-theoretic approach to minimizing the average number of lightpath probes to identify all possible network failure patterns. Specifically, we have established an isomorphic mapping between the fault-diagnosis problem in network management and the source-coding problem in Information Theory. This mapping suggests that the minimum average number of lightpath probes required is lower bounded by the information entropy of the network state and efficient source-coding algorithms (e.g., the run-length code) can be translated into scalable fault-diagnosis schemes under some additional probe feasibility constraint. Our analytical and numerical investigations yield a guideline for designing scalable fault-diagnosis algorithms: each probe should provide approximately 1-bit of state information, and thus the total number of probes required is approximately equal to the entropy of the network state.

To address the hardware cost of diagnosis, we also developed a probabilistic analysis framework to characterize the trade-off between hardware cost (i.e., the number of nodes equipped with Tx/Rx pairs) and diagnosis capability (i.e., the probability of successful failure detection and localization). Our results suggest that, for practical situations, the hardware cost can be reduced significantly by accepting a small amount of uncertainty about the failure status.

Thesis Supervisor: Vincent W.S. Chan
Title: Joan and Irwin Jacobs Professor of Electrical Engineering and Computer Science, and Aeronautics and Astronautics

Acknowledgments

In finishing my PhD thesis at MIT, I have incurred debts to many individuals, who have changed my life profoundly.

First and foremost, I would like to express my greatest gratitude to my thesis advisor, Professor Vincent W.S. Chan. This thesis (and my life) would not have been possible without his patience, guidance, inspiration and support. His emphasis on critical-thinking processes, communication skills and leadership has helped me through my PhD study and will be invaluable to my future career development.

I would also like to thank my thesis committee, Professor Muriel Medard, Mr. Eric Swanson and Mr. Naimish Patel. Their advice and constructive critics greatly improve the quality of this thesis. Moreover, this thesis cannot be finished without Eric's encouragement and the countless number of hours that he has spent with me to discuss my work.

During my PhD research, Professor Lizhong Zheng at MIT has been a great source of help and wisdom. His insights and coolness have always come to my rescue when I felt helpless. Dr. Terrence McGarty has also helped me tremendously by taking me out to the real world in his start-up adventure and sharing me with his experience and wisdom.

I am grateful to DARPA and NSF for the financial support to this research.

My life would have been different without help from a lot of my peers at MIT. My officemate, Guy Weichenberg, has generously spent numerous hours to help me with my research and life. His knowledge and patience have always been the first resource of help. Dr. Desmond Lun has been a great friend in all aspects. Lillian Dai has spent numerous hours to perfect my presentations. Moreover, my seven-year life at MIT is colorful with ”Chan's kids”(first at 35-438 and then at 32-D678). Anarupa, Andrew, Etty, Guy, James, Kyle, Lillian, Lizzy, Nick, Patrick, Roop and Serena, thank you all for making my life enjoyable at MIT. My gratitude also goes to all the friends along my way toward my PhD at MIT, for their companion and support.

I would like to thank all the staff in LIDS headquarters, whose diligence and help have made my life at MIT much easier.

I cannot end without thanking my parents and sister, for their unselfish love and support on which I can always rely throughout the ups and downs of my life.

Contents

List of Figures

List of Tables

Chapter 1

Fault Management Architecture

Owing to the recent explosion in internet traffic[58, 44, 19], optical fiber, with its vast transmission bandwidth ($\sim$35THz) [50], has emerged as the only realistic transmission medium for backbone networks. Moreover, all-optical networks [25], where data traverses lightpaths without any optical-to-electrical conversion, will be increasingly prevalent in future broadband networks as a result of its expected lower cost and full transparency to different signal formats and protocols. However, as in the case of other networks, all-optical networks are vulnerable to physical failures such as fiber cuts, switch node failures, optical amplifiers and transceivers breakdowns. These failures can lead to costly disruptions in communication, and their detection and localization can constitute a significant fraction of reoccurring network operating costs. To ensure specified levels of quality of service at an affordable cost, an efficient network management system - including scalable fault management capability - should be in place when all-optical networks are fully deployed in future. In this thesis, we focus on developing scalable fault management architecture, including fault-diagnosis algorithms that detect and localize failures in the optical layer and network survivability designs that provide robustness against network failures, for all-optical networks.

This chapter highlights a generic framework for fault management in all-optical networks. We first present a high-level fault-management architecture. Next, we elaborate on two crucial functions in fault management, i.e., fault diagnosis and network survivability. Specifically, in the context of fault diagnosis, we will propose a class of proactive fault-diagnosis schemes and characterize a trade-off between the diagnosis effort and the diagnosis delay with motivational examples. In the context of network survivability, we will compare performance of two alternative mechanisms (i.e., protection switching and lightpath diversity).

1.1 Fault Management System

1.1.1 Fault Management in Network Management Framework

For any network, network management system (NMS) is crucial to ensure efficient and continuous operations of the network, such that users of the network receive network services with the quality of service that they expect. This objective is achieved through five management functions provided by the NMS (as specified in [26]): fault management, configuration management, performance management, security management and account management. Fault management is responsible for detecting failures when they happen, identifying the faulty components, and restoring traffic that may be disrupted due to the failures. Configuration management deals with the set of functions associated with managing orderly changes in a network, including equipment management, connection management and adaptation management. Performance management deals with monitoring and managing the various parameters that measure the performance of the network. Security management covers a very broad range of security including physically securing the network, as well as controlling access to the network by the users. Account management is the function responsible for billing and for developing lifetime histories of the network components. By providing these five management functions, network management is also understood as OAM&P (Operations, Administration, Maintenance and Provisioning) [59].

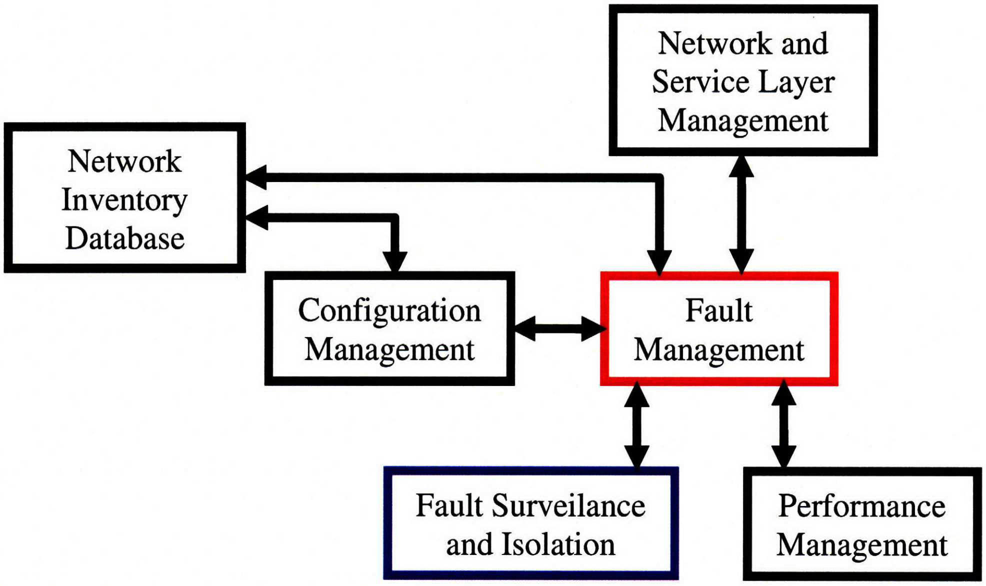

Network and Service Layer Management
Network Inventory Database
Configuration Management
Fault Management
Fault Surveilance and Isolation
Performance Management

Figure 1-1: Network management function map: fault management is the center of network management. (Adapted from [41])

Among all these five management functions, fault management serves as the hub of all these functions because the fault detection and localization subsystem (or the fault surveillance subsystem) provides information for other management functions [41]. In an optical network, the fault surveillance subsystem is responsible for monitoring the operation condition of each component, detecting the loss-of-light condition for fiber links, reporting these fault conditions to the fault management module. Network state information, acquired through the fault surveillance subsystem, is also forwarded via the fault management module to the configuration management module and the performance management module (as illustrated in Fig. 1-1). These modules then analyze the acquired fault conditions and use that information to update the network database that contains entries of each component in the network.

1.1.2 Fault Management Architecture

In this subsection, we present a generic system architecture for fault management in future all-optical networks.

To deal with failures in any network, fault management is normally expected to include the following five functionalities [59]:

1. Fault detection, which detects faults as quickly as possible, preferably before or at about the same time as users would notice it.

2. Fault localization, which identifies where the problem has occurred.

3. Service restoration, which reroutes the disrupted lightpaths to protecting lightpaths.

4. Identification of problem's root cause, which traces back the root cause of the problem.

5. Problem solution, which issues a trouble ticket[1] for the problem and initiates a process to fix the problem automatically or manually.

In Fig. 1-2, we illustrate a system architecture that provides the aforementioned five functionalities for fault management in all-optical networks. It normally includes three modules: a network element (NE) module, a data communication network (DCN) module and a network management system (NMS) module. In some cases, when network elements from multiple vendors are deployed, a tier of element management systems (EMS) are inserted between the network element modules and the network management system module, with the objective to provide a universal interface to the network management system.

Each network element module can be decoupled into two components: a hardware element under surveillance and a software agent to conduct specified management functions. In particular, the software agent is responsible for acquiring network state

[1] A trouble ticket is a notification to network engineers about information regarding to network problems.

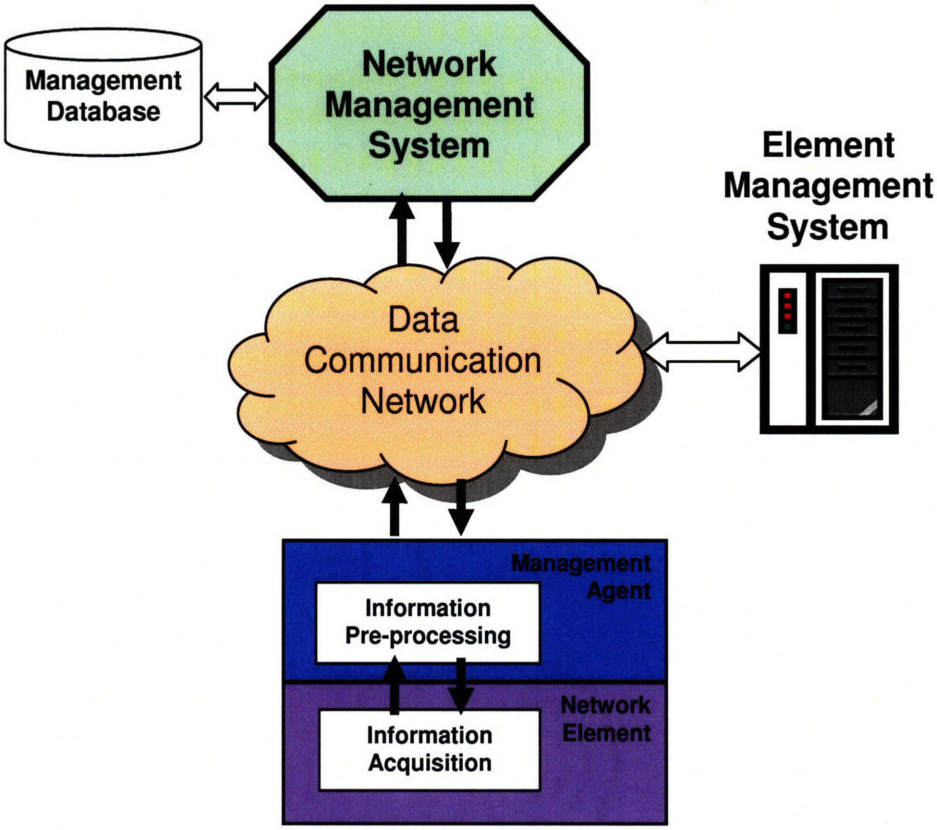

Management Database
Network Management System
Element Management System
Data Communication Network
Management Agent
Information Pre-processing
Network Element
Information Acquisition

Figure 1-2: Network management system architecture: a network element module and a network management module are interconnected by a data communication network module.

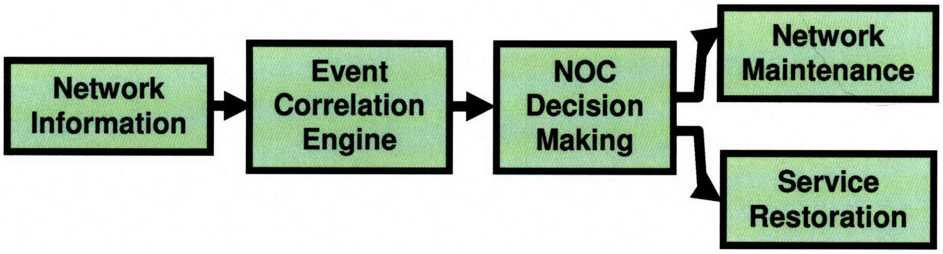


Figure 1-3: The work flow of the network management system: it takes input via the data communication network and makes NOC decisions.

information from the hardware element and pre-processing it before forwarding it to the network management system via the data communication network.

The data communication network connects the set of network element modules, the network management system module, and the set of element management modules if possible. For today's optical networks, it has been implemented in several ways [50]:

1. through a separate out-of-band network outside the optical layer;
2. through the optical supervisory channel (OSC) on a separate wavelength;
3. through the rate-preserving or digital wapper in-band optical channel layer overhead techniques.

The network management system, which usually resides in a centralized network operation center (NOC), is the brain of all the network management functions. Its work flow for fault management is illustrated in Fig. 1-3. Network state information, obtained through the data communication network from network element modules, is first fed into an event correlation engine. The event correlation engine then employs different techniques (to be explained later) to localize faults in networks and identify the root causes of those problems. The output of the event correlation engine is finally used to make network operating center decisions, such as network maintenance schedules that fix network problems, or service restoration decisions that dynamically re-route disrupted network traffics to other protecting lightpaths.

The event correlation engine employs different event correlation techniques for (i)detecting and filtering of events, (ii) correlating observed events to isolate and localize the fault either topologically or functionally, and (iii) identifying the cause of problems. Roughly speaking, existing correlation techniques can be classified into the following six categories[40]:

Rule-Based Reasoning This technique contains three components: a working memory, an inference engine and a knowledge base. The knowledge base contains knowledge as to (1) definition of a problem in the network and (2) action that needs to be taken if a particular condition occurs. The knowledge base is rule-based in the form of *if-then* or *condition-action*, containing rules that indicate which operations are to be performed when. The working memory contains the topological and state information of the network being managed, and recognizes when the network goes into a faulty state. The inference engine, in cooperation with the knowledge base, compares the current state with the left side of the rule-base and finds the closest match to output the right side of rule. The knowledge base then executes an action on the working memory.

Model-Based Reasoning This technique refers to an inference method based on a model of the physical world. In particular, an object-oriented software model is created for each managed network elements, and the relationship between network element objects is reflected in a similar relationship between models. The interaction between the physical network and the software model gives the opportunity to identify problems in the physical network.

Case-Based Reasoning This technique is similar to the rule-based reasoning, with the exception that the unit of knowledge is case (i.e., previously solved problem). The intuition of case-based reasoning is that situations repeat themselves in the real world and that what was done in one situation is applicable to others in a similar, but not necessarily identical, situation. It consists of four modules: input, retrieve, adapt and process, along with a case library. This approach uses knowledge, which is gained previously and stored in the case library, and extend it to the current situation.If the current situation, as received by the input module, matches one in the case library (as identified by the retrieved module), it is applied. If it does not, the closest situation is chosen by the adapt module and adapted to the current situation to solve the problem. The process module take the appropriate actions. Once the problem has been resolved, the newly adapted case is added to the library.

Codebook Correlation Model In this approach, problem events are viewed as messages generated by a system and encoded in set of alarms that they cause. The function of the correlator is to decode those messages to identify the problem. It follows that the coding technique has two phases. In the first phase, called the codebook selection phase, problems to be monitored are identified and the symptoms or alarms that each of them generates are associated with the problem. This phase results in a problem-symptom matrix. In the second phase, the correlator compares the streams of alarm events with the codebook and identifies the problem.

State transition graph model In this approach, a state transition graph is created to correlate events in a network. An action or a response from some previous action would change the state. If there is a problem in the system, we will arrive at a node in the graph that indicates a failure.

Finite state machine model It is a passive testing system based on the assumption that an observe agent is present in each node and reports abnormality to a central point. A failure in a node or a link is indicated by the state machine associated with the component entering an illegal state. A similarity between the finite state machine model and the state transition model is the state transitions. The main difference between them is that the former is a passive system and the latter is an active system.

As it will be explained later, the proactive fault diagnosis scheme developed in this thesis has properties of the codebook correlation model and the state transition

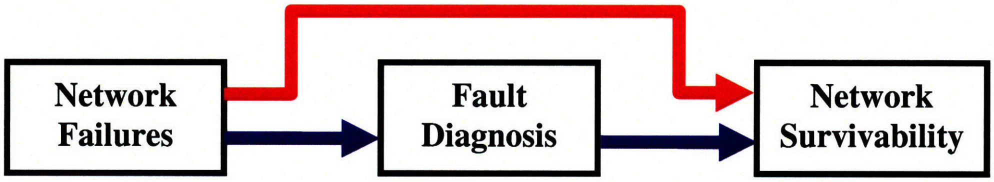


Figure 1-4: A simple view of the fault management system for optical networks includes two crucial functions to deal with network failures: fault diagnosis and network survivability. Fault diagnosis is responsible for acquiring network failure state information, and network survivability is responsible for maintaining the quality of network services with/without network failure state information.

graph model. Indeed, it can be considered as the state transition graph model with an online codebook generation process.

1.1.3 A Simple View: Fault Diagnosis and Network Survivability

Intuitively, when network failures happen, two tasks need to be performed in order to maintain the quality of service. The first task is to identify the failures. The second task is to design transport schemes that are robust, even without network failure information. Therefore, in this thesis, we take a simple view of the fault management system, focusing on two crucial functions to deal with network failures: fault diagnosis and network survivability, as illustrated in Fig. 1-4.

The failure diagnosis module is responsible for detecting and localizing network failures. In addition, fault diagnosis schemes should be designed with low overhead, short delay, and/or high accuracy. However, simultaneously optimizing these metrics generally is not possible, and thus trade-offs among them should be struck. In Section 1.2, we illustrate one trade-off between the diagnostic effort and the diagnostic delay, with some motivational example.

The network survivability module is responsible for maintaining the required quality of service, via service restoration or other mechanisms, in the event of network failures in all-optical networks. Survivability mechanisms could be either re-active (i.e., acting upon network failure state information) or passive (i.e., acting without network failure state information). Design alternatives will be presented in Section 1.3.

1.2 Scalable Fault Diagnosis Architecture

All-optical networks promise significant cost benefits, mainly due to optical switching of high data-rate lightpaths at intermediate network nodes, thereby reducing electronic processing costs. However, the absence of electronic processing capability at intermediate nodes results in challenges to fault detection and localization, which previously relies on the electronic processing capability at intermediate nodes(e.g., parity check bits in SONET/G.709 networks). It follows that, for all-optical networks, either optical signal is tapped out at each intermediate node for parity check or new mechanisms are needed to diagnose link/node failures. If tapping out signals were to be done, a lot of cost benefit of all-optical networks would be negated. In this thesis, we seek to develop scalable fault-diagnosis schemes for all-optical networks, by exploiting the unique property that optical signals are carried over lightpaths without being detected at intermediate nodes.

1.2.1 Two Alternative Fault Diagnosis Paradigms: Passive Monitoring vs. Proactive Probing

A fault-diagnosis system can be decoupled into three cascaded modules: (1) a network state information acquisition module, (2) a network state information transportation module and (3) a network state information processing module, as illustrated in Fig. 1-5. Each module provides its specific functionality for fault diagnosis. The network state information acquisition module is responsible for collecting information about internal network states, such as, optical power level, noise level, and etc, in the optical layer. The network state information transportation module transfers the network

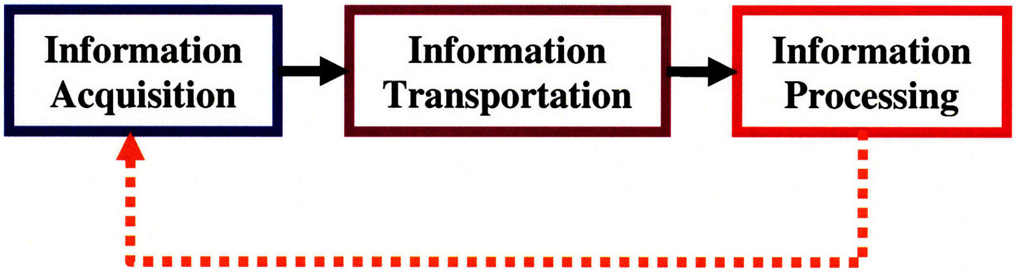


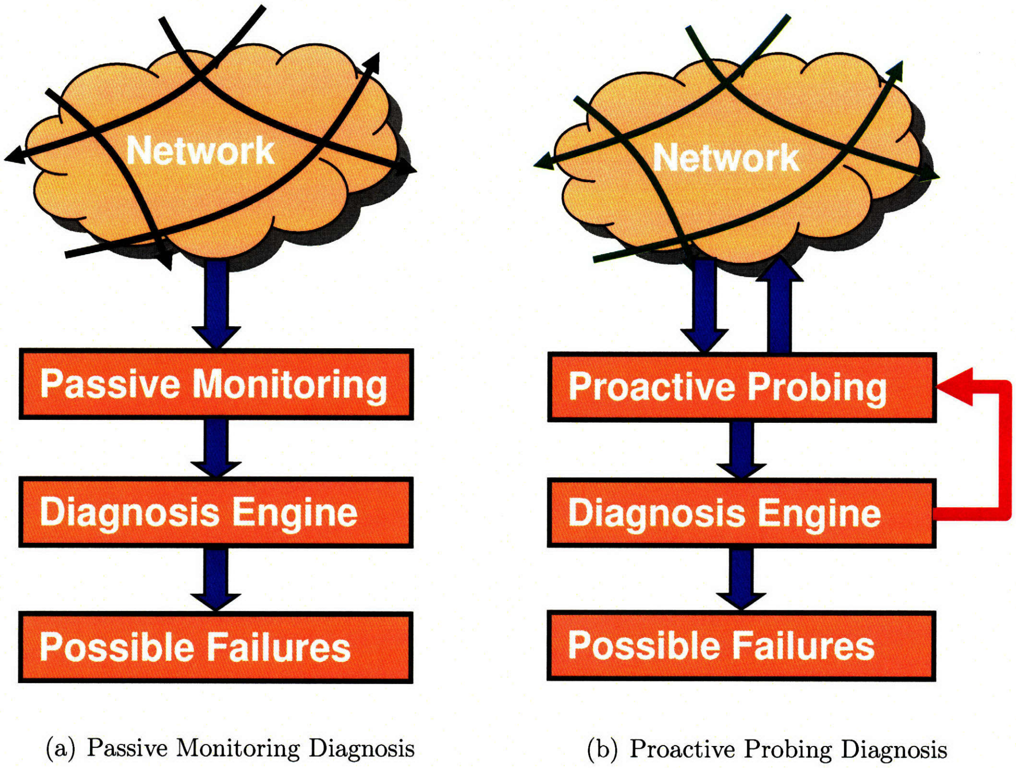
Figure 1-5: Fault diagnosis architecture. The fault-diagnosis system can be decoupled into three cascaded modules: network state information acquisition module, network state information transportation module and network state information processing module.

state information obtained at the acquisition module to the network state information processing module, which could be located at some centralized agent or a group of distributed agents. The network state processing module is responsible for analyzing the collected network state information to detect and localize possible failures.

The relationship among these three modules are illustrated in Fig. 1-5. In addition to the information flow from the acquisition module to the processing module via the transportation module, the feedback control from the processing module to the acquisition module provides an additional dimension to design a scalable fault surveillance system, which is one of the key sub-systems in a network management system [41].

Based on how network state information is acquired in the network state acquisition module, fault-diagnosis schemes can be classified roughly into two categories: a passive monitoring diagnosis paradigm and a proactive probing diagnosis paradigm[2], as illustrated in 1-6. In the passive-monitoring diagnosis paradigm (see Fig. 1-6(a)), network state information is acquired via passively monitoring existing traffics. In the proactive-probing diagnosis paradigm (see Fig. 1-6(b)), networks state information is acquired via proactively measuring optical probing signals.

[2] A third category might exist to combine both approaches by acquiring network state information through passively monitoring network traffics and proactively sending probing signals.



(a) Passive Monitoring Diagnosis

(b) Proactive Probing Diagnosis

Figure 1-6: Two alternative fault diagnosis paradigms based on the information acquisition mechanisms: a passive monitoring diagnosis paradigm vs. a proactive probing diagnosis paradigm. In the passive monitoring diagnosis paradigm, network state information is acquired through monitoring the existing traffics. In the proactive probing diagnosis paradigm, network state information is acquired through measurements of optical probing signals.

In current systems, the passive-monitoring fault-diagnosis paradigm has been deployed in SONET/G.709 networks, where network failures are identified by verifying the parity bits embedded in the overhead of data frames [50]. This approach is illustrated in Fig. 1-6(a). The passive monitoring module generates the events - alarms, warnings, parameters of network elements - as inputs to the fault-diagnosis engine. Using various inference algorithms or event-correlation techniques mentioned in Section 1.1.2 (e.g., neural networks [51] and Finite-state Machines [3]), the fault-diagnosis engine identifies a set of network elements whose failures may have caused the input events. Similar approach has also been proposed in [32] to diagnose network failures from network coding overhead bits[3]Because the monitoring module is decoupled from the fault-diagnosis engine, network architect can follow a "divide-and-conquer" approach in designing different modules separately, and thus reducing design complexity. In addition, the diagnosis scheme can leverage existing traffic, without incurring additional diagnosis traffic. However, the absence of feedback from the diagnosis engine to the monitoring module could entail tremendous inefficiency in fault-diagnosis process. For example, one single failure could trigger a large number of redundant alarms, all of which are fed into the fault-diagnosis engine. Combined with the network growth and faster switching speed, the redundancy in the input events could generate a large amount of management information. It can consume a fair amount of network source to transfer and store this large amount of management information, and thus limits its scalability in future all-optical networks. To make matters worse, because all measurements are piggybacked onto real traffic, the state information of infrequently used links might be obsolete when they are accessed. This could cause serious problems in some real-time applications with critical time deadlines [55], especially for dynamic all-optical networks.

Motivated by these shortcomings of the passive monitoring diagnosis paradigm, we focus in this thesis on proactive probing diagnosis schemes where optical probing signals are sent along a set of lightpaths and network failures are inferred through

[3] In network coding schemes, the coefficients used at intermediate nodes to linearly combine all the inputs are sent along the data to the destination. The destination can thus infer internal network states (e.g., link failures) by looking into the set of missing coefficients.

probing results of this set of lightpaths. The result of a lightpath probe is defined as the probe syndrome, indicating whether the probed lightpath is healthy or not. In this thesis, we adopt the following notations for probe syndromes: F for failure, S for success. *One design objective is to minimize the number of lightpath probes, so as to reduce the diagnosis effort*[4]. The proactive probing diagnosis paradigm is illustrated in Fig. 1-6(b), where the feedback from the fault-diagnosis engine to the proactive probing module (i.e., the event generator) provides the flexibility to design scalable proactive fault-diagnosis schemes that reduce the diagnostic effort and the diagnostic hardware cost.

The unique property of all-optical networks suggests that the proactive fault-diagnosis paradigm should be the natural choice for fault detection and localization in all-optical networks. In all-optical networks, optical signals traverse a lightpath without being detected and regenerated by intermediate nodes. This property permits lightpath probes to test the health of several links/nodes simultaneously, which can be used to reduce the diagnosis effort. To exploit such an opportunity, we focus on proactive probing diagnosis schemes in this thesis, with the objective to develop scalable proactive fault diagnosis schemes for dynamic all-optical networks.

Mathematically, the fault-diagnosis problem with the proactive probing diagnosis paradigm can be cast as a problem of group testing over graphs. As illustrated in Fig. 1-7, the network is abstracted as a graph in which the failure status of each node/link is modeled with a Bernoulli random variable. Probing signals are sent along a set of lightpaths and their measurements, defined as probe syndromes, are used to infer network state of health. This framework can be extended to model many other network-diagnosis applications by choosing appropriate state variables.

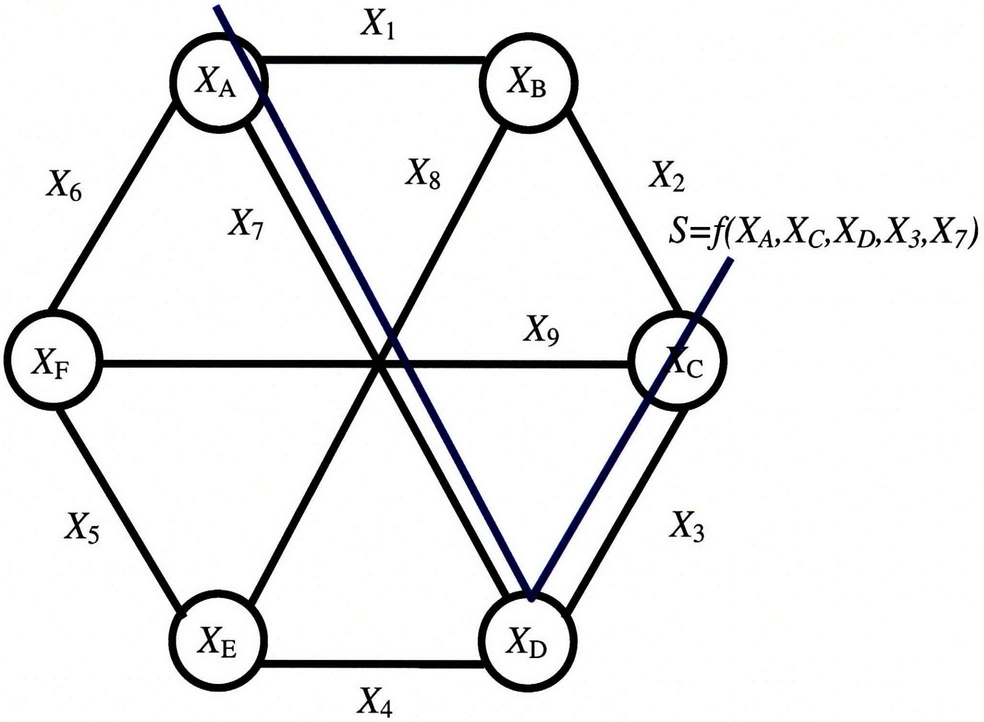


Figure 1-7: Group testing over graphs: the states of nodes and links are modeled by random variables (or random processes), and the outputs of probes are functions of node/link states covered by paths (or subgraphs).

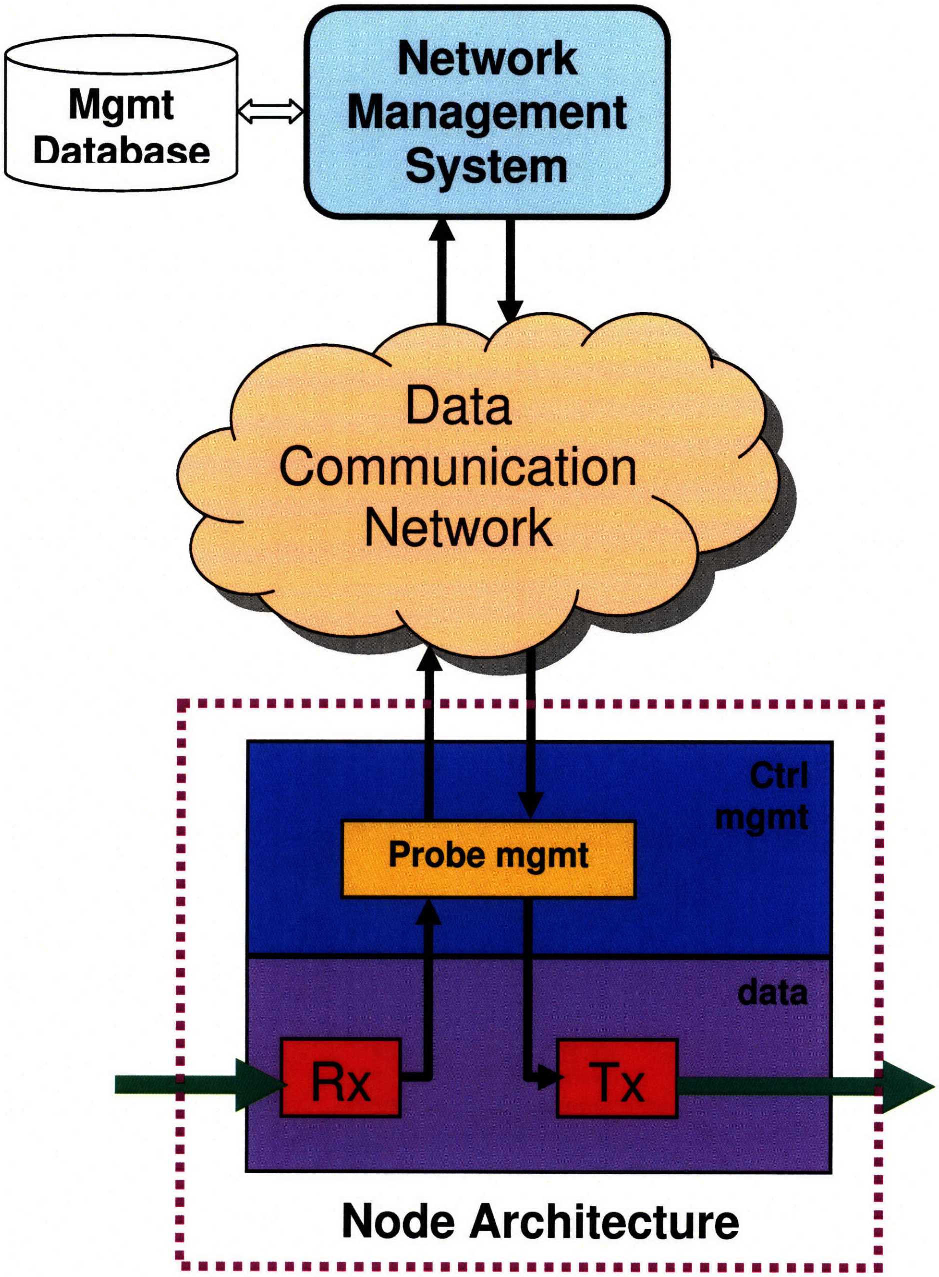


Figure 1-8: Proposed node architecture for proactive fault diagnosis schemes: Transmitter/Receiver pair for probe transmission and detection, software agent for probe syndrome processing.

1.2.2 Proposed Node Architecture for Proactive Fault Diagnosis Schemes

Under the proactive fault-diagnosis paradigm, some network nodes should have the capability to transmit and receive optical probing signals, and report probe syndromes. For this purpose, we propose a node architecture with diagnosis capability, as illustrated inside the dotted box of Fig. 1-8. In particular, the fault-diagnosis function is implemented via a pair of transmitter/receiver (Tx/Rx) at the data plane and a software agent at the control/management plane. The Tx/Rx pair at the data plane is responsible for transmitting optical probing signals along lightpaths and detecting probing signals in the optical layer. In general, transmitters and receivers do not have to pair up at the same node all the time. The coexistence of transmitters and receivers simplifies the fault management system architecture by having a uniform node architecture. The software agent at the control/management plane is responsible for probe management, including processing the detected probe signals to determine the message reported to the network management system and initiating probing signals according to probe signaling messages from the network management system.

In this thesis, we assume that dedicated Tx/Rx pairs are provisioned for fault diagnosis. An alternative assumption is to use idle transmitters and receivers in the network to send and detect optical probing signals, but there is a possibility that no such idle Tx/Rx pair is available. An implication of using dedicated Tx/Rx pairs for fault diagnosis that it incurs additional capital expenditure. It follows that *another design objective of scalable fault diagnosis schemes is to minimize the number of Tx/Rx pairs for fault diagnosis, so as to minimize the diagnostic hardware cost*[5].

[4] This design objective will be addressed in Chapter 2 and 3.

[5] This design objective will be addressed in Chapter 4.

1.2.3 Proactive Fault Diagnosis Schemes: Adaptive, Non-Adaptive and Multi-Step

Based on how probes are scheduled in proactive fault-diagnosis schemes, proactive fault-diagnosis schemes can be classified into three different categories:

Adaptive Diagnosis Scheme In an adaptive fault diagnosis scheme, individual optical probing signals are sent sequentially along a set of lightpaths over an all-optical network to probe its state of health. The network state (i.e., the failure pattern) is then inferred from the results of this set of end-to-end lightpath measurements (i.e., probe syndromes). Moreover, each successive probe is dynamically chosen among a set of permissible lightpath probes according to the set of previous probe syndromes, with the objective of minimizing the number of lightpath probes.

Non-adaptive Diagnosis Scheme In a non-adaptive fault diagnosis scheme, multiple optical probing signals are sent along a set of pre-determined lightpaths in parallel. The network state is then inferred from the set of probing syndromes. A brute-force non-adaptive diagnosis scheme is to test each individual link in the network for all possible link failures and the number of lightpath probes is equal to the number of links in the network.

Multi-step Diagnosis Scheme Multi-step diagnosis schemes carry the properties of both adaptive and non-adaptive diagnosis schemes. In multi-step fault diagnosis schemes, lightpath probes are scheduled in multiple sequential steps as in adaptive fault diagnosis schemes; at each step, multiple lightpath probes are sent in parallel as in non-adaptive diagnosis schemes.

As an example, we illustrate these three proactive fault diagnosis schemes in Fig. 1-9. For a linear network of 4 links, we assume that, if any failure happens, one and only one link failure occurs[6]. The adaptive diagnosis scheme is illustrated in Fig. 1-9

[6] Practically, one would not be able to make such a failure model assumption, because the upper bound on the number of failures is normally unknown. Here we assume that there a genie exists to tell the number of link failures if they occur.

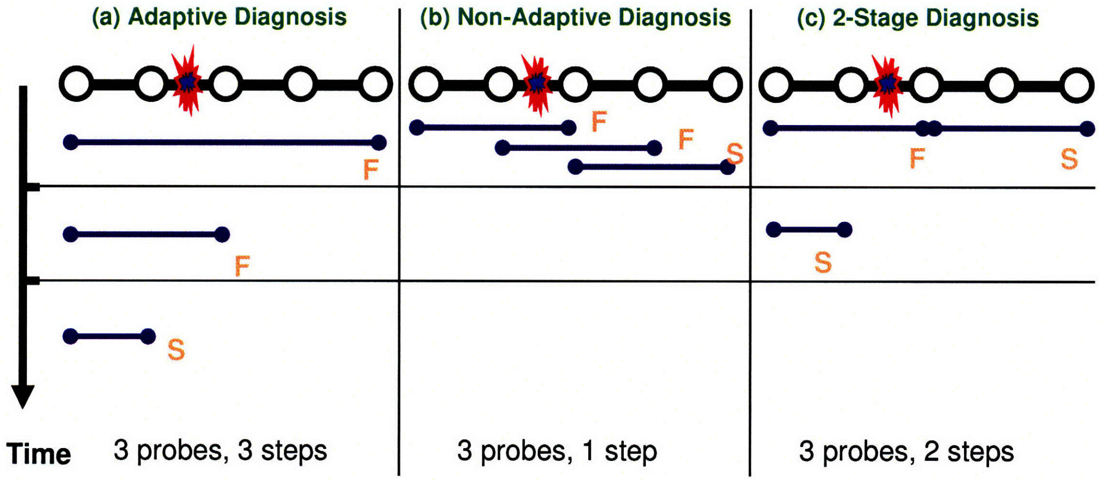


Figure 1-9: Illustrations of different proactive fault diagnosis schemes: (a) adaptive diagnosis, (b) non-adaptive diagnosis, and (c) multi (2)-step diagnosis. The number of lightpath probes can be considered as the diagnostic effort, and the number of steps can be considered as the diagnostic delay.

(a). In the first step, a probe of length 4 is sent and resulting in a probe syndrome of F (a failure). It suggests that some edge has failed. In the second step, a probe of length 2 is sent and resulting in a probe syndrome of F. The same process continues in the 3rd step and resulting in a probe syndrome of S (a successful transmission). This suggests that the second link has failed. The non-adaptive fault diagnosis scheme is illustrated in Fig. 1-9 (b). In this case, 3 probes are sent in parallel to uniquely identify the failure on the second link. Fig. 1-9 (c) illustrates a special case of multi-step diagnosis scheme (i.e., a 2-step diagnosis scheme). In this case, 3 probes are sent in two steps, where the first step has 2 probes and the second step has 1 probe.

As shown in this example, we are interested in two design metrics : i) the (average) number of lightpath probes, denoted as $\mathcal{L}_p$, and ii) the (average) number of steps, denoted as $\mathcal{T}_p$. Each diagnosis scheme is thus characterized by a tuple $(\mathcal{L}_p, \mathcal{T}_p)$. In fact, there is a trade-off between two design metrics, as explained in next subsection.

1.2.4 Trade-Off between Diagnostic Effort and Diagnostic Delay

For the proactive fault diagnosis paradigm, we are interested in two performance metrics : the diagnostic effort and the diagnostic delay. Both follow the design objectives of fault diagnosis schemes: (1) to make fault diagnosis schemes scalable and (2) to identify network failures as quickly as possible. In this subsection, both performance metrics are translated into practical design parameters, as follows.

The diagnostic effort refers to the amount of work expended in scheduling, transmitting and detecting optical probing signals and reporting probe syndromes. It is recurring, and is in proportion to the average number of lightpath probes, denoted as $\mathcal{L}_p$. It follows that, to make fault-diagnosis schemes scalable, we would like to minimize the average number of lightpath probes. In this thesis, Chapter 2 and 3 are dedicated to developing efficient fault diagnosis schemes that minimize the (average) number of lightpath probes.

The diagnostic delay can be interpreted as the average number of steps required to identify the network state, for different fault diagnosis schemes. For each probe, the delay could include three components: optical transmission delay, data communication network transmission delay, processing & scheduling delay. It can be shown that, in current optical networks, the probe delay is normally on the order of the data communication network transmission delay. Currently, the data communication network (e.g., a TCP/IP network) is normally carried over bandwidth-limited pipelines, such as a T1 line with a data rate of 1.544 Mbps. If each probe control message is carried over one IP packet with an average length of 500 bytes and standard deviation of 500 bytes[7], the data communication network transmission delay would be around 5ms with a standard deviation of 5ms. This is on the same order of the USA's coast-to-coast transmission delay ($\sim$ 10ms). It follows that, to a first order, the number of

[7] According to http://www.caida.org, the average length of an IP packet is around 500 bytes, with a standard deviation of 500 bytes. Here, we assume that the control message in the data communication network follows the same statistics.

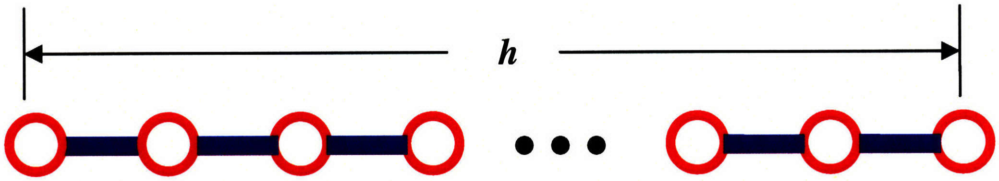


Figure 1-10: A line network of h edge. For illustration purpose, we assume that the number of simultaneous link failures is upper bounded by one.

diagnostic steps, denoted as $\mathcal{T}_p$, can be considered as a indicator of the fault diagnosis delay.

For proactive fault diagnosis schemes, we would like to minimize both the diagnostic effort (i.e., the average number of lightpath probes) and the diagnostic delay (i.e., the average number of diagnostic steps). However, these two objectives compete with each other. In one extreme, we can use as much resource as possible to diagnose failures and the delay could be just one step. For example, one can probe each individual link in the network in parallel to identify all possible link failures within one step, only to use the largest number of probes and the most Tx/Rx pairs. In the other extreme, one could use as little resource as possible and the delay would be longer. Intuitively, there is a trade-off between the diagnosis effort and the diagnosis delay. It is one of our objectives in this thesis to characterize this trade-off, and design optimal or near-optimal fault-diagnosis schemes.

In the following, we present a motivational example to highlight the trade-off between the diagnostic effort and the diagnosis delay, by comparing three alternative fault diagnosis schemes - adaptive, non-adaptive and 2-step fault diagnosis.

Example 1.1. *Given a linear network of h undirected edges, as shown in Fig. 1-10, it is assumed that there is one and only one failure if any failure has occured*[8].

[8] This assumption is contrived in that the number of simultaneous link failures is normally unknown. Moreover, our results on non-adaptive fault-diagnosis schemes depend on the fact that the fault-diagnosis scheme knows the upper bound on the number of simultaneous link failures. Nevertheless, the results obtained under this strong assumption provide some insights into more generalized cases where the number of link failures is unknown.

Table 1.1: Diagnostic Effort and Delay Comparison

	Adaptive	2-Step	Non-Adaptive
Effort: $\mathcal{L}_p$	$log_2 h$	$\sqrt{2h}$	$\frac{h}{2}$
Delay: $\mathcal{T}_p$	$log_2 h$	2	1

We would like to identify the faulty edge if there is one, via different proactive fault diagnosis schemes. The design objective is to minimize the number of lightpath probes for a given class of fault-diagnosis schemes (i.e., adaptive, non-adaptive and 2-step diagnosis schemes).

For different fault diagnosis schemes (i.e., adaptive, non-adaptive and 2-step diagnosis), we compare their diagnostic effort and diagnostic delay in Table 1.1. The optimal adaptive fault diagnosis scheme needs approximately $log_2 h$ probes on average, the optimal 2-step fault diagnosis scheme needs approximately $\sqrt{2h}$ probes on average[9]*, and the optimal non-adaptive fault diagnosis scheme requires $h/2$ probes. The diagnosis delay is indicated by the average number of probe steps in the fault diagnosis scheme, i.e., $log_2 h$ for adaptive diagnosis, 2 for 2-step diagnosis and 1 for non-adaptive diagnosis.*

Notice that, as the diagnosis delay (equivalently, the average number of diagnosis steps) increases, the minimum diagnostic effort (i.e., the average number of lightpath probes) decreases. Specifically, as the number of diagnostic steps increases from 1 (for non-adaptive diagnosis) to 2 (for 2-step diagnosis) to $\log_2 h$ (for adaptive diagnosis), the average number of lightpath probes decreases from $h/2$ (for non-adaptive diagnosis) to $\sqrt{2h}$ for 2-step diagnosis) to $\log_2 h$. This trend is shown in Fig. 1-11, as the diagnostic effort-delay curve is plotted for the case of $h = 16$.

This example suggests that a trade-off exists between the diagnosis effort and the diagnosis delay. In particular, the minimum diagnosis effort decreases as the diagnosis delay increases. This trade-off permits network architect to design specific

[9] If the length of probes in the first step is x, the total number of probes required is $\mathcal{L}(x) = \frac{x}{2} + \frac{h}{x}$, where the first term of $x/2$ comes from the number of probes in the second step and the second term of h/x comes from the number of probes in the first step. $\mathcal{L}(x)$ is minimized to be $\sqrt{2h}$ when $x = \sqrt{h/2}$.

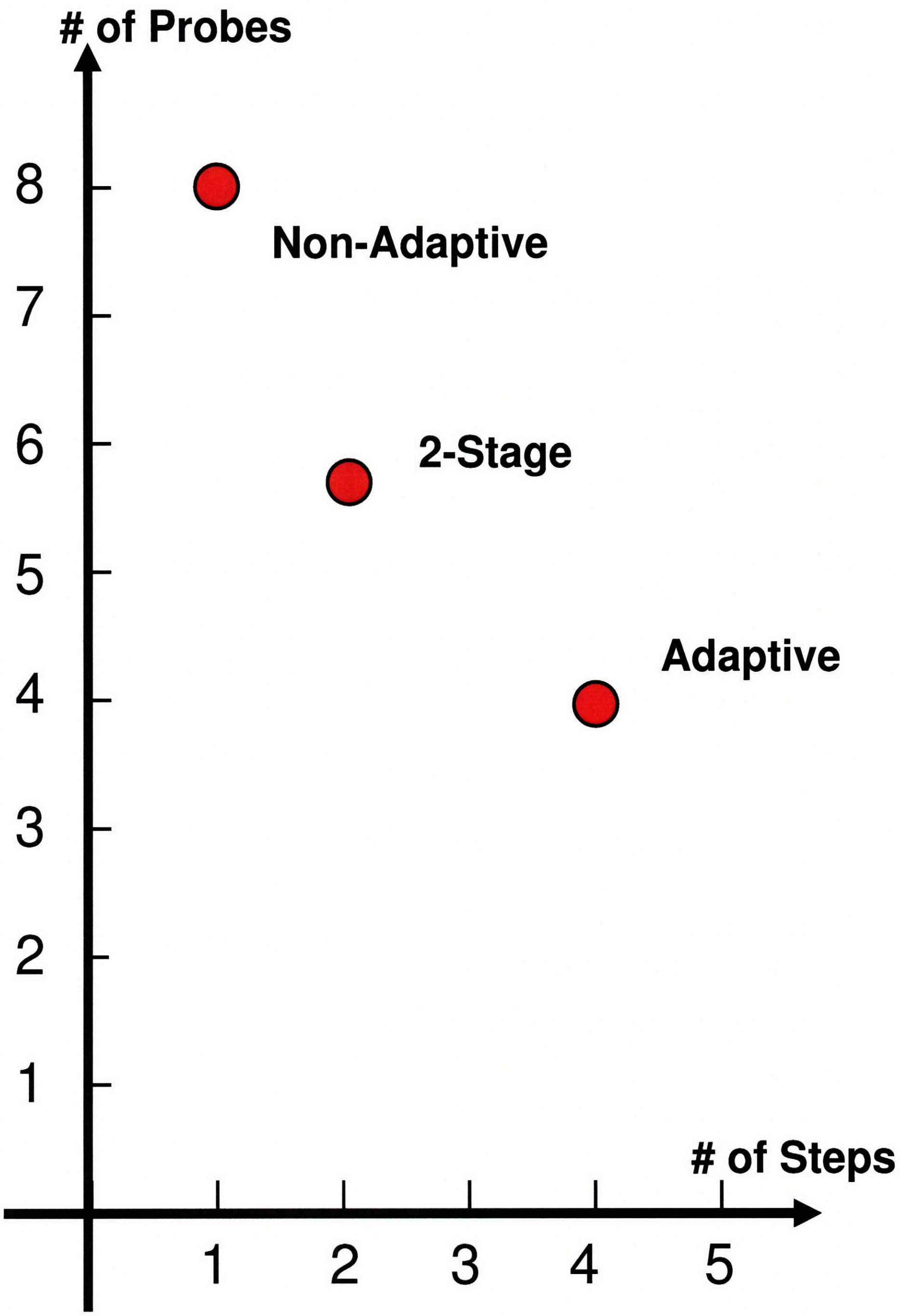


Figure 1-11: The diagnostic effort-delay trade-off for a line network with $h = 16$ edges. The diagnosis effort tends to decrease as the diagnosis delay increases.

fault diagnosis schemes for different applications. This example also motivates us to investigate the fault-diagnosis scheme for future dynamic all-optical networks with the objective of understanding the diagnostic effort-delay trade-off.

Theoretical research on scalable fault-diagnosis schemes should be complemented with additional research on some practical issues. One practical issue is where to place network nodes with diagnostic Tx/Rx pairs in the network. In Chapter 4, we look at this problem in an alternative setting. In particular, given a partial set of evenly-distributed network nodes equipped with diagnostic Tx/Rx pairs[10], we would like to maximize the probability of successful diagnosis. Another practical issue on scalable fault-diagnosis schemes is how to work around existing traffic in the network. In Chapter 5, we follow a "divide-and-conquer" strategy to develop fault-diagnosis schemes for different classes of lightpaths: i) lightpaths for existing traffic, ii) pre-computed lightpaths and iii) lightpaths computed online. One particular challenge here is that certain lightpath probes might not be feasible because existing traffic dictates how optical switches should be configured. In this case, we would infer and estimate internal network states through historical data and/or device failure models.

1.3 Network Survivability Architecture: Automatic Protection Switching vs. Lightpath Diversity

As illustrated in Fig. 1-4, network survivability, i.e., the capability to provide continuous network service through robust transport schemes in the presence of failures, is a critical function that all networks should provide. If these networks carry commercial, military and scientific traffic in super-high data rate, the interruption of network services for even a short period of time might have disastrous consequences [76].

For example, in the commercial Internet Service Provider (ISP) business, a carrier normally commits to providing a certain availability for the connection as part

[10] Such a policy might not be an optimal deployment of diagnostic Tx/Rx pairs in the network. However, for all-optical networks with symmetric graphs, we believe that the evenly-distributed policy should be close to the optimum.

of the service-level agreement between the carrier and its customer leasing a connection [50]. A common requirement is that the connection should be available 99.999% (five 9's)of the time. This requirement corresponds to a connection downtime of less than 5 minutes per year. The only practical way of obtaining 99.999% availability is to make the network survivable, that is, being able to continue providing service in the presence of failures. Therefore, network architect should design efficient network survivability schemes to guarantee network service as specified by any bilateral agreement or standard specification.

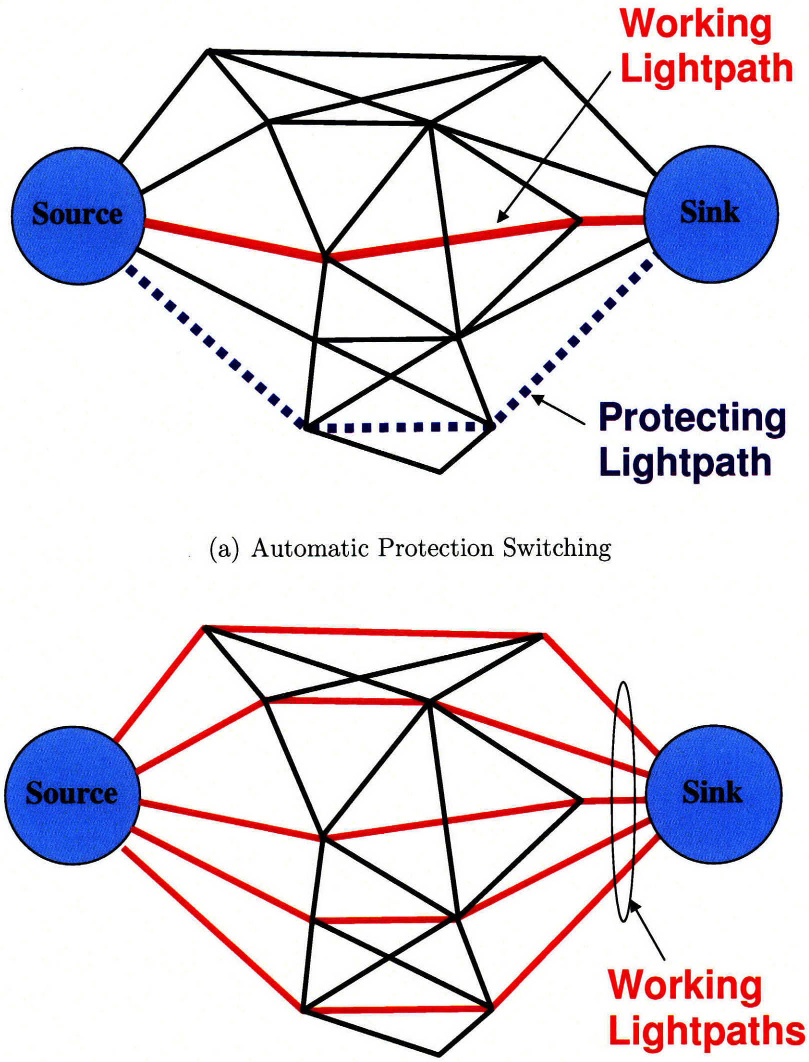
Survivability designs for all-optical networks are subjected to different requirements, for example, different restoration time requirements. In future all-optical networks, the lightpath demands can be classified into different restoration classes based on their different restoration time requirements [55]. It follows that the time delay is an important design metric for network survivability architecture.

Different restoration requirements dictate how network survivability should be designed. In particular, to meet the heterogenous requirements for different restoration applications, one cannot expect one single survivability scheme fit all, but to develop different survivability schemes for different situations. In this chapter, we focus on two classes of network survivability schemes: the automatic protection switching (APS) scheme, and the lightpath diversity (LD) scheme. Both schemes are illustrated in Fig. 1-12 and will be explained in next two subsections.

1.3.1 Automatic Protection Switching (APS)

Currently, the prevailing approach for network survivability is the automatic protection switching scheme, as implemented commercially in SONET/G.709 networks. In this scheme, as illustrated in Fig. 1-12(a), each primary working lightpath is protected by another secondary protecting lightpath. If the source-destination communication session over the working lightpath is interrupted by a failure, the failure is first detected and the communication is restored along the protecting lightpath.

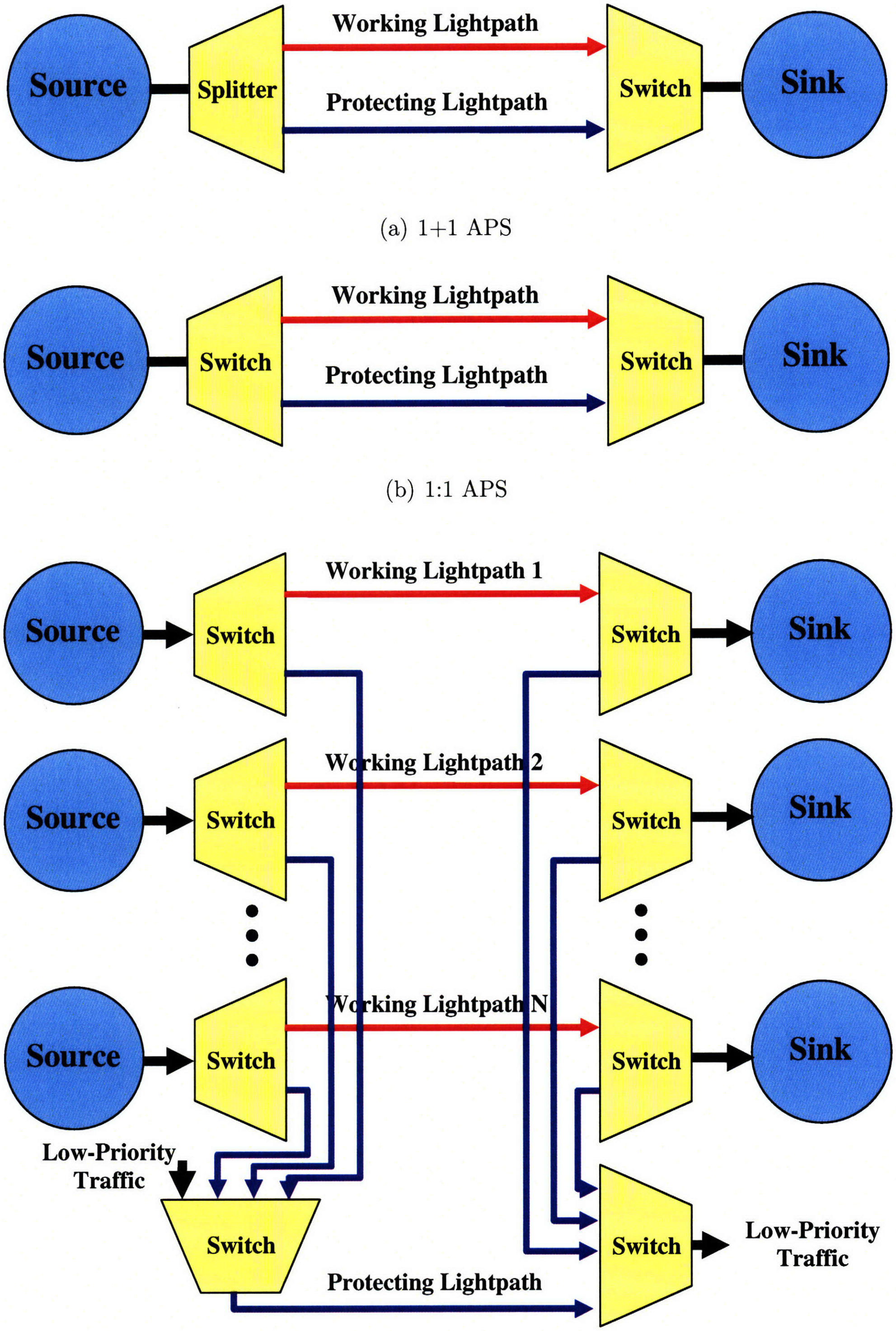
Depending on the assignment of protection resources, the automatic protection switching scheme has three main architectures: 1+1, 1:1 and 1:N. In the 1+1 auto-



(a) Automatic Protection Switching

(b) Lightpath Diversity

Figure 1-12: Two alternative network survivability schemes: (a) automatic protection switching scheme, (b) lightpath diversity scheme.



(a) 1+1 APS

(b) 1:1 APS

(c) 1:N APS

Figure 1-13: Three alternative automatic protection switching architectures: (a) 1+1, (b) 1:1 and (c) 1:N.

matic protection switching scheme, as illustrated in Fig. 1-13(a), traffic is transmitted simultaneously over two separate links (usually over disjoint routes) from the source to the destination. The destination simply selects one of the two links (each of which has its own receiver) for reception. If that link is cut, the destination simply switches over to the other link and continues to receive data. In the 1:1 automatic protection switching, shown in Fig. 1-13(b), there are still two fibers from the source to the destination. However, traffic is transmitted over only one fiber at a time, i.e., the working fiber. If that fiber is cut, the source and the destination both switch over to the other protection fiber. Under normal conditions, the protecting lightpath is either idle or used to carry low-priority traffic. In the 1:N automatic protection switching scheme, N working lightpaths share a single protecting lightpath. It operates similar to the 1:1 automatic protection switching scheme, except that the protecting capacity is shared among all the primary lightpaths and at any time only one primary lightpath can be protected.

One advantage of the automatic protection switching scheme is that its loss of bandwidth efficiency is limited. In fact, its bandwidth efficiency, defined as the ratio between the data bandwidth and the provisioned bandwidth, is lower bounded by $\frac{1}{2}$. With the 1:N automatic protection switching, the bandwidth efficiency is $\frac{N}{N+1}$, which is close to 1 when N is large. Therefore, the loss of bandwidth efficiency is bounded. However, this protection-switching mechanism can induce a rather long delay ($\sim$50-ms restoration time, the SONET standard [34]). Thus this scheme is inappropriate for some unique applications. For example, considering the service with super high data rate (>10Gbps), a short-time interruption can result in a large amount of data loss. In other critical applications (e.g. when the network is used for transporting control signals between the cockpit and control surfaces in an aircraft), the time-deadline of control-message delivery needs to be shorter than 1-ms and probably ten times faster in failure detection. This is faster than the speed at which most optical components can switch and protection-switching protocol can be executed.

1.3.2 Lightpath Diversity

For these applications with stringent restoration requirements, one could increase the speed of failure detection and lightpath switching to meet increasing data rates and critical time deadlines. However, such a brute-force approach complicates the system design and would include additional network cost. Instead, as illustrated in Fig. 1-12(b), the lightpath diversity scheme, which sends the same data through multiple lightpaths in different Shared-risk link groups [16], is a better alternative that can be implemented with current technologies. Chan and Parikh have explored this mechanism in [13, 45]. In that work they looked at a joint Data Link Control Layer and Transport Layer reliable message delivery scheme and have found significant merit for using path diversity efficiently via error correction coding techniques. In this thesis, we extend their work to a Physical Layer lightpath diversity mechanism, using an optimum signaling and detection scheme to optimize system performance and provide reliable end-to-end data delivery in the presence of failures (e.g., fiber cuts and node hardware failures).

The advantages of the lightpath diversity scheme are at least two-fold. First, because the entire mechanism is implemented at the Physical Layer, it provides a much faster response to failures than protocols that provide end-to-end reliability at higher layers, especially those that need feedback, such as the Transmission Control Protocol (TCP) at the Transport Layer. Second, it can be shown that the symbol error probability of multiple-lightpath transmission is significantly lower than that of single-lightpath transmission in medium and high signal-to-noise ratio regimes. In particular, for a source-destination pair connected by M lightpaths, the symbol error probability in the high signal-to-noise ratio regime is asymptotically equal to $\prod_{i=1}^{M} f_i$ (f_i is the failure probability of the lightpath.) This sets the asymptotic reliability limit of the multiple-lightpath transmission scheme. By choosing the number of lightpaths used, this limit can be made arbitrarily small compared to the asymptotic symbol error probability of using only a single lightpath between a source-destination pair.

Compared to the single lightpath transmission, one major disadvantage of the lightpath diversity scheme is that the same message is sent repeatedly through a group of disjointed lightpaths and thus degrades the throughput per channel use by a factor of M for an M-connected source-destination pair. This suggests that the bandwidth efficiency is $\frac{1}{M}$, which approaches zero as the number of lightpaths increases. However, in order to achieve ultra-reliable communication with low delay, for example, in an aircraft control network, we choose to sacrifice some bandwidth efficiency for reliability in a bandwidth-rich environment (e.g., optical fiber). In fact, multiple connections between any source-destination pair are necessary for reliable networks [64], and both parallel signaling and sequential signaling over multiple connections can realize high reliability. The lightpath diversity scheme satisfies this necessary condition by splitting each channel symbol and sending the fragments simultaneously through M disjointed lightpaths.

In Chapter 6, we will investigate the proposed lightpath diversity scheme from both a theoretical and an engineering perspective.

From the theoretical perspective, we characterize and optimize the error performance of the lightpath diversity system. First, we show that the bit error rate of the lightpath diversity schemes takes contributions from two sources: noise and failure. To reduce the noise effect, we would like to increase the signal-to-noise ratio per lightpath. To reduce the lightpath failure effect, we would like to use more lightpaths. Specifically, the bit error rate is given by

$$PB_{GA} = \{f + (1-f)\exp[-N_n(\sqrt{\Omega+1}-1)^2]\}^M, \tag{1.1}$$

where f is the lightpath failure probability, M is the number of lightpaths, N_n is the average number of noise photons received per bit, and Ω is the signal-to-noise ratio per lightpath. However, for a given amount of optical energy (per bit), the product of the signal-to-noise ratio per lightpath and the number of lightpaths is a constant,i.e.,

$$M \times \Omega = \frac{N_s}{N_n}, \tag{1.2}$$

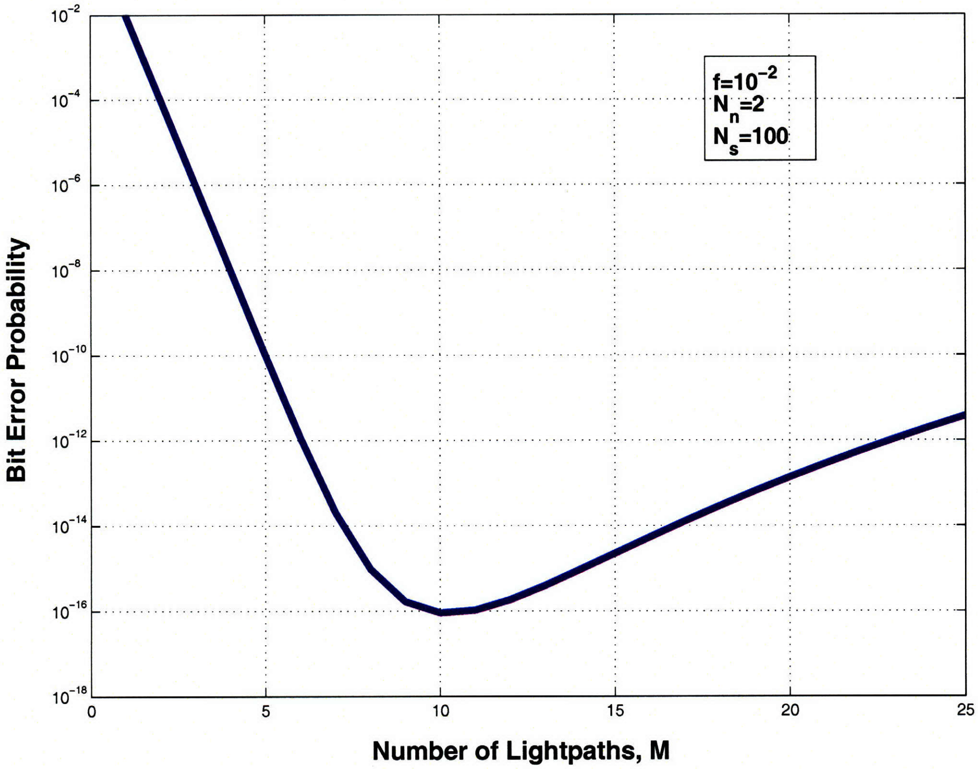


Figure 1-14: For a given amount of optical energy per bit, there is a trade-off between the SNR per lightpath and the number of lightpaths.

where N_s is the average number of signal photons per bit. Therefore, there is a trade-off between the number of the lightpaths used the signal-to-noise ratio per lightpath, for a given amount of optical energy (per bit), as illustrated in Fig. . At We then seek to balance this trade-off and derive an optimal operating point for the lightpath-diversity scheme.

From the engineering perspective, we develop a class of structured receivers and evaluate their error performance. First, we show a separation between the estimation function (responsible for estimating the lightpath failure state) and the detection function (responsible for deciding whether the transmitted bit is ONE or ZERO)in the optimal realizable receiver architecture. Due to the high complexity in the optimal receiver, we develop a class of simpler receivers, i.e., the equal-gain-combining receiver (EGC). Performance comparison indicates that the EGC receiver suffers some power penalty but provides significant reduction in complexity.

Chapter 2

Adaptive Fault Diagnosis Schemes

This chapter is dedicated to the design of adaptive fault-diagnosis schemes for all-optical networks. The design objective is to minimize the amount of diagnostic effort (i.e., the average number of lightpath probes) to identify all possible network failures[1].

2.1 Introduction

All-optical networks [25, 12], where data traverse along lightpaths without any optical-to-electrical conversion, will be increasingly prevalent in future broadband network due to its inherent large transmission bandwidth, lower cost, and transparency to different signal formats and communication protocols. However, similar to other networks, all-optical networks are also vulnerable to failures [66], such as fiber cuts and transmitter/receiver breakdowns. Moreover, there are new types of failures that are unique to all-optical networks - failures related to subtle changes in signal power, optical signal-to-noise ratio, cross-talk, Kerr effects, or other non-linear effects. These failures can result in the disruption of communication, and can be difficult to detect, localize and repair. Hence, when parts of a network are malfunctioning it is critical to locate and identify these failures as soon as possible. At the same time, the effort to detect and locate failures must be small to keep the network cost low. In this

[1] The content in this chapter has been published partially in journal paper [68] and conference papers [70, 69].

chapter, a family of adaptive fault-diagnosis schemes are proposed to exploit the unique property of all-optical WDM networks where optical signals are not usually detected at intermediate nodes along lightpaths (mostly for cost reasons).

According to the scale of their effect, failures in all-optical WDM networks can be classified into two categories. One type of failures affects individual wavelength, and is thus called as the wavelength-level failure. Examples of wavelength-level failures include transmitter/receiver failures in the case of one dedicated transmitter/receiver per wavelength, optical filter failures and individual channel failures of a frequency selective switch. The other type of failures affects all the lightpaths on an individual fiber, and is thus called the fiber-level failure. Examples of fiber-level failures include fiber cuts and EDFA breakdowns. From a graph theory perspective, one can attribute both categories to edge failures in a network (graph) topology. In this thesis, the "ON/OFF edge failure" is modeled as a binary-value function of value 0 if the required quality of transmission is met and of value 1 otherwise. Besides, all the failures that do not belong to the same risk group are considered independent. In real life, there may be failure correlation among risk groups due to physical proximity or accessibility from the same malicious attack entry point. In those cases the results in this chapter can still provide very useful upper and lower bounds to the diagnostic effort required to localize the failures.

Historically, since the fault diagnosis problem [48] was first proposed in 1967, it has been investigated extensively in electrical networks under a system diagnosis context [57, 6, 38]. In this context, most current research is focused on a "single hop" test model, i.e., signals are transmitted between adjacent nodes to determine whether failure occurs on the edge connecting them. The result of each test can be represented as one bit of diagnosis information: 1 or 0, corresponding to "failure" or "no failure". Indeed for SONET networks, each SONET link (single hop) checks the health of the link using parity checks within the SONET receiving chips[2]. However, in all-optical networks, this 'single-hop test' assumption will usually not be applicable

[2] We define a diagnosis scheme that tests each individual link in the network as a *link-wise diagnosis scheme.*

due to the unique property that optical signals are not typically detected at every (optically switched) intermediate node along the lightpath. For SONET networks, the network management system employs mechanisms such as BER measurement, optical trace and alarm management to perform fault detection and localization at each regenerator. In particular, these functionalities may be carried over various types of optical layer overhead [50], including pilot tone, subcarrier-modulated overhead, optical supervisory channel, rate-preserving overhead and digital wrapper overhead. To some degree, all these overheads are detected at some intermediate nodes along the lightpath. This, in fact, breaks the spirit of the transparency paradigm of all-optical networks and adds to the complexity and cost of future all-optical networks which do not need signal detection along a lightpath.

Currently, to diagnose failures in future all-optical wavelength-division multiplexing (WDM) networks, researchers typically consider an optical (channel) performance monitoring solution, where optical performance monitors are employed at a set of network nodes to watch for possible failures and report them to the network management system [24]. However, little work has been done to quantify the overhead cost and the amount of diagnostic effort that this monitoring solution might incur. Instead, most research literatures [74, 52] follow essentially the same design approach as their electrical counterparts, implicitly assuming that each network node, or even each active optical component such as optical amplifiers and OADMs (Optical Add-Drop Multiplexer), is equipped with a performance monitoring module which is active and reporting all the time. While this is an acceptable solution in the near-term since signal detection comes for free at every regeneration point, it is desirable to develop more efficient and less costly methods when the all-optical network paradigm is fully implemented and the network size grows significantly. Reduced complexity is good for the following reasons. First, the total amount of monitored information and signaling grows linearly with the number of network elements (i.e., network nodes and edges). The huge amount of management information, together with faster switching speeds in the network, complicates the network management system and stresses the limited capability of current network processing units. Therefore, a mechanism based on constant sensing and reporting of numerous individual active monitors[3] does not scale well with the size and tuning agility of future all-optical networks. Second, since each monitor only tests one component without taking into consideration of its failure statistics, the diagnostic effort (e.g., the required number of tests per unit time with the interval between monitoring drawn from QoS specifications including mean time to failure and failure statistics) of such a mechanism can be prohibitively high, limiting the efficacy and ultimately ubiquitous deployment of all-optical networks.

In this chapter, more efficient and elegant methods are sought to greatly lower the diagnostic effort for future all-optical WDM networks. In our proposed scheme, optical signals are sequentially sent along a set of lightpaths over an all-optical network to probe its state of health. The network state (i.e., failure pattern) is then inferred from the 'syndromes' of this set of end-to-end measurements. To keep the required number of probes small, each successive probe is dynamically chosen among the set of permissible probes according to the results of the previous tests. Under this generalized model, the traditional diagnosis mechanism based on single-hop probes is then a special case and will be proven to be rather inefficient compared to the proposed approach. In this chapter, a family of failure identification algorithms are developed to exploit the unique properties of all-optical networks to reduce the average number of diagnostic probes.

In all-optical networks, the fact that optical signals can be carried over a lightpath of a number of interconnected edges without necessarily being detected by the intermediate nodes allows "multi-hop" tests to probe several edges simultaneously. This technique can be used to greatly reduce the amount of diagnostic effort, as illustrated with the 3-node ring network in Fig. 2-1. In this example, it is assumed that each edge fails independently with probability of 0.1. If only 'single-hop' tests are allowed as in Fig. 2-1(b), the total number of tests to identify all edge states is 3 by employing three single-hop tests (A-B, B-A, C-A). Note that the number of tests required is independent of the edge failure probability and equal to the number of

[3] How fast updates are needed is correlated with how fast lightpaths are supposed to be set up initially and be restored if any problem arises.

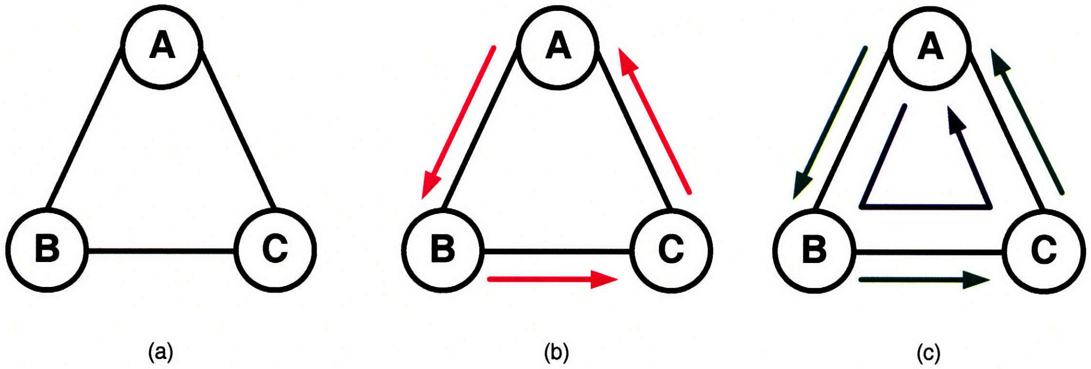


Figure 2-1: Comparison between diagnosis paradigms of electrical networks and all-optical networks: (a)three-node ring network; (b)diagnosis with three single-hope tests; and (c) diagnosis with one three-hop test and three single-hop tests.

edges in the topology. On the other hand, if multi-hop tests are allowed, one can first perform a three-hop test (A-B-C-A) as shown in Fig. 2-1(c). With a probability of $(1-0.1)^3 = 0.729$, all edges are found to be fault-free and the diagnosis is concluded with only one test. One can resort to the single hop tests only if there is at least one failure from the result of the first test, which has a probability of $1 - 0.729 = 0.271$. Thus, on the average, it requires only $0.729 \times 1 + 0.271 \times (1+3) = 1.813$ tests to fully diagnose this network. Intuitively, in most cases, the probability that a particular edge has failed is low when network diagnosis is performed; hence it makes sense to test several edges together. Here, reducing the average number of tests required for network diagnosis, which is used in this chapter as a measure of the diagnostic effort, or efficiency, of the diagnosis process.

This example suggests that the fault-diagnosis problem can be better understood from an information theoretic prospective. The network state can be viewed as a collection of binary valued random variables; each associated with an edge in the network, indicating failure/no failure on that edge. The objective of a fault-diagnosis algorithm is to use a number of tests, whose results, also called the 'syndromes', can be used to uniquely identify the network state. To put it simply, we use probes to dig out all the information hidden in the unknown network state. In the above example, with the single-hop tests, the result of each test is '0' (for no failure) with a

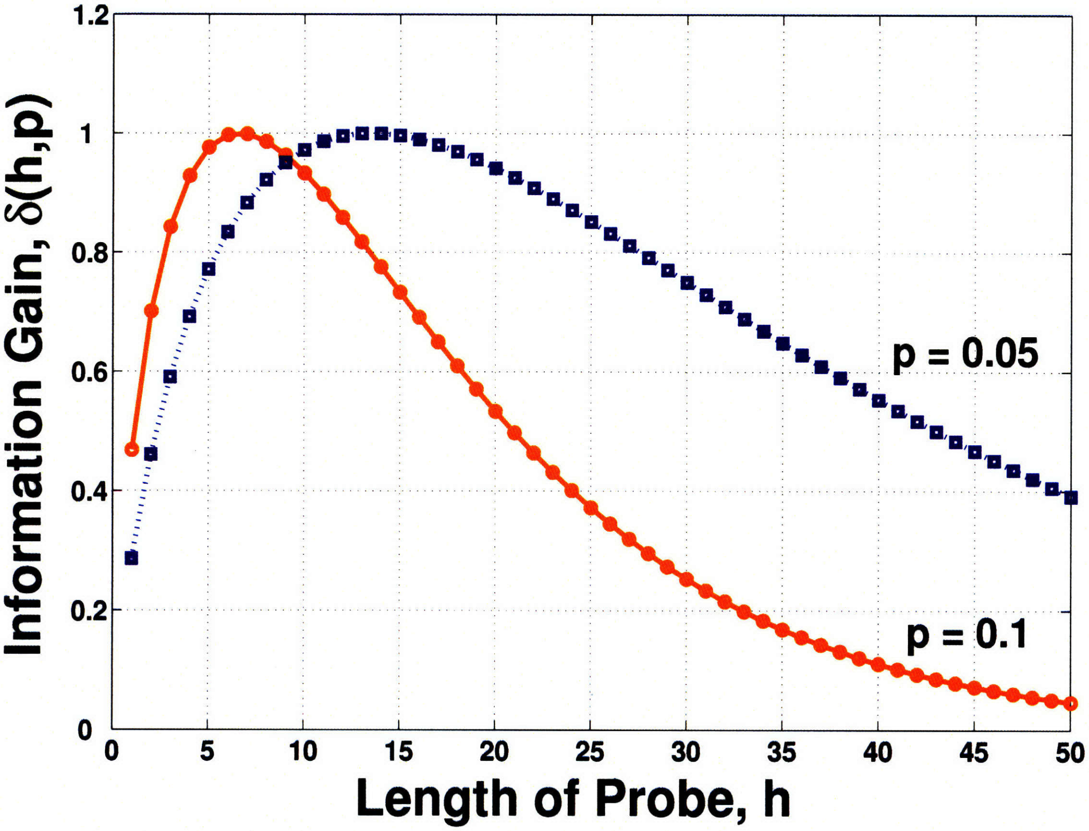


Figure 2-2: The amount of network state information provided by any probe is plot as a function of the length of probe, h.

probability of 0.9 and '1' (for failure) with a probability of 0.1. Thus the information about the network state provided by this test result is the binary entropy function of $H_b(0.1) = 0.469$ bits where $H_b(x) = -xlog_2x - (1-x)log_2(1-x)$. In comparison, the three-hop test (A-B-C-A) contains $H_b(0.271) = 0.843$ bits of information. The information contained in a three-hop test is obviously larger than that of the single-hop test, indicating that multi-hop tests are more informative than single-hop tests for this case. As a result, in the second approach, the network state can be identified by a smaller number of probes, or equivalently, the network state is represented by the test syndrome in a more efficient way. In other words, this can be viewed as encoding the network state with probe syndromes. In general, the amount of network state information provided by a lightpath probe of length h is given by

$$\delta(h,p) = -(1-p)^h \log_2(1-p)^h - [1-(1-p)^h] \log_2[1-(1-p)^h], \tag{2.1}$$

where p is the failure probability of each individual link. In Fig. 2-2, we plot the amount of network state information provided by any probe as a function of the length of the probe. This figure can be used to determine the efficiency of a probe in a general network. More importantly, this case study suggests that the design of efficient fault diagnosis algorithm is similar to the well-studied source-coding problem, whose goal is to use the minimum average number of bits to represent the source, which is also a collection of random variables. Intuitively, we would like to maximize the amount of network state information provided by each probe, so as to minimize the number of lightpath probes required.

By applying the above approach under a probabilistic failure model where each edge is assumed to fail independently with a prior failure probability, the following main results have been obtained. First, for all-optical networks with Eulerian topologies[4] under a probabilistic link failure model, we show that the fault-diagnosis problem in Network Management is mathematically equivalent to the source-coding problem in Information Theory. The isomorphic mapping suggests an entropy lower

[4] A Eulerian graph contains a path that passes through all the edges without repetition.

bound on the minimum average number of probes required and an information theoretic approach to translating efficient source coding algorithms into efficient fault diagnosis algorithms under the constraint that any lightpath probe can only traverse along a path in the graph[5]. In addition, a family of novel near-optimal polynomial time algorithms, i.e., the run-length probing schemes, have been developed based on run-length codes[27]. Analytical results reveal that its performance (i.e., the average number of lightpath probes required) is always within 5% more than the entropy lower bound and asymptotically approaches the entropy bound as the network becomes more reliable. Second, the run-length probing scheme has been extended to general non-Eulerian networks via two alternative mechanisms: (1)*the disjoint-trail decomposition approach* and (2) *the path-augmentation approach.* Finally, fault diagnosis for practical all-optical networks with both node and link failures has been investigated. To diagnose probabilistic node/link failures, a network transformation has been introduced to convert the original undirected graph into a directed graph: each link in the original graph is replaced by two parallel directed arcs in opposite directions, and each node of degree d is replaced by a $d \times d$ directed complete bi-partite graph [10], where any node in the left column is connected to any node in the right column via a directed arc. Under this transformation, both link and node failures in the undirected graph are mapped into arc failures in the directed graph. Depending on the relative dominance between link failure probability and node failure probability, different probing strategies are obtained through analytical and numerical investigations. Most importantly, analytical and numerical investigation reveals a guideline for efficient probing schemes: ***each probe should be designed to provide approximately 1-bit of information on the network state and the number of probes required is approximately equal to the information entropy of the network states***. Hence the complexity of optical network fault management is fundamentally related to the information entropy of the network state.

[5] This constraint is called the probe feasibility constraint in this thesis.

2.2 Adaptive Fault Diagnosis Problem

2.2.1 Probabilistic Failure Model

In this chapter, all-optical networks are abstracted as undirected graphs. An undirected graph G is an ordered pair of sets (V, E), where V is the set of nodes of size n, and E is the set of edges of size m. In this chapter, we would like to first focus on Eulerian network topologies that have at least one Euler trail [10], which is a sequence of interconnected edges containing all the edges in the topology without repetition. The results obtained are then extended to non-Eulerian topologies in Section 2.6.

In this chapter, the vulnerability of future all-optical networks is characterized initially by the following probabilistic failure model:

1. Nodes are invulnerable(the vulnerable node case will be treated in Section 2.8);

2. Edges are vulnerable, and assumed to fail independently with a prior probability of $p(0 \leq p \leq 1)$;

3. The states of the edges are assumed to stay unchanged over the duration of the fault diagnosis process.

For a given network topology, each edge can be labeled along an Euler trail with an index, $\beta = 1, 2, \ldots, m$. The state of the β^{th} edge is represented by a Bernoulli random variable F_β, called the edge state. Moreover, it is assumed that the edge states, $F_\beta, \beta = 1, 2, \ldots, m$, are statistically independent, and identically distributed with $\Pr(F_\beta = 1) = p$ for an edge failure and $\Pr(F_\beta = 0) = q$ for no failure with $q = 1 - p$.

The network state is defined as a realization of the set of edge states $\{F_\beta\}_{\beta=1}^{m}$, written as $s = f_1 f_2 \cdots f_m \in S = \{0, 1\}^m$. The set of all possible network states is denoted as S. Using the fact that all edges fail independently, one can obtain the prior probability of a particular realization of the network state $s = f_1 f_2 \cdots f_m$ as the product of prior probabilities of all edges, i.e.,

$$\Pr(s) = p^{\sum_{\beta=1}^{m} f_\beta} q^{m - \sum_{\beta=1}^{m} f_\beta}, \tag{2.2}$$

where the term $p^{\sum_{\beta=1}^{m} f_\beta}$ comes from the set of edges with failures, and the term $q^{m-\sum_{\beta=1}^{m} f_\beta}$ originates from the set of edge without failures.

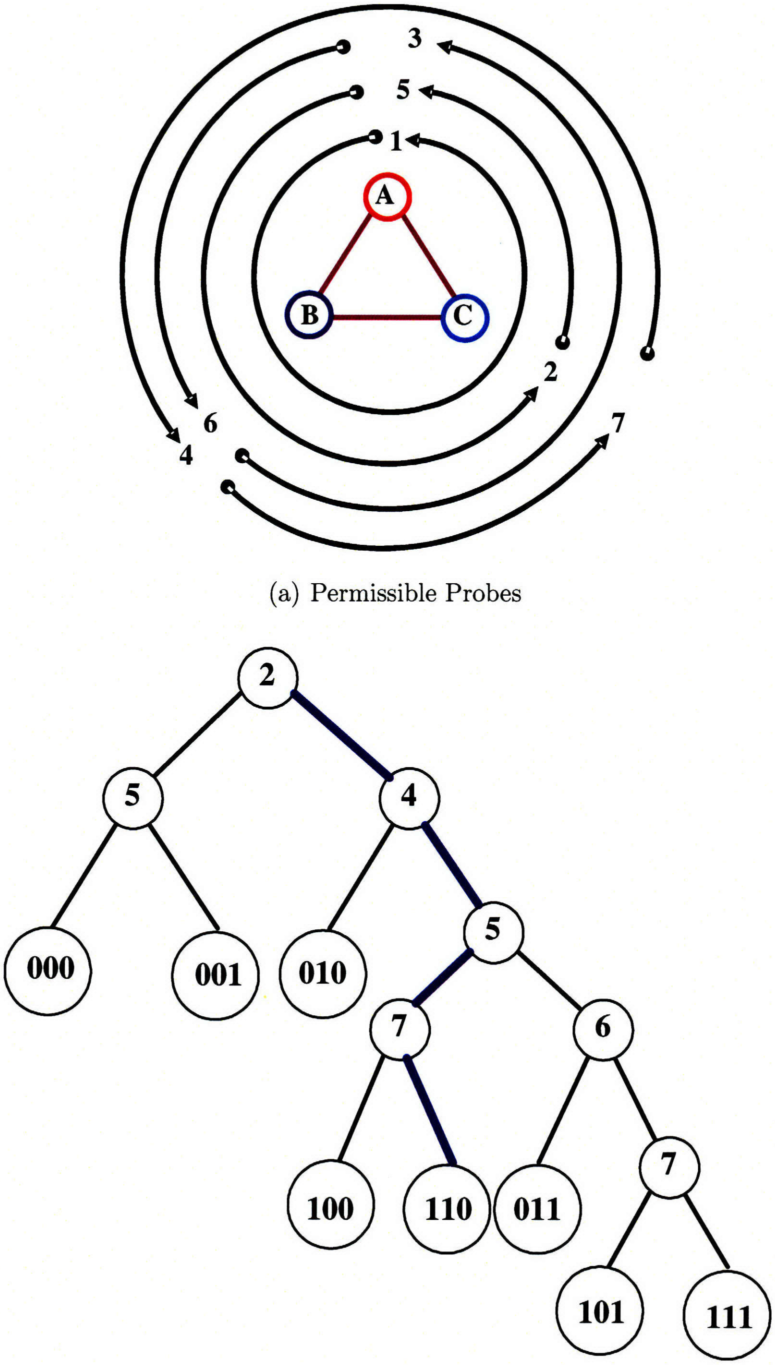
2.2.2 Adaptive Lightpath Probing Schemes

In this chapter, network states are diagnosed via the measurements of end-to-end probing signals. Specifically, each probe corresponds to sending an optical signal along some lightpath. This subsection illustrates a class of adaptive lightpath probing schemes.

A permissible *probe* t over an Eulerian network topology is a trail (a sequence of adjoined edges without repetition) over the graph. Physically, each trail corresponds to a lightpath. For a finite network, we can label each probe with an index $t \in T = \{1, 2, \ldots, |T|\}$ where $|T|$, the cardinality of the set T, is the number of distinct probes over the network. As an example, the 3-node ring topology has 7 permissible probes, as shown in Fig. 2-3(a).

The result of each probe is called the probe syndrome, denoted as r_t. When an optical signal is sent along a permissible lightpath probe, the signal either arrives at the destination when all the edges along the probe are ON, or never reaches the destination (or the quality of the signal is unacceptable) when any of the edges along the probe is OFF. In the former case, the probe syndrome is $r_t = 0$ (or, equivalently $r_t = S$); in the latter case, the probe syndrome is $r_t = 1$ (or, equivalently $r_t = F$).

A *probing scheme* π is a sequential employment of probes such that any network state can be identified. The successive probe can be sequentially determined according to the syndromes of previous probes. Due to this sequential decision making property, any probing scheme is equivalent to a **binary decision tree**, whose leaves are network states and inner nodes are probes. For example, a probing scheme for the 3-node ring network is shown in Fig. 2-3(b), where each inner node is labeled with the probe employed. We adopt the convention that at any inner node, if the probe syndrome is 0 (no failure), the subsequent probe is given in the left child; otherwise if the probe syndrome is 1, the probing process continues on the right child.



(a) Permissible Probes

(b) Probing Scheme

Figure 2-3: (a)Set of permissible probes over the three-node ring topology. Total number of probes is 7. Each probe is indexed with a number near the arrow. (b)Probing scheme (decision tree) for the three-node ring topology.

The set of all possible probing decision trees for the network topology G is denoted as $\Pi(G)$. Without loss of optimality, the following properties are assumed for any efficient probing scheme:

1. A probe will not be employed if its syndrome can be inferred from previous syndromes. For example, if a probe returns no failure, it means that no edge in that probe has failed; hence no probe that involves only a subset of these edges is performed thereafter.

2. When two probes are expected to reveal the same information, we would like to consistently choose the probe spanning fewer number of edges for convenience. For example, if the state of an individual edge is known, then one should not start or end a probe with this edge, since dropping it loses no information.

2.2.3 Fault-Diagnosis Problem Formulation

In this chapter, the design metric used to measure the proficiency of any fault-diagnosis scheme is its diagnostic effort, which is taken to be proportional to the average number of lightpath probes required to identify the network state. Our design objective is to minimize the average number of lightpath probes.

Each probe $t \in T$, if employed, is assumed to contribute one unit of diagnosis effort. Consequently, for a given fault-diagnosis scheme π, the effort to diagnose the state s, denoted by l_s^π , is equal to the number of lightpath probes from the root to the leaf node s in the probing decision tree π. We call it the probing depth of the state s. For example, as shown in Fig. 2-3(b), the probing syndrome of state 110 is 1101 and thus the probing depth is 4.

Given a probing scheme $\pi \in \Pi(G)$, the average number of probes required[6] is

$$\mathcal{L}_\pi = \sum_{s \in S} \Pr(s) l_s^\pi, \tag{2.3}$$

[6] The testing result of each probe is represented with 1 bit data (1/0). A similar objective has been pursued in [33], where Ho and Medard sought to minimize the number of network diagnosis information bits.

where $\Pr(s)$ is the prior probability of this state. Notice that the average number of probes scales with the size of network. To suppress the scaling effect, we focus on the average number of probes per edge, defined as

$$\bar{\mathcal{L}}_\pi = \frac{1}{m} \sum_{s \in S} \Pr(s) l_s^\pi, \tag{2.4}$$

where m is the number of edges in the network topology.

For a given network topology G, we would like to find the optimal probing scheme that minimizes the average number of probes per edge, and thus to minimize the diagnostic effort. Mathematically, it is formulated as the following optimization problem,

$$\begin{aligned} \min_{\pi} \quad & \bar{\mathcal{L}}_\pi = \frac{1}{m} \sum_{s \in S} \Pr(s) l_s^\pi, \\ s.t. \quad & \pi \in \Pi(G). \end{aligned} \tag{2.5}$$

The resulted minimum average number of probes per edge is written as

$$\bar{\mathcal{L}}^* = \min_{\pi \in \Pi(G)} \{ \frac{1}{m} \sum_{s \in S} \Pr(s) l_s^\pi \} = \frac{1}{m} \sum_{s \in S} \Pr(s) l_s^{\pi^*}, \tag{2.6}$$

where π^* is the optimum probing decision tree.

2.3 Optimum Fault-Diagnosis Schemes

In this section, we characterize some properties of the optimal probing schemes for Eulerian networks and derive the achievable performance of these schemes. The insights developed in this section will provide guidance for designing near-optimum diagnosis schemes.

2.3.1 Mapping between the Source-Coding Problem and the Fault-Diagnosis Problem

Structural similarity between these problems suggests that there is a mathematical mapping between the fault-diagnosis problem in Network Management and the source-coding problem in Information Theory, as to be shown in this subsection. This mapping provides theoretical and engineering insights for efficient network diagnosis scheme design.

Given a probing scheme $\pi \in \Pi(G)$, one can denote the probe syndrome of network state s as $r(s) = r(t_1^s)r(t_2^s)\cdots r(t_{l_s^\pi}^s)$, where l_s^π is the probing depth of state s , and $\{t_1^s, t_2^s, \ldots, t_{l_s^\pi}^s\}$ is the sequence of probes employed to identify state s. For example, the sequence of probes for state $s = 110$ in Fig. 2-3(b) is $\{2, 4, 5, 6\}$ and the probe syndrome is $r(s) = 1101$. For a given probing scheme π, we call the set of probe syndromes as $R(\pi) = \{r(s), s \in S\}$. In the following, we will show that the set of probe syndromes constitutes a uniquely-decodable code[7] for the set of network states [1].

Theorem 2.1. *For any valid probing decision tree $\pi \in \Pi(G)$, the set of probe syndromes $R(\pi) = \{r(s), s \in S\}$ forms a uniquely-decodable code for the set of network states S.*

Proof. We proof this theorem by contradiction. By the definition of a uniquely-decodable code[18], we know that each source symbol should be mapped into a different non-empty bit string.

If the set of probe syndromes $R(\pi)$ does not forms a uniquely-decodable code for the set of network states S, we can always find two distinguished network states that have the same probe syndrome. That is, there exist two network states s_1 and s_2, and $r(s_1) = r(s_2)$. In this case, the probing scheme π cannot distinguish between s_1 and s_2, and thus is not valid.

[7] A code is uniquely decodable if each source symbol is mapped into a different non-empty bit string.

Table 2.1: Similarity between Fault Diagnosis and Source Coding

Fault Diagnosis	Source Coding
Network states	Source symbols
Prior probability of states	Prior probability of symbols
Probe syndromes	Coded symbols
Average number of probes	Average code length

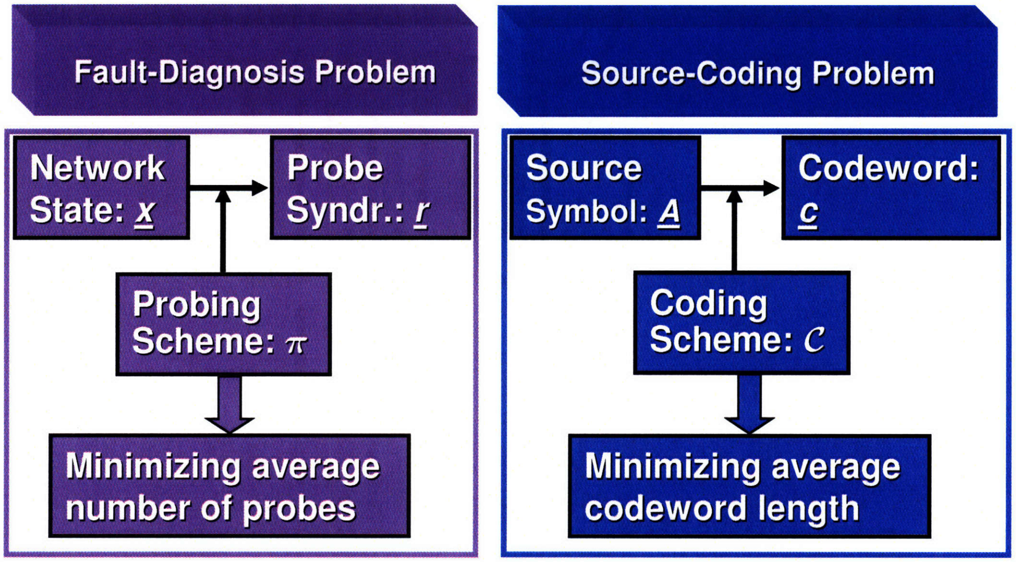

Fault-Diagnosis Problem
Network State: $\underline{x}$
Probe Syndr.: $\underline{r}$
Probing Scheme: π
Minimizing average number of probes
Source-Coding Problem
Source Symbol: $\underline{A}$
Codeword: $\underline{c}$
Coding Scheme: $\mathcal{C}$
Minimizing average codeword length

Figure 2-4: The structural similarity between the fault-diagnosis problem in network management and the source-coding problem in information theory suggests a mathematical mapping between them.

Therefore, for any valid probing scheme, the set of probe syndromes forms a uniquely-decodable code for the set of network states. □

Intuitively, there are mappings among various concepts between the fault-diagnosis problem and the source-coding problem, as illustrated in Table 2.1. The objective of the fault-diagnosis problem, i.e., to design a probing scheme mapping the set of network state into a set of probe syndromes such that the average syndrome length is minimized, is similar to the objective of the source-coding problem in Information Theory, i.e., to design a coding scheme mapping the set of source alphabets into a set of codewords such that the average codeword length is minimized. This structural similarity between the fault-diagnosis problem and the source-coding problem, as illustrated in Fig. 2-4, suggests an isomorphic mapping between them.

The mapping between the fault-diagnosis problem and the source-coding problem suggests that the rich set of results from the source-coding literature can be exploited to obtain the fundamental limits of the fault-diagnosis problem and good source-coding algorithms can be used to construct efficient fault-diagnosis schemes.

First, the mapping between the fault-diagnosis problem and the source-coding problem suggests a lower bound on the minimum average number of probes per link. It follows from the lossless source coding theorem [18] that the minimum average number of probes per link is lower bounded by the information entropy of individual link, i.e.,

$$\bar{\mathcal{L}}_{\pi}^{*} \geq H_b(p), \tag{2.7}$$

where $H_b(p)$ is the Shannon binary information entropy function. This fundamental limit can be understood intuitively as follows. The degree of uncertainty of the unknown network state can be represented by the information entropy of the network state. In the schemes considered here, each probe is either successful or fails and thus provides at most one bit of state information. It follows that the number of probes required should be larger than or equal to the information entropy of the network state.

Second, this mapping suggests an approach to designing efficient fault-diagnosis schemes by transforming (near)-optimal source-coding algorithms. However, not all source-coding algorithms, e.g. the optimal Huffman coding algorithm, can be transformed into fault diagnosis algorithms, due to the limitation imposed by the physical structure of lightpath probes. The solutions to both the source-coding problem and the fault-diagnosis problem can be understood as a sequence of YES/NO questions. In the source-coding context, questions can be asked about any subset of links in the network, which are not necessarily connected, and questions can be asked about a network state of mixed 1 and 0 regarding to the chosen subset of links. For example, the question like "Is Link 1 UP and link 3 DOWN and link 5 UP?" is allowed. On the other hand, in the fault-diagnosis context, not all of such questions are physically realizable probes, which can only probe consecutive links and ask questions whether all the links in the probe are UP, corresponding to one particular class of questions such as "Are links 2, 3, 4 all UP?" Thus, the nature of permissible probes imposes an extra restriction on the class of questions that can be asked. In our research, we refer to this restriction as the ***probe feasibility constraint***, and study the fault-diagnosis problem, or the equivalent source-coding problems, under this probe feasibility constraint.

2.3.2 Link-Wise Probing Schemes

As discussed in Section 2.1, the link-wise probing scheme, which probes each individual edge separately, is in general not optimal, especially when the failure probability of each edge, p, is small. On the other hand, if each edge fails with a high probability (which is unrealistic), the link-wise probing scheme becomes more efficient. The following theorem states the condition under which the link-wise probing scheme is optimal[8].

Theorem 2.2. *For any non-trivial network topology with a connected subgraph of more than one edge, the link-wise probing scheme is optimal if and only if the edge failure probability is larger than* $\frac{3-\sqrt{5}}{2}$ *(the golden ratio).*

The proof of this theorem is presented in Section A.1.

The theorem suggests that the link-wise probing schemes based on single-hop tests, as used in the electrical networks and some of the current optical monitoring schemes, are strictly sub-optimal in all-optical networks for $p < (3-\sqrt{5})/2 (\approx 0.382)$, which is the situation in most network monitoring scenarios.

According to the theorem, the link-wise probing scheme is optimal for the case of $p > 1/2$. As p increases to 1, the lower bound $H_b(p)$ decreases, while the optimal

[8] This theorem is similar to the break-point theorem of the group testing problem [63]: when the probability with which a sample is defected is higher than $\frac{3-\sqrt{5}}{2}$, it is optimal to test each individual sample to minimize the number of tests.

approach is always 1 probe per edge. Intuitively, for large values of p , if all the edges fail with a high probability, we could reduce the number of probes required if there were a probe to test the scenario where a collection of edges are all in OFF states. Since such a probe cannot be implemented, our information theoretical bound becomes less meaningful in the range of $p > 1/2$. Nevertheless, in almost all practical situations, the edge failure probabilities are small, thus in the remainder of this chapter, we always assume $p \leq 1/2$.

2.3.3 Optimal Probing Scheme for Lightpath with Single Failure

In this sub-section as a special case to illustrate the technique, we will focus on a linear network topology (i.e., bus) with h edges and only one failure if it happens. In an all-optical network context, this can be understood as the case where there is only one failure along a particular lightpath. Conditioning on the fact that there is one and only one faulty edge, each edge has a uniform distribution of being the faulty one. For this case, the optimum probing scheme to minimize the average number of probes (per edge) has been found in [73].

The optimum probing scheme works as follows. Given that the linear network topology has h edges and the number of faulty edges is 1, we first split the path of length h into two sub-paths of length h_l and h_r according to the following criteria:

$$h_l = g(h) = \begin{cases} 2^{\lfloor \log_2 h \rfloor - 1}, & \text{if } \; 2 \leq h \leq 3 \cdot 2^{\lfloor \log_2 h \rfloor - 1} \\ h - 2^{\lfloor \log_2 h \rfloor}, & \text{if } \; 3 \cdot 2^{\lfloor \log_2 h \rfloor - 1} < h \leq 2^{\lfloor \log_2 h \rfloor + 1} - 1 \end{cases} \tag{2.8}$$

and

$$h_r = h - h_l. \tag{2.9}$$

Next, the first sub-path of length h_l is probed. If the syndrome is 1, meaning the faulty edge is in the first sub-path, the scheme continues to split the first sub-path according to rule (2.8) and probe the resulted first sub-path. If the syndrome is 0

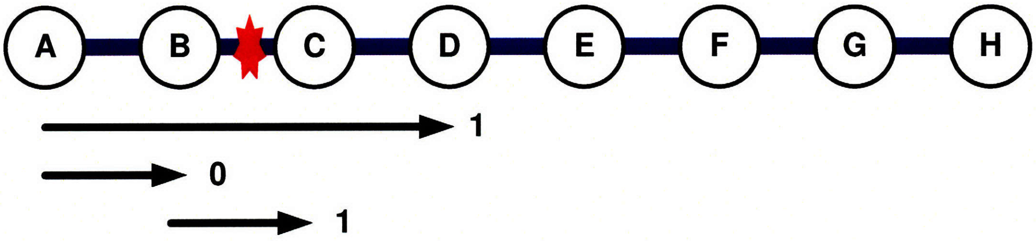


Figure 2-5: Optimal 2^α-splitting probing scheme for the linear network with seven edges and one failure at edge BC. The syndrome is 101.

meaning that the first sub-path is fault-free and the faulty edge is in the second sub-path, the scheme splits the second sub-path using (2.8) and probe the resulted first sub-path. This process continues until the faulty edge is located.

Such an optimum probing scheme can be understood to maximize the information gain of each probe with the concern of keeping a balanced probing decision tree. Intuitively, when dividing the path into two sub-paths of length h_l and h_r, respectively, it is desirable to cut the path into equal halves, thus the probability that the faulty edge is in the first sub-path is as close to 1/2 as possible. Such a probe provides as much information gain as possible. However, such approach is in fact only locally optimal: it makes the probe over the current h_l edges information efficient, but may cause the subsequent probes to be inefficient. In fact, the optimal splitting rule (2.8) can be identified by solving the following optimization problem,

$$h_l = \min_{\alpha^* = \arg\min |2^\alpha - (h - 2^\alpha)|} \{2^{\alpha^*}, h - 2^{\alpha^*}\}. \tag{2.10}$$

This optimization suggests that it is globally optimal to balance the lengths of two split sub-paths while making sure one of the sub-paths has a length of an integer power of 2. The resulted probing scheme is called the "2^α-splitting" probing scheme. Note that, in both local and global optimums, we are trying to balance the probabilities of syndrome 0 and syndrome 1, indicating that each efficient probe should provide approximately one bit of network state information.

To illustrate the "2^α-splitting" probing scheme, let us consider a linear topology with 7 edges. As shown in Fig. 2-5, we assume that the 2nd edge (BC) fails. In the first step, since $h = 7$, the splitting rule suggests that the length of next probe should be $h_l = 3$. The resulting probing syndrome is 1, indicating that the failure is within the first sub-path. In the second step, since $h = 3$, the length of the next probe should be $h_l = 1$. The probing syndrome is 0, indicating that the failure is within the second sub-path. Now, the length is $h = 2$. The scheme probes the first sub-path of length 1, resulting the probing syndrome of 1. Therefore, the edge BC fails. It follows that the probing algorithm outputs the syndrome 101 for the network state 0100000.

It is also important to observe that, if the problem is changed into the scenario where there is at least one faulty edge in the line network, and our objective is to locate the first (leftmost) one, the optimal probing scheme is exactly the same as the one described above, since the algorithm never tests a sub-path without knowing that every edge to the left is fault free. It turns out that this is crucial in developing the run-length probing algorithm in the next section.

2.4 Run-Length Probing Schemes

Mathematically, the optimization problem in (2.5) is, in fact, equivalent to designing an optimal binary decision tree for a decision problem. Previously, the design of optimal binary decision tree has been approached with well-established dynamic programming algorithms [47, 46]. However, it has been shown in [49] that the sequential diagnosis problem is Co-NP complete[9], meaning that the computational complexity of probing algorithms grows exponentially with the network size. From a practical point-of-view, it would be wise to develop simpler algorithms to design near-optimum probing schemes.

[9] A decision problem C is Co-NP-complete if it is in Co-NP and if every problem in Co-NP is polynomial-time many-one reducible to it. In complexity theory, the complexity class Co-NP-complete is the set of problems that are the hardest problems in Co-NP, in the sense that they are the ones most likely not to be in P. If you can find a way to solve a Co-NP-complete problem quickly, then you can use that algorithm to solve all Co-NP problems quickly.

In this section, as a trade-off between complexity and performance, we will develop a class of near-optimum probing schemes whose computational complexity is polynomial order of the network size. This class of near-optimum probing schemes have probe syndromes consisting of a string of run-length codes[28, 27, 60]. It can be shown that this probing scheme is asymptotically optimal in that it achieves the minimum average number of probes per edge for large enough networks. Furthermore, the run-length probing algorithm is easy to implement and its performance can be obtained in closed-form.

2.4.1 Introduction to Run-Length Codes

In this subsection, we describe the run-length code, which was first proposed by Golomb[28], and generalized by Gallager[27] and extended by Tanaka[60]. We will transform the run-length code into an efficient fault-diagnosis algorithm in next subsection.

For the source-coding problem, we are interested the set of source alphabets

$$\mathbf{Z} : \{z_i = 0^i 1 : i = 0, 1, \ldots\}, \tag{2.11}$$

(i.e., a run of i 0's followed by one 1) with a geometric probability distribution,

$$\Pr(z_i = 0^i 1) = (1-p)^i p, i \geq 0, \tag{2.12}$$

for some arbitrary p with $0 < p < 1$. This distribution arises in run-length coding, where one has an identical and independent binary source, with p being the probability of a ONE.

Instead of looking at the original set of source symbols $\mathbf{Z}$, we would like to first investigate a set of truncated source alphabets and extend the result into the original set of source symbols.

For any $0 < p < 1$, there exists a unique positive integer K such that

$$(1-p)^K + (1-p)^{K+1} \leq 1 < (1-p)^K + (1-p)^{K-1}. \tag{2.13}$$

Notice that the unique positive integer K from this inequality maximizes the information gain of any probe, given by (2.1). Solving this inequality, one can obtain the maximum probing length as,

$$K = \lceil -\log_{1-p}(2-q) \rceil. \tag{2.14}$$

Let us first look at the optimal source code for the set of truncated source alphabets,

$$\widetilde{\mathbf{Z}} : \{\widetilde{z}_i = 0^i 1 : i = 0, 1, \ldots K-1\}, \tag{2.15}$$

where each source alphabet has the following prior probability,

$$\Pr(\widetilde{z}_i = 0^i 1) = \frac{(1-p)^i p}{1-(1-p)^K}, \quad 0 \leq i \leq K-1. \tag{2.16}$$

It can be seen that each prior probability is the conditional probability given that $i < K$. Using the Taylor expansion, we obtain

$$\frac{(1-p)^i p}{1-(1-p)^K} = \sum_{j=0}^{\infty} p(1-p)^{i+jK}. \tag{2.17}$$

It follows that, each of the prior probabilities in (2.16) can be regarded as the accumulation of the probabilities of all the run-lengths in the original sources that are in the same equivalence class (under modulo K). As an example, for $K = 2$, the truncated source symbols are $\{1, 01\}$. The symbol $\{1\}$ represents the class of source alphabets $\{0^i 1 : i \mod 2 = 0\}$ in the original set of source symbols, and the symbol $\{01\}$ represents the class of source alphabets $\{0^i 1 : i \mod 2 = 1\}$ in the original set of source symbols. Therefore, the original source is decoupled into two types of source alphabet: $\widetilde{Z}\{0^i 1 : 0 \leq i \leq K-1\}$ and $\widehat{Z} = \{0^{jK} : j \geq 0\}$. This structure suggests that the optimal code for the original source must be the concatenation of the optimal code for the truncated source $\widetilde{Z}$ and the optimal code for the source set $\widehat{Z}$.

The optimal code for the truncated source $\widetilde{Z}$ is the Huffman code, which can be derived as follows. Notice from (2.13) that the sum probability of the two least likely source alphabets exceeds the probability of the most likely one[10]. Therefore, the lengths of the optimal codewords for the truncated source $\widetilde{Z}$ can differ at most by one, i.e., the difference between the largest length and the smallest one is not more than 1.

Now, let us derive the length distribution of the optimal codewords for the truncated source $\widetilde{Z}$. Since the tree for the optimal code is complete and the lengths can differ by at most one, the codeword length must be $\lfloor \log_2 K \rfloor$ and $\lfloor \log_2 K \rfloor + 1$ if K is not a power of two. Using the optimality condition of the Huffman code, we can assume that the length of first x codewords is $\lfloor \log_2 K \rfloor$, and the length of other $K - x$ codewords is $\lfloor \log_2 K \rfloor + 1$. Since the code tree is full at the height $\lfloor \log_2 K \rfloor$, we must have the following condition satisfied,

$$x + \frac{1}{2}(n - x) = 2^{\lfloor \log_2 K \rfloor}. \tag{2.18}$$

Solving (2.18), we obtain

$$x = 2^{\lfloor \log_2 K \rfloor + 1} - n. \tag{2.19}$$

This result indicates that the optimal code for the truncated source $\widetilde{Z}$ is to use codewords of length $\lfloor \log_2 K \rfloor$ for alphabets $\{\widetilde{z}_i : i < 2^{\lfloor \log_2 K \rfloor + 1} - n\}$, and codewords of length $\lfloor \log_2 K \rfloor + 1$, otherwise. Some of the properties of the optimal code tree for the truncated source $\widetilde{Z}$ are list below:

1. The code tree is complete[11] at height $\lfloor \log_2 K \rfloor + 1$. The height of a tree is the longest path from the root to any of its leaves.

[10] The probability of the most likely alphabet is $\frac{p}{1-(1-p)^K}$. The sum probability of the two least likely source alphabets is $\frac{p((1-p)^{K-1}+(1-p)^{K-2})}{1-(1-p)^K}$.

[11] A complete tree has no internal node with single child.

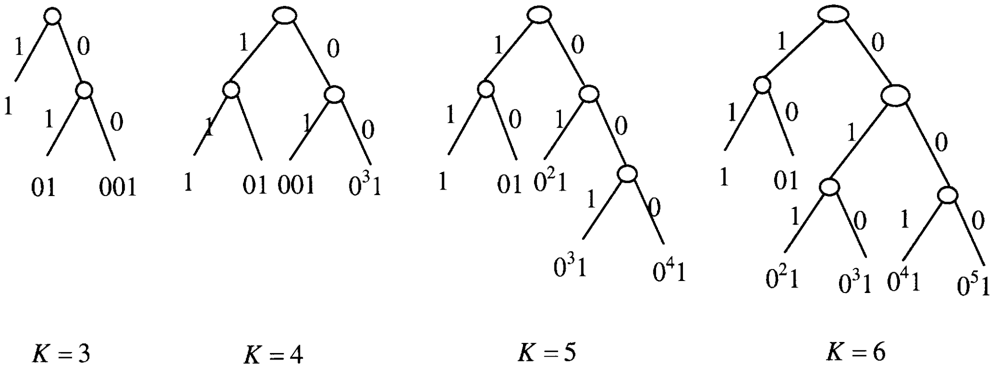


Figure 2-6: The optimal Huffman code tree for the truncated source $\widetilde{Z}$ with $K = 3, 4, 5, 6$.

2. The code tree is full at height $\lfloor \log_2 K \rfloor$. The $K - 2^{\lfloor \log_2 K \rfloor}$ leaves from the right are bifurcated to the height $\lfloor \log_2 K \rfloor + 1$.

3. From the left to the right, the leaves of the code tree can be labeled with $\{0^i 1 : 0 \leq i \leq K - 1\}$, which corresponds to the K source symbols in the truncated source.

In fact, this is not the only optimal code tree for the truncated source $\widetilde{Z}$. However, for any optimal code tree, we can change the branch labels such that the resulted code tree satisfies the above three properties. For any optimal code, the tree is full at height $\lfloor \log_2 K \rfloor$. It follows that we rearrange the first $2^{\lfloor \log_2 K \rfloor + 1} - K$ leaves from the left as $\{0^i 1 : 0 \leq i < 2^{\lfloor \log_2 K \rfloor + 1} - K\}$, without loss of optimality. The same procedure can be done to the leaves at height $\lfloor \log_2 K \rfloor + 1$. As an examples, Fig. 2-6 shows the optimal code trees for $K = 3, 4, 5, 6$. Without loss of optimality, we label all the left branches with 1 and all the right branches with 0. This label strategy is consistent with the syndromes of probes we employ at inner nodes of probing decision trees.

The average code length of the optimal code for the truncated source $\widetilde{Z}$ can be derived as,

$$\begin{aligned} \widetilde{L} &= \lfloor \log_2 K \rfloor \left(\sum_{i=0}^{k-1} \frac{p(1-p)^i}{1-(1-p)^K} \right) + (\lfloor \log_2 K \rfloor) \left(\sum_{i=k}^{K} \frac{p(1-p)^i}{1-(1-p)^K} \right) \\ &= \lfloor \log_2 K \rfloor + \frac{(1-p)^k - (1-p)^K}{1-(1-p)^K}, \end{aligned} \tag{2.20}$$

where $k = 2^{\lfloor \log_2 K \rfloor + 1} - K$ is the number of codewords with length $\lfloor \log_2 K \rfloor$.

The conjecture for the optimal code for the original source $\mathbf{Z}$ has been proved by Gallager in [27]. Indeed, the optimal code for the original source alphabet $z_i = 0^i 1 : i \geq 0$ with a Geometric distribution is a concatenation of two prefix codes: the optimal Huffman code for the alphabet $\widetilde{z}_{i'} = 0^{i'}1, i' = i \mod K$ in the truncated source $\widetilde{Z}$, and the unary code for the alphabet $0^{jK} : j = \lfloor i/K \rfloor$ in the source $\widehat{Z}$, i.e., $c(z_i) = c(\widetilde{z}_{i'})u(j)$. The unary code for the alphabet $0^{jK} : j = \lfloor i/K \rfloor$ is the codeword with j zeros followed by a single one, i.e., $u(j) = 0^j 1$. Without loss of optimality, we can reverse the order of the two codes, thus encoding the alphabet z_i into the unary code for 0^{jK} followed by the optimal codeword for the alphabet $\widetilde{z}_i$. Using this equivalent code structure, the encoder of the run-length code works as follows:

1. The encoder counts up to K zeros from the source and then produces a single zero at the output.

2. When a one appears from the source, the encoder produces a one, terminating the unary code.

3. The encoder produces the optimal codeword for the position of the incoming one within the sub-block of K digits.

The run-length code can be understood as an optimal code for the following modified source,

$$\mathscr{Z} = \{0^K, 0^i 1\}_{i=0}^{K-1}. \tag{2.21}$$

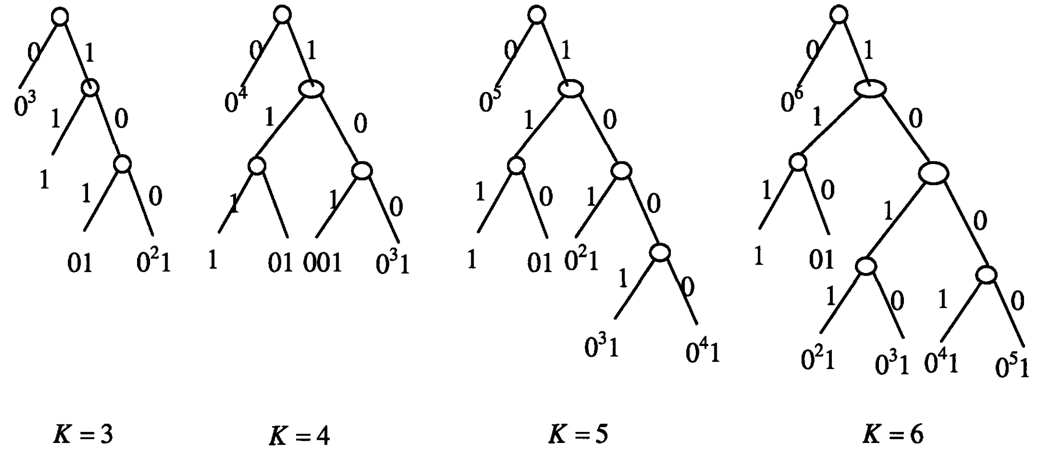


Figure 2-7: The run-length code trees for the source $\mathcal{Z}$ with $K = 3, 4, 5, 6$.

The prior probability distribution of the possible symbols in Z is given by

$$\Pr(0^i1) = (1-p)^i p, \quad 0 \le i \le K-1, \tag{2.22}$$

and

$$\Pr(0^K) = (1-p)^K. \tag{2.23}$$

For this set of source alphabets, the run-length code trees for $K = 3, 4, 5, 6$ are given in Fig. 2-7. The structure of these coding trees are simple: the codeword for 0^K is 0, and the codewords for $0^i1 : 0 \le i \le K-1$ are the concatenation of 1 and the optimal codewords for $\tilde{z}_i$. We call the resulting code as the order K run-length code.

In the rest of this subsection, we will calculate the code rate for the run-length code, which is defined as the average number of encoded bits per intermediate symbol. For a run-length code with sub-block size of K, the average number of pattern bits per intermediate symbol (i.e., the set of symbols $\mathcal{Z}$) is given by

$$\bar{l}_z = \sum_{i=0}^{K-1} (1-p)^i p + K(1-p)^K = \frac{1-(1-p)^K}{p}. \tag{2.24}$$

The average number of run-length coded bits per intermediate symbol is given by

$$\bar{l}_c = 1\cdot(1-p)^K+(1+\widetilde{L})(1-(1-p)^K) = (\lfloor\log_2 K\rfloor+1)(1-(1-p)^K)+(1-p)^k, \quad (2.25)$$

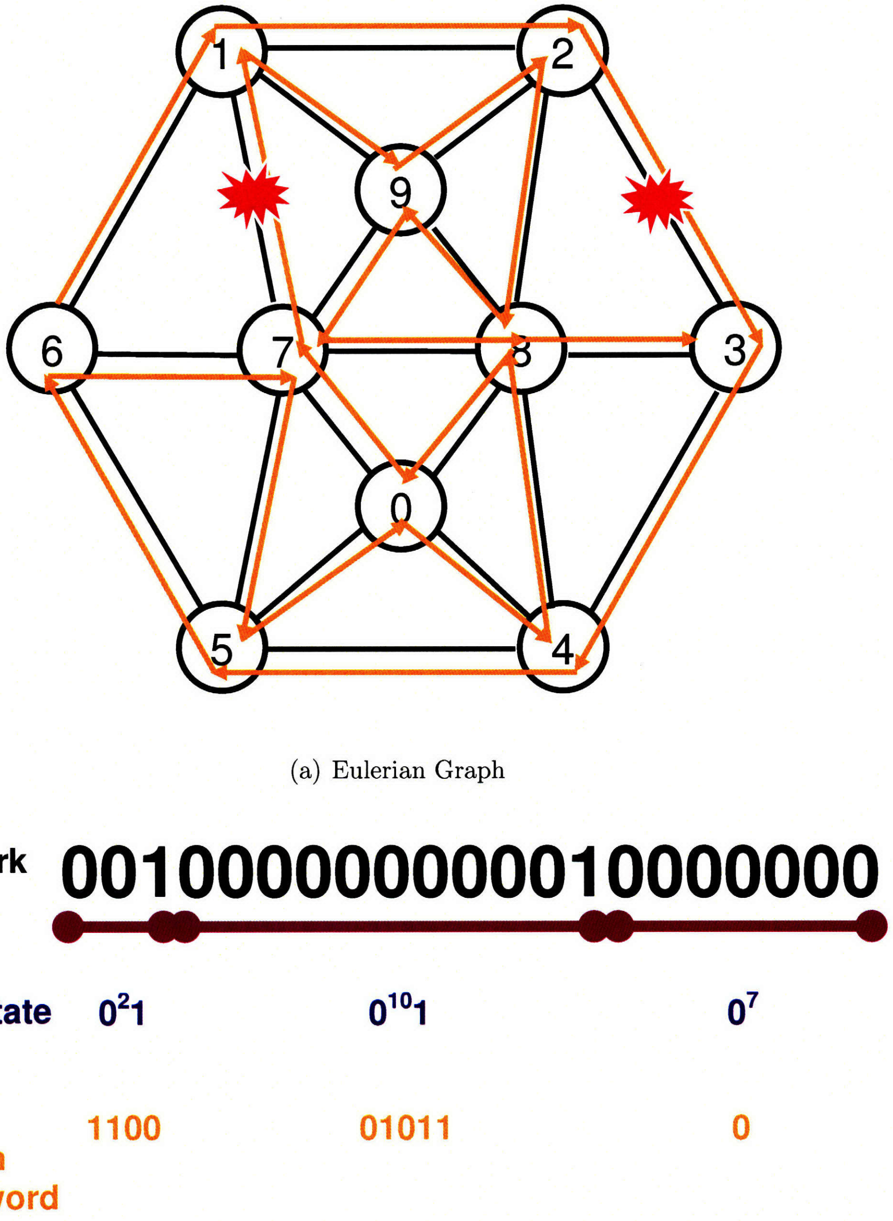
where $k = 2^{\lfloor\log_2 K\rfloor+1} - K$. Therefore, the coding rate of the order K run-length code is given by

$$\mathcal{R}(p) \triangleq \frac{\bar{l}_c}{\bar{l}_z} = p\left(\lfloor\log_2 K\rfloor + 1 + \frac{(1-p)^k}{1-(1-p)^K}\right). \quad (2.26)$$

2.4.2 Probing Schemes Based on Run-Length Codes

As a result of the mathematical mapping between the source-coding problem and the fault-diagnosis problem, we have indicated in Section 2.3 that efficient source-coding algorithms can be transformed into scalable fault-diagnosis schemes, under the probe feasibility constraint. In this subsection, we develop a class of scalable fault-diagnosis algorithms by transforming the run-length codes. In particular, the special structure in the run-length codeword makes it possible to transform the run-length coding algorithm into a fault-diagnosis scheme.

For an Eulerian network, we can introduce a natural order to any network state by indexing all the link states along an Euler trail in the network. Specifically, any network state must have the format of $0^{i_1}10^{i_2}1\cdots0^{i_L}$, where $i_1, i_2, \ldots, i_L$ are non-negative integers and 0^i means a run of i '0', and each of the segments, 0^i1, is called a sub-state. Since any probe can locate at most one faulty link at a time, each of such sub-states should be encoded separately. For example, for the Eulerian graph shown in Fig. 2-8(a), the network state can be decomposed into a set of sub-states, i.e., $\{0^21, 0^{10}1, 0^7\}$. This idea suggests that one should, instead of coding independent binary input streams, code the symbol set of $\mathbf{Z} = \{0^i1\}_{i=0}^{\infty}$ with a geometrical probability distribution. In the context of source coding, the optimal code for the set $\mathbf{Z} = \{0^i1\}_{i=0}^{\infty}$ with geometrical distributions has been shown as the run-length code [27], as introduced in Section 2.4.1. Alternatively, any network state $0^{i_1}10^{i_2}1\cdots0^{i_L}$ can be decoupled into a set of source symbols from the symbol set $\mathcal{Z} = \{0^K, 0^i1\}_{i=0}^{K-1}$. Notice that the last sub-state of 0^{i_L} can decoupled into a sequence of 0^K plus $0^{i_L \bmod K}$.



(a) Eulerian Graph

(b) Network State Decomposition

Figure 2-8: For an Eulerian graph, network state along any Euler trail can be decomposed into a sequence of source symbols in the set **Z**. It follows that we should encode each sub-state instead of the original network state.

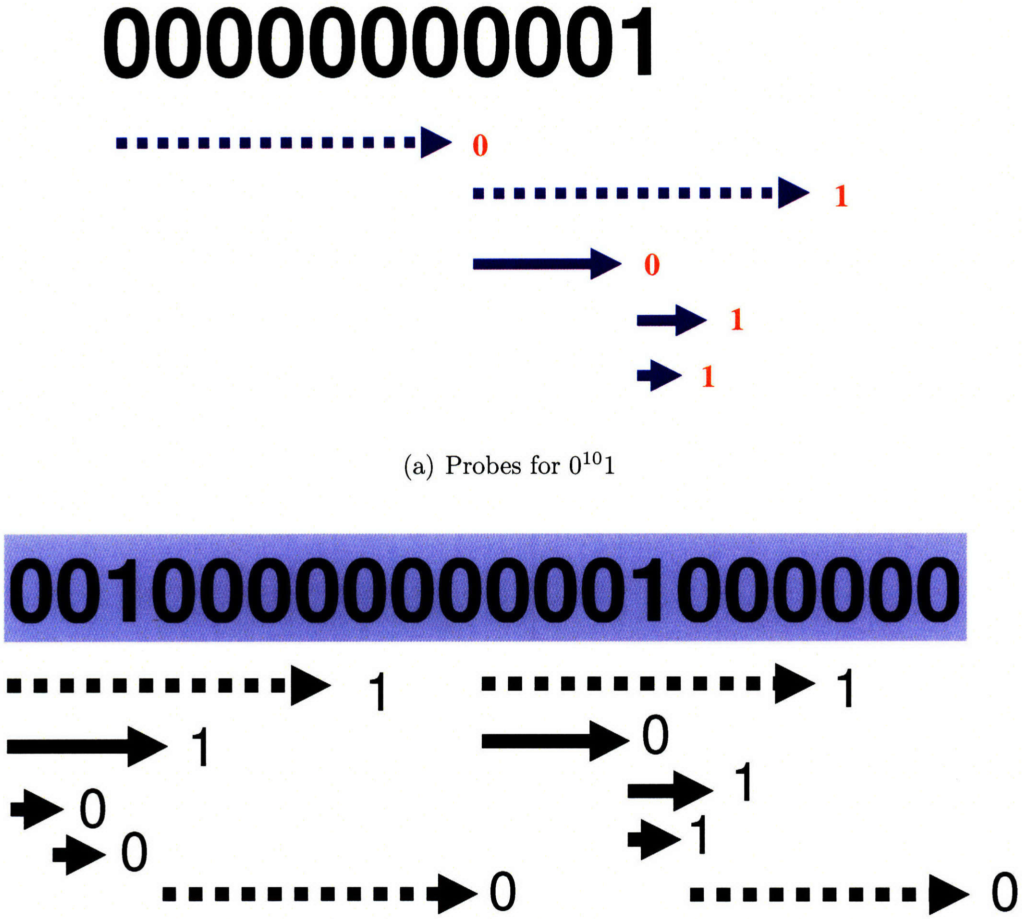
Notice that the symbol of $0^{i_L \mod K}$ is not included in the run-length source symbol set $\mathcal{Z}$.

The special structure of a run-length codeword permits us to translate the run-length coding algorithm into a corresponding fault diagnosis algorithm under the probe feasibility constraint. The run-length codeword of alphabet $0^i1 : i > 0$ is a concatenation of two prefix codes: the unary code for the integer $\lfloor i/K \rfloor$ followed by the Huffman code for the alphabet $0^{i \mod K}1$, where $K = \lceil -\log_{1-p}(2-p) \rceil$ is defined as the maximum probing length. These two codes can be transformed as follows:

- The unary code for an integer j is the codeword with j zeros followed by a single one, i.e., $u(j) = 0^j1$. In the fault diagnosis context, such a unary code can be implemented by sequentially sending $j + 1$ back-to-back probes, each spanning K edges along the Euler trail. The first j probes will return syndrome ZERO meaning that all the probed edges are fault-free, and the $(j + 1)^{th}$ probe will return ONE meaning that at least one of the edges in the $(j + 1)^{th}$ probe fails. For example, as shown in Fig. 2-9(a) for the sub-state $0^{10}1$ with $K = 7$, we send 2 back-to-back probes of length 7 along the Euler trail. The probe syndrome is 01, which is also the unary code for the integer $j = (10 \mod 7) = 1$. Intuitively, each probe of length K here maximizes the amount of network state information, for a given link failure probability.

- The Huffman codeword for the alphabet $0^k1 (k = i \mod K)$ can be implemented by the 2^α-splitting binary searching algorithm developed in [68, 73]. This algorithm balances the Huffman code tree and maximizes the amount of information revealed by each probe for an update conditional link failure probability. For example, as shown in Fig. 2-9(a) for the sub-state $0^{10}1$ with $K = 7$, the Huffman code for 0^31 is 011, which can be implemented with the 2^α-splitting binary searching algorithm.

Hence, the special structure of the run-length code guarantees its transferability to a corresponding fault diagnosis algorithm, called the '*run-length probing scheme*'.



(a) Probes for $0^{10}1$

(b) Run-Length Probing Scheme

Figure 2-9: Demonstration of run-length probing scheme over a network. It contains a sequence of concatenations of two phases: the fault detection phase (dotted lines) and the failure localization phase (solid lines).

Using the run-length code, we can derive the run-length probing scheme (RLPA). First, a probe is sent over a set of K consecutive edges along the Euler trail. If all the edges are fault-free, we move onto the next set of consecutive K edges along the Euler trail. If on the other hand the first probe suggests that there is at least one faulty edge in this group, we can employ the "2^{α}-splitting" probing scheme, described in Section 2.3 to locate the first faulty edge. The process resumes with another group of K edges along the trail right after this faulty edge. At the end of the Euler trail, we can simply encode any network sub-state of $0^i : 0 \leq i \leq K-1$ with a codeword 0. This can be achieved by a single probe to test the last i edges and the resulting performance degradation is negligible when the network is large enough. An example of the run-length probing scheme is illustrated in Fig. 2-9(b). A detailed description of this algorithm is given below.

Let P_i^j be a path (i.e., a permissible probe) that covers edge i to edge j over the Euler Trail. When that path is being probed, it is active. Let h_l denote the number of edges in the current active path, and h_r denote the number of edges in the subsequent active path which is to be probed if all the edges in the current active path are fault-free or the current active path has only one edge and it fails, and let i be the start point of the active path. The run-length probing scheme is given by the following pseudo-code.

To understand the run-length probing scheme pictorially, we illustrate the corresponding probing decision trees for different K's in Fig. 2-10. Note that, these trees are also the optimal Huffman code trees for the finite symbol set in (2.21) for different K's. It turns out that for the particular set $\mathcal{Z} = \{0^K, 0^i1\}_{i=0}^{K-1}$, the Huffman code can in fact be implemented under the probing feasibility constraint (i.e., any permissible probe should be a valid lightpath), as shown in Section 2.3 for the line network with single link failure. This should not be very surprising since we have already known (a) the Huffman code is always optimal for any given alphabet, (b) the algorithm above is optimal in locating the first faulty edge on a lightpath. The only missing logical step is that the symbol 0^K is always assigned to a length-one codeword, corresponding to a single probe. It can be shown that this is optimal from a coding perspective since

Algorithm 1 RLPS: Run-Length Probing Scheme

```
Set $i = 0$.
Set $h_l = h_r = K$.
while $i <= m$ do
  Probe the path $P_{i+1}^{i+h_l}$;
  if the syndrome $r(P_{i+1}^{i+h_l}) = 0$ then
    Set $i = i + h_l$, $h_l = h_r$ and $h_r = K$;
    if $i + h_r > m$ then
      Set $h_r = m - i$;
    end if
  else
    Set $h = h_l$,
    Set $h_l = g(h)$, [function $g(\cdot)$ is given by Eqn. 2.10],
    Set $h_r = g(h - h_l)$,
    if $h_l = 1$ then
      The edge $P_{i+1}^{i+h_l}$ fails,
      Set $h_l = h_r = K$, $i = i + 1$;
    end if
  end if
end while
```

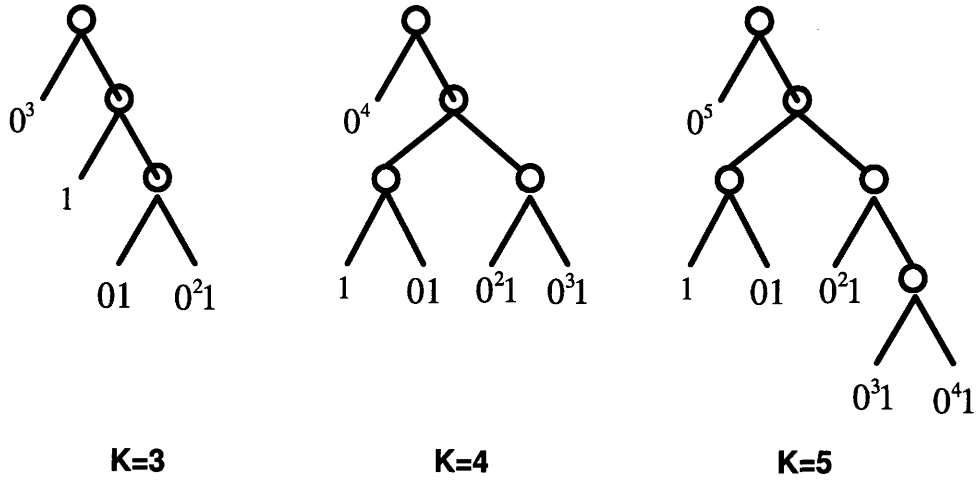


Figure 2-10: Run-length probing decision trees for $K = 3, 4, 5$, which are also the optimal Huffman code trees for the corresponding intermediate symbol sets $\mathbb{Z}$.

K is chosen such that the symbol 0^K is much more likely than the other symbols in Z. In fact, one can easily verify[12] that the probability of 0^K is larger than $2/5$, which assures the optimality of assigning to it a length 1 codeword [18].

In summary, the run-length probing algorithm is natural for the fault-diagnosis problem due to the following two reasons:

1. Each probe can at most locate one faulty edge, thus it makes sense to split the network state into sub-states and locate the faulty edges one-by-one;

2. The probing algorithm can achieve the information theoretical optimum in locating the individual faulty edges.

Note that the run-length probing algorithm is restricted on an Euler Trail of the network, and ignores other connections. In general, this restriction may seriously reduce the set of admissible probes, thus one cannot claim a general optimality of this algorithm over all possible sequential probing schemes. The run-length probing belongs to a class of 'nested' probing schemes. In a nested probing scheme, each successive probe includes only a subset of consecutive edges from the previous probe or a set of edges that are not tested in the previous probe. Within the class of nested probing schemes, the run-length probing scheme is optimal to minimize the average number of lightpath probes required. Wolf [72] has derived similar results under a totally different context of using group testing approach to resolve the conflict in multi-access communications, and showed that a similar scheme is optimal within the class of "nested" group testing algorithms [63].

2.4.3 An Algorithmic Interpretation

In this sub-section, we present an algorithmic description of the run-length probing scheme for Eulerian networks.

As a mirror of the concatenation structure in the run-length code, the run-length probing scheme alternates two phases (i.e., the *failure detection* phase and the *failure*

[12] Notice from (2.13), $(1-p)^K + (1-p)^{K-1} > 1$. It follows that $(1-p)^K > \frac{1}{2}\frac{1-p}{1-p/2} > 1/2$.

localization phase) to identify each faulty link along the Euler trail. In the failure detection phase, a detection probe is sent over a set of K (i.e., the maximum probing length) consecutive links along the Euler trail. This phase corresponds to the unary code in the run-length codeword. If all the links are fault-free, we move onto the next set of K consecutive links along the Euler trail. If on the other hand the detection probe returns the syndrome '1', the algorithm enters the failure localization phase. In this phase, given that there is some failure in the detection probe, the "2^{α}-splitting" binary searching algorithm [68, 73] is employed to locate the leftmost faulty link. This phase corresponds to the Huffman code in the run-length codeword. After the fault is localized, the algorithm resumes the failure detection phase by sending another probe spanning K links along the trail right after the failure.

As an illustration, Fig. 2-9 demonstrates how to employ the two-phase probing scheme for efficient fault diagnosis, where dotted line corresponds to the fault detection phase and solid line corresponds to the fault localization phase. Given an Eulerian network, one can first identify an Euler trail[13] in the network as indicated by the blue line in Fig. 2-8(a). This Euler trail introduces a natural structure to the network state, by indexing the link states along the trail, as illustrated in Fig. 2-9(b). The run-length probing scheme is then implemented along the Euler trail. In this example, we assume that the link failure probability equal to 0.1. It follows that the maximal probing length is 7. Therefore, the first probe should span 7 edges, as shown in Fig. 2-9(b). This probe corresponds to the fault detection phase. The resulted probing syndrome is "1", indicating some failure within the lightpath probe. The

[13] We can find a Euler trail in a graph G by following procedure:

- pick any vertex a and trace out a trail;
- let C be the cycle thus generated and let G' be the subgraph consisting of the remaining edges of G-C;
- because the original graph is connected, C and G' must have a common vertex, call it a';
- build a new cycle C' tracing through G' from a';
- incorporate C' into the cycle C at a' to obtain a larger cycle C';
- repeat the same process until no edge remains.

The complexity of this algorithm to identify a Euler trail is $\mathbf{O}(m)$, where m is the number of edges in the graph.

scheme continues in the fault localization phase, by using the 2^α-splitting probing scheme to locate the leftmost failure, as indicated in the solid probes. We first send a probe of length 3, and the resulting probe syndrome is ONE, meaning that there is a failure within the first three edges. We then send a probe of length 1 and the resulting probe syndrome is ZERO, meaning that the first edge is healthy and the failure happens between the second and the third edges. Finally, we send a probe of length 1 from the second edge, and the resulting probe syndrome is ZERO, meaning that the third edge fails. Once the probing scheme identifies the failure on the third link from the left, the algorithm resumes from the fourth link from the left. This process continues until all the link failures have been detected and localized, as illustrated in Fig. 2-9(b).

2.4.4 Performance of Run-Length Probing Schemes

In this subsection, the performance of the run-length probing scheme is characterized by taking advantage of its information theoretic interpretation.

For large networks (roughly speaking, $m \gg K$), the following Lemma characterizes the average number of probes per edge required for run-length probing schemes.

Lemma 2.1. *For a large Eulerian network whose link failures are modeled as identical and independent Bernoulli random variables with parameter p, the average number of probes per edge required by the run-length probing scheme to fully identify the network state, denoted as $\bar{\mathcal{L}}_{RLPA}$, can be approximated by the code rate of its corresponding run-length code, i.e.,*

$$\bar{\mathcal{L}}_{RLPA} \approx p \cdot \left(\lfloor \log_2 K \rfloor + 1 + \frac{(1-p)^k}{1-(1-p)^K} \right) \triangleq \mathcal{R}(p), \tag{2.27}$$

where $K = \lceil -\log_{1-p}(2-p) \rceil$ and $k = 2^{\lfloor \log_2 K \rfloor + 1} - K$.

This lemma can be proved using the results from the run-length code [27, 60], as shown in Section A.2.

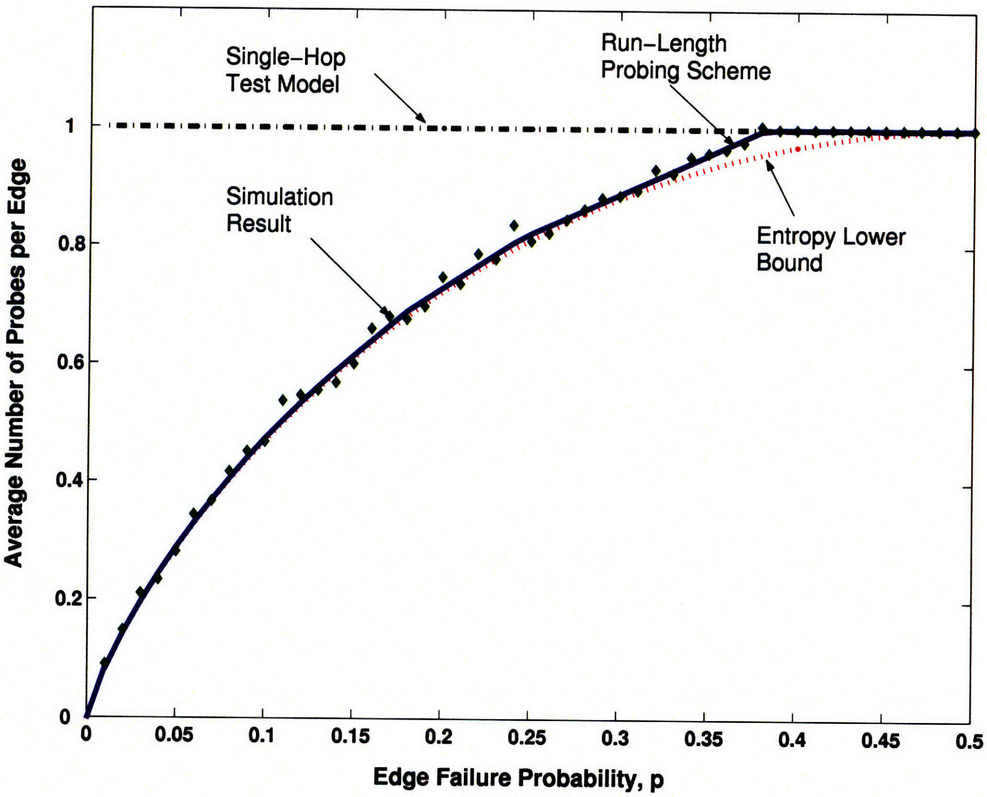


Figure 2-11: Simulated average number of probes per edge for an Eulerian topology with 50 edges is compare with the run-length code rate and the entropy lower bound. The performance of the run-length probing scheme is close to the entropy lower bound.

In Fig. 2-11, the approximate average number of probes per edge, i.e., $\mathcal{R}(p)$, is plotted as a function of the edge failure probability p, in comparison with the simulated results of the actual number of probes required, the performance of the link-wise probing scheme, and the entropy lower bound given in (2.7), for Eulerian networks with 50 edges. We have three observations from this plot. First, compared to the brute-force link-wise probing scheme, the run-length probing scheme provides a significant reduction in the diagnosis effort, especially when the network is relatively reliable. For example, when the link failure probability is $p = 10^{-5}$, the run-length probing scheme requires around 10^{-4} probes per edge. Compared to 1 probe per edge for the link-wise probing scheme, this is 4 order of magnitude more efficient.

Second, the plot indicates that $\mathcal{R}(p)$ is a good approximation of the actual performance of the run-length probing scheme over a broad range of reliability regime. This suggests that for a large Eulerian network ($m \gg K$) one can approximate the average number of probes for the run-length probing scheme as $\mathcal{L}_{\text{RLPA}} \approx m \cdot \mathcal{R}(p)$.

Finally, when the edge failure probability is small (of greater engineering interests), the average number of probes required is close to the entropy lower bound (a careful comparison will be presented next). For example, for an Eulerian network with 1000 edges and edge failure probability $p = 0.01$, the run-length probing scheme requires only 81.05 probes on average. Compared to the entropy lower bound of 80.79 probes, it requires only an additional number of 0.26 probes. As explained next, the performance of the run-length probing scheme actually approaches the entropy lower bound asymptotically.

To gain a clearer view of the relationship between the performance of the run-length probing scheme and the entropy lower bound, we define the information inefficiency (or, probing inefficiency) of the probing scheme as the ratio between the extra number of probes per edge (compared to the entropy lower bound) and the entropy of each edge, i.e.,

$$\varepsilon(p) = \frac{\bar{\mathcal{L}}_{\pi} - H_b(p)}{H_b(p)}, \tag{2.28}$$

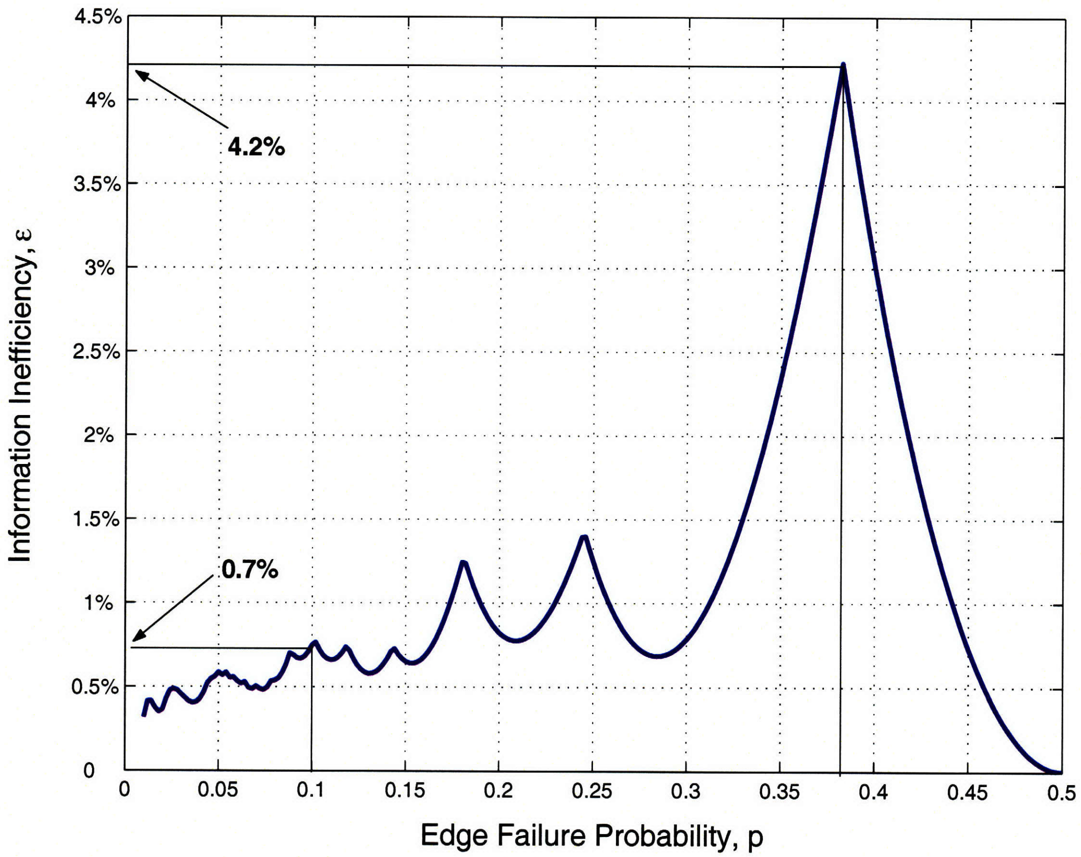


Figure 2-12: Information inefficiency of run-length probing schemes for different edge-failure probabilities. It is uniformly upper bounded by 5% and tends to decrease as the link failure probability decreases.

where $\bar{\mathcal{L}}_\pi$ is the average number of probes per edge of any probing decision tree $\pi \in \Pi(G)$.

In Fig.2-12, the information inefficiency of the run-length probing scheme, i.e., $\varepsilon(p) = \frac{\mathcal{R}(p)-H_b(p)}{H_b(p)}$, is plotted as a function of the link failure probability. Notice that, when the edge failure probability p decreases, the run-length algorithm becomes more efficient, with the fluctuation due to the change of the choice of the maximum probing length K, which takes on only integer values. In particular, if the edge failure probability is less than 0.1, the average number of probes per edge of the run-length probing scheme, for large networks, is upper bounded by

$$\bar{\mathcal{L}}_{\text{RLPA}} \leq 1.007 H_b(p), \tag{2.29}$$

which is only 0.7% higher than the entropy lower bound. Moreover, the difference between the achieved performance and the entropy lower bound is uniformly bounded. In the range of $p \in (0, 0.5]$, the worst case, as shown in [60], occurs at $p = (3-\sqrt{5})/2$, where

$$\bar{\mathcal{L}}_{\text{RLPA}} \approx 1.0423 H_b(p), \tag{2.30}$$

meaning that the actual performance of the run-length probing scheme is less than 5% larger than the lower bound.

Based on(2.7), (2.29) and (2.30), the performance of the run-length probing scheme is bounded by the following inequalities,

$$H_b(p) \leq \bar{\mathcal{L}}_{\text{RLPA}} \leq [1 + \epsilon(p)] H_b(p), \tag{2.31}$$

where $\varepsilon(p)$ tends to decrease with smaller edge failure probability and we can approximate $\varepsilon(p) < 0.01$ for $p \leq 0.1$ and $\varepsilon(p) < 0.05$ for $0.1 < p \leq 0.5$. This indicates that the performance of the run-length probing scheme is always less than 5% larger than the entropy lower bound.

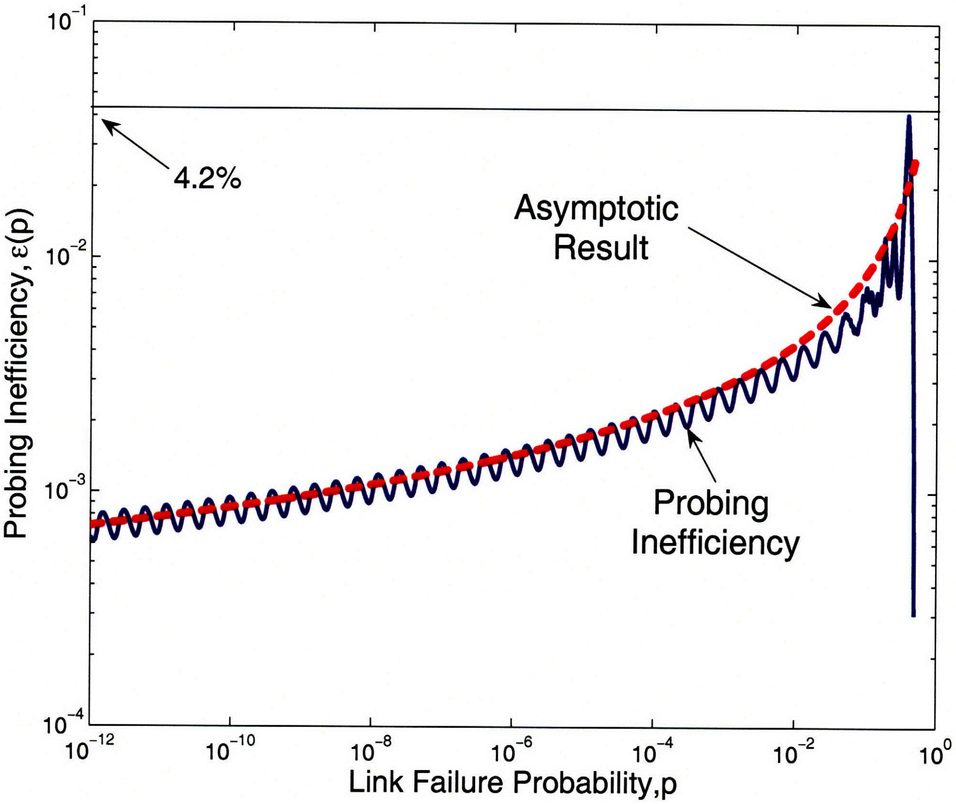


Figure 2-13: The information inefficiency of run-length probing schemes is uniformly upper bounded and approaches zero asymptotically when the link failure probability decreases.

Moreover, as illustrated in Fig. 2-12, the information inefficiency tends to decrease as the link failure probability decreases. Asymptotically, it can be shown[14] that

$$\lim_{p \to 0} \epsilon(p) \approx \frac{c}{-\log_2 p}, \tag{2.32}$$

where $c = 2 - (\log_2 e + \log_2 \log_2 e) \approx 0.029$ is a constant. As illustrated in Fig. 2-13, (2.32) shows that the probing inefficiency approaches zero asymptotically as the failure probably decreases, and thus the run-length probing scheme is asymptotically optimal when the network is relatively reliable. In practical networks with fairly reliable components, both the upper and the lower bounds in (2.31) are reduced to the entropy of individual link, suggesting that the run-length probing scheme is asymptotically optimum for large Eulerian all-optical networks.

Finally, the performance of the run-length probing scheme can be used as an upper bound for the minimum average number of probes per edge. The convergence of both upper bound and lower bound to the entropy indicates that the minimum probing effort approximately equals to the entropy of the network states. This result suggests that,

- ***in an efficient probing scheme (e.g., the run-length probing scheme), each probe is designed to provide approximately one bit of state information.***

Since the amount of unknown information in the network state is equal to the entropy,

- ***the number of probes required to identify the network state is approximately equal to the entropy of network state.***

2.5 Greedy Probing Schemes

The information theoretic perspective in the context of fault diagnosis is in general also very useful in understanding, comparing, and improving the network probing

[14] In [27], Gallager has shown that, when $p \to 0$, $\mathcal{R}(p) - H_b(p) \to p[2 - (\log_2 e + \log_2 \log_2 e)]$. At the same time, when $p \to 0$, $H_b(p) \to -p \log_2 p$. It follows that $\lim_{p \to 0} \epsilon(p) \approx \frac{c}{-\log_2 p}$.

schemes based on existing heuristics. In particular, we will in this section study the fault diagnosis design based on dynamic programming approaches.

In fact, it can be shown that the optimum fault-diagnosis problem is equivalent to the optimal binary decision tree design. It follows that the adaptive fault-diagnosis problem for general network topology is Co-NP Complete [49], as pointed out in Section 2.4. As a compromise, various sub-optimal greedy algorithms [54] are proposed based on local optimization heuristics, under the dynamic programming approach. The performance of such heuristic algorithms is usually studied only via simulations. With the information theoretic viewpoint of the problem, it is natural to connect these problems to their counterparts of source coding problems with dynamic programming approaches, which have been thoroughly studied for decades.

In this section, we will first review the dynamic programming formulation of the network-diagnosis problem, and then focus on a particular greedy algorithm that maximizes the local information gain at each stage [54]. Finally, we will compare the performance of this greedy scheme with that of the run-length algorithm to gain more insights.

2.5.1 Dynamic Programming Formulation of Adaptive Fault Diagnosis

In this subsection, the optimal adaptive fault diagnosis problem is formulated as a dynamic programming problem.

First some useful notations are introduced to facilitate the formulation. The design of optimal fault-diagnosis algorithms is equivalent to the design of optimal binary decision trees. For a decision tree of π, the set of inner nodes is denoted as $\mathcal{I}_\pi$. Let $\varsigma \in \mathcal{I}_\pi$ denote one of the inner nodes, and P_ς the probability that ς is reached. It follows that P_ς equals to the sum of the prior probabilities of the network states that are descendants of the node ς [75]. Let t_ς be the probe employed at inner node ς , $\Pr(0|\varsigma)$ and $\Pr(1|\varsigma)$ be the probabilities that this test returns "0" and "1", corresponding to the probabilities that the network state lies in the left or right sub-

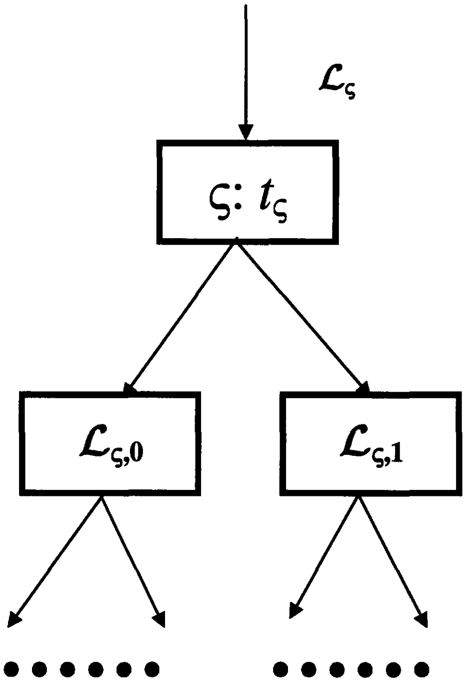

$\mathcal{L}_\varsigma$
ς: t_ς
$\mathcal{L}_{\varsigma,0}$
$\mathcal{L}_{\varsigma,1}$

Figure 2-14: Dynamic programming illustration: at each node, we choose a probe to minimize the number of probe in the subtree starting from the node.

trees of inner node ς, respectively. Furthermore, let $\mathcal{L}_\varsigma$ be the average number of successive probes required when the inner node is reached.

Now to design the optimal diagnosis algorithm with the minimum average number of probes, it is required, at each inner node ς (as illustrated in Fig. 2-14), to choose a probe t_ς to minimize

$$\mathcal{L}_\varsigma = 1 + \Pr(0|\varsigma) \times \mathcal{L}^*_{\varsigma,0} + \Pr(1|\varsigma) \times \mathcal{L}^*_{\varsigma,1}, \tag{2.33}$$

where $\mathcal{L}^*_{\varsigma,0}$ and $\mathcal{L}^*_{\varsigma,1}$ are the minimum average number of probes required by the left and right sub-trees from the inner node ς, respectively. In particular, taking ς as the root of the entire tree, the solution of this optimization problem gives the optimal fault diagnosis scheme.

Note that the difficulty of such a problem comes from the fact that the optimization problems at different steps are coupled. In choosing t_ς, one needs to cater for the future optimizations of $\mathcal{L}^*_{\varsigma,0}$ and $\mathcal{L}^*_{\varsigma,1}$. As a result, the computational complexity of this problem grows exponentially with the number of edges m. Some results of using dynamic programming in designing binary decision trees can be found in [47, 46].

Now from an information theoretic perspective, the performance, in terms of the average number of probes, can be computed from the local information efficiencies as follows. For a given probing tree π, the average number of probes required to reach the leaves can be computed as

$$\begin{aligned}
\mathcal{L}_\pi &= \sum_{s\in S} \Pr(s) \times (\text{number of probes to reach state } s) \\
&= \sum_{s\in S} \Pr(s) \times (\text{number of ancestors of } s) \\
&= \sum_{\varsigma\in\mathcal{I}_\pi} \Pr(\varsigma) \times 1 \qquad (2.34)
\end{aligned}$$

On the other hand, one can write $H(\varsigma)$ as the amount of information in bits, obtained by employing the probe t_ς as node ς is reached, that is,

$$H(\varsigma) \triangleq H_b(\Pr(0|\varsigma), \Pr(1|\varsigma)), \qquad (2.35)$$

where $H_b(p,q) = -p \times \log_2 p - q \times \log_2 q$ is the information entropy function. By running this fault-diagnosis algorithm, one can always find out the network state, which contains on the average $m \cdot H_b(p)$ bits of information, and can be viewed as the sum of the information obtained in each step, i.e.,

$$m \cdot H_b(p) = \sum_{\varsigma\in\mathcal{I}_\pi} \Pr(\varsigma) \times H(\varsigma). \qquad (2.36)$$

Hence the total inefficiency of the algorithm, in terms of the average number of probes required in excess of the information minimum $m \cdot H_b(p)$ is

$$\mathcal{L}_\pi - m \cdot H_b(p) = \sum_{\varsigma\in\mathcal{I}_\pi} \Pr(\varsigma) \times (1 - H(\varsigma)). \qquad (2.37)$$

where $1 - H(\varsigma)$ is referred as the local inefficiency of the algorithm π at the inner node ς. Intuitively, one probe is used to return only $H(\varsigma)$ bits of information. Hence the difference between the two measures is the information inefficiency of employing this probe, and the weighted sum of the inefficiency over the tree gives the total number of extra probes required by the given probing scheme. To minimize the average number of probes, the greedy approach chooses a probe to minimize the local inefficiency at all the inner nodes.

2.5.2 Greedy Probing Algorithms

In order to design efficient network-diagnosis algorithms, it is desirable to minimize the local inefficiency at each stage. Intuitively, by always asking the question to which the answer is completely without bias, one would expect to figure out the network state with fewer questions. This corresponds to making the left and right sub-trees as balanced as possible, i.e., to chose a probe t_ς to minimize

$$t_\varsigma^* = \arg\min_{t_\varsigma} |\Pr(0|\varsigma) - \Pr(1|\varsigma)|. \tag{2.38}$$

Such intuition of balancing the probabilities of the outcomes of a probe is in general very useful. For example, the same design principle has been used to design the maximum probing length K for the run-length algorithm. For the first probe over K links, the probabilities of UP and DOWN are, respectively, q^K and $1 - q^K$. It can be shown that the choice of in inequality (2.13) indeed minimizes the difference between these two probabilities.

It is important to note that such an approach, by maximizing the local information gain $H(\varsigma)$, may not necessarily be the globally optimum choice. As an example, in the example of locating a single failure in Section 2.3, it is globally optimal to split the path as in (2.8), to make sure that the length of one of the sub-paths is an integer power of 2. On the other hand, a greedy design based on local optimizations would simply split the path into equal halves.

The greedy algorithm presented above to maximize the local information gain is in fact one of many variations [54]. Such algorithms are sometimes preferred due to their conceptual simplicity. Although the run-length algorithm has the same or-

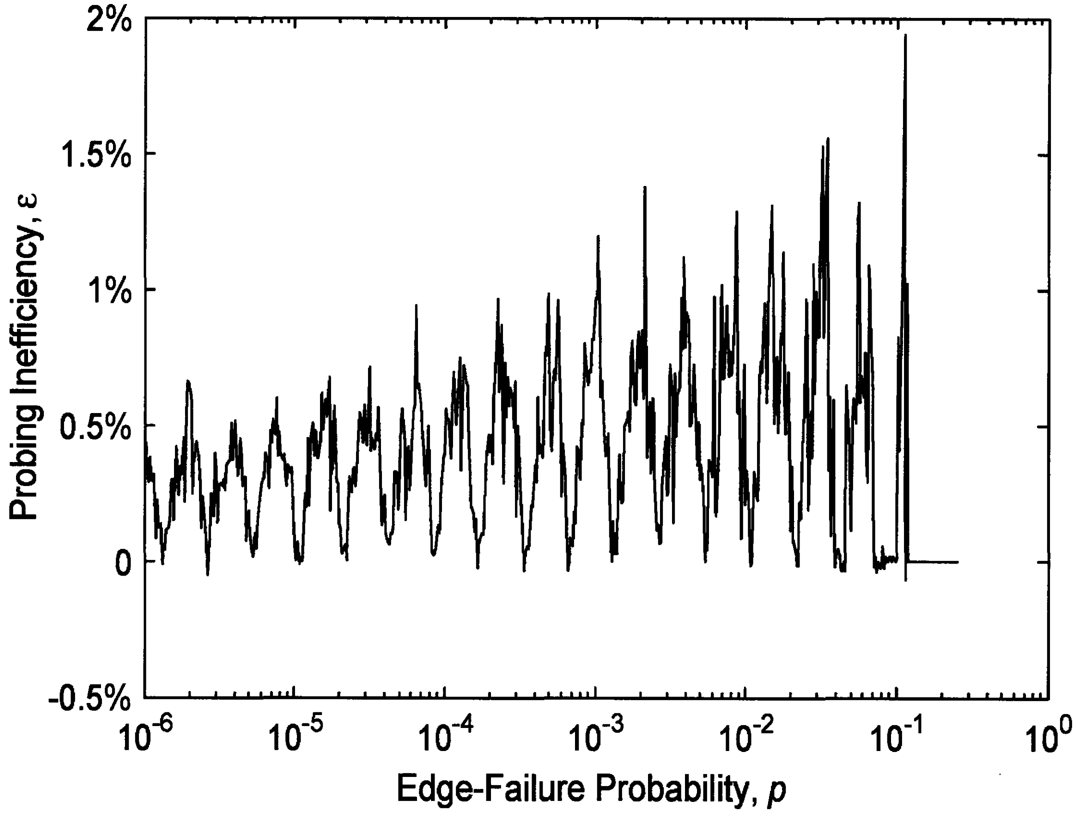


Figure 2-15: Performance comparison between the run-length probing scheme and the greedy probing scheme. The run-length probing scheme always outperforms the greedy probing scheme, with the exception that both schemes have the same performance when the maximum probing length K is an integer power of 2.

der of computational complexity as these algorithms, it provides better performance compared to the greedy algorithms, as illustrated in next subsection.

2.5.3 Performance Comparison Between Run-Length Probing Schemes and Greedy Probing Schemes

Using Monte Carlo method, we simulate the performance of the greedy probing algorithms and compare it to that of the run-length probing algorithm in this sub-section.

To compare these two algorithms in a finer scale, the probing inefficiency of the greedy probing scheme over the run-length probing scheme, defined as

$$\epsilon(p) = \frac{\bar{\mathcal{L}}_{GPA} - \bar{\mathcal{L}}_{RLPA}}{\bar{\mathcal{L}}_{RLPA}}, \tag{2.39}$$

is plot as a function of the link failure probability p in Fig. 2-15. In (2.39), $\bar{\mathcal{L}}_{RLPA}$ and $\bar{\mathcal{L}}_{GPA}$ are the average number of probes per link for the run-length probing algorithm and the greedy probing algorithm, respectively.

Notice that for some range of link failure probability, both the run-length probing scheme and the greedy probing scheme have the same average number of probes per link. It can be verified that this happens for all the link failure probabilities such that the maximum probing length K is an integer power of 2. Under this scenario, splitting a path of length K automatically gives sub-paths with lengths as powers of 2, hence the local and global optimums coincide. On the other hand, when the link failure probability is in the range such that K is not an integer power of 2, the greedy algorithms are strictly sub-optimal. As a result, in Fig. 2-15, there is a periodic pattern in the log plot: when p is such that $K(p)$ equals to a power of 2, the probing inefficiency is equal to 0; as increases or decreases such that $K(p)$ does not equal a power of 2, the probing inefficiency is strictly non-zero.

Therefore, although both probing schemes have the same computational complexity, the run-length probing scheme provides some saving in diagnostic effort over the greedy probing scheme. However, the difference between the two algorithms is quite limited. Intuitively, this is because that the global optimum solution always makes sure that all but one sub-path have lengths of powers of 2, in which case the greedy algorithm is also optimum.

Notice that, to develop the run-length probing schemes, we have made two assumptions. First, it is assumed that the network contains a Euler trail. Second, only link failures happen and nodes are robust. Practical networks normally cannot meet these two assumptions. In the rest of this chapter, these two assumptions will be relaxed to investigate fault diagnosis for practical all-optical networks.

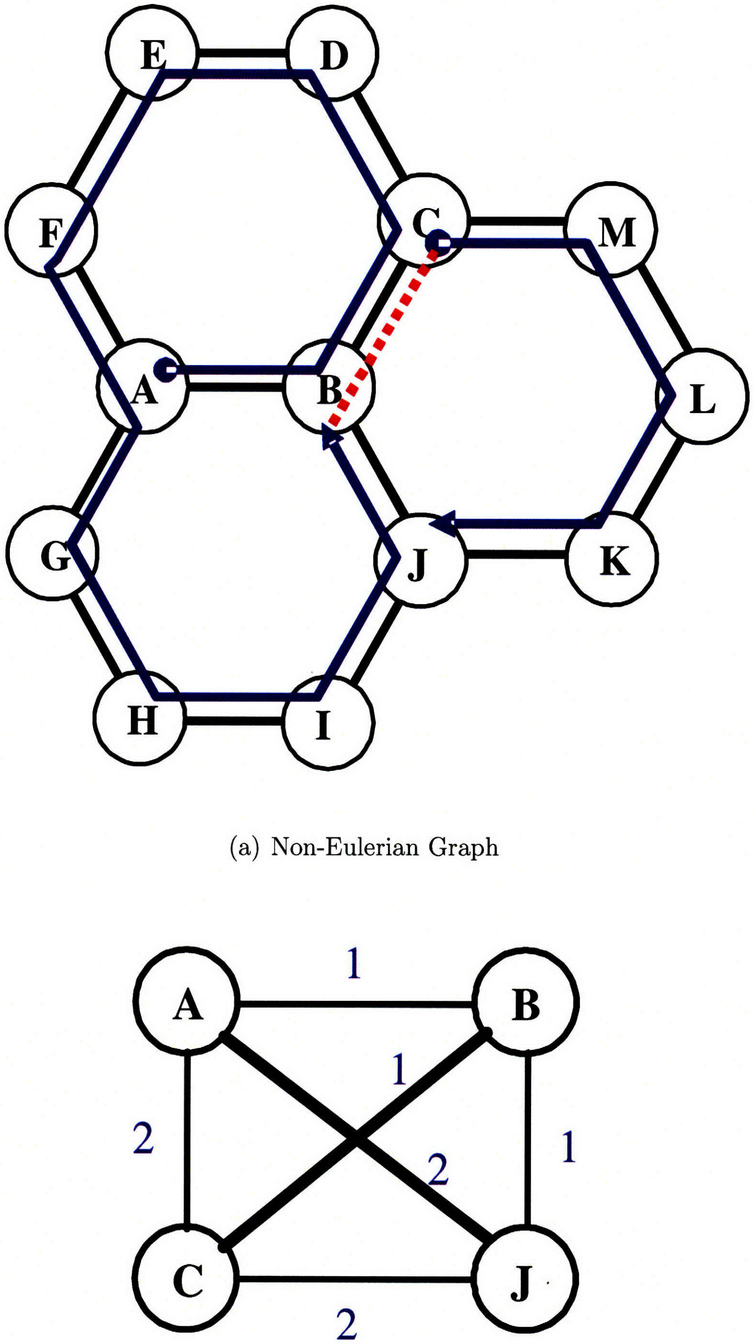
2.6 Fault Diagnosis for All-Optical Networks with Non-Eulerian Topologies

The limitations of run-length probing scheme sometimes make it hard to apply to realistic all-optical networks. To employ the run-length probing scheme, we had assumed in Section 2.4 that the network is Eulerian. This requires that all (or except two) the nodes in the network have even degrees [10]. However, practical all-optical networks may not satisfy this condition and thus the run-length probing schemes cannot be applied directly. In this section, two alternative approaches are proposed to extend the run-length probing scheme to non-Eulerian topologies. Their corresponding performance are characterized analytically.

2.6.1 Disjoint-Trail Decomposition Approach

The first approach is based on the idea of decomposing the Non-Eulerian graph into a set of link-disjoint trails (a trail in a graph is a sequence of interconnected links without repetition.) It follows the following two-stage procedure.

First, any non-Eulerian graph can be decomposed into a set of link-disjoint trails, among which no two trails share the same link. The set of link-disjoint trails can be identified via a sequential deletion procedure as follows. One can start from any node and walk along the graph until he has to pass some link twice. The set of passed links forms a trail, and are deleted from the graph. The same trail deletion process is continued from any other node of non-zero degree until the graph is empty. For example, in Fig. 2-16(a), the sequential deletion procedure results in two link-disjointed trails in the non-Eulerian network, i.e., trail A-B-C-D-E-F-G-H-I-J-B and trail C-M-L-K-J. Notice that the minimum number of link-disjoint trails is fixed although the length of each trail might vary. It is desirable to keep each decomposed trail long enough so that the loss of efficiency is insignificant.



(a) Non-Eulerian Graph

(b) Complete Graph M

Figure 2-16: Fault diagnosis for Non-Eulerian Graphs: (a)Each non-Eulerian graph can be decomposed into a set of non-overlapping trails. (b)The complete graph M to identify the minimum set of replicated links.

Second, after the decomposition step, the run-length probing scheme can be applied to each link-disjoint trail sequentially. The network state is uniquely identified after all the trails have been probed.

However, the decomposition could potentially penalize the performance of the run-length probing scheme. In particular, the decomposition could potentially break one sub-state 0^i1 into two sub-states of $0^{i'}$ and $0^{i-i'}1$ on two link-disjoint trails. The number of probes to identify sub-state 0^i1 is at least less than the number of probes to identify two sub-states of $0^{i'}$ and $0^{i-i'}1$, where the additional number of probes is upper bounded by 1. If the number of individual link-disjointed trails is n_T , the average number of probes per link is given by

$$\mathcal{R}(p) \leq \bar{\mathcal{L}}_{RLPA} \leq \mathcal{R}(p) + \frac{n_T}{m}, \tag{2.40}$$

where $\mathcal{R}(p)$ is defined in (A.12). Since each link-disjoint trail reduces the number of odd-degree nodes by two, one can conclude[15] that $n_T = n_o/2$, where n_o is the number of odd-degree nodes in the network, and thus the upper bound becomes $\mathcal{R}(p)+n_o/2m$.

Specifically, it is possible to derive a tighter bound for the class of non-Eulerian regular topologies considered in [29]. A graph is said to be regular of degree d if the degrees of all the nodes are equal to d. For example, the d-nearest neighbors Graph, the symmetric Hamilton Graph and the Moore Graph (with the fully-connected graph as a special case) are the most popular regular graphs considered for all-optical network architectures. The non-Eulerian property suggests that degree d is odd and thus $n_o = n$. Notice that for a regular graph of degree d, the handshake property suggests $n/2m = 1/d$. It follows that, for a non-Eulerian regular graph of degree d, the average number of probes per link is given by

$$\mathcal{R}(p) \leq \bar{\mathcal{L}}_{RLPA} \leq \mathcal{R}(p) + \frac{1}{d}. \tag{2.41}$$

[15] In [71], Theorem 1.2.33 states that, for a connected nontrivial graph with exactly $2k$ odd vertices, the minimum number of trails that decompose it is $\max\{k, 1\}$.

For cost-optimized architectures of all-optical networks with optical cross-connect (OXC) switches, Guan and Chan [29] have recently shown that, under the assumption of all-to-all uniform traffic, the optimal node degree d asymptotically approaches infinity as the network size (in particular, the number of nodes) approaches infinity while their ratio approaches zero. It follows that, for a cost-optimally architected all-optical network, the upper bound in (2.41) converges to the lower bound, indicating that the run-length probing scheme is asymptotically optimum for large non-Eulerian regular networks with cost-optimized architectures.

2.6.2 Path-Augmentation Approach

An alternative approach converts the non-Eulerian graph into an Eulerian graph by replicating a minimum set of links. It also exhibits a two-stage procedure.

First, in any network, one can replicate each link once along the shortest path between any two nodes of odd degree to make their degrees even. The shortest path between two odd-degree nodes is called as an augmenting path and the above replicating operation as a path augmentation. Notice that the path augmentation does not change the degree parity of any other nodes along the augmenting path (specifically, their degrees are increased by 2.), but reduces the number of odd-degree nodes in the network by two. Since the number of odd-degree nodes in a finite network is always even due to the handshake property (i.e., the sum of node degrees is even) [10], one can convert any non-Eulerian graph into an Eulerian graph via a finite number of path augmentations.

Second, after the path-augmentation step, the run-length probing scheme can be applied along the nominal Euler trail in the resulting Eulerian graph. Upon termination, all the link states have been identified except that a set of redundant links have been probed more than once. If possible, to reduce the diagnosis effort, one can skip those redundant links whose states have been identified previously.

Moreover, to save the fault diagnosis cost, one would like to minimize the number of replicated links resulted from the path-augmentation step, via the following minimum-weight perfect matching approach. This approach includes the following four steps:

1. an all-pair shortest-path algorithm (for example, the Floyd-Warshall algorithm [2]) is run to identify the set of all-pair shortest paths among the set of odd-degree nodes in the original graph (e.g., six distinct shortest paths for the set of odd-degree nodes A, B, C, J in Fig. 2-16(a));

2. a complete graph M(i.e., Fig 2-16(b)) is created with the set of odd-degree nodes (i.e., A, B, C, J) and the weight of each link as the length of the shortest path connecting the two nodes in the original graph (e.g., the weight of link AJ is 2 because the shortest path connecting node A and node J in Fig. 2-16(a)is A-B-J);

3. a minimum-weight perfect matching algorithm (a perfect matching of a graph is a subset of links in the graph that touch all the nodes exactly once [2], which can be identified by the Edmonds' blossom algorithm [2]) is run over graph M to obtain a perfect matching (e.g., AJ, BC is the minimum weight perfect matching in Fig. 2-16(b));

4. the original network G is augmented along the set of paths chosen by the resulted minimum perfect matching except for the augmenting path with the maximum weight, because a graph with two odd-degree nodes is Eulerian. As a result, path B-C is augmented in Fig 2-16(a) via the dotted link.

The approach has two distinguished advantages. First, the augmented graph is always Eulerian so that the run-length probing scheme is applicable. For example, in the augmented graph of Fig. 2-16(a), we can identify a nominal Euler trail[16], i.e., trail A-B-C-D-E-F-G-H-I-J-B-C-M-L-K-J, which passes link B-C twice. Second, the number of replicated links is usually small compared to the number of links in the original network. Notice that the number of replicated links is 1 in our example,

[16] A Euler trail over a Eulerian graph can be found by first decoupling the graph into a set of edge-disjoint cycles and then connecting them together. A formal proof of this algorithm can be found in [10](page 17).

which is significantly less than the number of links (14 in this case) in the graph. This observation can be made rigorously for Non-Eulerian regular graphs as follows.

Specifically, for the set of non-Eulerian regular topologies considered for all-optical networks in [29], in particular, the symmetric Hamilton Graph, one can obtain a tight upper bound on the performance. Notice that each Hamilton graph contains a Hamilton path (a path containing each node exactly once) of size n. One minimum weight perfect matching consist of alternating edges along the Hamilton path, and thus the number of replicated links is $n/2$. Hence, for a regular Hamilton graph of degree d, the average number of probes per link under the path-augmentation approach is bounded by

$$\mathcal{R}(p) \leq \bar{\mathcal{L}}_{RLPA} \leq \mathcal{R}(p)\left(1 + \frac{1}{d}\right), \tag{2.42}$$

where $\mathcal{R}(p)$ is defined in (A.12). For cost-optimized architecture whose optimal node degree asymptotically approaches infinity as the network size (in particular, the number of nodes) tends to infinity [29], the upper bound in (2.42) converges to the lower bound, verifying that the run-length probing scheme is asymptotically optimum for large non-Eulerian regular networks with cost-optimized architectures.

2.7 Network Transformation from Undirected Topologies to Directed Topologies

Another limitation of the run-length probing scheme is that it can only identify link failures. However, in practical all-optical networks, node failures are also possible. In this section, we will first introduce failure models for optical links and optical nodes of all-optical networks, and then introduce a transformation that converts link/node failures in original undirected network topologies into arc failures in transformed directed network topologies. The transformation makes the run-length probing scheme applicable for both link and node failures in all-optical networks.

2.7.1 Optical Links and Link Failure Model

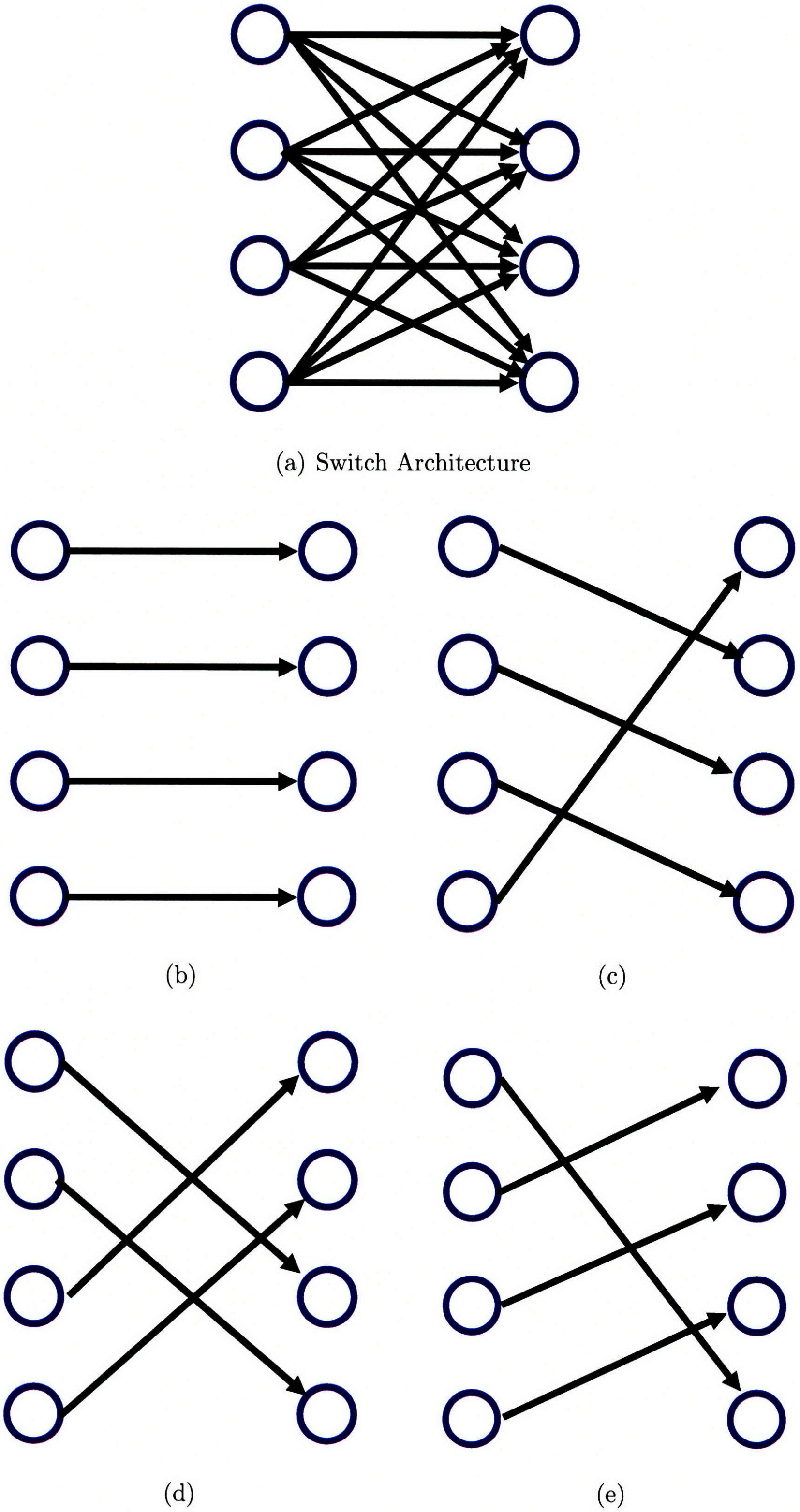
In optical networks, bidirectional communication between adjacent nodes is typically achieved by means of two parallel optical fibers that propagate optical signals in opposite directions. If connectivity is of interest, an optical link may be abstracted as an undirected graph edge in an undirected graph. On the other hand, if physical failures are of interest, an optical link is more appropriately modeled as a pair of contra-directional arcs in a directed graph. In the following, the latter abstraction is adopted.

It is assumed that each directed optical link fails independently with probability of p ($0 \leq p \leq 0.5$) over an interval of time, which represents the time duration between fault diagnoses. This assumption of statistical independence among failures is reasonable when "normal" operation of the network is considered, because the equipment (e.g., optical amplifiers and in-line filters, and etc) of each arc operates independently from the equipment of other arcs. In the event of a catastrophic failure, however, this model is not applicable and other approaches to ensure network reliability, such as lightpath diversity [66], can be used.

2.7.2 Optical Nodes and Node Failure Model

The model for optical nodes is based on all-optical switches that are responsible for optically routing signals from input fibers to output fibers. Assume that each network node of degree d is equipped with a $d \times d$ optical switch fabric, switching the optical beam from each input port to any desired output port, as shown in Fig 2-17(a). It is further assumed that each input/output port of the optical switch is equipped with a pair of low-cost transceivers[17] (economically viable due to the VCSEL technology [15]), whose state of health is locally monitored and reported to the network management system upon polling. This research focuses on the active

[17] As a comparison, we assume that no such transceivers are equipped for optical amplifiers. Accordingly, we need to investigate the transmitter/receiver deployment for fault diagnosis in Chapter 4.



(a) Switch Architecture

(b)

(c)

(d)

(e)

Figure 2-17: Optical network node model: (a) an illustration of 4x4 optical switch fabric; (b)-(e) an illustration of some non-blocking directed configurations of a 4x4 optical switch fabric.

components (e.g., the mirrors in MEMS optical switches) in the switch fabric, which could fail from manufacture defects and/or fatigue from normal use.

Under these assumptions, each node i of degree d with a $d \times d$ optical switch fabric can be modeled by a directed bipartite graph, defined as follows:

1. d virtual input nodes correspond to all the input ports of the switch, denoted as $i_k^I, k = 1, 2, \ldots, d$;

2. d virtual output nodes correspond to all the output ports of the switch, denoted as $i_k^O, k = 1, 2, \ldots, d$;

3. Each virtual input node is connected to all the virtual output nodes via d directed arcs, as shown in Fig. 2-17(a).

For each node of degree d, there exists d different mutually exclusive and collectively exhaustive non-blocking directed configurations[18], each comprising a set of d directed arcs from input nodes to output nodes where no two arcs share the same source/destination nodes. For example, Fig. 2-17(b)-(e) shows some of the possible configurations for a node of degree 4. At any instance, the switch can take one and only one non-blocking configuration. Therefore, one can use one sample non-blocking directed configuration at a time to model the corresponding network node for the purpose of fault diagnosis.

In an analogy to the link failure model, an independent failure model is assumed for each configuration of the optical switch: each input-output connection in the switch fabric fails independently with probability q ($0 \leq q \leq 0.5$). Using different non-blocking configurations, one can create d different topologies to diagnose all the connections in a regular graph of degree d. In this research, we would like to focus on one instance of network topology and develop fault diagnosis schemes which can be easily extended to all the different network topologies. This simplified node failure

[18] Notice that the total number of non-blocking directed configurations is $d!$. The number of orthogonal non-blocking directed configuration is d, because the number of individual input-output connections is d^2 and the number of input-output connection in any non-blocking directed configuration is d. In fact, one can obtain the set of d orthogonal non-blocking directed configurations by following the same cyclic decomposition as illustrated in Fig. 2-17.

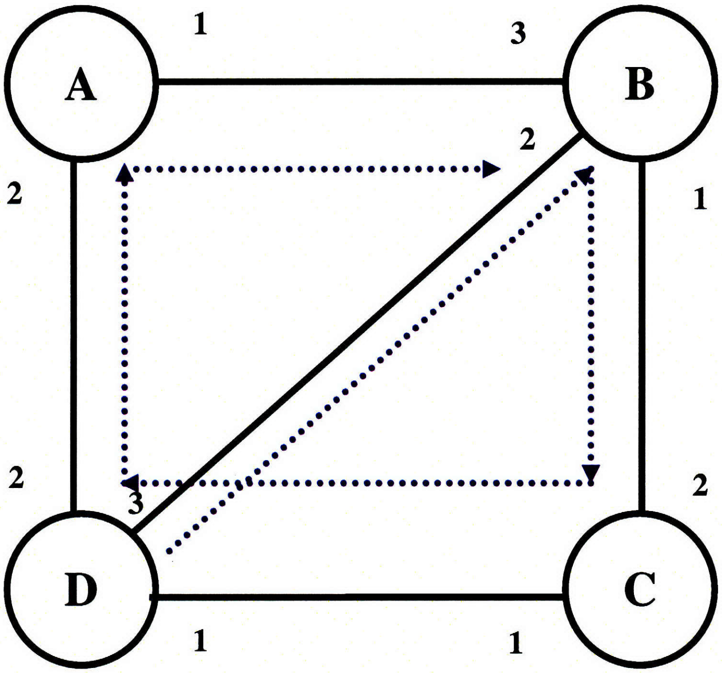


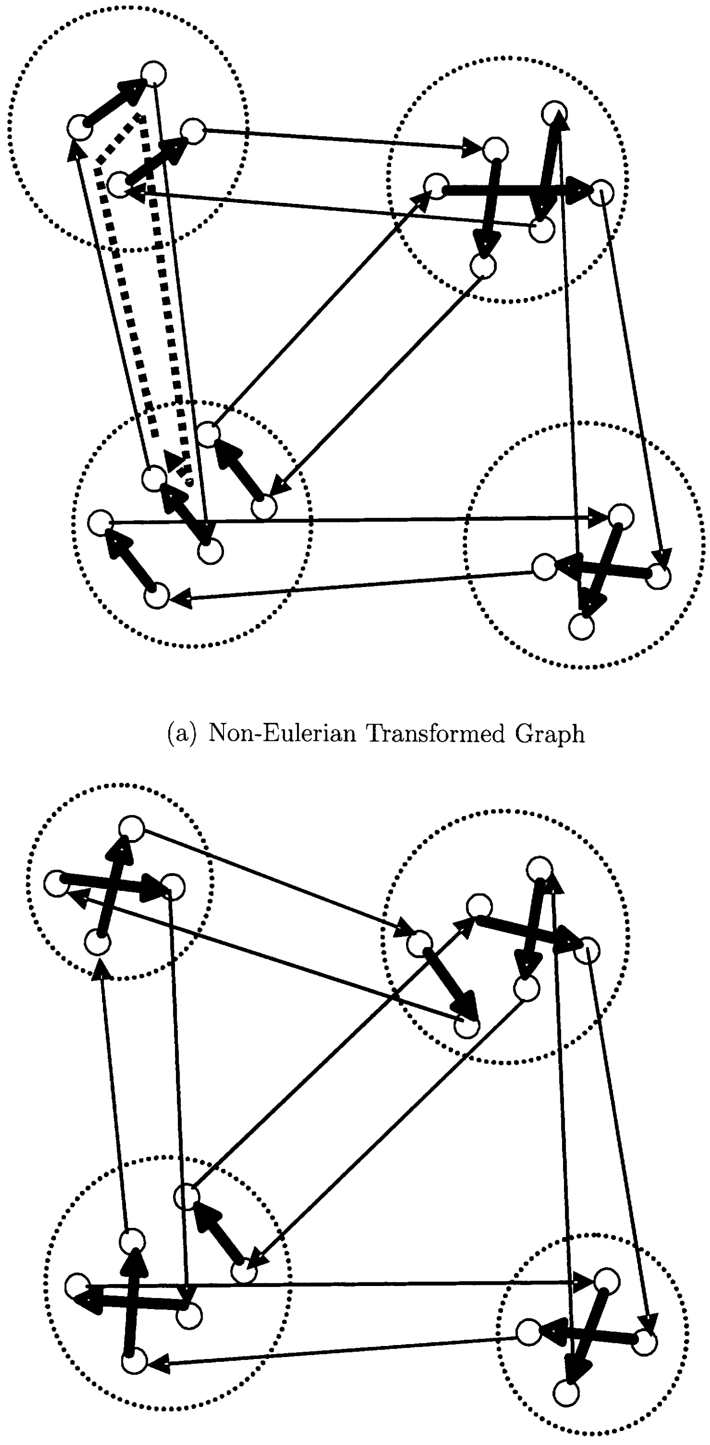
Figure 2-18: An original undirected graph: the Euler trail is illustrated with a blue dotted line.

model captures the essence of practical switching node failures, and more practical node failure models can be addressed by appropriate extension of this simple model.

2.7.3 Network Transformation from Undirected Topologies to Directed Topologies

The run-length probing scheme developed in section 2.4 can only diagnose link failures, whereas, nodes are also vulnerable to failures in practical all-optical networks. To diagnose failures in all-optical networks with directed optical links and possible node failures, a transformation can be used to convert node failures into link failures:

1. All the links of each node in the undirected topology are indexed with an integer, as shown in Fig 2-18.



(a) Non-Eulerian Transformed Graph

(b) Eulerian Transformed Graph

Figure 2-19: Transformation from undirected graphs into directed graphs: (a)Non-Eulerian transformed graph and (b)Eulerian transformed graph.

2. Each link (i, j) in the undirected graph is replaced by two directed arcs, $i \longrightarrow j$ and $j \longrightarrow i$, in opposite directions.

3. Each network node of degree d is replaced with an empty bipartite graph comprised of $2d$ nodes (i.e., two columns of indexed nodes without any arc connecting them).

4. For each link (i, j) in the original graph, if its index in node i is k and its index in node j is l, the output node i_k^O is connected to the input node j_l^I with the directed arc $i \longrightarrow j$, and also the output node j_l^O is connected to the input node i_k^I with the directed arc $j \longrightarrow i$.

5. For each node, an appropriate directed configuration is chosen such that the transformed graph is Eulerian (details will be elaborated later in this section.)

Fig. 2-18 and 2-19 demonstrate how the network transformation described above converts an undirected graph into a directed graph. In particular, Fig. 2-18 depicts the original undirected graph, and Figures 2-19 depicts two different directed graphs resulted from choosing different network node configurations.

The proposed transformation provides two properties that are crucial for application of the run-length probing scheme.

First, the transformed directed graph can be made Eulerian or be decoupled into a set of directed cycles. Upon transformation, each virtual node in the resulted graph has a degree of 2 with one for in-degree and one for out-degree. It follows from the Euler's Theorem[19] [10] that the resulted graph contains one Eulerian trail or a set of arc-disjoint disconnected cycles. Indeed, the existence of an Euler cycle in the directed graph depends on how the configurations are chosen for all the network nodes. As shown in Fig. 2-18, the original graph has an Euler trail of 4-2-3-4-1-2. The directed graph in Fig. 2-19(b) is decomposed into two arc-disjoint cycles and is thus non-Eulerian. Alternatively, in Fig. 2-19(a), the Eulerian property of the graph is maintained by appropriately choosing the configurations of all the nodes. In general,

[19] A non-trivial connected graph has an Euler circuit if and only each vertex has even degree.

the set of appropriate node configurations for Eulerian graphs can be identified as follows. After step 2 of replacing each link in the undirected graph with two parallel directed arcs, the resulting directed graph has an Euler trail since the in-degree of each node is equal to its out-degree. For example, in Fig. 2-18, the Euler trail passes node 1 from its link 2 to its link 1. It follows that node 1 should be configured as the cross state as in Fig. 2-19(a), instead of the through state as in Fig 2-19(b). As a result, Fig. 2-19(a) is Eulerian, while Fig. 2-19(b) is non-Eulerian. In the former case, the run-length probing scheme can be applied to the resulted directed Euler cycle to identify all the failures. In the latter case, the run-length probing scheme can be applied to each of the arc-disjoint cycles sequentially. In Section 2.6, we have shown that the performance of the run-length probing scheme over a set of link-disjoint trails is still close to the entropy lower bound. Therefore, the rest of our analysis focuses on the case where the resulted graph is Eulerian, and the resulted obtained can shed light onto the case of a set of arc-disjoint cycles[20].

Second, through the proposed transformation, both links and nodes in the undirected graph are mapped into directed arcs in the directed graph. Any directed arc connecting two virtual nodes in different switches corresponds to a directed optical fiber link in the all-optical network, which shall be called a link arc. Any directed arc connecting two virtual nodes in the same switch corresponds to a switching component (e.g., a MEMS mirror) in the all-optical network, which shall be called a node arc (or equivalently switch arc). For an original undirected graph of m links and n nodes and a chosen switch configuration, the directed graph has $2m$ links arcs, and $2m$ switch arcs. In this way, the transformation maps both link and node failures in the original graph into arc failures in the transformed graph, which can be identified by the run-length probing scheme as shown in the next section.

[20] However, we are not yet clear whether the performance is related to how we decompose each node into a set of non-blocking configurations. For a node of degree d, the number of different decompositions is $(d-1)!$.

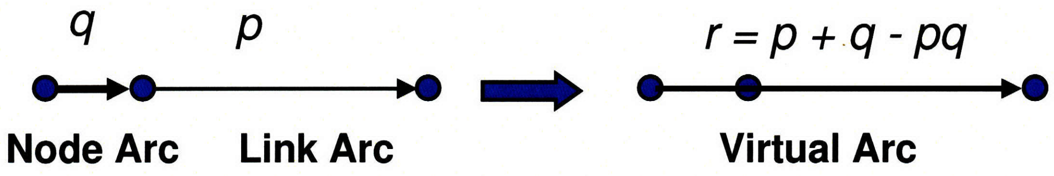


Figure 2-20: A virtual arc is formed by combining a node arc with an adjacent link arc.

2.8 Run-Length Probing Schemes for All-Optical Networks with Link/Node Failures

In this section, a four-stage process is proposed to employ the run-length probing scheme for fault diagnosis to practical all-optical networks with probabilistic link/node failures through the proposed network transformation.

1. In the first stage, one can employ the transformation approach to obtain the directed network topology, and then identify a directed Euler trail in the transformed graph. Notice that link arcs and switch arcs appear alternatively along any Euler trail. Without loss of generality, one can assume that the directed Euler trail starts from a link arc and ends with a switch arc.

2. In the second stage, since the failure probability of each arc along the Euler trail is heterogeneous, our proposed solution is to combine an adjacent switch arc and link arc into a virtual arc with failure probability of $r = p + q - pq$. This combination results in a directed Euler trail of length $2m$, in which the failure probability of each virtual arc in the Euler trail is homogenous. Hence, the run-length probing scheme is now applicable.

3. In the third stage, one employ the run-length probing scheme along the directed Euler trail to identify all the faulty virtual arcs. For a reasonably large network,

the average number of probes per virtual arc is approximately equal to:

$$\bar{\mathcal{L}}^{arc} = \begin{cases} \mathcal{R}(r), & \text{if} \quad 0 < r < \frac{3-\sqrt{5}}{2} \\ 1, & \text{if} \quad r \geq \frac{3-\sqrt{5}}{2}, \end{cases} \tag{2.43}$$

where $\mathcal{R}(r)$ is defined in (A.12). Note that when the virtual arc failure probability is higher than $\frac{3-\sqrt{5}}{2}$ (the golden ratio), the run-length probing scheme always probes each virtual arc individually. Hence, in this case, the average number of probes per arc is always equal to 1.

After the third stage, among all the $2m$ virtual arcs, the average number of failures is $2m(p+q-pq)$. If we find that one virtual arc fails, there are three possible failure scenarios: (1) a single switch arc failure with probability of $(1-p)q/r$, (2) a single link arc failure with probability of $p(1-q)/r$, or (3) a combined switch/link arc failure with probability of pq/r.

4. In the fourth stage, additional probes can be deployed, using the built-in lasers in the optical switch, to determine which of the above three scenarios has occurred for each faulty virtual arc. It can be shown that, if $p > q$, the switch arc should be probed first, otherwise, the link arc should be probed first. Under such a diagnosis strategy, the average number of additional probes for each faulty virtual arc is given by:

$$\bar{\mathcal{L}}^{c} = 1 + \min\{\frac{p}{r}, \frac{q}{r}\}. \tag{2.44}$$

where the first term of 1 comes from the default probe, and the second term comes from the case when the probe syndrome of the default probe is F and then we have to probe the other arc.

The performance of run-length probing schemes can be obtained by combining our efforts in first identifying all the faulty virtual arcs, and then determining the sources of failure for each faulty virtual arc. In particular, the average number of probes per

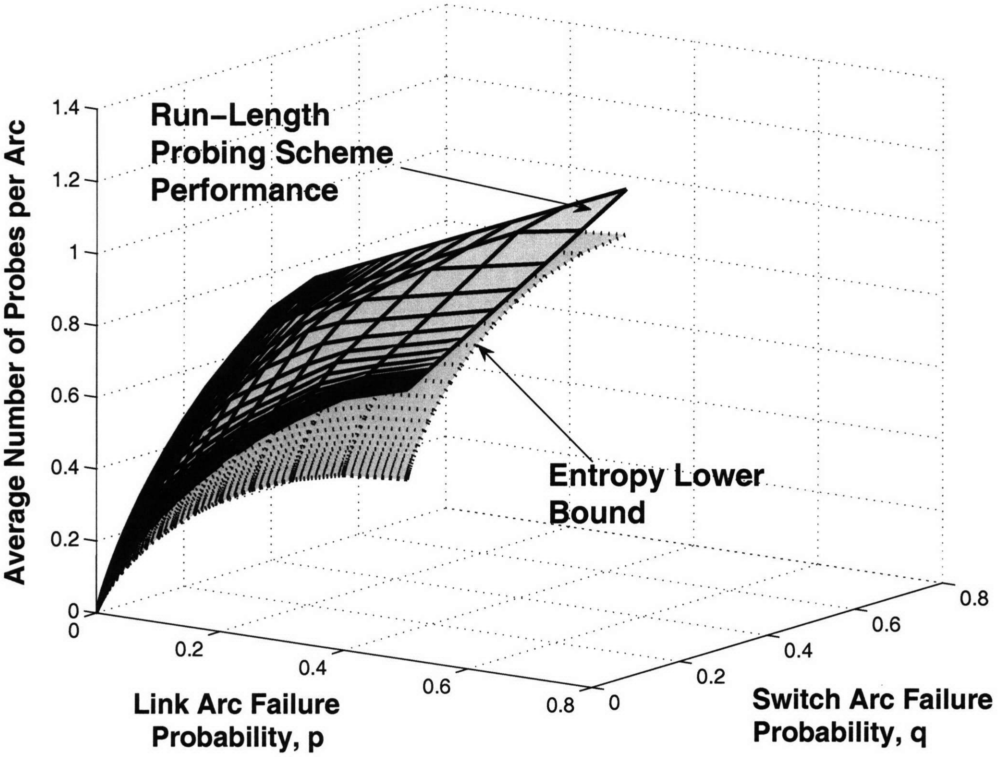


Figure 2-21: The average number of probes per component is compared to the entropy lower bound, for different value of p and q. The performance of the run-length probing scheme is close to the entropy lower bound.

directed arc (or vulnerable component) is given as

$$\begin{aligned}\bar{\mathcal{L}}_{RLPA} &\approx \frac{1}{4m}[2m\bar{\mathcal{L}^{arc}} + 2m(p+q-pq)\bar{\mathcal{L}^{c}}] \\ &= \frac{1}{2}[\bar{\mathcal{L}^{arc}} + (p+q-pq)\bar{\mathcal{L}^{c}}] \qquad (2.45)\end{aligned}$$

Meanwhile, suggested by the fault-diagnosis/source-coding mapping, the average number of probes per arc is lower bounded by the information entropy of an individual arc, i.e.,

$$\bar{\mathcal{L}}_{RLPA} \geq \frac{1}{2}[H_b(p) + H_b(q)] \triangleq \mathbf{H}(p,q), \qquad (2.46)$$

where $H_b(x) = -x\log_2(x) - (1-x)\log_2(1-x)$ is the entropy function.

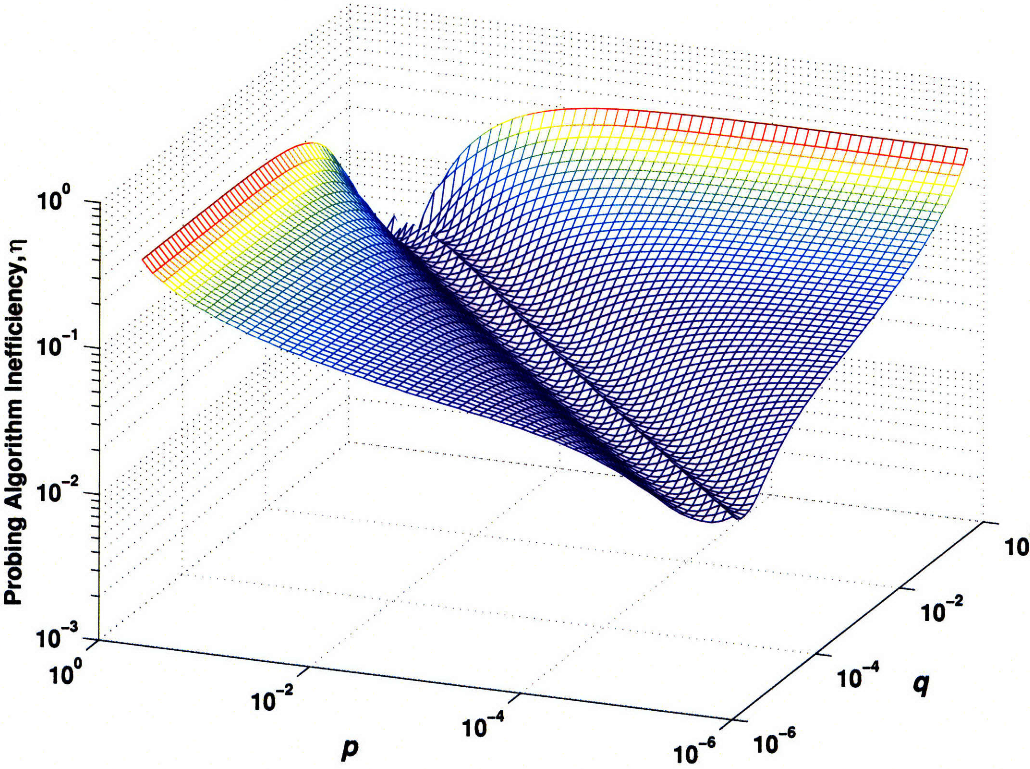


Figure 2-22: The probing algorithm inefficiency is plotted for different link arc failure probability and switch arc failure probability pairs.

The performance (2.45) is compared with the entropy lower bound (2.46) in Fig. 2-21. An immediate observation from Fig. 2-21 is that the average number of probes per arc is close to the entropy lower bound, as expected from the previous results on the near-optimality of the run-length probing scheme. This observation also lends support to the proposed approach to fault diagnosis involving network transformations. A second observation, from Fig. 2-22, is that the probing algorithm inefficiency, which is defined as the ratio between the number of additional probes compared to the entropy lower bound and the entropy lower bound, i.e.,

$$\eta(p,q) = \frac{\bar{\mathcal{L}}_{RLPA} - \mathbf{H}(p,q)}{\mathbf{H}(p,q)}, \tag{2.47}$$

increases as the difference between the link arc failure probability and the switch arc failure probability increases. This can be understood as follows. When the difference between p and q increases, one kind of failure occurs more likely than the other. The general approach treats both the link arc failure and the switch arc failure equivalently. As a result, one pay the penalty for not exploiting in the algorithm the fact that one type of failure dominates the other. The third observation, from Fig. 2-22, is that when p is equal to q and both approach zero, the probing algorithm inefficiency converges to zero much slower than the link-failure case [68]. In this case, we have $r^* = 2p - p^2$. This can be shown as the following equation,

$$\begin{aligned}
\eta(p,p) &= \frac{\frac{1}{2}[\mathcal{R}(r^*) + r^*\bar{\mathcal{L}}^c] - H_b(p)}{H_b(p)} \\
&= \frac{\frac{1}{2}[\mathcal{R}(r^*) - H_b(r^*)] + \frac{1}{2}r^*\bar{\mathcal{L}}^c - H_b(p)}{H_b(r^*)} \cdot \frac{H_b(r^*)}{H_b(p)} \\
&= [\frac{1}{2}\varepsilon(r^*) + \frac{1}{2}\frac{r^*\bar{\mathcal{L}}^c}{H_b(r^*)} + \frac{1}{2} - \frac{H_b(p)}{H_b(r^*)}] \cdot \frac{H_b(r^*)}{H_b(p)} \\
&\overset{p\to 0}{=} \varepsilon(r^*) + \frac{\bar{\mathcal{L}}^c}{-\log_2 r^*},
\end{aligned} \tag{2.48}$$

where $\lim_{p\to 0} \frac{H_b(r^*)}{H_b(p)} = 2$ and $\bar{\mathcal{L}}^c > 1$. As $p \to 0$, the first term in (2.48) approaches to zero as indicated in (2.32), and the second term in (2.48) approaches zero. However, the second term is much larger than the first term. It follows that the convergence to zero is much slower than the link-failure case. In fact, if $p = q$, the link/node failure diagnosis problem is equivalent to the link failure diagnosis problem with twice as many links. It would be better to treat switch arcs and link arcs on equal basis, and thus employ the run-length probing scheme along an Euler trail of $4m$ links. The combination of switch arcs and link arcs as virtual arc definitely sacrifices performance when the failure probability is fairly low, because the two-stage probing procedure is different from the optimal run-length probing scheme.

In summary, the numerical analysis suggests the following rules of thumb for applying the run-length probing scheme to all-optical networks with probabilistic link/node failures. First, when the link failure probability is equal to the switch failure probability, it is better to treat them equivalently and employ the run-length probing scheme over an Euler trail of $4m$ links. Second, when one type of failures dominates, we should focus on the dominant type of failures. Finally, for all other cases between the aforementioned two extremes, we should adopt the proposed virtual arc approach.

Once we have finished one instance of the network topology, we can continue to diagnose other instances of the network topology, with the additional knowledge of all the link states. We can merge two virtual nodes connected by any link arc and deploy the run-length probing scheme over the resulting graphs.

2.9 Summary

Optical switching, in replacement of electronic switching, of high data-rate lightpaths at intermediate nodes has been widely touted as the key enabling technology for economically scalable future data networks. Less widely acknowledged, however, are the challenges with respect to fault detection and localization entailed by this replacement of electronic switching with optical switching in future all-optical networks. Presently, fault detection and localization techniques, as implemented in SONET/G.709 networks, rely on electronic processing at intermediate nodes for bit-level parity checks. To adapt these techniques to all-optical networks, optical signals need to be tapped out at intermediate nodes for parity checks, which would significantly diminish the cost advantages of optical switching.

In this chapter, we present new scalable fault-diagnosis approaches specifically tailored to all-optical networks, with the objective of keeping the diagnostic effort low. Instead of the passive paradigm based on parity checks, we propose a proactive lightpath probe paradigm: carefully chosen optical probe signals are sequentially sent along lightpaths in the network, and the network state of health is inferred via the set of end-to-end measurements from lightpath probes. The design objective of our proposed fault diagnosis schemes is to minimize the number of probes in order to keep the network operating cost low.

We have initiated an information-theoretic approach to the fault-diagnosis problem. Specifically, we established a mapping between the fault-diagnosis problem in network management and the source-coding problem in Information Theory, which suggests an entropy lower bound on the minimum average number of probes required and an information-theoretic approach to translating efficient source coding algorithms into efficient fault diagnosis schemes. Our results—including an asymptotically optimal probing scheme—provide insights into the reduction of fault management overhead costs for all-optical networks, as well as the relationship between information entropy and network management.

Chapter 3

Non-Adaptive Fault Diagnosis Schemes

In parallel to Chapter 2 for adaptive fault diagnosis schemes, we investigate in this chapter non-adaptive fault diagnosis schemes for all-optical networks[1]. In non-adaptive fault diagnosis schemes, all the lightpath probes are sent in parallel within one step. Our design objective is to minimize the number of lightpath probes, so as to keep the diagnostic effort low. Mathematically, the non-adaptive fault-diagnosis problem can be cast as a special case of the group-testing-over-graphs problem, i.e., the problem of combinatorial group testing over graphs.

3.1 Introduction

Instead of the passive paradigm based on parity checks in SONET/G.709 networks, we have proposed, in Chapter 1, a proactive fault diagnosis paradigm in [68]: optical probing signals are sent along a set of lightpaths to test the health of the network, and probe syndromes (i.e., results of the probes) are used to differentiate failure patterns. The design of proactive fault-diagnosis schemes for all-optical networks, illustrated in Chapter 1, bears two key objectives: (i) detecting faults quickly, and (ii) keeping the

[1] The content in this chapter is based on the joint work with Nicholas J.A. Harvey, Mihai Patrascu and Sergey Yekhanin at CSAIL MIT., and has been published in [30].

diagnosis effort low. The importance of objective (i) stems from the current SONET standard [34], in which the 50-ms restoration time leaves little room for fault detection and localization. This will probably be reduced further in future all-optical networks to avoid large amount of data loss during a short period of communication disruption. Hence, when parts of a network are malfunctioning, it is critical to locate and identify these failures as soon as possible.

The two design objectives could be tightly related to two parameters of proactive fault-diagnosis schemes (i.e., the number of lightpath probes and the number of diagnostic steps), as illustrated in Chapter 1. First, the number of lightpath probes could serve as the manifestation of fault management effort. In particular, each probe requires certain amount of effort in both network management/control plane (e.g., signaling) and data plane (e.g., transmission and detection) that otherwise could be used to generate revenue. In addition, each probe results in one bit of management information, whose transportation, storage and processing consumes additional network resources. Second, under the assumption that each step takes approximately equal amount of time to a first order, the number of diagnostic steps indicates how fast the fault pattern could be identified. In this thesis, we exploit two alternative designs for sending probes (i.e, *adaptive* probing, and *non-adaptive* probing) to balance these two objectives.

Previously in Chapter 2, we have investigated adaptive fault diagnosis schemes [68, 70, 69, 46], in which probing signals are sequentially sent to probe the health of the network until the failure pattern is identified. Owing to its sequential nature, successive probes can be chosen according to previous probe syndromes, and thus the number of probes required is usually quite small and approaches to the theoretical limit of the information entropy of the network state. We have shown in Chapter 2 that the average number of probes is lower bounded by the information entropy of the network state. Based on information theoretic insights, we have also developed the run-length probing scheme and proved its performance to be within 5% of the entropy lower bound. However, the number of diagnostic steps might be quite large for some network failure patterns and/or in some large networks.

To keep the number of diagnostic steps small, in this chapter, we consider an alternative non-adaptive approach [48] to diagnose failures in all-optical networks. Instead of sending optical probing signals sequentially, a pre-determined set of probing signals are sent in parallel to probe the network state of health. In addition, compared to the probabilistic failure model (i.e., each link fails independently and no upper bound on the number of failures) used in Chapter 2, we assume a worst-case failure model in that the number of simultaneous failures is upper bounded by a constant. Under such a framework, our design objective is to minimize the number of parallel probes for non-adaptive fault diagnosis schemes, specifically by exploiting the fact that the number of simultaneous failures is upper bounded. Practically, one would not be able to make such a failure model assumption, because the upper bound on the number of failures is normally unknown. We hope to shed some light on the more practical case by investigating this idealised case[2].

In this chapter, we cast the non-adaptive fault diagnosis problem mathematically as the problem of combinatorial group testing (CGT) on graphs. In the classical group testing problem[20], defected samples are identified through a set of parallel testings on different combinations of unknown samples. This field has a wide variety of practical applications, such as HIV screening, DNA testing, MAC design, and much more [17]. It has also been used in network management applications (see, e.g., [5]), but only to a limited degree. We believe that CGT is a powerful tool that can be used in a wide variety of network failure detection contexts, and we hope that our work will inspire its use more widely. In the framework of group testing on graphs, the valid tests are determined by the structure of a graph. In the all-optical network context, this graph corresponds to the network topology, and the constraint on valid tests is due to the obvious condition that lightpaths can only traverse interconnected edges. To the best of our knowledge, this is a novel framework for CGT, and we believe that it deserves further study. We formally analyze the number of tests needed for certain interesting classes of graphs, and even arbitrary graphs (with performance depending

[2] In the idealised case, we can assume a genie who reveals the number of simultaneous failures to the fault diagnosis scheme.

on the topology). In some cases, we can derive the upper- and lower-bounds that have the same order, on the number of tests needed. Our algorithms have a common theme, which suggests a practical rule-of-thumb for efficient fault diagnosis schemes: a fault-free sub-graph in the network topology should be identified, and used as a "hub" to diagnose other failures in the network.

The remainder of this chapter is organized as follows. In Section 3.2, we present the non-adaptive fault diagnosis problem. In Section 3.3, this problem is cast as the combinatorial group testing problem on graphs. In Section 3.4, we describe algorithms and lower bounds for various classes of regular network topologies: linear networks, complete networks, grid networks. In Section 3.5, we consider trees and arbitrary graphs, and obtain efficient algorithms when the diameter is small and/or the graph does not have small cuts. Section 3.6 concludes this chapter.

3.2 Non-Adaptive Fault Diagnosis Paradigm

3.2.1 Permanent Link Failure Model

As in previous chapters, all-optical networks in this chapter are abstracted as undirected graphs. An *undirected graph* G is an ordered pair of sets (V, E), where V is the set of nodes, and E is the set of edges, which are unordered pairs of nodes. The number of nodes is n and the number of edges is m. The terms links and edges are used interchangeably in this chapter.

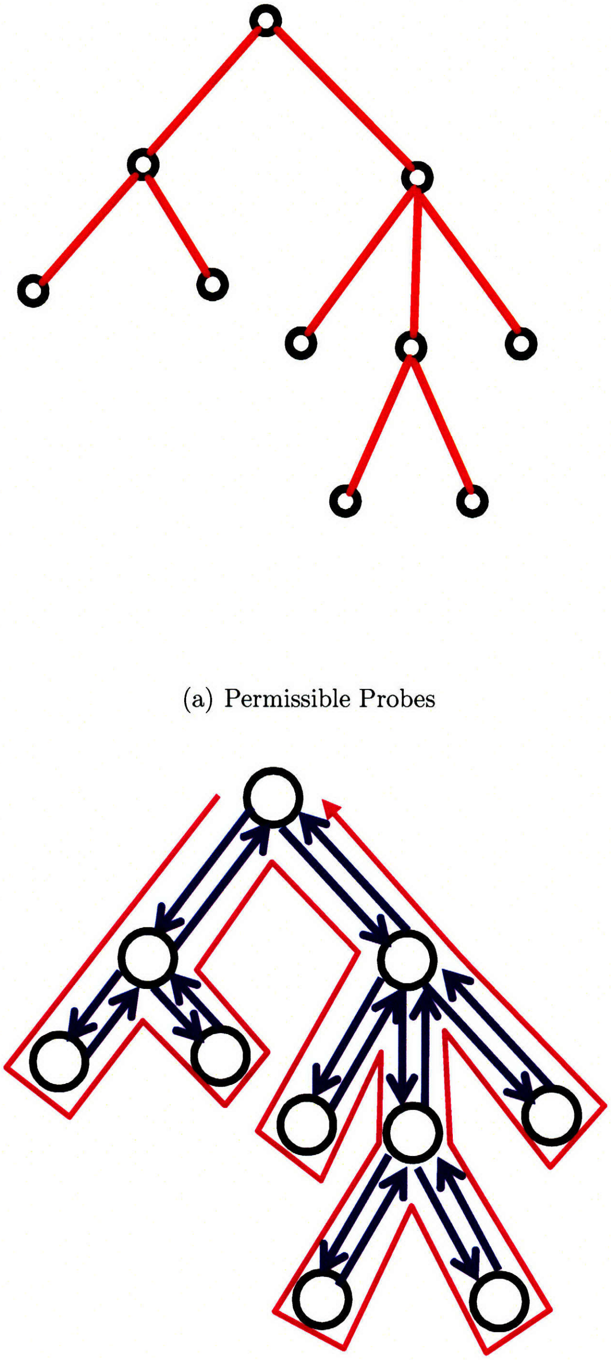
Moreover, we assume that links fail and nodes do not. Insights from this limited case could facilitate to address fault diagnosis for both node and link failures. In addition, this chapter consider a static failure model, i.e., an edge is either *failed* or *intact*, and the failure status does not change over the period of diagnosis. Since it is unlikely that numerous edge failures happen simultaneously, we assume that the number of edges failures is upper bounded by a constant $s(\leq m)$ at any instant. In this chapter, it is generally allowed for s to be arbitrary, although the case of $s = 1$ is often central. An alternative view of this combinatorial failure model is that network architects might be only interested in identifying up to some number of failures in the network, and classifying any network state with more failures as one "big" failure state.

3.2.2 Non-Adaptive Fault Diagnosis Schemes

In this chapter, network failures are detected and localized by sending optical probing signals along certain lightpaths to determine the network's state. We assume that a *probe* in the network corresponds to a walk (a sequence of adjacent edges, allowing repetitions) in the corresponding graph. Physically, each probe corresponds to a lightpath in the network. For example, a walk in the graph can constitute a sub-tree in the graph as in Fig. 3-1(a), which can be translated to a lightpath in practical all-optical networks as in Fig. 3-1(b). In Fig. 3-1(a), the network is abstracted as undirected graph, whose nodes correspond to the optical switches and links correspond to the optical fibers. In practical all-optical networks, each link represents two parallel optical fibers transmitting signals in opposite directions. As shown in Fig. 3-1(b), we can replace each link in Fig. 3-1(a) by two directed arcs in opposite directions. In this way, each walk can be implemented as a probe by sending a diagnosis signal along the directed lightpath, as illustrated in Fig. 3-1(b). Moreover, to avoid the potential fiber loop lasing effect [37], a physically feasible probe must satisfy one additional property: each network link is traversed at most once in each direction. We call such a probe a permissible probe. The probes generated by fault diagnosis algorithms in Section 3.4 and 3.5 are all permissible probes.

As in the adaptive fault diagnosis schemes presented in Chapter 2, when an optical signal is sent along a given lightpath, the signal will arrive at the destination if all edges along the lightpath are intact. Otherwise, if there is at least one failed edge on the lightpath, the signal never reaches the destination (or the quality of the signal is unacceptable). The result of each probe is called the *probe syndrome*, denoted as $r = 0$ if the probing signal arrives successfully; and $r = 1$ otherwise.

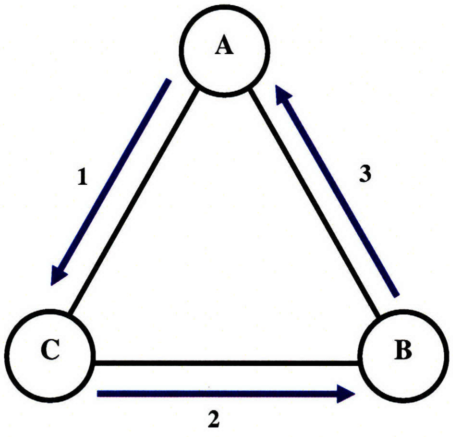
A *non-adaptive fault diagnosis scheme* is a method for sending optical signals (i.e., probes) along a set of pre-determined lightpaths in parallel such that up to s edge



(a) Permissible Probes

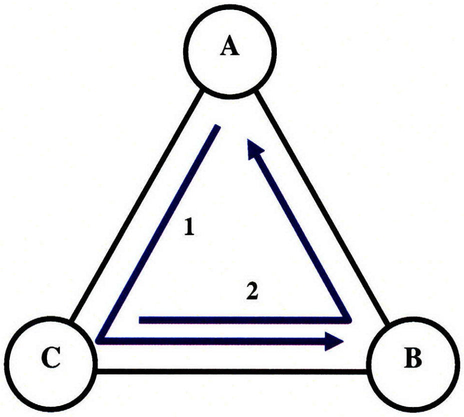
(b) Lightpath Probes

Figure 3-1: Lightpath probe model for non-adaptive fault diagnosis schemes: (a)any walk over graph is a permissible probe, (b) a walk over undirected graph can be implemented with a lightpath in practical all-optical networks.



	AB	BC	CA	Φ
1	**1**	**0**	**0**	**0**
2	**0**	**1**	**0**	**0**
3	**0**	**0**	**1**	**0**

(a) Link-Wise Probing Scheme



	AB	BC	CA	Φ
1	**1**	**1**	**0**	**0**
2	**0**	**1**	**1**	**0**

(b) Multi-Hop Probing Scheme

Figure 3-2: Two non-adaptive fault diagnosis schemes for the 3-node ring network, with their associated diagnosis matrix. The number of simultaneous link failures is upper bounded by 1.

failures can be identified by examining the set of probe syndromes. For example, as shown in Fig. 3-2, both sets of probes can identify any single edge failure. One would prefer Scheme (b) to Scheme (a) since Scheme (b) uses less probes than Scheme (a). Indeed, to keep the fault diagnosis effort low, we would like to develop efficient non-adaptive fault diagnosis schemes using the minimum number of probes.

3.3 Combinatorial Group Testing on Graphs

In this section, we present the theoretical background on combinatorial group testing (CGT) and its connection to the non-adaptive fault diagnosis problem.

The general CGT problem is defined as follows. Consider a set S of m elements, each of which is either intact or failed. The maximum number of failed elements is bounded by s, which is considered to be small relative to m. It is allowed to perform group tests of the following form: specify a subset $t \subseteq S$, run the test on t, and learn if there is at least one failed element in t. The objective is to discover all faulty elements, while using the smallest possible number of group tests. It has been pointed out in [20, 43] that the combinatorial group testing problem is isomorphic to the superimposed code problem [36] in Information Theory. Interested readers could refer to [36, 21, 22] for more in-depth description the superimposed code problem.

Let $T^*(m, s)$ denote the minimum number of non-adaptive group tests needed to locate up to s failed elements in a set of size m. It is obvious that $T^*(m, s) \leq m$, since one can test each element individually. The total number of failure patterns is $N(m, s) = \sum_{k=0}^{s} \binom{m}{k}$, so the minimum number of probes[3] needed to distinguish between these patterns is at least $\log_2 N(m, s)$. Hence, $\log_2 N(m, s) \leq T^*(m, s) \leq m$. In particular, if $s = 1$, the minimum number of non-adaptive probes needed is bounded as follows:

$$\log_2(m + 1) \leq T^*(m, 1) \leq m. \tag{3.1}$$

[3] Given that $T^*(m, s)$ probes are deployed, the maximum number of distinguishable patterns is $2^{T^*(m,s)}$. This number must be larger than or equal to the total number of failure patterns of $N(m, s)$, i.e., $2^{T^*(m,s)} \geq N(m, s)$. It follows that $T^*(m, s) \geq \log_2 N(m, s)$.

Table 3.1: Diagnosis matrix for the logarithmic testing procedure (LTP) with $m = 7$. Columns correspond to elements to be tested, and rows correspond to tests.

	0	1	2	3	4	5	6	7
1	0	0	0	0	1	1	1	1
2	0	0	1	1	0	0	1	1
3	0	1	0	1	0	1	0	1

For arbitrary s and sufficiently large m, it has been shown in [21, 23] that $T^*(m, s)$ can be bounded[4] as,

$$\frac{s^2}{2\log_2 s}\log_2 m(1+o(1)) \leq T^*(m,s) \leq s^2 \log_2 m \log_2 e(1+o(1)). \tag{3.2}$$

Any non-adaptive combinatorial group testing algorithm with $T(m, s)$ tests can be expressed as a testing matrix C with $T(m, s)$ rows and $N(m, s)$columns, where each row corresponds to a group test and each column corresponds to a failure pattern. One can set $c_{ij} = 0$ if group test i would fail under failure pattern j; otherwise, $c_{ij} = 1$. As a simple illustration, consider the case of $s = 1$ and $m = 7$; the testing matrix is shown in Table 3.1. In this case, the algorithm performs three group tests. The elements involved in these tests are respectively $\{4, 5, 6, 7\}$, $\{2, 3, 6, 7\}$ and $\{1, 3, 5, 7\}$. If element i has failed, the results of the tests are identical to column i, which is the binary representation of i. If no element has failed, all tests return zero. Thus $T(7, 1) = 3$, which corresponds to the lower bound of (3.1).

A similar construction yields an efficient procedure to find a single failed element in any group of m elements. This procedure plays an important role in the fault diagnosis algorithms of Section 3.4 and 3.5. The construction involves a matrix with $\lceil \log(m + 1) \rceil$ rows (corresponding to the tests) and $m + 1$ columns (corresponding to the $m+1$ possible failure patterns). Column 0 corresponds to the scenario in which all elements are intact, and column i ($i = 1, \ldots, m$) corresponds to the scenario in which

[4] $f(n) = O(g(n))$ means that there exists a constant c and integer N such that $f(n) \leq cg(n)$ for all $n > N$. $f(n) = \Omega(g(n))$ means $g(n) = O(f(n))$. $f(n) = \Theta(g(n))$ means both $f(n) = O(g(n))$ and $g(n) = O(f(n))$.

the element i has failed. We set column i of the matrix to be the binary representation of i. Each row corresponds to a group test which tests the subset of objects which have a 1 entry in the row of the diagnosis matrix. It is easy to see that if item i has failed then the outcome of the tests will be precisely the binary representation of i. In the rest of this chapter, we refer to this procedure as the *logarithmic testing procedure* (LTP).

The non-adaptive network fault diagnosis problem can be formulated as a non-adaptive combinatorial group testing problem, under some additional constraints. In particular, in this formulation of the non-adaptive fault diagnosis problem, there are up to s edge failures among the set of m network edges. A set of permissible probes are sent concurrently to test whether any edge of the corresponding walk has failed. It follows that the non-adaptive fault diagnosis problem is equivalent to a non-adaptive combinatorial group testing problem, under the constraint that the group test can be performed only if it corresponds to a permissible probe. This variant of CGT is called the problem of *combinatorial group testing on graphs.* This chapter addresses the non-adaptive fault diagnosis problem by proving several results concerning combinatorial group testing on graphs.

3.4 Efficient Fault Diagnosis for Regular Networks

In this section, we present efficient non-adaptive fault diagnosis algorithms for network topologies with different classes of regular graphs, and characterize the minimum number of non-adaptive probes required to identify up to s failed edges in graph G. The minimum number of non-adaptive probes required is denoted as $L^*(G, s)$.

3.4.1 Networks with Line or Ring Topologies

Line topologies[5] are used mostly for distribution networks in optical networks. Ring topologies are also widely used and are largely similar to linear networks, from a fault diagnosis perspective.

Consider a line network consisting of n nodes, indexed by integers $0, 1, \ldots, n-1$. The edges are $\{i, i+1\}$ for $0 \leq i \leq n-2$. For line networks, one can establish the following result.

Theorem 3.1. *The minimum number of non-adaptive probes to locate up to a single edge failure in a line network of n nodes, denoted as $L^*(G, s = 1)$, is precisely $\lceil n/2 \rceil$, i.e.,*

$$L^*(G, s = 1) = \lceil n/2 \rceil = \Theta(n). \tag{3.3}$$

Proof. Let t be an arbitrary probe in a line network. Let a be the node with the smallest index that is contained in t , and b be the node with the largest index contained in t. Note that probe t is equivalent to a path from node a to node b. One can use the notation $t = [a, b]$ and call $a(b)$ the head (tail) of t.

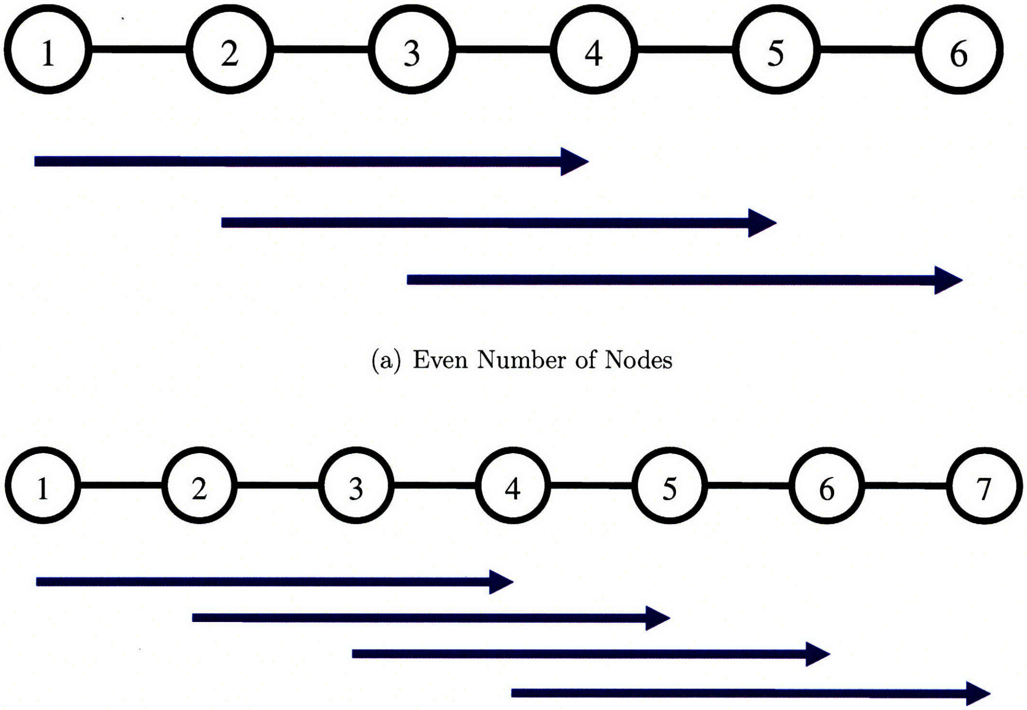
First, consider the lower bound of L^*. Let $P = \{t_1, \ldots, t_l\}$ be a set of probes that can detect a single edge failure. Suppose $2L^* < n$; then there exists a node i that is neither a head or a tail of any test t_j. Considering the following two cases:

- $i = 0$ or $n-1$: In this case, no probe t_j includes an edge that is adjacent to node i. Therefore, the probe algorithm cannot identify whether the edge adjacent to node i has failed or not.

- $1 \leq i \leq n-2$: In this case, every test t_j either contains both edge $\{i-1, i\}$ and edge $\{i, i+1\}$, or contains neither. Therefore, the probe algorithm cannot distinguish between the case when edge $\{i-1, i\}$ has failed and the case when edge $\{i, i+1\}$ has failed.

In both cases, one can arrive at a contradiction and conclude that is a necessary condition.

[5] Strictly speaking, line graphs are not regular. However, they can be approximated by ring graphs that are regular.



(a) Even Number of Nodes

(b) Odd Number of Nodes

Figure 3-3: Optimal fault-diagnosis schemes for line networks with different number of nodes: (a)Even number of nodes, and (b) Odd number of nodes.

Now, let us proceed to the upper bound of L^*. Consider the probe test t_j , where $t_j = [j, i + \lfloor n/2 \rfloor$ for $0 \leq j \leq \lceil n/2 \rceil - 1$, as illustrated in Fig. 3-3 for $n = 6$ (see Fig. 3-3(a)) and $n = 7$ (see Fig. 3-3(b)). Clearly, every edge e belongs to some test t_j. Therefore all one needs to show is that, for every pair of edges $e_1 \neq e_2$, there is a test t_j that contains exactly one of the edges. This will imply that, given all the probe syndromes, one can locate the faulty edge or decide that no failure has occurred. Let $e_1 = [t_1, h_1]$ and $e_2 = [t_2, h_2]$. Without loss of generality, we assume $h_1 \leq t_2$. Consider the following two cases:

- $h_1 \geq \lceil n/2 \rceil$: In this case, the test $[h_1 - \lfloor n/2 \rfloor, h_1$ contains e_1 but not e_2.
- $h_1 < \lceil n/2 \rceil$ In this case, either the test $[h_1, h_1 + \lfloor n/2 \rfloor]$ or the test $[\lceil n/2 \rceil - 1, n-1]$ contains e_2 but not e_1.

This completes the proof. □

This $\Theta(n)$ bound for line networks is much larger than the lower bound of $\log(n)$ in (3.1). Intuitively, the low connectivity of the line/ring network topology restricts the possible tests to such an extent that testing becomes inefficient. Note that with a line network, s becomes irrelevant (one can handle any s with $m = n - 1$ probes). It can be shown that the same result can be proved for ring networks by simply cutting the ring network into a line network at any node.

3.4.2 Networks with Fully-Connected Topologies

This sub-section deals with the non-adaptive fault diagnosis problem for all-optical networks whose topologies are fully-connected (i.e., complete graphs). For a topology of n nodes, denoted K_n, each node is connected to all other nodes in the network, resulting in $m = n(n-1)/2$ edges in the network. The case $n = 5$ is illustrated in Fig. 3-4. For such a network, we have followed a "trial-and-error" approach to develop an optimal non-adaptive fault diagnosis algorithm, as illustrated in Algorithm 2.

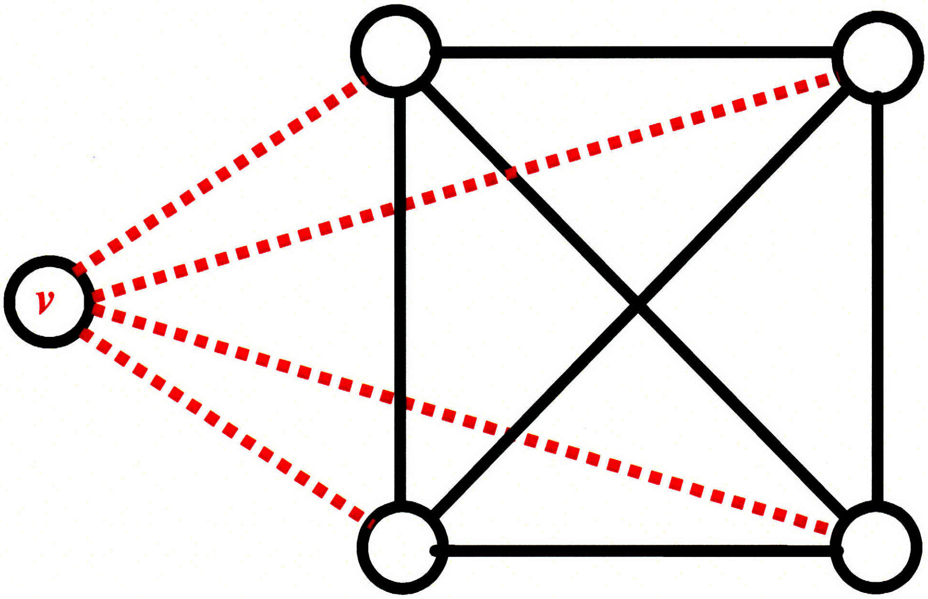


Figure 3-4: The complete graph with $n = 5$, where node v and its neighborhood are used to route probes.

Algorithm 2 Testing for a single failure in complete networks

Step 1a:

Arbitrarily pick a node v and define its neighborhood sub-graph $b(v)$ as the $n-1$ edges that connect it to all other nodes. As shown in Fig. 3-4, the neighborhood is a star centered at node v.

Step 1b:

Perform the LTP on the sub-graph $b(v)$. Each LTP test becomes a valid probe due to the star topology.

Step 2:

Perform the LTP on the subgraph obtained by deleting node v. The sub-graph $b(v)$ is used to route the probes as needed.

Notice that, although we separate the probes into different steps, all the probes are actually sent out in parallel. The correctness[6] of Algorithm 2 can be proved as follows. If the network topology did not impose any constraints on the choice of probes (that is, if an arbitrary subset of edges formed a permissible probe), then one could directly apply the LTP procedure, using the individual edges as elements to be tested. Unfortunately, the topology restricts the choice of probes to sequences of adjacent edges, so the probes are chosen more carefully. At a high level, the approach is first to identify a fault-free sub-graph, then to use this sub-graph to route the probes for an LTP procedure.

Theorem 3.2. $\Theta(\log n)$ *probes are necessary and sufficient to identify a single edge failure in a fully connected network with nodes.*

Proof. Algorithm 2 uses two LTPs, of size $n-1$ and size $(n-1)(n-2)/2$ respectively, and therefore the total number of probes required is $\mathbf{O}(\log n)$. Combining this result with the lower bound of (3.1), one can establish that $\Theta(\log n)$ probes are necessary and sufficient to identify a single edge failure in a fully connected network with nodes. □

3.4.3 Networks with 2-D Grid Topologies

The sub-section considers two-dimensional grid networks[7] of size $\sqrt{n} \times \sqrt{n}$. Such structures are also commonly used as interconnection networks [39]; in the context of all-optical networks, they are sometimes called Manhattan Networks. Fig. 3-5 illustrates the case of $n = 25$. Using a "trial-and-error" approach, we have developed an optimal non-adaptive fault diagnosis scheme for 2-D grid networks, as illustrated in Algorithm 3. Notice that, although there are different steps in the algorithm, all the probes are sent out simultaneously.

The correctness of Algorithm 3 is shown in Appendix B.1. As with Algorithm 2, the strategy is first to identify a fault-free sub-graph (either column 1 or row 1), and

[6] In theoretical computer science, correctness of an algorithm is asserted when the algorithm does what it is supposed to do, with respect to a specification.

[7] Strictly speaking, 2-D grid graphs are not regular, but they can approximated by 2-D torus graphs that are regular.

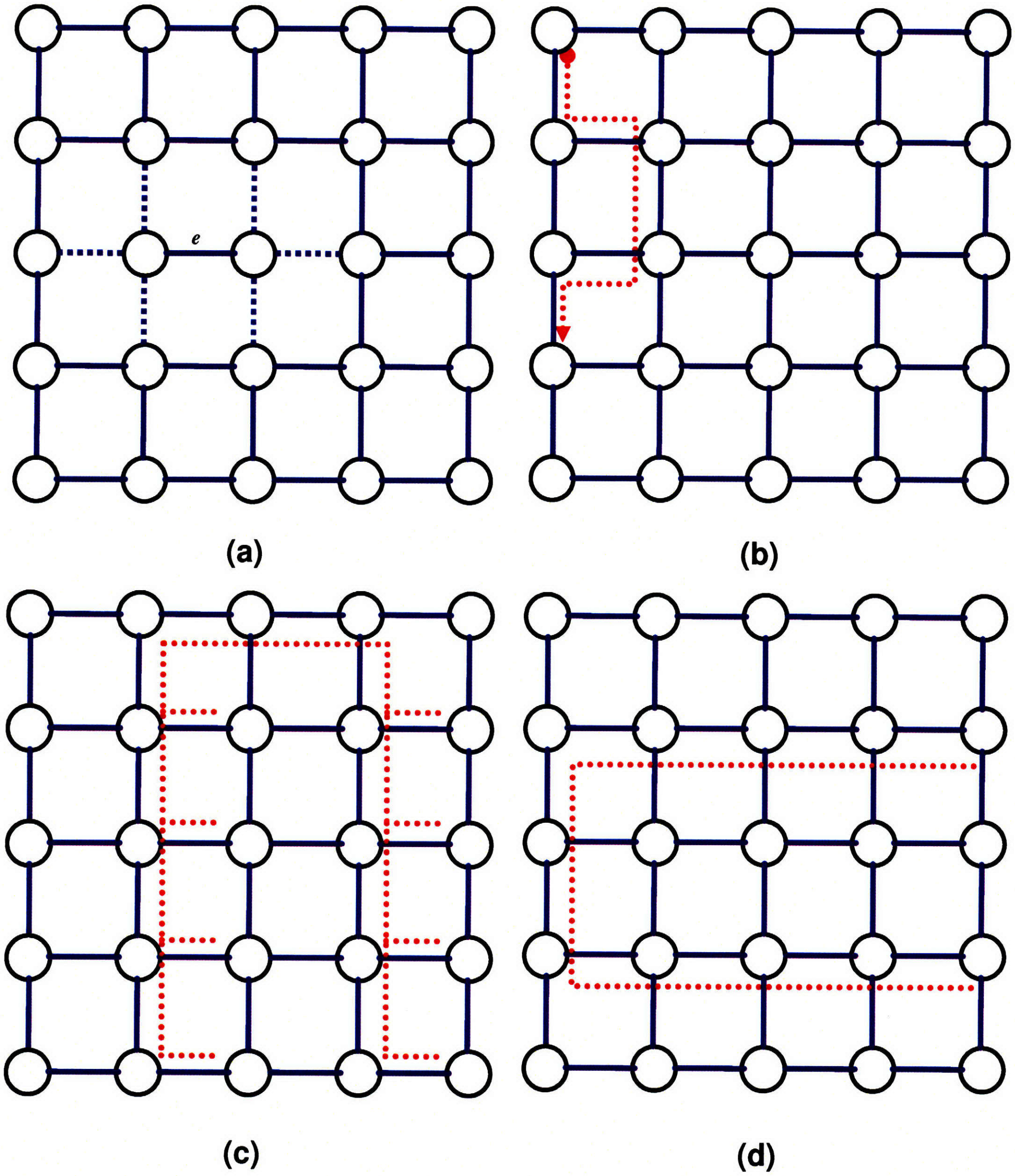


Figure 3-5: (a) A 2-D grid with 25 nodes. If at most 7 failures are allowed, then the failure of edge e cannot be detected efficiently by non-adaptive tests. (b) A single probe to test edge 1 and edge 3 on Column 1. (c) A single probe to test Column 2 and Column 4. (d) Single probe to test the 2nd edge on all rows and the 4th edge on all rows.

Algorithm 3 Testing for a single failure in 2-D grid networks

Step 1a:
Test all edges in column 1 using a single probe.

Step 1b:
Perform the LTP on the edges in column 1 using edges between column 1 and column 2 and edges in column 2 to route the probes as necessary. Fig. 3-5(b) illustrates a single probe to test edge 1 and edge 3 in column 1, numbering the edges in increasing order from top to bottom.

Step 2a:
Test all edges in row 1 using a single probe.

Step 2b:
Perform the LTP on the edges in row 1 using edges between row 1 and row 2 and edges in row 2 to route the probes as necessary. (This is similar to Step 1b.)

Step 3a:
Perform the LTP on row 2 through row $\sqrt{n}$. This step differs from Steps 1b and 2b in that an entire row is treated as a single element for testing purposes. The edges in column 1 are used to route between rows. Fig. 3-5(c) illustrates a single probe to test row 2 and row 4.

Step 3b:
Perform the LTP on the individual row edges (the elements are $s_1, \dots, s_{\sqrt{n}}$, where $s_i = \{i_{th}\text{edge of row}j : 2 \leq j \leq \sqrt{n}\}$). The column edges and the edges of row 1 are used to route between rows. Fig. 3-5(d) illustrates a single probe to test the 2nd edges in all rows and the 4th edge in all rows.

Step 4a:
Perform the LTP on column 2 through column $\sqrt{n}$, in a manner analogous to Step 3a.

Step 4b:
Perform the LTP on the column edges, in a manner analogous to Step 3a.

then to use the fault-free sub-graph to route the necessary probes required by the LTPs.

Theorem 3.3. $\Theta(\log n)$ *probes are needed to identify a single edge failure in a 2-D grid network of size* $\sqrt{n} \times \sqrt{n}$.

Proof. Algorithm 3 uses only 6 LTPs, each over a set of $\sqrt{n}$ elements, plus two additional probes. It follows that the total number of probes used is only $\mathbf{O}(\log n)$. Combining this result with the lower bound of (3.1), we have established that $\Theta(\log n)$ probes are needed to identify a single edge failure in a 2-D grid network of size $\sqrt{n} \times \sqrt{n}$. □

In general, if multiple failures can occur simultaneously, more probes are needed. This phenomenon can be intuitively explained as follows. An edge e can hide behind a small cut which separates it from the rest of the network. If all the edges of this cut have failed, the only way to test whether edge has also failed is to probe the edge by itself. Theorem 3.4 explains this phenomenon formally.

Theorem 3.4. *If at least 7 failures can occur,* $\Theta(n)$ *probes are needed to identify all the edge failures in a 2-D grid network.*

Proof. As illustrated in Fig. 3-5(a), the minumum cutset that separates any single edge (e.g., edge e) has an order to 6. If all the edges in the cutset have failed, the only way to test whether edge e has also failed is to probe edge e itself. However, the identity of edge e is not known when the algorithm chooses its probes, due to the non-adaptive nature of the algorithm. Therefore, the algorithm can only know whether edge e has also failed if it performs $\Omega(m) = \Omega(n)$ probes. Combining this with the upper bound of (3.1) completes the proof. □

3.5 Efficient Fault Diagnosis for Arbitrary Topologies

Efficient testing algorithms for arbitrary graphs and trees are presented in this section. The algorithms depend on the diameter[8] and/or the edge connectivity [9]of the graph. On practical networks, one would expect the diameter to be relatively small, and the connectivity to be large (for failure resilience).

3.5.1 Networks with Well-Connected Topologies

As shown in Section 3.4, identifying multiple failed edges in some networks (e.g., 2-D grid networks) requires exponentially more probes than required for a single failed edge. This high complexity is caused by edge failures that can hide behind small cuts. One might conjecture that this phenomenon does not occur in graphs with sufficiently high connectivity. The following theorem proves such a result.

Theorem 3.5. *If a graph G contains $s+1$ edge-disjoint spanning trees[10], the minimum number of non-adaptive probes required to identify up to s failed edges, i.e., $L^*(G,s)$, is bounded by $T^*(m,s) \leq L^*(G,s) \leq \mathbf{O}(s \cdot T^*(m,s))$, where $T^*(m,s)$is as defined in Section 3.3.*

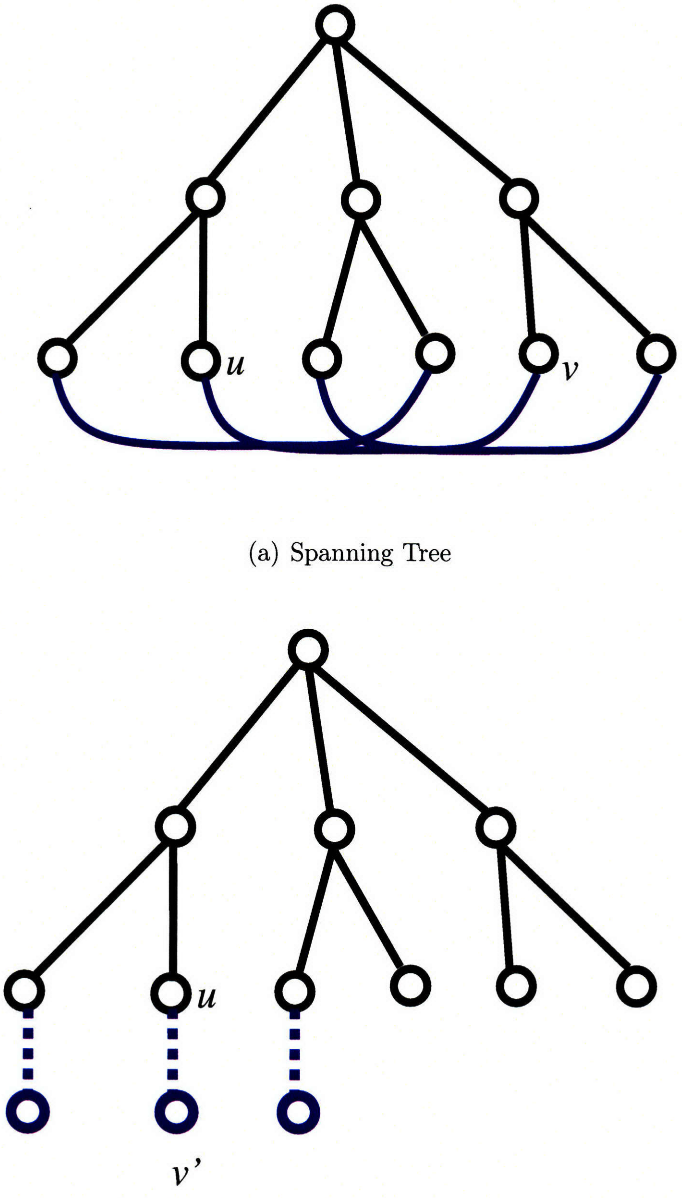
Proof. The lower bound is immediate since the non-adaptive fault diagnosis problem is polymorphic to the combinatorial group testing problem with an additional restriction on the feasible probes.

Let us proceed to the upper bound now. If the graph G contains $s+1$ edge-disjoint spanning tree, at least one of the spanning trees, call it G_T, contains no edge failures according to the Pigeon Hole Principle. A single probe suffices to test if all edges of a tree are intact, therefore one can identify G_T using only $s+1$ probes. As illustrated in Fig. 3-6, for every non-tree edge $\{u,v\}$ (Fig. 3-6(a)), one can create a virtual node

[8] The diameter of a graph is the maximum shortest distance between any two nodes in the graph.

[9] Edge-connectivity means the minimum cardinality of any subset of edges whose removal disconnects the network.

[10] A spanning tree of a graph is an acyclic sub-graph containing all the nodes.



(a) Spanning Tree

(b) Non-tree Edges Transformation

Figure 3-6: Transformation from non-tree edges in the original graph into tree-edges in the new graph.

$v'$and replace $\{u, v\}$ with $\{u, v'\}$ (Fig. 3-6(b)). After this transformation, all non-tree edges are at the bottom of G_T, i.e., they have height zero.

Now consider these non-trees edges as the elements to be tested, and one can use any CGT algorithm to test them by using the fault-free spanning tree G_T as the hub. Pick a root for G_T arbitrarily; one can think of the CGT algorithm as running at this root node. By the choice of G_T, the path from the root to each of the non-tree edges contains no failures. The CGT algorithm produces a sequence of tests, each of which specifies a set of elements to test. For each such set, one sends a probe from the root node which traverses the tree and visits only the non-tree edges in the specified set. Therefore a probe fails if and only if one of the elements in the corresponding CGT test has failed. The results of these probes are returned to the CGT algorithm, and it identifies the failed edges.

To summarize, the optimal non-adaptive CGT algorithm can be applied to the set of non-tree edges, using the edges of G_T to route from the root to the non-tree edges. This approach uses $\mathbf{O}(T^*(m, s))$ probes. Since we have to perform these tests for all $s + 1$ trees, $\mathbf{O}(T^*(m, s))$ probes are sufficient. □

We now illustrate this theorem by comparing it to our earlier results. A 2-D grid network has edge-connectivity 2, since the corner nodes have degree only 2. Therefore Theorem 3.5 yields no result for 2-D grids. On the other hand, consider a 2-D torus, i.e., a grid in which the edges wrap around. Such a graph is shown in Fig. 3-7(a). Any 2-D torus has edge connectivity 4, so it has two disjoint spanning trees. An example of two spanning trees in a 2-D torus is shown in Fig. 3-7(b). As consequences of Theorem 3.5, one can have the following two corollaries.

Corollary 3.1. *In a 2-D torus with m edges,* $\Theta(\log m)$*probes are sufficient to identify a single edge failure.*

Proof. A 2-D torus has two edge-disjoint spanning tree, as illustrated in Fig. 3-7. When $s = 1$, the number of probes required is bounded by

$$T^*(m, s = 1) \leq L^*(G, s = 1) \leq \mathbf{O}(1 \cdot T^*(m, s = 1)). \tag{3.4}$$

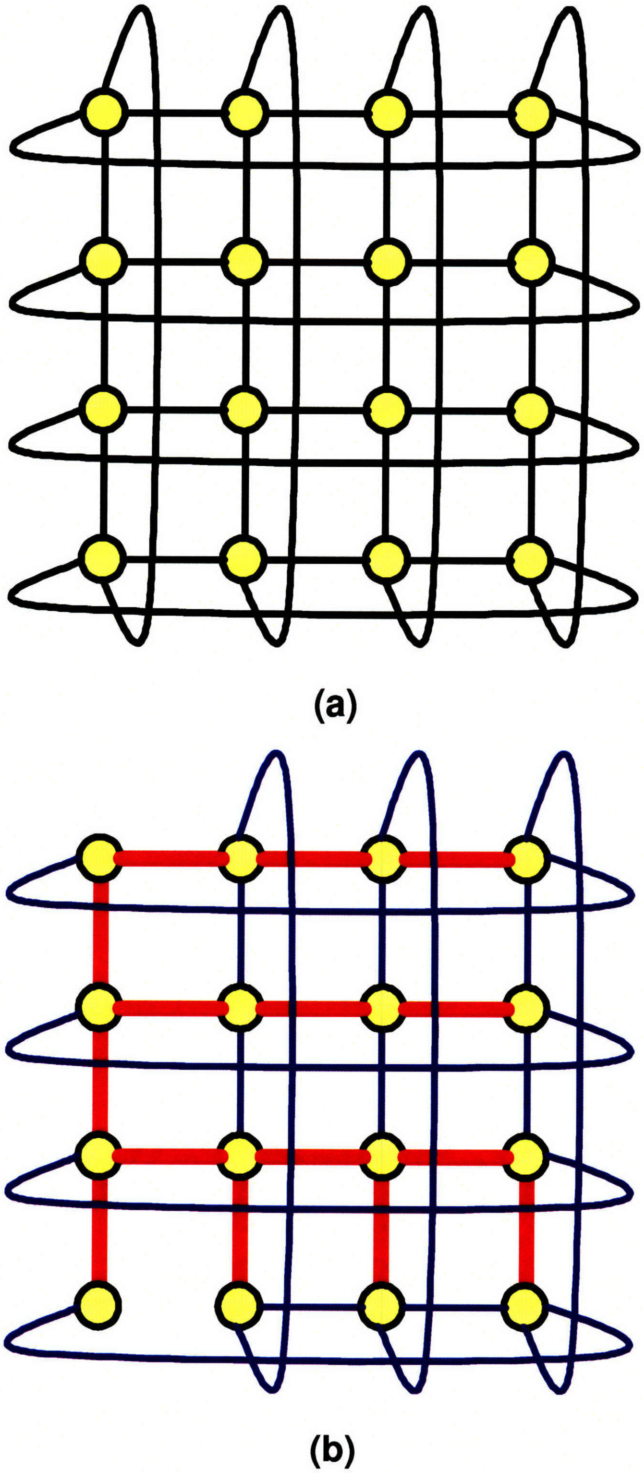


Figure 3-7: (a)A 2-D torus of size 4x4. (b) Two edge-disjoint spanning trees contained in the 2-D torus.

This suggests that $L^*(G, s = 1) = \Theta(T^*(m, s = 1))$. From the definition of LTP, we know that $T^*(m, s = 1) = \Theta(\log m)$. It follows that $L^*(G, s = 1) = \Theta(\log m)$. □

Corollary 3.2. *In a complete (i.e., fully connected) network with n nodes, $\mathbf{O}(s \cdot T^*(m, s))$ probes are sufficient to identify up to $s \leq (n-3)/2$ failed edges.*

Proof. According to the Tutte-Nash-Williams theorem [42, 62], a graph with edge connectivity of at least $2(s+1)$ has at least $s+1$ edge-disjoint spanning trees.

For a fully-connected graph with n nodes, the node degree is $n-1$. This corollary follows from Theorem 3.5 by having $n - 1 \geq 2(s+1)$. □

Theorem 3.5 also suggests the following general paradigm for applying classical CGT procedures (such as LTP) to combinatorial group testing problems on graphs.

- Preprocessing

  1. Identify $s+1$ edge-disjoint connected sub-graphs. Each sub-graph will be used in turn as a "hub" to reach the edges of the graph outside itself.
  2. For each hub, use a CGT algorithm to generate tests for the set of edges outside it.

- Probing the network non-adaptively

  3. For each hub, verify that its edges are intact.
  4. For each hub, each test from Step 2 is implemented by a permissible probe as follows: the probe traverses the interior of the hub, and steps out only onto the neighboring edges that are to be tested. Note that, assuming the hub is intact, the probe fails if and only if one of the edges to be tested has failed.

- Diagnosis

  5. Since there are at most s failures and $s+1$ edge-disjoint hubs, at least one contains no failed edge. Such a hub can be identified based on the results of Step 3. All other hubs are ignored by the diagnosis algorithm.

6. Run the CGT algorithm on the results of Step 4 for the good hub, thus identifying all failed edges.

It can be seen that Algorithm 2 is a special case of this general procedure with $s = 1$. Similar fault diagnosis algorithms can be designed for other regular networks of degree d.

3.5.2 Networks with Tree Topologies

In this subsection, we consider networks with tree topologies, and obtain bounds in terms of the diameter D.

Theorem 3.6. *For any tree G^T , when $s = 1$, we have:*

$$\Omega(\log n) \leq L^*(G^T, 1) \leq \mathbf{O}(D \cdot \log n), \tag{3.5}$$

where D is the diameter of the graph G^T.

The proof of Theorem 3.6 is given in Appendix B.2.

3.5.3 Networks with Arbitrary Topologies

In this sub-section, we address the fault diagnosis problem for networks with arbitrary topologies. The main result is summarized as follows.

Theorem 3.7. *If a graph G contains s edge-disjoint spanning trees $T_1, \ldots, T_s$, then the minimum number of non-adaptive probes to identify up to s failed edges is upper bound by*

$$L^*(G, s) \leq \mathbf{O}(s \cdot T^*(m, s) + \sum_{i=1}^{s} L^*(T_i, s = 1)). \tag{3.6}$$

Proof. Under the given assumptions, one of the following statements must be true:

1. there is exactly one failure in each of the spanning tree;

2. there is at least one spanning tree having no edge failure, while some spanning tree has multiple failures.

For Case 1, one can use $L^*(T_i, s = 1)$ probes to find the failure in the spanning tree T_i. For Case 2, we can use the fault-free spanning tree as a hub to diagnose at most s failures among the remaining edges, by using the non-tree edge transformation illustrated in the proof of Theorem 3.5. This needs $T^*(m, s)$ probes. Since the fault-free tree is unknown beforehand, we need to do it to all the spanning trees. Thus the total number of probes required in this case is $s\dot{T}^*(m, s)$. □

Theorem 3.7 implies an upper bound for arbitrary graphs as follows.

Corollary 3.3. *For an arbitrary graph G and $s = 1$, we have:*

$$L^*(G, 1) \leq \mathbf{O}(D + \log^2 n), \tag{3.7}$$

where D is the diameter of the graph.

Proof. Choose the spanning tree to be a shortest path tree from an arbitrary starting node. This guarantees that the depth of the tree is at most the diameter of G. It follows from Theorem 3.6 that $L^*(T_i, s = 1) = \mathbf{O}(D + \log^2 n)$ and from the LTP that $T^*(m, s = 1) = \log n$. □

3.6 Conclusion

In this chapter, we focused on the proactive fault-diagnosis framework, in which a set of probes are sent along lightpaths to test whether they have failed; the network failure pattern is identified using the results of the probes. In particular, a non-adaptive probing design, where all the probes are sent in parallel, are investigated. The key objective of our design is to minimize the number of probes sent, in order to minimize the diagnostic effort.

The non-adaptive fault diagnosis problem for all-optical networks is equivalent to the combinatorial group testing problem on graphs. In the latter problem, probes can only be sent over walks over the graph, and therefore such probes correspond to lightpaths in all-optical networks. Under this framework, we develop in this chapter develops efficient fault diagnosis algorithms for different classes of network topologies. In particular, we assume an upper bound on the number of simultaneous link failures and exploit this fact to obtain upper and/or lower bounds on the number of non-adaptive lightpath probes needed and derive optimal or near-optimal non-adaptive fault-diagnosis schemes for different classes of network topologies. The non-adaptive fault diagnosis algorithms proposed in this chapter share a common theme: *a fault-free sub-graph should be identified in the network and serve as a hub to route other necessary probes to diagnose failures in the network.*

Our research in this chapter has some limitations. First, we assumed that the number of simultaneous link failures is upper bounded by a known number s. In practical optical networks, this number is normally unknown, in which case the solution would be to test each individual link. Our rationale is that the number of simultaneous link failures cannot be too large for a reasonably reliable network. Second, under the non-adaptive fault-diagnosis paradigm, we provided solutions only to a limited set of topologies, including ring, tree, 2-D grid, well-connected graphs. Future work should focus on extending this framework to more generalized mesh networks. Finally, we had not fully exploited the connection between the non-adaptive fault-diagnosis problem and the superimposed code problem. A deeper understanding of this connection could potentially reveal more insights about the non-adaptive fault-diagnosis problem.

Although this research was presented in the context of all-optical networks, we believe that our methods based on combinatorial group testing on graphs can be employed in other network contexts to solve fault diagnosis problems.

Chapter 4

Hardware Provisioning for Proactive Fault Diagnosis Schemes

This chapter addresses the diagnostic hardware cost for proactively fault diagnosis schemes, i.e., the diagnostic transmitter/receiver (Tx/Rx) cost[1]. Our investigation suggests that the hardware cost can be reduced significantly by accepting a small amount of uncertainty about failure status.

As pointed in Chapter 1, we are interested in two design metrics for the proactive fault-diagnosis schemes: i) the diagnostic effort (i.e., the number of lightpath probes) and ii) the diagnostic hardware cost (i.e., the number of transmitters and receivers for diagnosis purpose. Previously in Chapter 2 and 3, we have established a theoretical framework to minimize the number of lightpath probes and have developed asymptotically optimal fault diagnosis schemes to keep the diagnostic effort low, [68, 69, 70]. At the same time, due to the unique cost structure of optical networks[2], the hardware cost, i.e., Tx/Rx pairs needed to transmit and detect optical probing signals, contributes a significant portion of the fault-diagnosis cost. In this chapter,

[1] The content in this chapter has been accepted to 2008 IEEE International Conference in Communications[67].

[2] In optical networks, transmitters and receivers are usually expensive. The price of an optical transmitter could range from a few hundred dollars (for single fixed wavelength) to several thousand dollars (for wide-band tunable lasers).

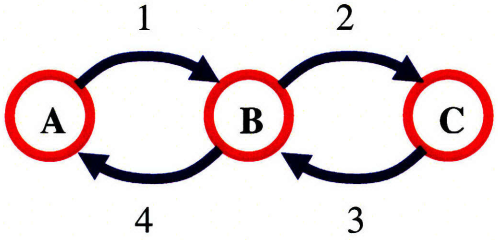


Figure 4-1: A motivational example: the trade-off between the cumulative diagnosability probability and the number of nodes equipped with diagnostic Tx/Rx pairs.

we aim to characterize this Tx/Rx hardware cost and understand its implications on practical network design.

In particular, we develop a probabilistic framework to investigate the diagnostic Tx/Rx cost for the proactive fault diagnosis paradigm. As a benchmark, we first show that all the network nodes should be equipped with diagnostic Tx/Rx pairs in order to identify all possible network states. This result prompts us to investigate the impact on diagnosis performance when only a small fraction of nodes is equipped with diagnostic Tx/Rx pairs. Our analytical results suggest a trade-off between the number of nodes equipped with diagnostic Tx//Rx pairs and the diagnosis capability. The metric we employ for the diagnosis capability is the probability of all identifiable network states, defined as *the cumulative diagnosability probability.* This trade-off can be illustrated via the following example.

Consider a line network with 3 nodes and 4 directed arcs, where nodes do not fail and arcs fail independently with probability p, as illustrated in Fig. 4-1. First, if only one node is equipped with a Tx/Rx pair (either A, B or C), one can only diagnose the network state with zero arc failure, and the cumulative diagnosability probability is $(1-p)^4$. Second, if two nodes are equipped with Tx/Rx pairs (e.g., node A and C), the identifiable network state set[3] is $\{\Phi$, $\{1\}$, $\{2\}$, $\{3\}$, $\{4\}$, $\{1,2\}$, $\{3,4\}$, $\{1,4\}$,$\{2,3\}\}$, where Φ denotes the network state with zero arc failures and 1 denotes the network state with arc 1 failure, and so on. In this arrangement, only a subset of the network states with two arc failures can be uniquely identifiable. The

[3] A formal proof of this result will be presented in a generalized case in Section 4.2.

cumulative diagnosability probability is $(1-p)^4 + 4p(1-p)^3 + 4p^2(1-p)^2$. Finally, if all the nodes are equipped with diagnostic Tx/Rx pairs, any network state can be identified and thus the cumulative diagnosability probability is 1. It is evident from this example that the cumulative diagnosability probability increases as the number of nodes equipped with Tx/Rx pairs increases.

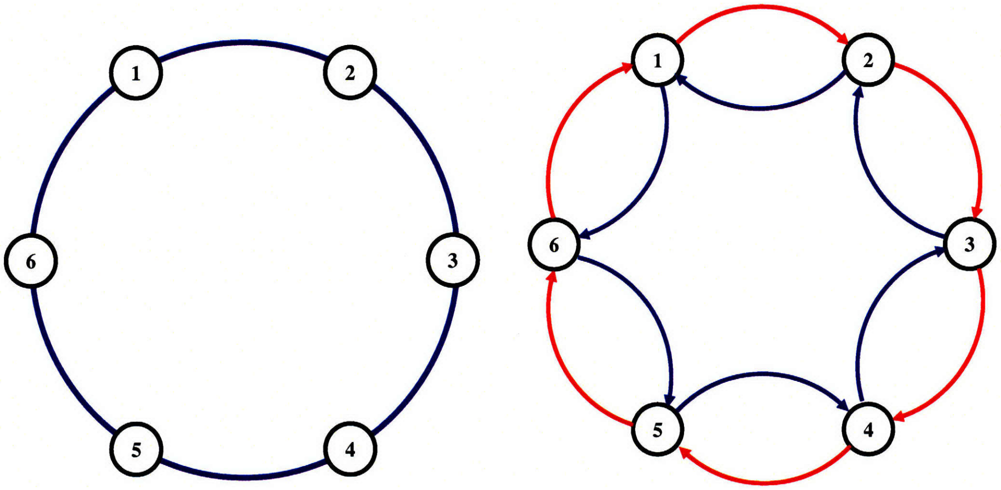
This example suggests an opportunity to reduce the diagnostic cost, especially the diagnostic Tx/Rx hardware cost, by accepting a reduced cumulative diagnosability probability. In particular, when the network is relatively reliable, only a small fraction of nodes equipped with Tx/Rx pairs is needed to provide a high diagnosis fidelity. It follows that a significant portion of the worst-case fault diagnosis hardware cost can be saved in exchange for an acceptable amount of uncertainty about the network's state.

This chapter is organized as follows. In Section 4.1, we present the proactive fault diagnosis architecture for all-optical networks including the network model, the fault diagnosis cost model, and the probabilistic analysis framework. In Section 4.2, we derive the cumulative diagnosability probability for any ring network by decomposing the network into a set of canonical linear networks with Tx/Rx pairs at both end nodes, and characterize the trade-off between the number of nodes equipped with Tx/Rx pairs and the cumulative diagnosability probability for ring networks. In Section 4.3, the trade-off for mesh networks is characterized via two alternative approaches: the cutset-based approach and the Euler-Trail-based approach.

4.1 Proactive Fault Diagnosis Architecture for All-Optical Networks

4.1.1 Fault Diagnosis Model

In this section, we present some highlights of the fault-diagnosis model that are immediately related to this chapter, including the topology model, the network failure model and the lightpath probing model.



(a) Undirected Graph

(b) Directed Graph

Figure 4-2: Network topology for all-optical networks: (a) undirected graph, and (b) directed graph. Each undirected link in the undirected graph is replaced by two directed arcs in the opposite directions, to illustrate bidirectional connection between adjacent nodes.

In general, all-optical networks are abstracted as undirected graphs. An undirected graph G is a pair of sets (V, E) , where V is the set of network nodes of size n, and E is the set of optical links of size m. For example, Fig. 4-2(a) illustrates an optical network with 6 nodes arranged in a ring structure. However, in practice, connections between adjacent nodes are bidirectional and are usually achieved via two parallel optical fibers transmitting optical signals in opposite directions. To capture this practical constraint, we replace each undirected edge in the undirected graph with two directed arcs in opposite directions. It follows that the original undirected graph is transformed into a directed graph, as illustrated in Fig. 4-2(b). The number of arcs in the directed graph is $2m$.

We assume in this chapter that nodes are invulnerable (the node failure case can be investigated via similar approaches as in [69]), and that arcs fail independently with probability p ($0 \leq p \leq 0.5$). Moreover, we assume that the state of an individual arc does not change over the duration of the fault-diagnosis process. Therefore, each arc state can be modeled by a Bernoulli random variable, taking value 1 with probability p for arc failure, and value 0 with probability $1 - p$ for no failure. A network state

$s \in S$ is referred to as a realization of all arc states, where $S = \{0, 1\}^{2m}$ denotes the set of all possible network states.

To detect and localize possible arc failures, we adopt the adaptive fault-diagnosis paradigm, based on the proactive lightpath-probe mechanism developed in [68, 69, 70]. In particular, optical probing signals are sequentially sent along a set of permissible lightpaths in the network and network failures are identified through the set of probe results. The result of each probe is called the *probe syndrome*, denoted as $r_t = 0$ if all the arcs along the probed lightpath are UP (no failure) and the probe signal arrives successfully; and $r_t = 1$ if any of the arcs along the probed lightpath is DOWN (at least one failure) and the probe signal never reaches the destination.

4.1.2 Design Metrics for Fault Diagnosis Schemes

As illustrated in Chapter 1, we are interested in two design metrics for fault diagnosis schemes: the diagnostic effort (i.e., the number of lightpath probes) and the diagnostic Tx/Rx hardware cost. Optical Tx/Rx pairs are used in the data plane for probe transmission and detection. This part of the diagnostic cost is a one-time cost and is proportional to the number of nodes equipped with diagnostic Tx/Rx pairs. The diagnostic effort indicates the effort expanded to scheduling, transmitting and detecting optical probes and reporting probe syndromes. The diagnostic effort is recurring and is proportional to the number of lightpath probes deployed to identify the network state.

For each design, there is some trade-off associated with it. When the diagnostic effort (i.e., the number of lightpath probes) is of interest, we characterize a trade-off between the diagnostic effort (i.e., the number of lightpath probes) and the diagnostic delay (i.e., the number of diagnostic steps), via exploiting three alternative diagnosis paradigms in Chapter 2 and Chapter 3. In this chapter, our concern is to minimize the diagnostic Tx/Rx cost (i.e., the number of nodes equipped with diagnostic Tx/Rx pairs). Specifically, we will investigate a trade-off between the fraction of nodes equipped with diagnostic Tx/Rx pairs and the cumulative diagnosis probability (i.e., the probability of successful diagnosis).

4.1.3 Probabilistic Analysis Framework

To identity all possible network states, any fault diagnosis scheme has to diagnose the network state with all the arcs failing simultaneously. This, in turn, requires the diagnosis scheme to be able to probe each directed arc individually, which can be achieved only if each node in the network is equipped with a pair of diagnostic transmitter and receiver. It follows that, for a network of n nodes and m links (or equivalently $2m$ arcs), the number of nodes equipped with diagnostic Tx/Rx pairs is

$$n_d = n, \tag{4.1}$$

in order to identify all possible network states. However, the hardware cost of such a worst-case approach could be prohibitively high and limits its application for future all-optical networks.

The fact that the probability mass is not evenly distributed among all network states provides us an opportunity to reduce the diagnostic hardware cost, with little loss in diagnosis capability. Due to the probabilistic arc failure model, some network states can occur with extremely small probability. However, in the worst-case analysis, the diagnosis scheme has to identify these network states by paying a high cost. Here, we propose a probabilistic analysis under which the objective of fault diagnosis is to identify the majority of network states by deploying less Tx/Rx pairs than the number of nodes in the network. This is similar to the lossy source coding problem in Information Theory [18] by encoding only the "typical sets".

The probabilistic analysis works as follows. We assume that n_d nodes are equipped with diagnostic Tx/Rx pairs. The fraction of network nodes equipped with Tx/Rx pairs is then defined as

$$\eta = \frac{n_d}{n}, \tag{4.2}$$

where $0 < \eta \leq 1$. For a given subset of nodes equipped with diagnostic Tx/Rx pairs, the set of all network states, denoted as S, is partitioned into two mutually exclusive and collectively exhaustive subsets: the set of identifiable network states (S_I), and the set of unidentifiable network states (S_U), with $S = S_I \cup S_U$. We define the *cumulative diagnosability probability* as the sum probability of all the network states in the set of identifiable network states, i.e.,

$$\beta_D(n, n_d, p) = \sum_{s \in S_I} \Pr(s), \tag{4.3}$$

where $\Pr(s) = p^i(1-p)^{2m-i}$ is the probability of any network state with $0 \leq i \leq 2m$ arc failures. Similarly, we define the *cumulative undiagnosability probability* as the sum probability of all the networks in the set of unidentifiable network states,

$$\alpha_F(n, n_d, p) = \sum_{s \in S_U} Pr(s), \tag{4.4}$$

The example of a 3-node linear network suggests a trade-off between the cumulative diagnosability probability (or the cumulative undiagnosability probability) and the number of nodes equipped with diagnostic Tx/Rx pairs. That is, the cumulative diagnosability probability increases as the number of nodes equipped with diagnostic Tx/Rx pairs increases and more network states can be uniquely identified. In the rest of this chapter, we characterize this trade-off for ring networks and mesh networks, and develop useful insights for engineering designs.

4.2 Efficient Tx/Rx Deployment for Ring Networks

In this section, we present a systematic approach to calculate the cumulative diagnosability probability for any ring network with a subset of nodes equipped with Tx/Rx pairs, by decomposing the network into a set of canonical linear networks, both end nodes of which are equipped with diagnostic Tx/Rx pairs. For example, in Fig. 4-2(b), if node 1 and node 4 are equipped with diagnostic Tx/Rx pairs, the network can be decoupled into two canonical linear networks, i.e., 1-2-3-4 and 4-5-6-1. In both canonical linear works, only end nodes are equipped with diagnostic Tx/Rx pairs. Therefore, we can first derive the cumulative diagnosability probability of canonical

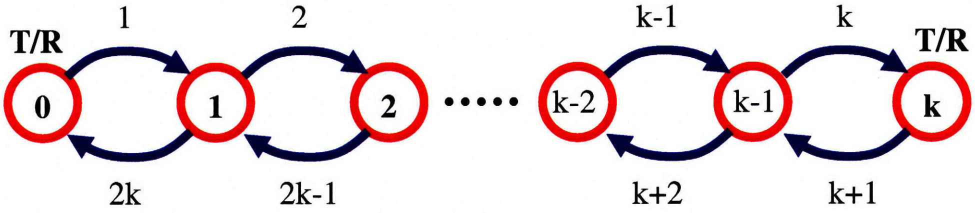


Figure 4-3: The canonical linear network with $k+1$ nodes and $2k$ arcs: nodes at both ends are equipped with diagnostic Tx/Rx pairs.

linear networks, and then synthesize the cumulative diagnosability probability for any ring network with a subset of nodes equipped with Tx/Rx pairs. Using this result, we then characterize the trade-off between the target cumulative diagnosability probability and the required fraction of nodes equipped with diagnostic Tx/Rx pairs.

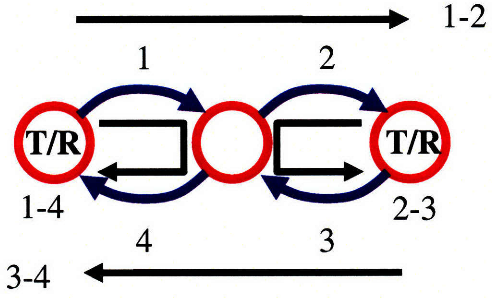
4.2.1 Canonical Network Analysis: Linear Network with Diagnostic Tx/Rx Pairs at Both End Nodes

In this subsection, we consider a canonical linear network with $k+1$ nodes and $2k$ unidirectional arcs. As illustrated in Fig. 4-3, only the two end nodes (i.e., node 0 and node k) are equipped with diagnostic Tx/Rx pairs.

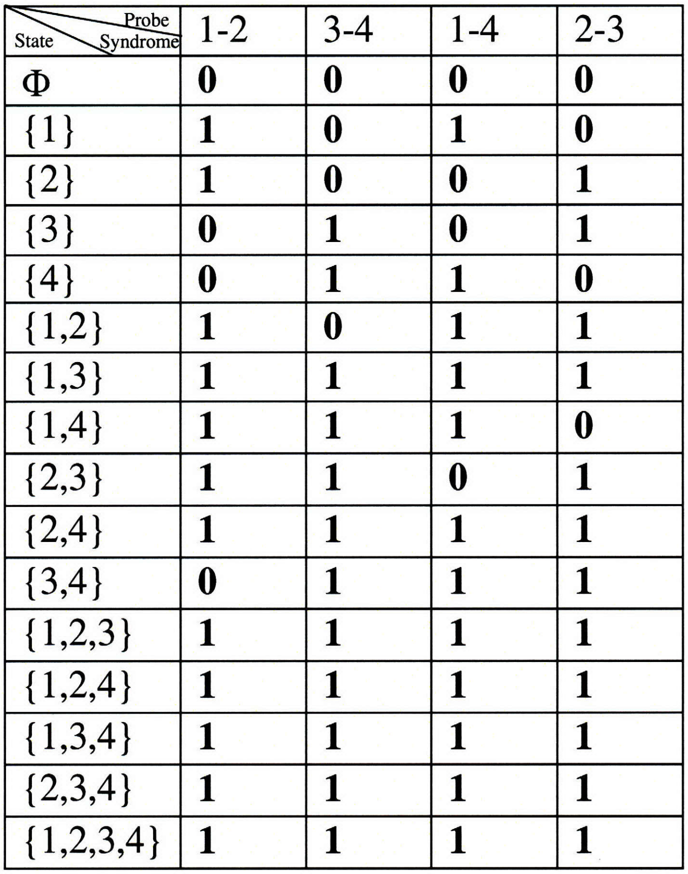
Let us first look at the case of $k = 2$, as illustrated in Fig. 4-4(a). There are 4 possible lightpath probes, i.e., 1-2, 3-4, 1-4 and 2-3. Using these four lightpath probes, we derive a diagnosis matrix as in Fig. 4-4(b), whose columns correspond to probes and rows correspond to network states. A network state is diagnosable if and only if it has a unique probe syndrome. From the diagnosis matrix, we conclude that any network state containing 3 or more edge failures is not diagnosable, and the diagnosable network states can be classified into the three types:

1. network state with zero arc failure: Φ.

2. network states with one arc failure: $\{1\}$,$\{2\}$,$\{3\}$,and $\{4\}$.



(a) Canonical Linear Network: k=2



Probe State Syndrome	1-2	3-4	1-4	2-3
Φ	**0**	**0**	**0**	**0**
{1}	**1**	**0**	**1**	**0**
{2}	**1**	**0**	**0**	**1**
{3}	**0**	**1**	**0**	**1**
{4}	**0**	**1**	**1**	**0**
{1,2}	**1**	**0**	**1**	**1**
{1,3}	**1**	**1**	**1**	**1**
{1,4}	**1**	**1**	**1**	**0**
{2,3}	**1**	**1**	**0**	**1**
{2,4}	**1**	**1**	**1**	**1**
{3,4}	**0**	**1**	**1**	**1**
{1,2,3}	**1**	**1**	**1**	**1**
{1,2,4}	**1**	**1**	**1**	**1**
{1,3,4}	**1**	**1**	**1**	**1**
{2,3,4}	**1**	**1**	**1**	**1**
{1,2,3,4}	**1**	**1**	**1**	**1**

(b) Diagnosis Matrix

Figure 4-4: (a)canonical linear network for $k = 2$: all possible lightpath probes, (b) the corresponding diagnosis matrix: each column corresponds to a probe, and each row corresponds to a network state. Each network state is uniquely diagnosable if and only if its corresponding row is unique.

3. a subset of the network states with two arc failures[4]: {1,4},{2,3},{1,2} and {3,4}. Notice that the first two network states correspond to both directions of one directional link, the last two network states correspond to two arc failures in two consecutive arcs in the same direction.

For any canonical linear network with $k \geq 3$ and two Tx/Rx pairs at each end, we notice that any network state containing 3 or more arc failures is not diagnosable. For all the network states containing 2 or less arc failures, it can be reduced to the canonical linear network with $k = 2$. It follows that, for any canonical linear network, three types of failure patterns can be identified with adaptive fault diagnosis schemes[5].

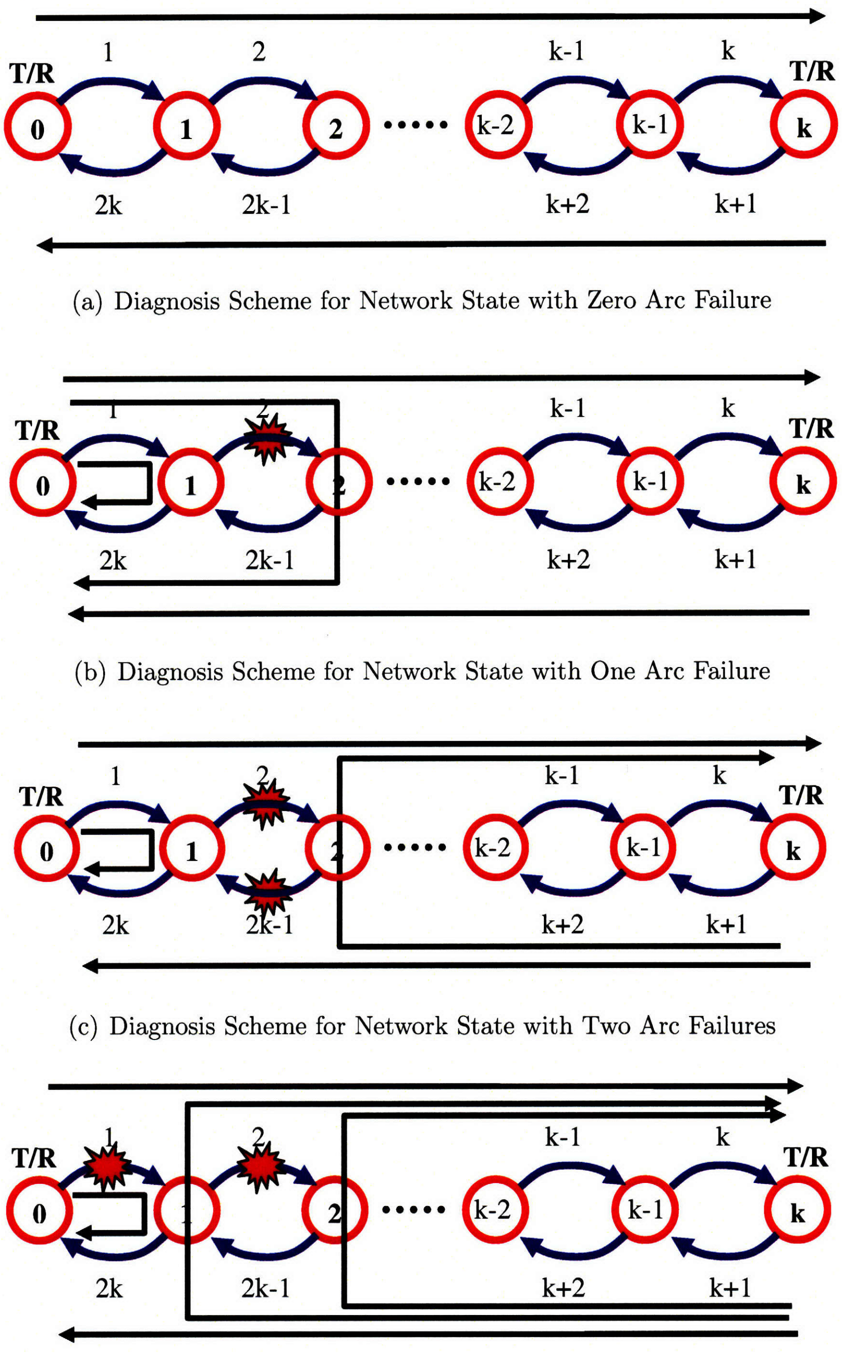
1. The first type of identifiable failure patterns contains network states with zero arc failure. The number of network states in the first category is 1 and the probability of that network state is $(1-p)^{2k}$. This network state can be identified by a probing scheme illustrated in Fig. 4-5(a), where two probes are sent from one end to the other end.

2. The second type of identifiable failure patterns contains network states with a single arc failure. The number of network states in the second category is $2k$ and the probability of such network state is $p(1-p)^{2k-1}$. This type of network states can be identified by a probing scheme illustrated in Fig. 4-5(b), where the two probes from one end to the other end detect the failure and the binary searching algorithm is used to localize the failure.

3. The third type of identifiable patterns contains a subset of the network states with two arc failures. In particular, among all the $k(k-1)/2$ network states with two arc failures, the following two classes of failure patterns, i.e., failure patterns with two arc failures in both directions of one bidirectional link (i.e., arc failures at {1, 2k}, {2,2k-1},..., {k-1,k+2} or {k,k+1}, and failure patterns with two arc failures in two consecutive arcs in the same direction (i.e., arc

[4] Note that network states {1,3} and {2,4} are indistinguishable and thus not diagnosable

[5] Here the objective is to identify possible network states, instead of minimizing the number of probes at in [68, 69, 70]



(a) Diagnosis Scheme for Network State with Zero Arc Failure

(b) Diagnosis Scheme for Network State with One Arc Failure

(c) Diagnosis Scheme for Network State with Two Arc Failures

(d) Diagnosis Scheme for Network State with Two Arc Failures

Figure 4-5: Diagnosis schemes for canonical linear networks.

failures at {1,2}, {2,3}, ..., {k-1,k}, {k+1, k+2}, ..., {2k-1,2k}. The two classes of failure patterns can be identified by probing scheme illustrated in Fig. 4-5(c) and Fig. 4-5(d), respectively. In both cases, any network state in this category can be uniquely identified by two probes from node 0 to node k and from node k to node 0 to detect, followed by two binary searching procedures from both ends to localize.. The total number of network states in the third category is $3k-2$ and the probability of such network state is $p^2(1-p)^{2k-2}$.

It follows that we can obtain the cumulative diagnosability probability for the canonical linear network as

$$\beta_D^{\dagger}(k,p) = (1-p)^{2k} + 2kp(1-p)^{2k-1} + (3k-2)p^2(1-p)^{2k-2}, \tag{4.5}$$

for $k \leq 1$ and $0 < p < 1$. Notice that the ratio between the number of identifiable network states with two arc failures and the number of network states with two arc failures, i.e., $\frac{3k-2}{k(k-1)/2}$, is on the order of $1/k$ when k is large enough. When the arc failure probability is small, the contribution of the subset of identifiable network states with two arc failures is negligible. However, when the arc failure probability is high, we need to keep the length of the canonical linear network small so that the contribution of this subset of identifiable network states with two arc failures is kept insignificant.

4.2.2 Cumulative Diagnosability Probability for Ring Networks

In this subsection, the cumulative diagnosability probability for a ring network is derived by decomposing it into a set of canonical linear networks.

Consider a ring network with n nodes, among which a subset of n_d nodes are equipped with diagnostic Tx/Rx pairs. Notice that the ring network can be decoupled into n_d canonical linear networks, both end nodes of which are equipped with diagnostic Tx/Rx pairs. For example, for the ring network in Fig. 4-6, if node 1,

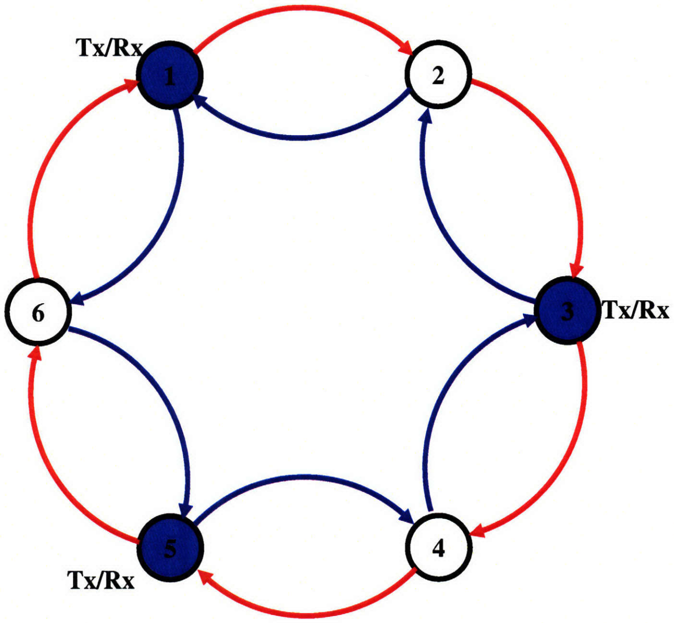


Figure 4-6: When a subset of nodes are equipped with diagnostic Tx/Rx pairs, the network can be decomposed into a set of non-overlapping canonical linear networks.

node 3 and node 5 are equipped with diagnostic Tx/Rx pairs, the network can be decomposed into three canonical linear networks, i.e., 1-2-3, 3-4-5 and 5-6-1. We denote the length of each canonical linear network as k_i, $i = 1, 2, \ldots, n_d$. Using the cumulative diagnosability probability for the canonical linear network, we can synthesize the cumulative diagnosability probability for the ring network as

$$\beta_D(n, n_d, p) = \prod_{i=1}^{n_d} \beta_D^{\dagger}(k, p), \tag{4.6}$$

where $\beta_D^{\dagger}(k, p)$ is defined in (4.5).

For a given number of nodes equipped with Tx/Rx pairs, it is natural to maximize the cumulative diagnosability probability by optimally distributing them among all the network nodes. We have not yet derived the optimum Tx/Rx distribution, but have assumed that the set of n_d diagnostic Tx/Rx pairs are evenly distributed among all the network nodes and derive the cumulative diagnosability probability under such a deployment policy. Although the uniform distribution policy may not be optimal, it is a reasonable starting point, especially for symmetric graphs.

Under the uniform Tx/Rx deployment policy, the length of each decomposed canonical linear networks is made as equal as possible and the length of each canonical linear network could be k^* and $k^* + 1$, where $k^* = \lfloor n/n_d \rfloor$. Moreover, the number of decomposed canonical linear networks[6] with length k^* is $(k^* + 1)n_d - n$, and the number of decomposed canonical linear networks with length k^*+1 is $n - k^* n_d$. Notice that, when $\frac{n}{n_d}$ is an integer, all the decomposed canonical linear networks have the same length of k^*. It follows that the cumulative diagnosability probability is given by

$$\beta_D(n, n_d, p) = \{\beta_D^\dagger(k^*, p)\}^{(k^*+1)n_d - n}\{\beta_D^\dagger(k^* + 1, p)\}^{n-k^* n_d} \tag{4.7}$$

where the first term results from the decomposed canonical linear networks of length k^* and the second term is due to the decomposed canonical linear networks of length $k^* + 1$.

In practice, the cumulative diagnosability probability of (4.7) can be further approximated as a function of the fraction of nodes equipped with Tx/Rx pairs. For the special case that n/n_d is an integer, the cumulative diagnosability probability of 4.7 would be reduced to be $\beta_D(n, \eta, p) = \{\beta_D^\dagger(\eta^{-1}, p)\}^{n\eta}$. In general, using the approximation $\beta_D^\dagger(k^*, p) \approx \beta_D^\dagger(k^* + 1, p)$, we can approximate the cumulative diagnosability probability as

$$\beta_D(n, \eta, p) \approx \{\beta_D^\dagger(\eta^{-1}, p)\}^{n\eta} \tag{4.8}$$

Therefore, for the rest of this chapter, we use (4.8) to approximate the cumulative diagnosability probability for ring networks.

4.2.3 Diagnostic Cost-Performance Trade-off

In this sub-section, we characterize the trade-off between the diagnostic hardware cost (i.e., the number of nodes equipped with diagnostic Tx/Rx pairs) and the diagnostic

[6] Let x be the number of canonical linear networks with length k^* and y be the number of canonical linear networks with length k^*+1. First, we have $x+y = n_d$. Second, we have $x \cdot k^* + y \cdot (k^* + 1) = n$. Solving these two equations, we obtain $x = (k^* + 1)n_d - n$ and $y = n - k^* n_d$.

performance (i.e., the cumulative diagnosability probability). Our results demonstrate that the diagnostic hardware cost can be reduced significantly by accepting some reasonable amount of uncertainty about the network state.

For practical engineering design, we would like to calculate the fraction of nodes equipped with Tx/Rx pairs required to provide a target cumulative diagnosability probability (or a tolerable cumulative undiagnosability probability). Indeed, for a given cumulative diagnosability probability of β_D, we can identify the minimum fraction of nodes equipped with diagnostic Tx/Rx pairs by exhaustively searching over (4.8). Specifically, we can increase the number of nodes equipped with diagnostic Tx/Rx pairs gradually until the cumulative diagnosability probability is higher than our target.

In Fig. 4-7, we plot the required faction of nodes equipped with Tx/Rx pairs, for different target cumulative diagnosability probabilities, as a function of the arc failure probability, for a ring network with 100 nodes. Notice that all the curves share a similar "S" shape, with two thresholds. In one extreme, when the arc failure probability is small, the number of nodes with Tx/Rx pairs is either 1 or 2. In the other extreme, when the arc failure probability is high, the required fraction of nodes equipped with Tx/Rx pairs is close to 1. Between these two extreme cases, there is a transition phase from a small fraction of nodes equipped with diagnostic Tx/Rx pairs to a large fraction of nodes equipped with diagnostic Tx/Rx pairs.

These observations can be understood as follows. The cumulative diagnosability probability in (4.8) can be expanded as

$$\beta_D(\eta) = (1-p)^{2n} + 2np(1-p)^{2n-1} + \mathbf{O}(p^2), \tag{4.9}$$

where $\mathbf{O}(p^2)$ denotes a polynomial of p with an order of at least 2. Notice that each term in (4.9) corresponds to one class of identifiable network states. The first term of $(1-p)^{2n}$ corresponds to the subset of network states with zero arc failure. The second term of $2np(1-p)^{2n-1}$ corresponds to the subset of network states with a single arc

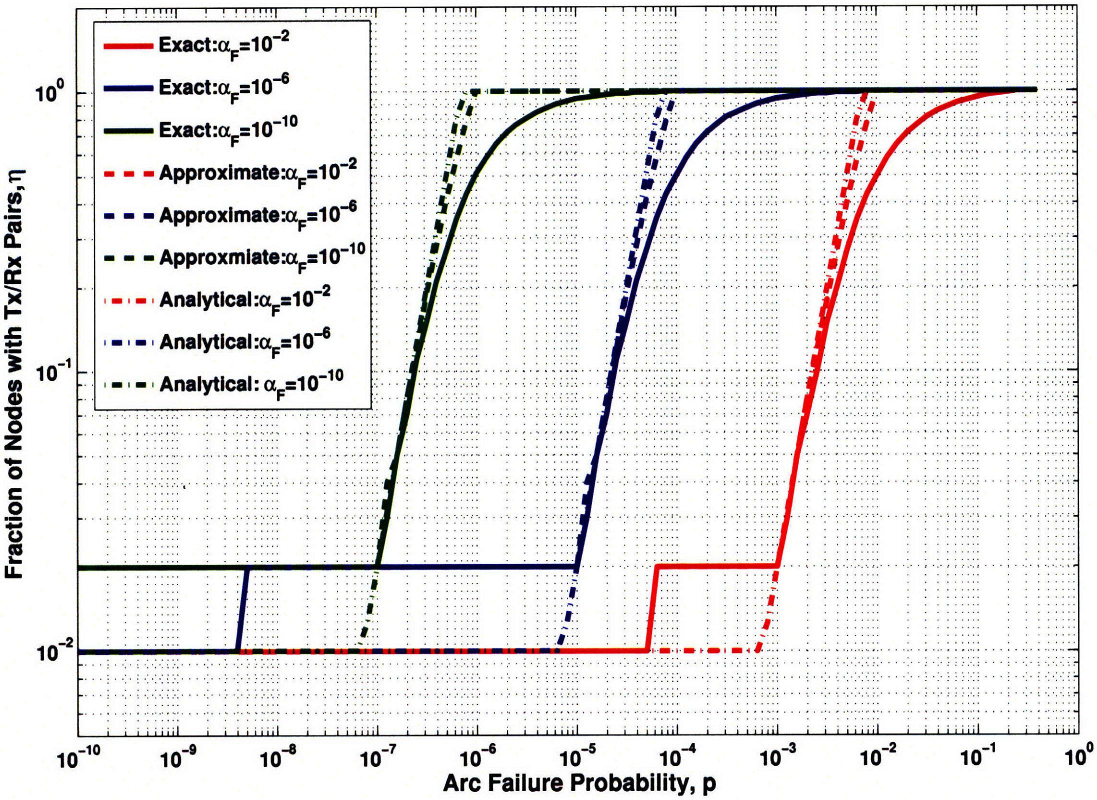


Figure 4-7: The required fraction of nodes with diagnostic Tx/Rx pairs for different target cumulative diagnosability probabilities is plot against the arc failure probability. They share similar "S" shapes. The exact results are compared with the approximate results and the analytical results. The number of nodes in the ring network is 100.

failure. The third term corresponds to the subset of network states with two or more arc failures. The significance of these terms depends on the arc failure probability.

In one extreme, when the arc failure probability is small, the cumulative diagnosability probability is first dominated by the first term and then by the first two terms. In the former case, when the target cumulative diagnosability probability is less than $(1-p)^{2n}$, it is sufficient to diagnose the network state without any arc failure with only one Tx/Rx pair. In the latter case, when the target cumulative diagnosability probability is less than the sum of the first two terms, it is sufficient to diagnose the subset of network states containing zero or a single arc failure, achieved by two diagnostic Tx/Rx pairs. Therefore, there exist two thresholds as the arc failure probability increases, as shown in Fig. 4-7.

In the other extreme, when the arc failure probability is high, the probability mass of all network states is mostly contributed by networks states with multiple arc failures. In this case, almost all of the nodes have to be equipped with Tx/Rx pairs in order to identify the subset of network states with multiple arc failures.

Between these two extreme cases, for a target cumulative diagnosability probability, the required fraction of nodes equipped with Tx/Rx pairs increases as the arc failure probability increases. In this regime, we hypothesize that, the cumulative diagnosability probability in each decomposed canonical linear network is dominated by the subset of network states with zero and a single arc failure. To verify this hypothesis, we can approximate the cumulative diagnosability probability in (4.8) as,

$$\beta_D(\eta) \approx \{(1-p)^{2\eta^{-1}} + 2\eta^{-1}p(1-p)^{2\eta^{-1}-1}\}^{n\eta}, \tag{4.10}$$

where the contribution from the subset of identifiable network states with two arc failures in each decomposed linear network is suppressed. In Fig. 4-7, we also plot the fraction of nodes equipped with Tx/Rx pairs to provide a target cumulative diagnosability probability, obtained by exhaustively searching over (4.10). We observe that, the approximation is very close to the exact solution derived from (4.8), especially when the arc failure probability is small. With this approximation, the required fraction of nodes equipped with Tx/Rx pairs can be derived by solving the following equation,

$$\{(1-p)^{2\eta^{-1}} + 2\eta^{-1}p(1-p)^{2\eta^{-1}-1}\}^{n\eta} = 1-\alpha_F, \tag{4.11}$$

for a tolerable cumulative undiagnosability probability of α_F. Taking ln on both sides of (4.11), we obtain the following equation,

$$n\eta \ln\{(1-p)^{2\eta^{-1}} + 2\eta^{-1}p(1-p)^{2\eta^{-1}-1}\} = \ln(1-\alpha_F). \tag{4.12}$$

Using a Taylor expansion for the left hand side of (4.12), we obtain the left hand side of (4.12) as

$$\begin{aligned} LHS &= n\eta(2\eta^{-1}-1)\ln(1-p) + n\eta\ln(1-p+\frac{2}{\eta}p) \\ &= n(2-\eta)\ln(1-p) - n\eta\ln(1+(\frac{2}{\eta}-1)p) \\ &\approx -n(2-\eta)p + n(2-\eta)p - n\eta\cdot\frac{p^2}{2}(\frac{2}{\eta}-1)^2 \\ &\approx -n\eta\cdot\frac{p^2}{2}\cdot\frac{4}{\eta^2} = -\frac{2np^2}{\eta}, \end{aligned} \tag{4.13}$$

where the first approximation is due to Taylor expansion (i.e., $\ln(1+x) = x - \frac{x^2}{2} + \mathbf{O}(x^3)$), and the second approximation is due to $2\eta^{-1}-1 \approx 2\eta^{-1}$. Similarly, using a Taylor expansion for the right hand side of (4.11), we obtain

$$RHS = \ln(1-\alpha_F) \approx -\alpha_F. \tag{4.14}$$

Substituting (4.13)and (4.14) into (4.12), we obtain the following equation

$$\frac{2np^2}{\eta} \approx \alpha_F. \tag{4.15}$$

Solving (4.15), we can approximate the required fraction of nodes equipped with diagnostic Tx/Rx pairs as

$$\eta^*(\alpha_F) \approx \frac{2np^2}{\alpha_F} \tag{4.16}$$

for small α_F and $1/n \leq 2np^2/\alpha_F \leq 1$. In Fig. 4-7, we also plot the required fraction of nodes equipped with diagnostic Tx/Rx pairs to provide a tolerable cumulative undiagnosability probability, based on the analytical result in (4.16). Notice that the analytical result matches the exact solution from (4.8) closely, especially when the arc failure probability is small.

4.3 Efficient Tx/Rx Deployment for Mesh Networks

In this section, we address the problem of efficient Tx/Rx deployment for mesh networks via two alternative approaches. One approach progressively identifies all the network states with up to κ link failures. The other approach extends our results for ring networks to Eulerian graphs.

4.3.1 Cutset-based Approach

For a given mesh network of n nodes and m links, we can order all the network states, based on the number of link failures contained in the network state, into a sequence of disjoint subsets, $S_0, S_1, \ldots, S_m$, where S_i denotes the set of network states containing i link failures. The probability of all network states with i link failures is

$$P_i = \binom{m}{i} p^i (1-p)^{m-i}, \tag{4.17}$$

for $i = 0, 1, \ldots, m$.

For a target cumulative diagnosability probability β_D, starting from the set S_0, we can progressively identify all the network states with up to κ link failures by deploying more diagnostic Tx/Rx pairs, so that the probability of all the identifiable sets of network states is larger than the target diagnosability probability, i.e.,

$$\sum_{i=1}^{\kappa} P_i \geq \beta_D. \tag{4.18}$$

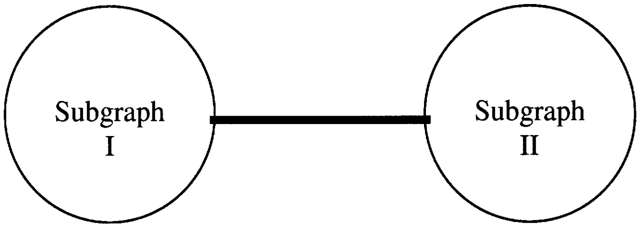


Figure 4-8: One Tx/Rx pair has to be deployed in each of non-trivial subgraphs, resulted from any cutset.

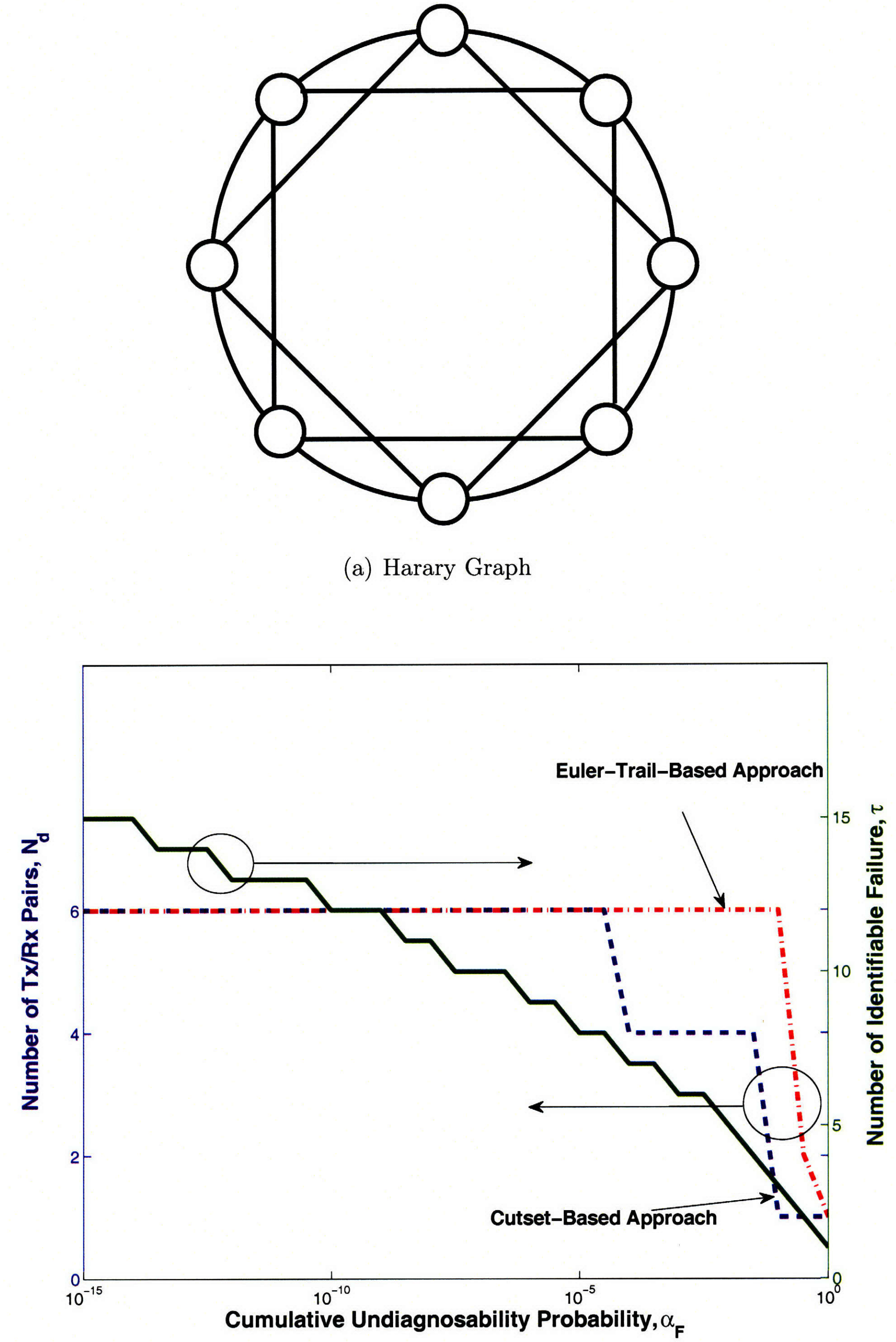
Solving the inequality (4.18), we obtain κ^*, which is the largest number of link failures that we need to identify[7].

The number of Tx/Rx pairs required can be determined by the following cutset approach[8]. In order to identify the network state set S_i (i.e., all the network states with i link failures), we need to deploy one Tx/Rx pair in each nontrivial subgraph (i.e., containing at least one link), resulted from any edge cutset of order i . As shown in Fig. 4-8, each non-trivial subgraph resulted from any cutset has to be equipped with a Tx/Rx pair. Otherwise, we cannot identify the state of all the links in the cutset that connect these two subgraphs. It follows that the efficient Tx/Rx deployment problem can be translated into the following combinatorial problem: *for an integer number κ, what is the minimum set of nodes in a graph such that a node from the minimum set exists in each nontrivial subgraph resulted from any cutset with an order up to κ ?* We call this problem the **efficient Tx/Rx deployment problem**.

As an example, we consider a Harary graph with 8 nodes and node degree $d = 4$ as illustrated in Fig. 4-9(a). We plot the required number of nodes equipped with diagnostic Tx/Rx pair, n_d , and the number of identifiable link failures, κ, as a function of the cumulative undiagnosability probability α_F in Fig. 4-9(b). We can see that the required number of nodes equipped with diagnostic Tx/Rx pairs decreases

[7] However, we have not yet found an approach to solve the inequality (4.18) analytically. Here, we assume that the inequality is solved numerically.

[8] A cutset of any graph is the set of links, whose removal results in a disconnected graph (i.e., a collection of connected subgraphs).



(a) Harary Graph

(b) Trade-Off

Figure 4-9: (a)Harary graph with 8 nodes and 16 links. (b) The number of identifiable links failures and the required number of nodes equipped with diagnostic Tx/Rx pairs as a function of cumulative undiagnosability probability.

as the cumulative undiagnosability probability increases. However, we have not yet derived any analytical results for generalized mesh networks.

The challenge here comes from two sources. First, we have not been able to solve the inequality (4.18) analytically to obtain κ^*. Second, for a generalized mesh network, the efficient Tx/Rx deployment problem is a NP-hard problem in the worst case[9]. Therefore, we will seek an alternative approach, based on ring network results, in the next sub-section.

4.3.2 Euler-Trail-based Approach

The analysis for ring networks can be extended to derive performance bounds for network topologies with an embedded ring structure, such as Eulerian graphs (an Eulerian graph contains a path that passes through all the links without repetition.) Non-Eulerian graphs can be approximated well with Eulerian graphs by a path augmentation approach [70].

In particular, as illustrated in Fig. 4-10, all the links in an Eulerian graph can be re-arranged into a ring network by replicating each node with $d/2$ virtual nodes, where d is the node degree. Under such a network transformation, our analysis for ring networks can be applied directly. However, the transformation suppresses a rich set of possible probing paths in the original network. It follows that the derived cumulative diagnosability probability is a lower bound, i.e.,

$$\beta_D(\eta) \geq \{(1-p)^{\eta^{-1}} + \eta^{-1}p(1-p)^{\eta^{-1}-1}\}^{\frac{1}{2}nd\eta}, \tag{4.19}$$

for the transition phase. Due to the bidirectional graph model used here, (4.19) is different from (4.10). This result, in turn, suggests that the resulting fraction of nodes equipped with Tx/Rx pairs for any target cumulative diagnosability probability is an upper bound on the required fraction of nodes equipped with diagnostic Tx/Rx pairs. Specifically, if the tolerable cumulative undiagnosability probability is α_F, we have

[9] When $\kappa \geq 4$, this problem can be converted into the vertex cover problem in graphs with maximum vertex degree of 3, since any edge can be turned into a connected component by deleting all other edges adjacent to its endpoints.

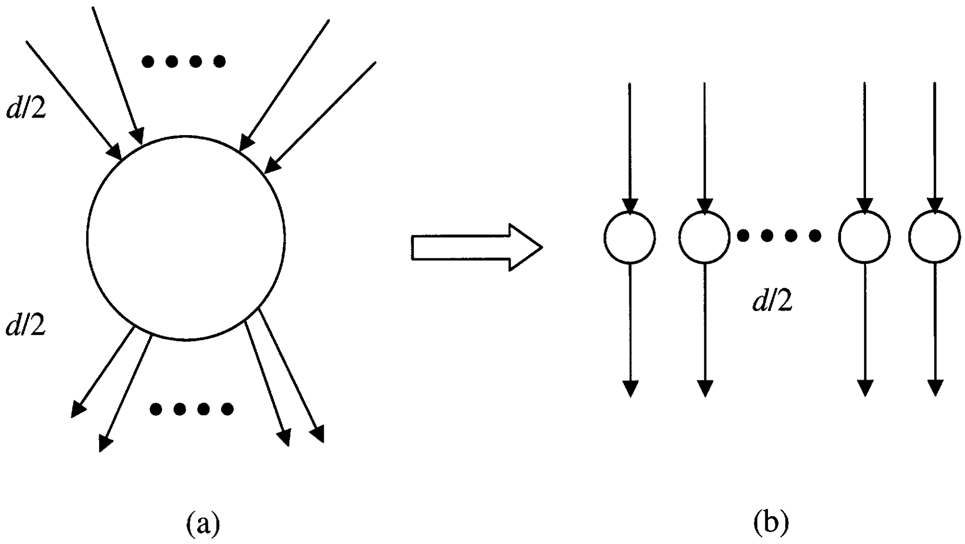


Figure 4-10: Node replication approach: (a) a node of degree d has $d/2$ in degree and $d/2$ out degree, (b) the node is replicated with $d/2$ virtual nodes, each of which has 1 in degree and 1 out degree.

the following inequality,

$$\{(1-p)^{\eta^{-1}} + \eta^{-1}p(1-p)^{\eta^{-1}-1}\}^{\frac{1}{2}nd\eta} = 1 - \alpha_F. \tag{4.20}$$

We can take ln on both sides of (4.20), and obtain the following simplified inequality,

$$\frac{1}{2}nd\eta \ln\{(1-p)^{\eta^{-1}} + \eta^{-1}p(1-p)^{\eta^{-1}-1}\} = \ln(1 - \alpha_F). \tag{4.21}$$

Using a Taylor expansion for the left hand side of (4.21), we obtain the left hand side of (4.21) as

$$\begin{aligned} LHS &= \frac{1}{2}nd\eta(\eta^{-1}-1)\ln(1-p) + \frac{1}{2}nd\eta\ln(1-p+\frac{1}{\eta}p) \\ &= \frac{1}{2}nd(1-\eta)\ln(1-p) + \frac{1}{2}nd\eta\ln(1+(\frac{1}{\eta}-1)p) \\ &\approx -\frac{1}{2}nd(1-\eta)p + \frac{1}{2}nd(1-\eta)p - \frac{1}{2}nd\eta\cdot\frac{p^2}{2}(\frac{1}{\eta}-1)^2 \\ &\approx -\frac{1}{2}nd\eta\cdot\frac{p^2}{2}\cdot\frac{1}{\eta^2} = -\frac{4ndp^2}{\eta}, \end{aligned} \tag{4.22}$$

where the first approximation is due to Taylor expansion, and the second approximation is due to $\eta^{-1} - 1 \approx \eta^{-1}$. Similarly, using a Taylor expansion for the right hand side of (4.21), we obtain

$$RHS = \ln(1 - \alpha_F) \approx \alpha_F. \tag{4.23}$$

Substituting (4.22)and (4.23) into (4.21), we obtain the following equation

$$-\frac{4ndp^2}{\eta} \approx -\alpha_F. \tag{4.24}$$

Solving (4.24), we obtain that the required fraction of nodes equipped with diagnostic Tx/Rx pairs is upper bounded by,

$$\eta^*(\alpha_F) \leq \frac{ndp^2}{4\alpha_F} \tag{4.25}$$

where α_F is the tolerable cumulative undiagnosability probability. Notice that the required fraction of nodes equipped with diagnostic Tx/Rx pairs decreases as the cumulative diagnosability probability increases. The result of (4.25) is plot in Fig. 4-9(b) for the Harary graph of 8 nodes and node degree 4. Notice that the number of nodes equipped with Tx/Rx pairs is larger than the result from the cutset approach, because rich connection in mesh networks is not exploited in the Euler-Trail-based approach.

The tightness of these performance bounds depends on both the arc failure probability and the node degree. When the arc failure probability is small and/or the node degree is small, these bounds are expected to be tight. In particular, when the arc failure probability is small, the cumulative diagnosability probability in each decomposed network is dominated by network states with zero and a single arc failure. When the node degree is small, the benefit of additional node degree is not significant enough to change the order of magnitude. However, when the node degree is large, these bounds could be loose. In this case, the rich set of connections in the mesh network of degree larger than 2 should be explored to identify failure patterns with multiple arc failures, and thus reduce the number of nodes equipped with Tx/Rx pairs.

4.4 Conclusion

In this chapter, we built upon our previous research on proactive lightpath probing schemes to investigate the cost-effective Tx/Rx deployment for probe transmission and detection in all-optical networks. We developed a probabilistic framework to characterize the trade-off between the number of nodes equipped with diagnostic Tx/Rx pairs and the cumulative diagnosability probability. Our investigation suggested that the diagnostic hardware cost can be reduced significantly by accepting a reasonable amount of uncertainty about network failure status.

For future research, we would like to extend this analysis to all-optical networks with other mesh topologies. Other possible future work is to develop optimal Tx/Rx deployment schemes to maximize the cumulative diagnosability probability for a limited number of diagnostic Tx/Rx pairs.

Chapter 5

Fault Diagnosis Made Practical

This chapter addresses practical issues of fault diagnosis for all-optical networks. First, we will investigate the fault detection and localization problem for optical wavelength-division multiplexing (WDM) networks with multiple wavelength channels. This investigation suggests that the deployment of this type of proactive fault diagnosis schemes will depend on the type of failures (i.e., fiber-level failure vs wavelength-level failure). Second, we will focus on fault diagnosis for dynamic all-optical network with existing traffic by classifying all lightpath services into three categories and developing fault-diagnosis scheme for each category.

5.1 Fault Diagnosis for All-Optical WDM Networks with Multiple Wavelength Channels

In this section, we will extend run-length probing schemes to fault detection and localization for all-optical wavelength-division multiplexing (WDM) networks with multiple wavelength channels. For WDM networks with multiple wavelength channels, network failures can be roughly classified into two categories: fiber-level failures and wavelength-level failures. The relevant importance of these two categories of network failures dictates how the fault diagnosis scheme should be deployed.

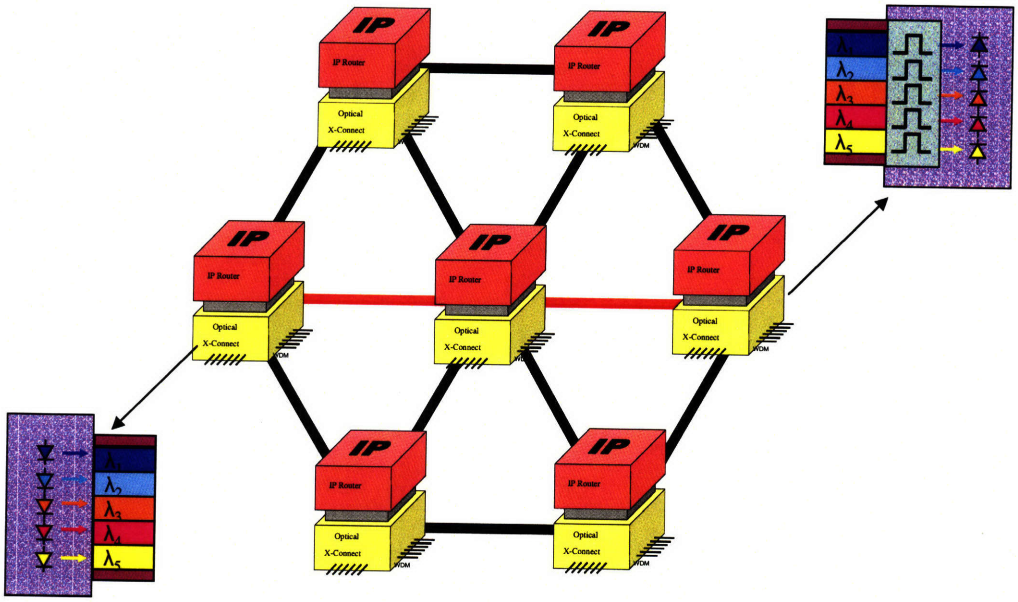


Figure 5-1: An illustration of all-optical WDM Network with multiple wavelength channels: optical lightpath traverses from the source to the destination without being processed electronically at intermediate nodes.

5.1.1 Two Types of Failures: Fiber-Level vs. Wavelength-Level

Fig. 5-1 illustrates an all-optical WDM network, where each lightpath traverses across the network without being processed electronically at any intermediate nodes. Without loss of generality, it is assumed that there are W wavelengths per fiber in an all-optical WDM network. In the optical layer, the network is subjected to different kinds of failures. According to the scale of their effect, these failures can be classified into two categories. One category is the wavelength-level failure, which affects a particular wavelength channel. For example, transmitter/receiver failures with one dedicated transmitter/receiver per wavelength and single-bandwidth optical filter, or single-channel frequency-selective switch failures, belong to this category. The other category is the fiber-level failure, which affects all the wavelength channels within an individual fiber, such as fiber cuts, EDFA breakdowns and transmitter/receiver failures in the case of only one tunable transmitter/receiver per fiber (which rarely happens).

Notice that, although that the wavelength-level failure and the fiber-level failure are statistically independent, all the wavelength channels passing through an EDFA fail simultaneously when the EDFA fails. This suggests that failures in different wavelength channels on the same fiber are dependent in that knowing one particular wavelength channel fails reveals some information about the failures of other wavelength channels. Therefore, the fault diagnosis algorithm for practical all-optical WDM networks must consider inter-dependence among failures in different wavelength channels.

5.1.2 General Approach

As mentioned previously, the application of the run-length probing scheme over practical all-optical networks depends on the relative dominance between wavelength-level failures and fiber-level failures. In other words, the relationship between p_F (i.e., the prior probability of individual fiber-level failure) and p_W(i.e., the prior probability of individual wavelength-level failure) determines how the run-length probing scheme should be implemented over practical all-optical WDM networks.

In one extreme, for an all-optical WDM network where the wavelength-level failure dominates the fiber-level failure (i.e., $p_W \gg p_F$), we can view the network as a graph where each physical link is represented with W parallel edges. On one hand, if wavelength converters are deployed in the network, we can apply the run-length probing scheme over the hyper-graph and the average number of probes required is approximated by the entropy lower bound, i.e, $W \cdot m \cdot H_b(p_W)$. On the other hand, if no wavelength conversion is allowed at any node, this hyper-graph can be decoupled into W graphs, each of which is exactly the same as the original network topology and represents one particular wavelength plane. Upon this separation of different wavelength planes, we can employ the run-length probing scheme over each wavelength plane graph. This is called the wavelength-level implementation. For a reasonable large network (i.e., $m \gg K$ where K is the maximal probe length determined by the wavelength failure probability), the average number of probes required by the run-length probing scheme can be approximated by $W \cdot m \cdot H_b(p_W)$.

In the other extreme, for an all-optical WDM network where the fiber-level failures dominate the wavelength-level failures (i.e., $p_W \ll p_F$), we can view the network as a graph where each physical link is represented with one edge and employ the run-length probing scheme over an Euler trail of the resulted graph. This is called the fiber-level implementation. For a large network (i.e., $m \gg K$ where K is the maximal probe length determined by the fiber failure probability), the average number of probes required is approximately equal to $m \cdot H_b(p_W)$.

Between these two extreme cases, for an all-optical WDM network subjected to a comparable (in terms of probability of occurrence) mixture of both fiber-level failures and wavelength-level failures, we can still use the fault-diagnosis/source-coding equivalence to obtain a useful lower bound for the minimum average number of probes required as the information entropy of network states, i.e,

$$\mathcal{L}^* \geq m \cdot H_b(F_1, F_2, \ldots, F_W), \tag{5.1}$$

where F_i's are dependent random variables indicating states of wavelength channel i's , and $H_b(\cdot)$ is the information entropy function. The entropy function $H_b(F_1, F_2, \dots, F_W)$ can be calculated through the summation of a sequence of conditional entropy functions, i.e.,

$$H_b(F_1, F_2, \dots, F_W) = \sum_{i=1}^{W} H_b(F_i|F_1, \dots, F_{i-1}). \tag{5.2}$$

However, It is not yet clear whether the entropy lower bound (5.1) can be achieved, or if achievable, how we can develop probing schemes to achieve this lower bound.

As a brute-force approach, we can view the network as a graph where each physical link is represented with parallel edges and employ the run-length probing scheme over an Euler trail of the resulted graph. The performance of this wavelength-level implementation, which is hard to obtain due to the complicated dependence among failures in different wavelength channels, can certainly serve as an upper for the minimum average number of probes required by an optimal probing scheme. However, since the wavelength-level implementation does not consider the dependencies among failures in different wavelength channels, the run-length probing scheme is not optimum in general. The same conclusion that the information entropy is a lower bound and the run-length probing scheme might not be near-optimum, can be extended to a more general failure model which accommodates dependent failures and/or heterogeneous failures. In the next subsection, we propose three alternative approaches to solve the fault-diagnosis problem for all-optical WDM networks with a comparable mixture of link-level and wavelength-level failures.

5.1.3 Three Proposed Approaches

To deal with the mixed failure case (i.e., the link-level failure probability and the wavelength-level failure probability are of the same order), we propose the following three approaches. These approaches are expected to perform better than the brute-force approach (i.e., the wavelength-level implementation).

The first approach is similar to the brute-force approach, except that the link failure probability of unprobed wavelength planes is updated once any wavelength plane has been finished with probes. The scheme works as follows. First, the network is decoupled into W wavelength planes, whose failure probabilities are denoted as $p_{i,l}^{(0)} = p_F + p_W - p_F p_W$, for $i = 1, 2, \dots, W$ and $l = 1, 2, \dots, m$, where (0) denotes the initial failure probability. Second, the fault-diagnosis scheme randomly starts with one wavelength plane, whose link failure probability can be calculated as $p_{1,l}^{(0)} = p_F + p_W - p_F p_W$. In this case, the independence among all link failures is still valid. Once the diagnosis for this wavelength plane is finished, the link failure probability on all the unfinished wavelength planes is updated with the conditional failure probability, i.e., $p_{i,l}^{(1)}$, where the superscript (1) indicates that one wavelength plane has been diagnosed and it will be increased by 1 after each wavelength plane has been diagnosed. Specifically, for any link whose first wavelength channel is of no failure, we have $p_{i,l}^{(1)} = p_W$. For any link whose first wavelength channel has failed, we have $p_{i,l}^{(1)} = p_W + (1 - p_W)\frac{p_F}{p_F + p_W}$ The challenge here is that the updated failure probability for any given wavelength plane is inhomogeneous among all the edges[1]. The same process continues until all the wavelength planes are probed.

The second approach declares a fiber-level failure once the number of wavelength-level failures identified in that fiber passes some threshold. The threshold is chosen to strike a balance between the saving of additional probes and the penalty of false declaration of fiber-level failures. The trade-off can be understood through the following example of a single fiber with W wavelength channels and one optical amplifier(OA). The OA fails with probability of p. Once OA fails, all the wavelength channels fails. In addition, each wavelength channel fails independently with probability q (mostly from transmitter/receiver, filter failures). All the wavelength channels are probed sequentially. If any wavelength channel is good, one declares that the OA is good and continue to probe the rest of wavelength channels. On the other hand, if the first k wavelength channels are found bad, one declares that the OA fails and stops the probe process. The diagnosis cost includes two parts: the probing cost and the false-alarm penalty. Let us assume that each probe costs c_p, and if a good OA is

[1] In this case, we can follow the proven guideline for efficient fault diagnosis: each probe should provide approximately one bit of state information.

declared as bad, it incurs a penalty cost of c_f. Under such assumptions, the objective is

$$C(k) = kc_p + c_f \Pr(\text{OA is good } |\text{first } k \text{ wavelength channels fail}), \tag{5.3}$$

where the false-alarm probability can be derived as

$$\Pr(\text{OA is good } |\text{first } k \text{ wavelength channels fail}) = \frac{(1-p)q^k}{(1-p)q^k + p}. \tag{5.4}$$

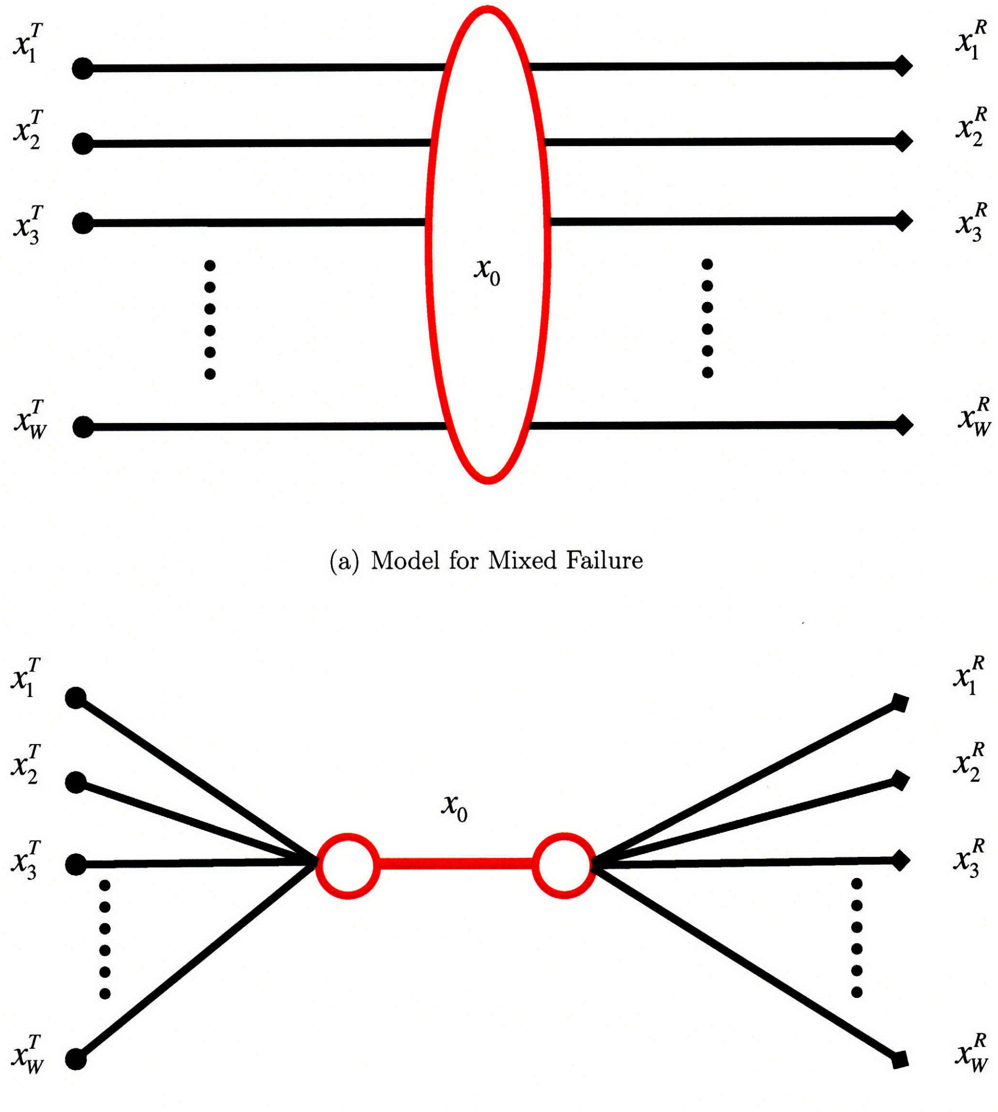
Therefore, the optimal threshold can be identified by solving the following optimization problem,

$$\begin{aligned} \min_k \quad & C(k) = c_p k + c_f \frac{(1-p)q^k}{(1-p)q^k + p}, & (5.5)\\ s.t. \quad & 1 \leq k \leq W. & (5.6) \end{aligned}$$

Let us first look at two extreme cases. On one extreme, when $c_p \gg c_f$, we have $k^* = 1$. On the other extreme, when $c_p \ll c_f$, we have $k^* = W$. Between these two extreme cases, we can approximate the optimal k^* as

$$k^* \approx \frac{\ln\left(\frac{c_p}{c_f}\frac{1-p}{p}\frac{1}{\ln q^{-1}}\right)}{\ln q^{-1}}. \tag{5.7}$$

The third approach is based on a network transformation that separates the fiber-level failure and the wavelength-level failures. A graphical model for the mixed failure case is illustrated in Fig. 5-2(a), where x_0 indicated the fiber-level failure, and the set $\{(x_i^T, x_i^R) : i = 1, 2, \ldots, W\}$ indicate the wavelength-level failures. All of these indicators are random variables with binary values. When each wavelength channel is probed, the syndrome is an OR function of x_i^T, x_0 and x_i^R. The difficulty here is that the indicator x_0 involves all the wavelength channels and thus wavelength channel failures are dependent. Here, we propose a graphical transformation to decouple these dependencies, as illustrated in 5-2(b). In the transformation, the fiber-level failure is represented by an additional bridge link between the two set of wavelength-level failure indicators. However, it also comes with some penalty: no diagnosis Tx/Rx can



(a) Model for Mixed Failure

(b) Transformation for Mixed Failure

Figure 5-2: (a)Graphical model for mixed failure (b)Graphical transformation for mixed failure.

be deployed at the two end points of the fiber-level link. This changes the model of our original assumption. Under the new model, we have to solve the fault diagnosis problem with the assumption that only a limited set of nodes can be equipped with diagnosis Tx/Rx. We have addressed a similar problem in Chapter 4.

5.2 Fault Diagnosis for All-Optical Networks with Existing Traffic

In this section, we investigate the fault-diagnosis problem for all-optical networks with existing traffic. In particular, we take a "divide-and-conquer" strategy by classifying lightpaths into three classes and developing fault-diagnosis schemes for each class of lightpaths separately.

5.2.1 Lightpath Service Requirements

In future dynamic all-optical networks, we believe that the lightpath service will have different requirements in restoration and setup.

First, existing lightpath services present stringent restoration requirements in terms of time deadlines. This suggests that primary lightpaths should be monitored constantly, and secondary lightpaths should be monitored and be ready as soon as primary lightpaths are interrupted.

Second, dynamic lightpath services exhibit a wide range of set-up requirements in terms of blocking probability and time deadline. This suggests that lightpath demands should be met via a combination of off-line and online routing and wavelength assignment (RWA) algorithms. Hence, fault-diagnosis algorithms should be designed to cater for both cases.

5.2.2 Breakdown of Lightpath Services

For dynamic all-optical networks, we can classify the lightpath services into three categories:

1. **Class 1**: currently lit lightpaths
2. **Class 2**: pre-computed but unlit lightpaths, including i) backup lightpaths of currently lit ones and ii) unlit pre-computed lightpaths with stringent set-up time requirements[2]
3. **Class 3**: unknown lightpaths computed by online RWA algorithms with relaxed set-up time requirement

To meet aforementioned lightpath service requirements, difference fault diagnosis strategies have to be developed for different classes of lightpath service, as illustrated in next subsection.

5.2.3 Fault Diagnosis for Different Classes of Lightpath Services

For Class 1 lightpaths, a lit lightpath is under constant monitoring at the receiver node. Once it fails, the destination node would initiate, through the network management system, an adaptive fault diagnosis process to identify the failure as soon as possible. Two alternative implementations of the run-length probing scheme can be deployed to identify the failure along each failed lightpath. In the case that intermediate nodes does not have self-monitoring capability via tapping-out, the network management system could signal the source node to initiate lightpath probes to different intermediate nodes sequentially, according to the run-length probing scheme (i.e., the 2^{α}-splitting probing scheme), and the faulty link(s) can be identified based on the probe syndrome. Alternatively, if intermediate nodes have self-monitoring capability via tapping out, either the network management system can poll the network element management module for management information, or the network element management module can send alarms to the network management system. In the former case, the poll sequence is determined by the run-length probing scheme, with the objective of minimizing the number of polls. In the latter case, the run-length

[2] Two other students in Prof. Vincent Chan's group, Bishwaroop Ganguly and Anarupa Ganguly, are doing research on probing and resource reservation along this class of lightpaths.

probing scheme suggests an efficient way to process the large amount of alarm information by sequentially checking whether some specific alarms are present, with the objective to minimize the amount of network processing unit (NPU) facility.

For Class 2 lightpaths, to meet the stringent time requirements for fast restoration and very fast service setup, the network management system aggressively monitors the states of the pre-computed but unlit lightpaths in a periodic schedule. The schedule period is determined by the lightpath restoration and set-up requirement, could be as short as 100mS.

We suggest two alternative monitoring procedures for Class 2 lightpaths. In the first approach, the set of links covered by all unlit lightpaths forms a subgraph of the original network, on which our previously established fault-diagnosis schemes will be deployed periodically. Intuitively, this approach is preferable when the number of pre-computed but unlit lightpaths is large so that the overlapping among different lightpaths is large enough for the run-length probing scheme to save the number of probes. In the second approach, each lightpath is monitored individually[3]. Once some lightpath has problem, the run-length probing scheme is applied to that specific lightpath to identify the failure. Intuitively, this approach is preferable for the case when the number of pre-computed but unlit lightpaths is relatively small so that the run-length probing scheme over the expanded sub-graph is not efficient. It is an interesting future work to characterize the optimality conditions for both approaches.

For Class 3 lightpaths that are not computed until the connection request has arrived, the network management will conduct fault diagnosis over unassigned wavelength channels on a coarse time scale. However, the existence of lit lightpaths poses some technical challenges for such network-wise fault diagnosis. In particular, existing lightpaths could render some network elements inaccessible for diagnosis. For example, as illustrated in Fig. 5-3, for a network node equipped with an OXC, if a lightpath connects input port i and output port j, the non-blocking requirement precludes any probing traffic accessing the set of connections originating at node i

[3] Supervised by Prof. Vincent Chan, Anarupa Ganguly is working on this periodic probing scheme for pre-computed but unlit lightpaths.

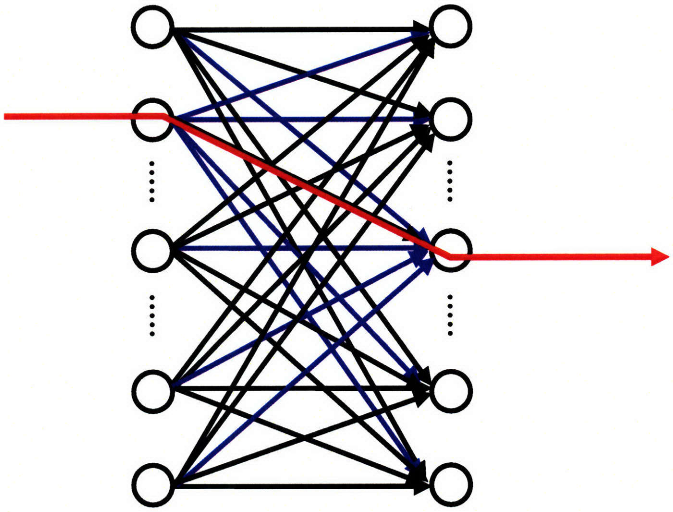


Figure 5-3: The existing lightpaths could render some network elements inaccessible for diagnosis.

or terminating at node j. For such network elements, we will develop algorithms to estimate/predict their current states based on their previous states and information revealed by previous probes, or by scheduling probes on them when their associated lit lightpaths have been torn down. As in the previous discussion, the remaining set of diagnosable network elements spans a sub-network, on which our previously established adaptive/non-adaptive fault diagnosis schemes would be periodically deployed to monitor their state of health.

Because of this limitation resulted from existing traffic, network components can be decoupled into three subsets: components that can be monitored by existing traffic (denoted as S_{et}), components that can be monitored by probing traffic (denoted as S_{pt}) and components that cannot be monitored at all (denoted as S_{uo}). For example, for a $d \times d$ optical switch, the total number of connections is d^2. If there is l ($l \leq d$) existing lightpaths through the switch, the number of connections that can be monitored by existing traffic is l, the number of connections that can be monitored by probing traffic is $(d-l)^2$ and the number of connections that cannot be observed is $d^2 - (d-l)^2 - l$. Moreover, for a regular network of n nodes and degree d, the total number of switch connections in the network is nd^2. If the average load of each switch is ρ, the number of lightpaths for existing traffic on each switch is ρd. In this case, the number of switch connections monitored by existing traffic is $n\rho d$, the number of switch connections that can be monitored by probing traffic is $n(d - \rho d)^2$, and the number of switch connections that cannot be monitored is $nd^2 - n(d - \rho d)^2 - n\rho d$. The total number of connections (denoted as $N_{monitored}$) that can be monitored either through existing traffic or through probing traffic is given by

$$N_{monitored} = n(d - \rho d)^2 + n\rho d. \tag{5.8}$$

It can be shown from (5.8) that $N_{monitored}$ is minimized when $\rho^* = 1 - \frac{1}{2d}$. Therefore, we can conclude that, as the average load increase, the number of observable connections decreases first and then increases again after it goes above $1 - \frac{1}{2d}$.

Chapter 6

Network Survivability: Lightpath Diversity

This chapter proposes using spatial diversity via multiple node-disjoint[1] lightpaths at the optical layer to achieve ultra-reliable communication between any source-destination pair in all-optical networks[2]. Compared to the automatic protection switching (APS) scheme for network survivability, the lightpath diversity scheme trades the abundance of bandwidth in optical fibers for ultra-reliable communication. Our design objective of lightpath diversity is to optimize the resource efficiency while providing the required reliability.

6.1 Introduction

When deployed, all-optical networks will trigger an architectural revolution for future broadband networks by eliminating all optical-to-electrical conversions along a lightpath [25, 12]. Originally proposed to exploit the huge bandwidth within the low attenuation transmission window of optical fibers to meet the exponential growth of traffic demand, optical networks have been evolved to provide other highly desirable features including wavelength switching, dynamic reconfigurability and improved re-

[1] Node-disjoint implies link-disjoint.

[2] The content in this chapter has been published in [66].

liability. These enhanced features can support highly reliable services that can transport, for instance, aircraft control signals between the cockpit and control surfaces over lightweight all-optical networks.

However, similar to other networks, all-optical networks are also vulnerable to different categories of failures. One kind of failure is physical component failure, for example, fiber cut and node hardware failure. Even if all network components are reliable individually, the communication between a source-destination pair can be interrupted by soft failures due to network problems, such as congestion, buffer overflow, and routing algorithm oscillations. In this chapter, we will focus on the problem of achieving ultra-high reliability in all-optical networks for some special applications that may have to support services with critical time deadlines.

To support ultra-reliable communication in all-optical networks, two mechanisms can be used to counteract the aforementioned failures: automatic protection switching and simultaneous lightpath-diversity. Currently, the prevailing approach is the protection switching scheme, as implemented commercially in Synchronous Optical Network (SONET) based networks. In this scheme, if a source-destination communication session is interrupted by a failure, a detection algorithm first identifies the failure, and then communication is switched to another dedicated or shared backup connection. However, this protection-switching mechanism can induce a rather long delay (e.g., $\sim$ 50-ms restoration time, a SONET standard [34]). Thus this scheme is inappropriate for some applications. For example, considering the service with super high data rate ($10Gbps$ or higher), a short-time interruption can result in a large amount of data loss. In other critical applications (e.g. when the network is used for transporting control signals between the cockpit and control surfaces of an aircraft), the time-deadline of control-message delivery needs to be shorter than 1-ms and probably ten times faster in failure detection. This is faster than the speed at which most optical components can switch and protection switching protocols can be executed. For such applications, instead of increasing the speed of failure detection and lightpath switching to meet increasing data rates and critical time deadlines, multiple-path diversity is a better alternative that can be implemented with current technologies. Chan and Parikh have explored this mechanism in [13, 45]. In that work they looked at a joint Data Link Control Layer and Transport Layer reliable message delivery scheme and have found significant merit for using path diversity efficiently via error correction coding techniques. In this work, their work will be extended to a Physical Layer lightpath diversity mechanism, using an optimum signaling and detection scheme to optimize system performance and provide reliable end-to-end data delivery in the presence of failures (e.g., fiber cuts and node hardware failures).

The advantages of the proposed reliable transmission scheme, which is based on spatial diversity via multiple disjoint lightpaths belonging to different shared-risk groups, are at least two-fold. First, because the entire mechanism is implemented at the Physical Layer, it provides a much faster response to failures than protocols that provide end-to-end reliability at higher layers, such as the Transmission Control Protocol (TCP) at the Transport Layer using feedback and time-outs. Second, as will be shown in this chapter, the symbol error probability of multiple-lightpath transmission is significantly lower than that of single-lightpath transmission in medium and high signal-to-noise photon rate ratio regimes. In particular, for a source-destination pair connected by M lightpaths, the symbol error probability in the high signal-to-noise photon rate ratio regime is asymptotically equal to $\prod_{i=1}^{M} f_i$ (f_i is the failure probability of the lightpath.) This is the asymptotic reliability limit of the multiple-lightpath transmission scheme. By choosing the number of lightpaths used, this limit can be made arbitrarily small compared to the asymptotic symbol error probability of using only a single lightpath between a source-destination pair.

Compared to the single lightpath transmission, one major disadvantage of the lightpath diversity scheme is that the same message is sent repeatedly through a group of disjointed lightpaths and thus degrades the throughput per channel use by a factor of M for an M-connected source-destination pair. However, in order to achieve ultra-reliable communication with low delay, for example, in an aircraft control network, it is reasonable to sacrifice some bandwidth efficiency for reliability in a bandwidth-rich environment(e.g., optical fibers). In fact, multiple connections between any source-destination pair are necessary for reliable networks [64], and both parallel signaling and sequential signaling over multiple connections can realize high reliability. The lightpath diversity scheme satisfies this necessary condition by splitting each channel symbol and sending the fragments simultaneously through M disjointed lightpaths. Another potential disadvantage of a lightpath diversity scheme is that more energy may have to be used than a single lightpath scheme. However, the error probability of any single lightpath scheme is bounded from below by the lightpath failure probability f. In order to achieve an error probability below f, it is necessary to use more than one lightpath to lower the asymptotic error limit. In this work, using optimum signaling and detection, the number of lightpaths will be chosen to optimize energy efficiency and reduce the amount of additional necessary optical energy to the minimum.

In this chapter, the proposed ultra-reliable transmission mechanism will be investigated from both a theoretical and an engineering perspective. From the theoretical perspective, we will characterize and optimize the error performance of the lightpath diversity system. From the engineering perspective, we will develop a class of structured receivers and evaluate their error performance. The remainder of this chapter is organized as follows. In Section 6.2, we will formulate the detection problem and introduce the structured receiver architecture. The error probability of the lightpath-diversity system will be characterized via an idealized receiver in Section 6.3. This benchmark result is called the 'genie-aided' receiver limit which is a lower bound for practical receivers. In Section 6.4, the system is optimized via (1) minimizing the error probability for a given amount of optical energy, and (2) minimizing the total optical energy for a target error probability. In Section 6.5, we will illustrate the trade-off between implementation complexity and error performance in the receiver design. Also in this section, the architecture of the optimal receiver is derived, and its error probability bound is obtained and compared with the 'genie-aided' receiver limit. One sub-optimal receiver, the equal-gain-combining receiver, is developed in Section 6.6. Its error probability bound is also calculated and compared with the 'genie-aided' receiver limit.

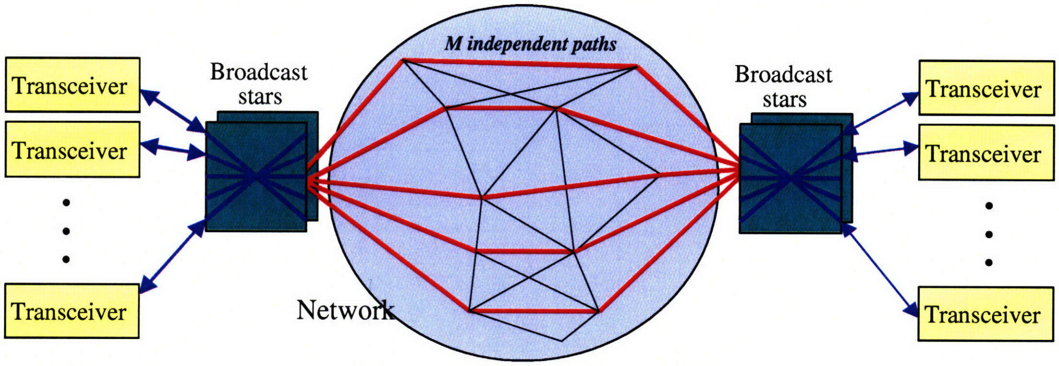


Figure 6-1: Network model for an M-connected source-destination pair in a densely connected all-optical networks [14].

6.2 Lightpath Diversity Problem

6.2.1 Network Model

It is assumed that the physical topology of the optical network under consideration has dense enough connections such that M node-disjointed light-paths can be found between some source-destination pair, as shown in Fig. 6-1 [64, 65]. All the lightpaths must belong to different shared-risk groups to justify the following independent failure model. Each lightpath can be modeled as a discrete additive-noise channel with UP and DOWN states. In particular, for the i^{th} lightpath, the DOWN state corresponds to a disconnected lightpath and occurs with probability f_i , and the UP state occurs with probability $1-f_i$ and corresponds to a viable lightpath. Mathematically, the input-output relation of the channel can be expressed as $Y_i = F_iX_i + N_i$, as shown in Fig. 6-2, where X_i and Y_i are the input and output, F_i is the lightpath state indicator function which is a Bernoulli random variable with $\Pr\{F_i = 0\} = f_i$ and $\Pr\{F_i = 1\} = 1 - f_i$, and N_i is the additive noise (zero if no optical amplifier is used). For a given source-destination pair, one can define a lightpath state vector $\mathbf{F} = (F_1, F_2, \ldots, F_M)^T$, where the components F_i's are independent Bernoulli random variables. The source-destination pair is also characterized by a delay vector $\tau = (\tau_1, \tau_2, \ldots, \tau_M)^T$ where each component τ_i is the delay of the i^{th} lightpath, an

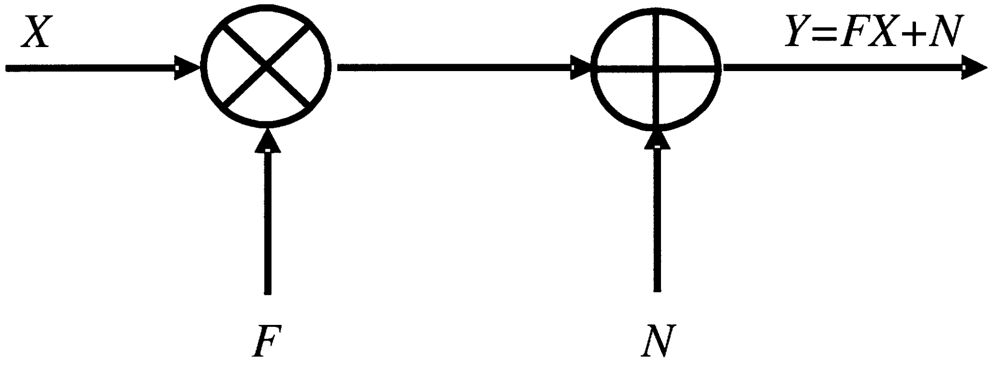


Figure 6-2: Discrete channel model of an individual lightpath. X is the input, Y is the output, F is a Bernoulli random variable indicating the lightpath state, and N is the noise.

attenuation vector $\mathbf{L} = (l_1, l_2, \ldots, l_M)^T$ where each component l_i is the attenuation of the i^{th} lightpath, and a noise vector $\mathbf{N} = (N_1, N_2, \ldots, N_M)^T$ where each component N_i is the noise of the i^{th} lightpath (including additive detection noise if present).

In this chapter, Binary Pulse-Position Modulation (BPPM) is used to simplify the receiver implementation by not having to adaptively set the decision threshold as in the case of On-Off-Keying (OOK) modulation. The modulated signal is split into M parts. Each part is sent over an independent lightpath to the receiver. At the destination node, the received optical signals are either combined optically before detection, or individually detected and electrically combined for symbol-by-symbol decisions. With a photon-counting receiver, the photo-event count at the receiver's output obeys Poisson statistics [4] if the optical signal is generated by a single-mode laser. The expected photo-event arrival rate λ (the mean number of photo-event per unit time) is determined by the received optical power (i.e., energy per bit, given the bit rate). In this work, the received optical power is a random variable due to the random channel model as illustrated in Fig. 6-2. Thus the photo-event process at the detector output can be modeled by a Doubly-Stochastic Point Process [56].

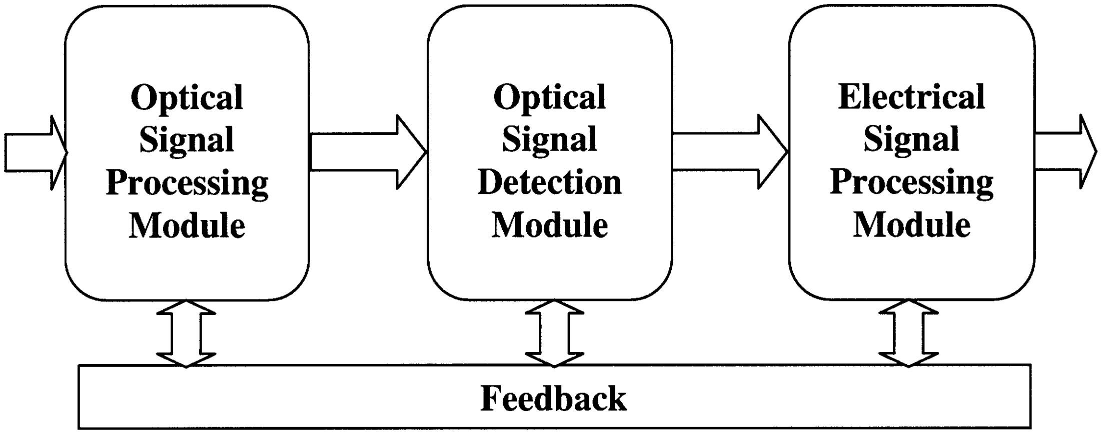

Optical Signal Processing Module
Optical Signal Detection Module
Electrical Signal Processing Module
Feedback

Figure 6-3: Structure receiver architecture. The receiver is divided into three cascaded modules: an optical signal processing module, an optical detection module and an electrical processing module.

6.2.2 Structured Receiver Architecture

One can design an optical receiver using two different approaches. Due to the quantum nature of weak optical signals, one approach is to use a full quantum description of the receiver, and optimize it over the class of physically realizable measurements [31]. Quantum receivers are optimum in energy efficiency. However, they are complicated and hard to realize with current electrical and optical components. In this thesis, a "structured" or "semi-classical" approach [35] is pursued. Although structured receivers without feedback can suffer a 3-dB loss of energy efficiency over optimum receivers for binary signaling, they are much simpler and easier to implement with current technologies. The architecture of all possible structured receivers can be divided into three cascaded processing modules, as illustrated in Fig. 6-3: an optical signal processing module, an optical detection module, and an electrical signal processing module. The three modules must be jointly optimized to achieve a globally optimum performance. Causal feedbacks among these blocks are also permissible, which can make structured receivers achieve the quantum limit for binary signaling [53]. However, due to their complexity, we will not consider them here.

6.3 System Characterizations

In this section, the symbol error probability of the lightpath-diversity scheme will be characterized with an exponentially tight upper bound. It is assumed that, at the destination node, an idealized receiver obtains the lightpath state vector $\mathbb{F}$ from a "genie" (i.e., the receiver has information of the channel states.) At the optical signal processing module, optical delay lines are used to compensate for delay variations among different lightpaths (fiber delays can also be replaced by time delays in the electrical signal processing stage if M parallel detectors are used.) At the optical detection module, the photo-events at the output of the M detectors are recorded for symbol decisions. At the electrical processing module, one apply the Maximal Likelihood (ML) decision to the vector output of the detectors to make optimal symbol-by-symbol decisions.

Under the general network model given in Section 6.2, the optimum receiver is complex, as derived in Section 6.5. The analysis and results would not provide much insight into the signaling and detection schemes due to the heterogeneity of individual lightpath. Here we will make the simplifying assumption of homogeneous lightpaths, resulting in a simpler derivation and the results will provide much better insight into the proposed transmission scheme:

- *All the lightpaths are assumed to be homogeneous and independent; i.e.,* $f_1 = f_2 = \cdots = f_M = f$ *and* $N_1 = N_2 = \cdots = N_M$. Although some generality is lost due to this assumption, results based on this assumption will provide better insight for the optimization of the proposed transmission scheme. Under this assumption, a uniform energy (per bit) allocation algorithm at the transmitter is optimal (See Appendix C.1). Otherwise, the optimal energy (per bit) allocation algorithm can be obtained by solving a complicated convex optimization problem.

- *All the attenuation parameters are assumed to be equal and normalized to one.* Note that this result can be generalized to the unequal attenuation case by solving a complicated convex optimization problem. The detail of this optimization problem is beyond the scope of this thesis, and thus suppressed here.

6.3.1 Photo-Event Counting Processes

With the BPPM signaling and the uniform energy (per bit) allocation, the optical signal power over the i^{th} lightpath is either

$$P_i^{(0)}(t) = \begin{cases} \frac{P_s}{M}, & 0 \leq t \leq \frac{T}{2} \\ 0, & \frac{T}{2} \leq t \leq T \end{cases} \tag{6.1}$$

for hypothesis H_0 (i.e., symbol "0"), or

$$P_i^{(1)}(t) = \begin{cases} 0, & 0 \leq t \leq \frac{T}{2} \\ \frac{P_s}{M}, & \frac{T}{2} \leq t \leq T \end{cases} \tag{6.2}$$

for hypothesis H_1 (i.e., symbol "1"). In both cases, P_s is the average output power of the laser, and T is the symbol time.

The received optical signals can be corrupted by amplifier noise if optical amplifiers are used. The noise process is assumed to receive contributions from many spatial-temporal modes, and the probability of two successive noise-driven photo-events coming from the same spatial-temporal mode is close to zero. It follows that the Weak Photon-Coherence Assumption holds and one can approximate the noise-driven photo-event process with a point process of a constant rate λ_n equal to its mean [11]. This approximation is accurate within about 1-dB for a single channel. With M channels and many amplifiers in cascade, one can expect the approximation to be even better. Consequently, taking into account of the noise, the photo-event rate at the output of the detector is either

$$\lambda_i^{(0)}(t) = \begin{cases} \frac{F_i\lambda_s}{M} + \lambda_n, & 0 \leq t \leq \frac{T}{2} \\ \lambda_n, & \frac{T}{2} \leq t \leq T \end{cases} \tag{6.3}$$

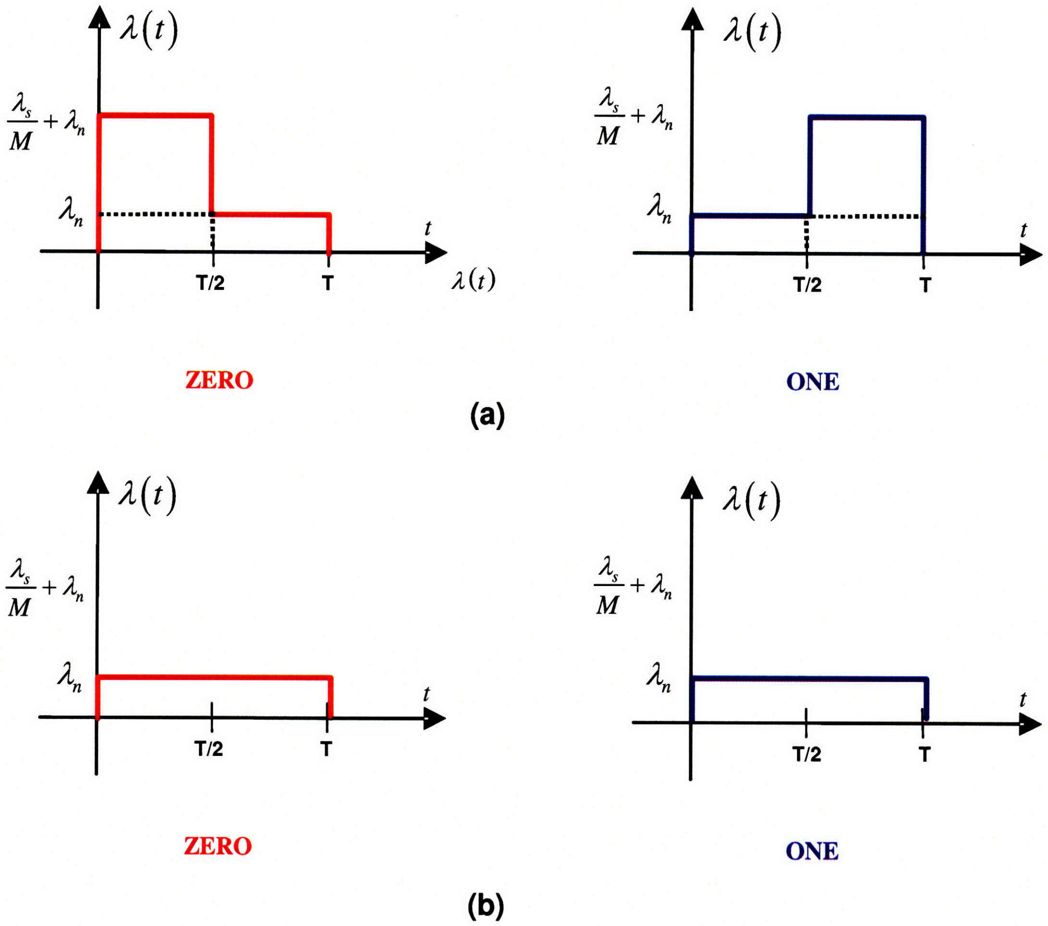


Figure 6-4: Detected photo-event rates for two hypotheses (H_0 and H_1) with BPPM signaling: (a) When the lightpath is UP, the detected rate is the sum of the signal rate (λ_s/M) and the noise rate (λ_n). (b) When the lightpath is DOWN, the detected rate is only the noise rate (λ_n).

for hypothesis H_0, or

$$\lambda_i^{(1)}(t) = \begin{cases} \lambda_n, & 0 \leq t \leq \frac{T}{2} \\ \frac{F_i\lambda_s}{M} + \lambda_n, & \frac{T}{2} \leq t \leq T \end{cases} \tag{6.4}$$

for hypothesis H_1. In both cases, $\lambda_s = \eta P_s/h\nu(\eta$ is the quantum efficiency of the detector, $h\nu$is the photon energy)[3] is the rate of the signal photo-event process with an average signal power of P_s, and F_i is a Bernoulli random variable with parameter $1-f$. Fig. 6-4 shows the rates of the photo-event counting process for (a)$F_i = 1$ and (b)$F_i = 0$. For a given hypothesis, the photo-event process is a Doubly-Stochastic Point Process due to its random rate parameter.

6.3.2 Optimum Detection Rule

In this section, we assume that there is a genie telling the failure status of each lightpath. In this case, if $m \leq M$ lightpaths are UP during the symbol duration, one can re-index them from 1 to m for a "genie-aided" receiver. Under this scenario, the optimal decision rule is the same as the detection rule for the scenario with m perfectly reliable lightpaths [61], i.e.,

$$\sum_{i=1}^{m} k_{i1} \underset{\hat{H}=H_1}{\overset{\hat{H}=H_0}{\gtrless}} \sum_{i=1}^{m} k_{i2}, \tag{6.5}$$

where k_{i1} and k_{i2} are photo-event counts during $[0, T/2]$ and $[T/2, T]$ over the lightpath, respectively.

6.3.3 Symbol Error Probability Bound

In this section, we derive an exponentially tight upper bound for the error probability of the genie-aided receiver via a two-step procedure:

[3] The attenuation coefficient has been normalized to one and thus is suppressed in this chapter.

1. the Chernoff Bound of the error probability conditioning on the number of UP lightpaths is calculated; and
2. the overall error probability upper bound is calculated by averaging the conditional error probability bound over the distribution of the number of UP lightpaths.

Given that m lightpaths are UP during the transmission, the conditional error probability is defined as

$$\begin{aligned}\Pr(\varepsilon|m) &= p_0 \Pr\left[\sum_{i=1}^{m} k_{i1} \leq \sum_{i=1}^{m} k_{i2}|H_0, m\right] + p_1 \Pr\left[\sum_{i=1}^{m} k_{i1} \geq \sum_{i=1}^{m} k_{i2}|H_1, m\right] \\ &= \Pr\left[\sum_{i=1}^{m} k_{i1} \leq \sum_{i=1}^{m} k_{i2}|H_0, m\right] \qquad (6.6)\end{aligned}$$

where p_0, p_1 are probabilities of sending the "ZERO" or "ONE" bit, and the second equality is due to the symmetry of binary pulse-position modulation and $p_0 = p_1 = 1/2$ for equiprobable digital source. Since the closed form solution of $\Pr(\varepsilon|m)$ is involved with summation of infinite numbers of terms, the exponentially tight Chernoff Upper Bound [61] is used here,

$$\begin{aligned}\Pr\left[\sum_{i=1}^{m} k_{i1} \leq \sum_{i=1}^{m} k_{i2}|H_0, m\right] &\leq \mathbb{E}_{s>0}\left\{e^{s\left(\sum_{i=1}^{m} k_{i2} - \sum_{i=1}^{m} k_{i2}\right)}|H_0, m\right\} \\ &= e^{mN_n(e^s-1)+\left(\frac{mN_s}{M}+mN_n\right)(e^{-s}-1)} \qquad (6.7)\end{aligned}$$

where $N_s = T\lambda_s/2$ is the average data-driven photo-event count of duration $T/2$ with binary pulse-position modulation and $N_n = T\lambda_n/2$ is the average noise-driven photo-event count per 1/2 bit. Since the inequality is valid for any value of $s > 0$, the bound can be tightened by minimizing the right hand side of (6.7),

$$\begin{aligned}\Pr(\varepsilon|m) &\leq \min_{s>0} \exp\{mN_s(e^s-1) + (\frac{mN_s}{M} + mN_n)(e^{-s}-1)\} \\ &= \exp\{-m(\sqrt{\frac{N_s}{M} + N_n} - \sqrt{N_n})^2\} \qquad (6.8)\end{aligned}$$

where the minimum is achieved when $e^s = \sqrt{1 + N_s/(MN_n)}$.

The overall error probability is then obtained by averaging the conditional error probability (6.8) over the distribution of the number of UP lightpaths, m ,

$$\Pr(\varepsilon) = \sum_{m=0}^{M} \Pr(\varepsilon|m)\Pr(m). \tag{6.9}$$

The number of UP lightpaths can be written as $m = \sum_{i=1}^{M} F_i$. It can be verified that m has a binomial distribution of

$$\Pr(m) = \frac{M!}{m!(M-m)!}(1-f)^m f^{M-m}. \tag{6.10}$$

Substituting (6.8) and (6.10) into (6.9), one can obtain

$$\Pr(\varepsilon) \leq \sum_{m=0}^{M} \frac{M!}{m!(M-m)!}(1-f)^m f^{M-m} e^{-m\psi(N_s,N_n,M)} \triangleq PB_{GA} \tag{6.11}$$

where $\psi(N_s, N_n, M) = (\sqrt{N_s/M + N_n} - \sqrt{N_n})^2$. Note that the right hand side of (6.11) has the form of the characteristic function of the random variable m . Using the fact that the characteristic function of a binomial random variable $X \sim B(n, 1-f)$ is $[f + (1-f)e^{jv}]^n$ [9], one can obtain the upper bound of the overall error probability as

$$PB_{GA} = \left[f + (1-f)e^{-(\sqrt{N_s/M+N_n}-\sqrt{N_n})^2}\right]^M. \tag{6.12}$$

For a sanity check, if $f = 0$, (6.12) turns out to be

$$PB_{GA} = \exp\{-(\sqrt{N_s + MN_n} - \sqrt{MN_n})^2\}, \tag{6.13}$$

which is the error probability bound for the source-destination pair connected by M reliable lightpaths. Note that f is the probability of the lightpath being DOWN where the error probability is equal to 1, and $1 - f$ is the probability of the lightpath being UP where the error probability is bounded by the term $e^{-\psi(N_s,N_n,M)}$, which is the Chernoff Bound for the error probability of a single lightpath with N_s/M

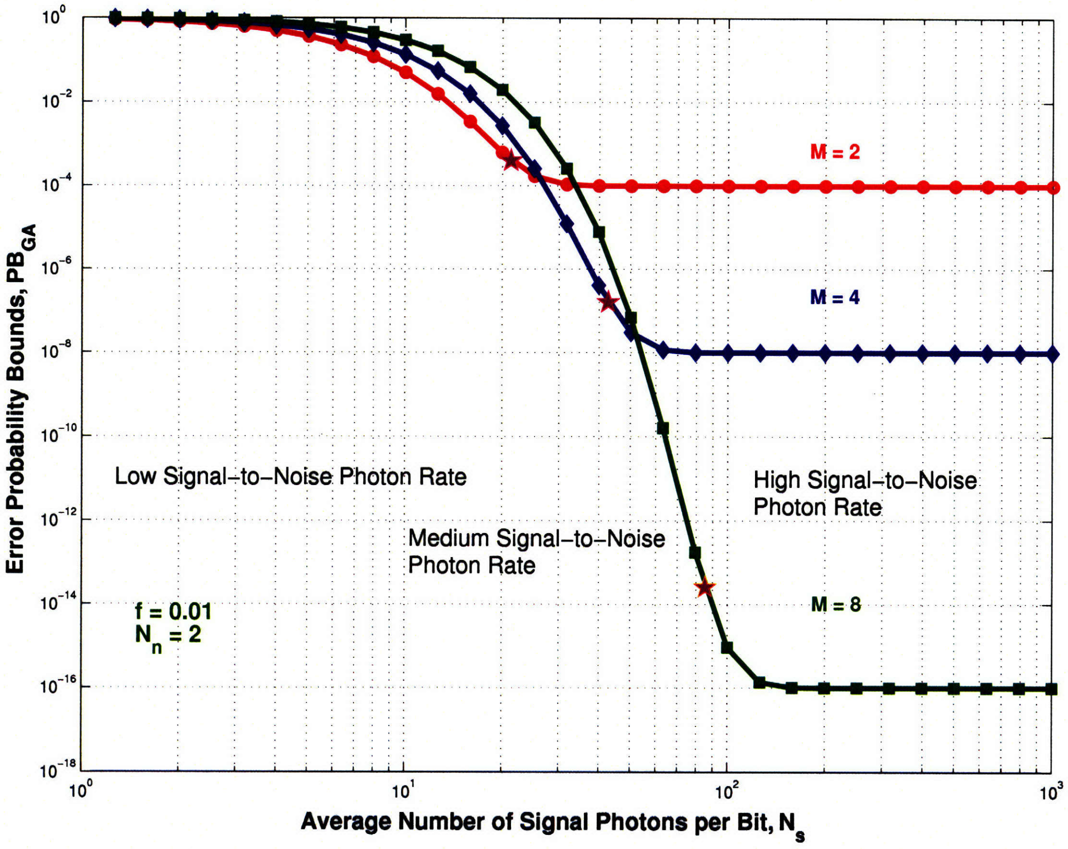


Figure 6-5: Error probability bounds for the idealized receiver with different number of lightpaths exhibit different characteristics in different signal-to-noise photon rate ratios. The star indicates the optimal operating point for each chosen number of lightpaths. $f = 0.01$ and $N_n = 2$.

signal photons per a bit and N_n noise photons per half a bit. It follows that the term $f + (1 - f)\exp\{-\psi(N_s, N_n, M)\}$ is the Chernoff Bound of the expected error probability for a single lightpath with signal power $P = \frac{N_s}{MN_s}$ and failure probability f. The overall error probability bound is obtained by reducing this expected error probability of a single lightpath to its M^{th} power, which can be defined as the lightpath diversity gain.

The error probability upper-bound (6.12) is plotted in Fig. 6-5, where the error curves exhibit different characteristics in three different signal-to-noise photon rate ratio regimes.

In the low signal-to-noise photon rate ratio regime, the lightpath-diversity mechanism has an inherently poor error performance, and thus is of no engineering interest.

In particular, if one let $f \rightarrow 0$ and $\Omega = N_s/(MN_n) \ll 1$, the error probability bound is reduced to

$$PB_{GA} \approx \exp\{-\frac{N_s^2}{4MN_n}\}. \tag{6.14}$$

The error exponent decreases if more lightpaths are used, which suggests that it actually hurts to use lightpath diversity in the low signal-to-noise photon rate ratio regime.

In the high signal-to-noise photon rate ratio regime, the error probability curves converge to error floors because the effect of lightpath failures dominates that of the amplification and detection noises. In fact, if one let $N_s/M \rightarrow \infty$ and $N_s/M \gg N_n$ for a fixed f, the symbol error probability becomes

$$PB_{GA} \approx f^M. \tag{6.15}$$

This verifies that the error floor phenomenon corresponds to the event in which the source-destination pair is disconnected from each other, with probability of f^M. This result suggests that topologies with a small probability of disconnection are preferable for reliable networks [64, 65]. Moreover, due to this saturation property, one cannot improve the error performance by simply increasing the signal-to-noise photon rate ratio. Thus, it is inefficient in energy utilization to work in the super-high signal-to-noise photon rate ratio regime.

In the medium-to-high signal-to-noise photon rate ratio regime, the error performance depends on the number of lightpaths (M, the lightpath diversity gain) and the signal-to-noise photon rate ratio. After some algebraic manipulations, (6.12) can be written as

$$PB_{GA} = \{f + (1-f)\exp[-N_n(\sqrt{\Omega+1}-1)^2]\}^M, \tag{6.16}$$

where the signal-to-noise photon rate ratio is given by $\Omega = N_s/MN_n$. As shown in (6.16), in order to achieve a lower error probability, we want to increase the number of lightpaths M and the signal-to-noise photon rate ratio simultaneously. However, for a given amount of optical energy per bit (signal photons per bit, N_s), it can be seen that

$$M \times \Omega = \frac{N_s}{N_n}. \tag{6.17}$$

This indicates that the number of lightpaths M and the signal-to-noise photon rate ratio Ω are two factors competing for a limited amount of optical energy. Therefore, one needs to balance this trade-off to optimize the system performance and improve the energy efficiency, which will be addressed in the next section.

6.4 System Optimizations

The output optical energy (per bit) of the transmitter is limited by physical constraints such as laser construction. One needs to utilize this limited amount of optical energy efficiently. As indicated in last section, the energy efficiency can be improved over the choice of the number of lightpaths for different objective functions.

6.4.1 Minimizing Error Probability for Limited Amount of Optical Energy

Given a limited amount of optical energy, the number of lightpaths can be chosen to minimize the error probability. Equivalently, one can minimize the error probability bound PB_{GA} since this bound is exponentially tight. It can be formulated as the following nonlinear optimziation problem,

$$\begin{aligned} \min \quad & G(M) = [f + (1-f)e^{-\psi(N_s,N_n,M)}]^M \\ s.t. \quad & M \in \mathbb{N}. \end{aligned} \tag{6.18}$$

Instead of finding the exact solution, one can relax the integer constraint, and assume M is a positive real number to solve the approximate problem without the integer constraint. Note that the minimum of $G(M)$ without the integer constraint is a lower bound of the minimum of $G(M)$ with the integer constraint. If $0 < f < 1/2$

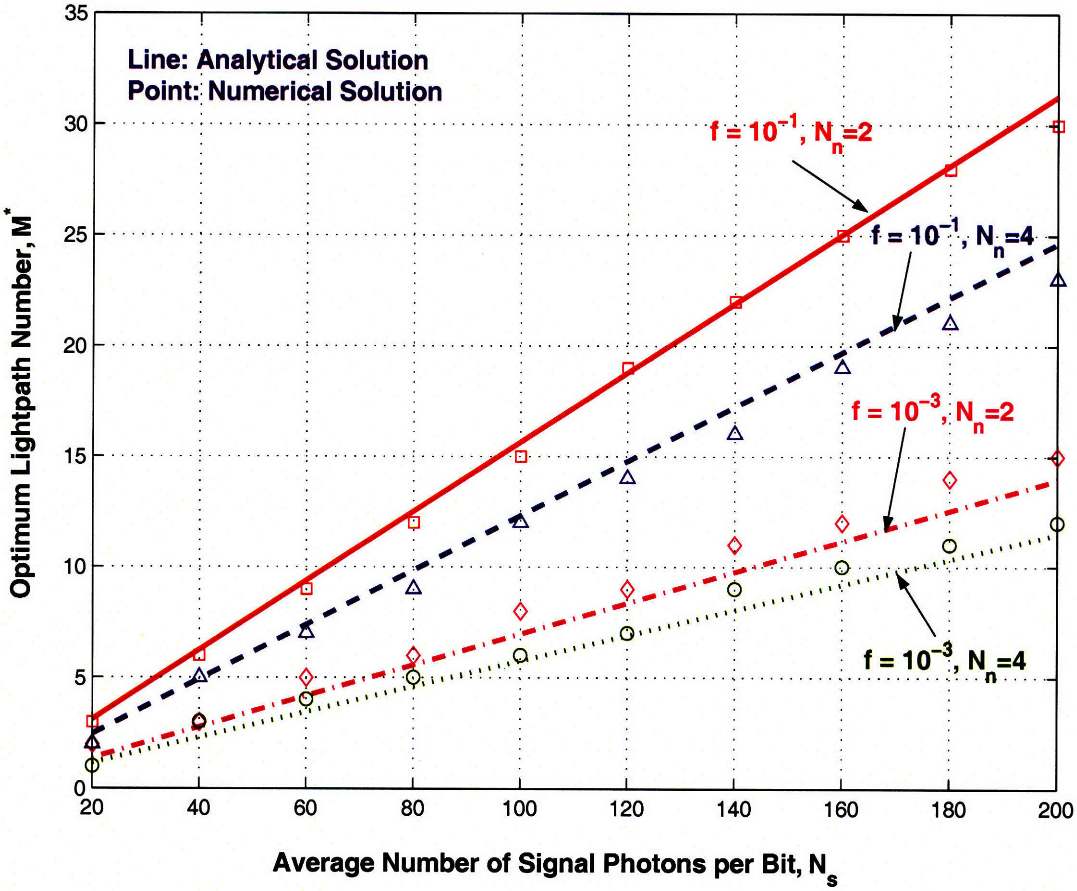


Figure 6-6: Optimal number of lightpath M^* is plotted against the average number of signal photons per bit N_s. As a comparison, the results from the exhaustive search algorithm is plotted.

(we are interested in this region since practical networks seldom have $f > 1/2$), the optimum lightpath number M^* (See Appendix C.2) is approximated by

$$M^* \approx \frac{N_s}{\zeta(f, N_n)}, \tag{6.19}$$

where $\zeta(f, N_n) = \ln(1/f - 1) + 2\sqrt{N_n}\sqrt{\ln(1/f - 1)}$. Considering the boundary condition that the minimum number of lightpath is one, the optimal number of lightpaths is given by $M^* = \max\left\{1, \frac{N_s}{\zeta(f, N_n)}\right\}$. In this chapter, we are interested in the non-constant part, i.e., $\frac{N_s}{\zeta(f, N_n)}$. For comparison, we have also found the optimal integer M^* by using an exhaustive search algorithm. In Fig. 6-6, the results from both the exhaustive search algorithm (bullets) and the analytical solution (lines) are plotted against different signal energy levels, i.e., the average number of signal photons per bit N_s. The analytical results match the numerical results well.

According to (6.19) and Fig. 6-6, the optimum lightpath number M^* decreases with higher noise energy per slot since one wants to maintain a certain level of signal-to-noise photon rate ratio, and also decreases with more reliable lightpaths since one has no incentive of using light-path diversity if the lightpath is reliable. Moreover, the optimum lightpath number M^* increases linearly with the transmitted energy per bit N_s. This suggests that *each lightpath requires a fixed optimal average number of signal photons per bit*, i.e.,

$$\frac{N_s}{M^*} = \ln(\frac{1}{f} - 1) + 2\sqrt{N_n}\sqrt{\ln(\frac{1}{f} - 1)}, \tag{6.20}$$

which is fully determined by the parameters of the lightpath, i.e., the lightpath failure probability f and the noise level N_n. When the lightpath is very reliable (i.e., $f \ll 1$), this number is asymptotically equal to $\ln(f^{-1}) + 2\sqrt{N_n}\sqrt{\ln(f^{-1})}$. This asymptotic result suggests that the optimal average number of photons per lightpath increases with higher noises and more reliable lightpaths. This is because, under these two scenarios, one needs more optical energy per lightpath to bias the lightpath at the optimum operating point, which will be addressed next.

Substituting (6.19) into (6.12), the minimum error probability bound is approximated by

$$PB^*_{GA} \approx (2f)^{M^*}. \tag{6.21}$$

This indicates that, at the optimum operating point where the number of lightpaths are optimally chosen to minimize the error probability bound for a given amount of optical energy (per bit), *each lightpath is biased to have an effective error probability of* $2f$, which is close to the saturation error probability of f for an individual lightpath. This result implies that the optimum operating point lies in the medium-to-high signal-to-noise photon rate ratio regime but near the high signal-to-noise photon rate ratio regime. This also justifies why each lightpath requires more photons when the lightpath is more reliable. In fact, when the lightpath failure probability f decreases, we need to increase the signal-to-noise photon rate ratio to sustain an effective error probability of $2f$.

Furthermore, if $f \ll 1$, the minimum error probability bound is approximated by

$$PB^*_{GA} \approx \exp\{-\frac{1}{1+2\sqrt{\frac{N_n}{\ln(f^{-1})}}}N_s\}. \tag{6.22}$$

The error exponent decreases linearly with the optical energy (per bit). This again suggests that the optimum strategy works at the medium-to-high signal-to-noise photon rate ratio regime and near the high signal-to-noise photon rate ratio regime; otherwise, the error exponent would depend quadratically on the total signal energy transmitted in low signal-to-noise photon rate ratio regime [56]. This is verified in Fig. 6-5 where the points marked by stars near the high signal-to-noise photon rate ratio regime correspond to optimum operating points for different optical energy per bit indicated in the horizontal axis. Also shown in (6.22), the ratio $N_n/\ln(f^{-1})$ determines the minimum error probability bound for a limited amount of optical energy (per bit)N_s. This says that both the noise and the lightpath failure probability exponent contribute equally in determining the optimum operating point. Finally, the asymptotic minimum error probability bound (6.22) approaches zero when one increases the optical energy (per bit). This implies that one can eliminate the saturation effect in the super-high signal-to-noise photon rate ratio regime if one chooses the number of lightpaths optimally.

6.4.2 Minimizing Energy Consumption for Target Error Probability

In this section, we will minimize the total transmitted optical energy (per bit) for a target error probability. Since the Chernoff Bound is exponentially tight, one can minimize the transmitted optical energy (per bit) for an equivalent target error probability bound. This is actually the dual of the problem of minimizing the error probability for a limited amount of optical energy.

For a given amount of optical energy (per bit) N_s, if one substitutes (6.19) into (6.21), one can approximate the minimum error probability bound approximated by

$$P_b \approx \exp\{-N_s \Theta(f, N_n)\}, \tag{6.23}$$

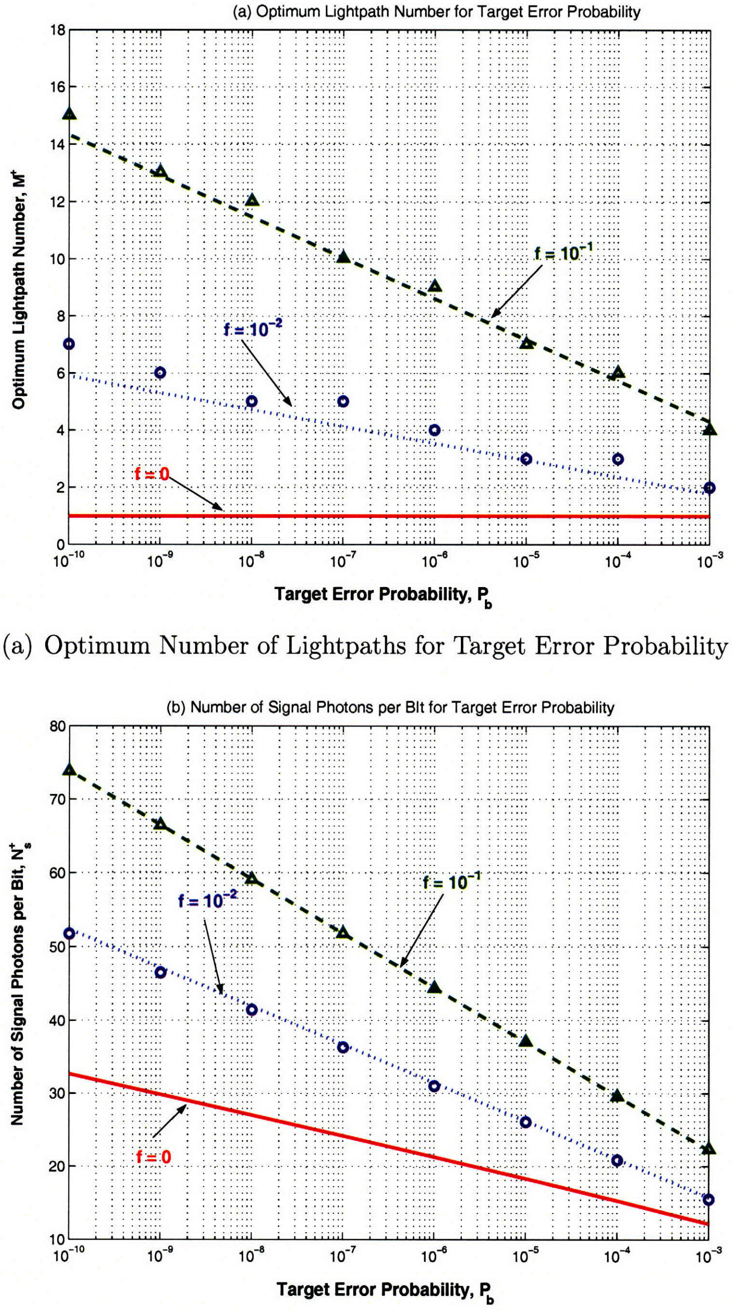
where $\Theta(f, N_n) = -\ln(2f)/\zeta(f, N_n) > 0$. Using (6.23), one can obtain the required minimum optical energy (per bit) for a target error probability bound P_b , given by

$$N_s^\dagger = \frac{-\ln(P_b)}{\Theta(f, N_n)}. \tag{6.24}$$

Substituting (6.24) into (6.19), the optimal number of lightpaths to minimize the transmitted energy (per bit) is obtained as

$$M^\dagger = \frac{\ln(P_b)}{\ln(2f)} = \log_{2f}(P_b). \tag{6.25}$$

In Fig. 6-7, the optimum lightpath number $M^\dagger$ and the minimum optical energy (per bit) $N_s^\dagger$ are plotted against different target error probabilities using the Chernoff bound, according to the analytical solutions (6.25) and (6.24). As a comparison, the numerical results from an exhaustive search algorithm are also labeled as points. Also



(a) Optimum Number of Lightpaths for Target Error Probability

(b) Number of Signal Photons per Bit for Target Error Probability

Figure 6-7: Number of lightpaths is chosen optimally to minimize the total energy per bit for target error probability bounds. (a) shows the optimum number of lightpaths for various target error probability bounds. (b) shows the minimum optical energy per bit for various target error probability bounds. The lines are for analytical solutions, and the bullets for numerical solutions. We choose $N_n = 1$ for all the cases.

in Fig. 6-7, the case of $f = 0$ is plotted for reference. If the lightpath is perfectly reliable (i.e., $f = 0$), lightpath diversity is not used since using more lightpaths only increases the total noise and degrades the error performance as suggested by (6.14).

As shown in Fig. 6-7, both the optimal number of lightpaths and the minimum optical energy (per bit) increase with lower target error probabilities, because each lightpath is biased at the optimum operating point to have error probability of $2f$ by requiring an optimum average number of photons (per bit). Notice also from Fig. 6-7 that more lightpaths are needed to achieve a target error probability bound when the reliability of individual lightpath deteriorates.

In (6.25), one cannot directly observe the effect of noise in determining the optimum lightpath number $M^{\dagger}$. As implied by (6.21), at the optimum operating point, each lightpath is biased to have error probability of $2f$, which is independent of the noise N_n. At the same time, in order to work at the optimum operating point, each lightpath requires an optimum average number of signal photons (per bit) given by (6.20). Therefore, when the noise increases, instead of requiring more lightpaths, we increase the total optical energy (per bit) to maintain the signal-to-noise photon rate ratio and thus bias each lightpath to have an effective error probability of $2f$. In fact, if one lets $f \ll 1$, the required minimum optical energy (per bit) is approximated by

$$N_s^{\dagger} \approx -\ln(P_b)\left[1+2\sqrt{-\frac{N_n}{\ln(f)}}\right]. \tag{6.26}$$

This says that, if one increases N_n , the required minimum optical energy (per bit) increases to bias each lightpath to maintain an effective error probability of $2f$ and thus the target error probability bound is achieved without requiring more lightpaths.

6.5 Optimum Realizable Receivers

Pragmatic engineering design is basically a trade-off between implementation complexity and symbol error probability. In general, in order to achieve a lower error probability, the receiver needs to estimate states of all the lightpaths for symbol decisions. This joint estimation and detection approach can result in a complicated receiver structure. On the other hand, a simpler receiver uses simpler lightpath state estimators or does not estimate the lightpath states at all, and thus usually has a higher error probability. To highlight this trade-off, two extreme cases for the complexity-error trade-off are explored in this chapter:

1. *The optimal receiver*: It has the lowest symbol error probability, but has the most complicated receiver architecture. This will be investigated in this section.
2. *The equal-gain-combining receiver*: The receiver architecture is much simpler. However, the error performance is sub-optimum since it does not exploit all the available information at the receiver. This will be investigated in the next section.

Between the two extreme cases are other reasonably good sub-optimal receivers. Their error performance is usually better than that of the equal-gain-combining receiver, and worse than that of the optimal receiver. On the other hand, their complexity falls between the most complicated optimal receiver and the simplest equal-gain-combining receiver. One of the research objectives in this chapter is to see how these receivers perform in different signal-to-noise photon rate ratio regimes and generalize rules of thumb to balance the complexity-error trade-off in practical optical receiver design.

In this section, we will first find the optimal counting receiver under the following framework. At the optical signal processing module, optical delay lines are used to compensate for delay variations among different lightpaths (fiber delays can also be replaced by time delays in the electrical processing stage since we will use parallel detectors); at the detection module, photon-counting receivers are used to record the photo-event times for symbol decisions; at the electrical processing module, to minimize the symbol decision error probability, a Maximum Likelihood (ML) detector is used to make symbol decisions based on the recorded photo-event time statistic.

6.5.1 Optimum Receiver Architecture

Let us start with the calculation of the likelihood functions for both hypotheses. For the i^{th} channel, let (k_{i1}, k_{i2}) be photo-event counts during the first half bit interval $[0, T/2]$ and the second half bit interval $[T/2, T]$, and

$$(\mathbf{t}_{i1}, \mathbf{t}_{i2}) = (t_1, t_2, \dots, t_{k_{i1}}, t_{k_{i1}+1}, t_{k_{i1}+2}, \dots, t_{k_{i1}+k_{i2}}) \tag{6.27}$$

be the corresponding photo-event time statistic. The conditional distribution density functions of the time statistic at the lightpath output, as derived in [56], are given by

$$p(\mathbf{t}_{i1}, \mathbf{t}_{i2}|H_0) = \left[\prod_{j=1}^{k_{i1}} (\hat{F}_i^{(0)}(t_j)\frac{\lambda_s}{M} + \lambda_n)\right] (\lambda_n)^{k_{i2}} e^{-\frac{\lambda_s}{M}[\int_0^{T/2} \hat{F}_i^{(0)}(t)dt]-2N_n} \tag{6.28}$$

and

$$p(\mathbf{t}_{i1}, \mathbf{t}_{i2}|H_1) = (\lambda_n)^{k_{i1}} \left[\prod_{j=1}^{k_{i2}} (\hat{F}_i^{(1)}(t_{j+k_{i1}})\frac{\lambda_s}{M} + \lambda_n)\right] e^{-\frac{\lambda_s}{M}[\int_{T/2}^{T} \hat{F}_i^{(1)}(t)dt]-2N_n} \tag{6.29}$$

where the minimum mean squared error (MMSE) causal estimate of the lightpath state for hypotheses H_j $(j = 0, 1)$ is given by

$$\hat{F}_i^{(j)} = \begin{cases} \mathbb{E}[F_i^{(j)}|H_j, N_t = 0], & N_t = 0 \\ \mathbb{E}[F_i^{(j)}|H_j, N_t = k, \mathbf{t}_{i1}, \mathbf{t}_{i2}], & N_t = k \geq 1 \end{cases} \tag{6.30}$$

and N_t is the number of photo-events over $[0, t]$. As derived in Appendix C.3, these estimators are given by

$$\hat{F}^{(0)}(t) = \frac{1}{1 + \frac{f}{1-f}\exp(\frac{\lambda_s}{M}t)(1+\Omega)^{-N_t}}, t \in [0, \frac{T}{2}], \tag{6.31}$$

where N_t is the number of photo-events over $[o, t]$, and

$$\hat{F}^{(1)}(t) = \frac{1}{1 + \frac{f}{1-f}e^{\frac{\lambda_s}{M}(t-\frac{T}{2})}(1+\Omega)^{-(N_t - N_{T/2})}}, t \in [\frac{T}{2}, T], \tag{6.32}$$

where N_t is the photon count over $[0, t]$, and $N_{T/2}$ is the photon count over $[0, T/2]$ of the same realization of the photo-event process.

Note that the photo-event time statistics of the lightpaths are independent because the lightpaths belong to different shared-risk groups. It follows that the overall conditional distribution density functions can be written as

$$p(\mathbf{t}_1, \mathbf{t}_2 | H_0) = \prod_{i=1}^{M} p\mathbf{t}_{i1}, \mathbf{t}_{i2} | H_0) \tag{6.33}$$

and

$$p(\mathbf{t}_1, \mathbf{t}_2 | H_1) = \prod_{i=1}^{M} p\mathbf{t}_{i1}, \mathbf{t}_{i2} | H_1) \tag{6.34}$$

where $\mathbf{t}_1, \mathbf{t}_2 | H_0) = (\mathbf{t}_{11}, \mathbf{t}_{21}, \ldots, \mathbf{t}_{M1}, \mathbf{t}_{12}, \mathbf{t}_{22}, \ldots, \mathbf{t}_{M2})$ is the overall photo-event time statistics. Using (6.33) and (6.34), the log likelihood-ratio can be written as

$$\begin{aligned} \ln \Lambda\{\mathbf{t}, \mathbf{N_1}, \mathbf{N_2} : 0 \le t \le T\} &= \ln \frac{p(\mathbf{t}_1, \mathbf{t}_2, \mathbf{N}_1, \mathbf{N}_2 | H_0)}{p(\mathbf{t}_1, \mathbf{t}_2, \mathbf{N}_1, \mathbf{N}_2 | H_1)} \\ = \sum_{i=1}^{M} &\left[\sum_{j=1}^{k_{i1}} \ln(1 + \hat{F}_i^{(0)}(t_j)\Omega) - \frac{\lambda_s}{M} \int_0^{T/2} \hat{F}_i^{(0)}(t) dt \right] \\ - \sum_{i=1}^{M} &\left[\sum_{j=1}^{k_{i2}} \ln(1 + \hat{F}_i^{(1)}(t_{j+k_{i1}})\Omega) - \frac{\lambda_s}{M} \int_{T/2}^{T} \hat{F}_i^{(1)}(t) dt \right] \end{aligned} \tag{6.35}$$

where $\Omega = \lambda_s / M\lambda_n = N_s / MN_n$ is the signal-to-noise photon rate ratio. After some algebraic manipulations, one can obtain the maximum likelihood detection rule as

$$\sum_{i=1}^{M} \left[\sum_{j=1}^{k_{i1}} \ln(1 + \hat{F}_i^{(0)}(t_j)\Omega) - \frac{\lambda_s}{M} \int_0^{T/2} \hat{F}_i^{(0)}(t) dt \right] \underset{\hat{H} = H_1}{\overset{\hat{H} = H_0}{\gtrless}} \sum_{i=1}^{M} \left[\sum_{j=1}^{k_{i2}} \ln(1 + \hat{F}_i^{(1)}(t_{j+k_{i1}})\Omega) - \frac{\lambda_s}{M} \int_{T/2}^{T} \hat{F}_i^{(1)}(t) dt \right] \tag{6.36}$$

For a sanity check, assume all the lightpaths are UP, i.e., $\hat{F}_i^{(0)}(t) = \hat{F}_i^{(1)}(t) = 1$, during the symbol transmission, the decision rule (6.36) turns out to be

$$\sum_{i=1}^{M} k_{i1} \underset{\hat{H}=H_1}{\overset{\hat{H}=H_0}{\gtrless}} \sum_{i=1}^{M} k_{i2}, \tag{6.37}$$

Note that the detection rule (6.37) is identical to the detection rule for the case with invulnerable lightpaths [61].

Note that each received photon is weighed by the scaling factor $\ln(1 + \hat{F}(t_j)\Omega)$ which depends on the lightpath state estimate at the photon arrival time. If the estimate of the lightpath state is large meaning that the possibility of the lightpath being UP is high, the scaling factor is large since it is more likely that the photon comes from the signal, not the noise. On the contrary, one assigns a small scaling factor to the photon if the lightpath state estimate is small. In particular, if one estimates that the lightpath is DOWN, the scaling factor is equal to zero since the photon must come from noise and thus should not be taken into consideration for detection.

Moreover, Detection rule (6.36) indicates a fundamental decomposition of functions in the optimal receiver structure, which is generalized as the separation theorem of detection in [56]. In particular, the receiver consists of two separable operation modules, i.e., estimators for lightpath states and signal processing modules for hypothesis testing, as shown in Fig. 6-8. This separation property suggests that one may be able to replace the complicated optimal lightpath state estimator with some simpler heuristic state estimators to reduce the receiver complexity without modifying the receiver structure. This idea often performs well in practice and yields near-optimal policies in dynamic programming [8]. Therefore, it is expected that the error performance with sub-optimal lightpath state estimators is not degraded significantly, which indeed is true, as will be shown in next sub-section.

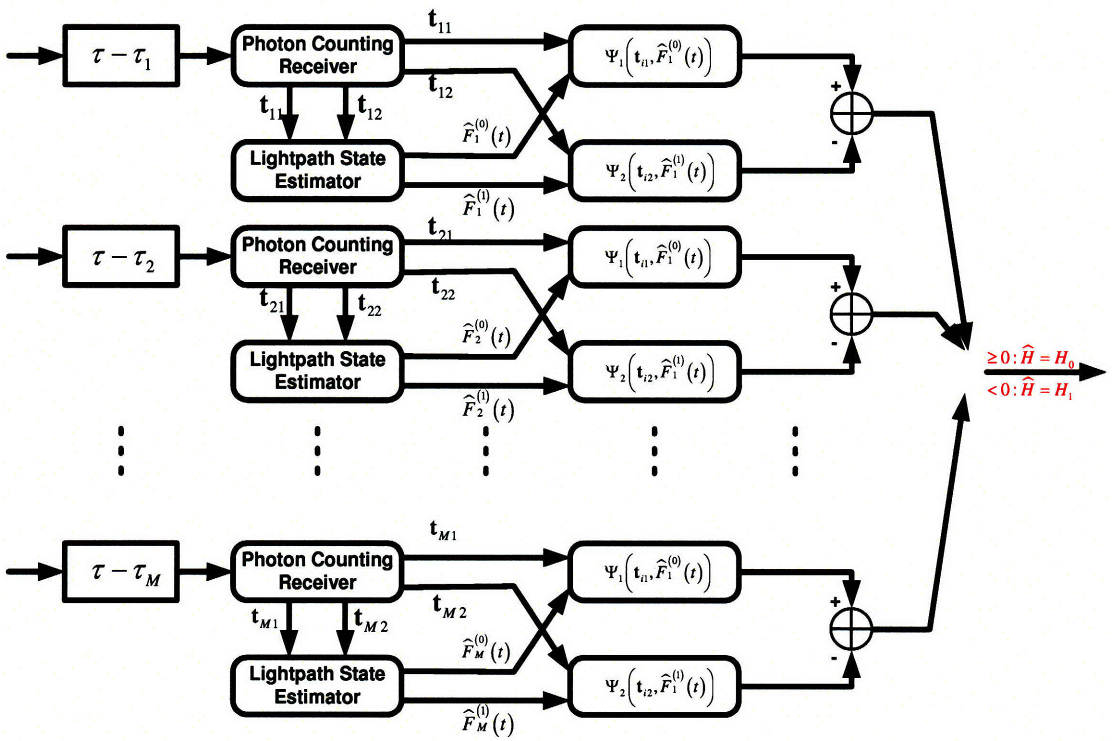


Figure 6-8: Optimal receiver architecture. $\Psi_1(\mathbf{t}_{i1}, \hat{F}_i^{(0)}(t)) = \sum_{j=1}^{k_{i1}} \ln(1 + \hat{F}_i^{(0)}(t_j)\Omega) - \frac{\lambda_s}{M} \int_0^{T/2} \hat{F}_i^{(0)}(t)dt$, where $\mathbf{t}_{i1}$ are photo-event time statistics and $\hat{F}_i^{(0)}(t)$ are channel sate estimators under H_0. $\Psi_2(\mathbf{t}_{i2}, \hat{F}_i^{(1)}(t)) = \sum_{j=1}^{k_{i2}} \ln(1 + \hat{F}_i^{(1)}(t_{j+k_{i1}})\Omega) - \frac{\lambda_s}{M} \int_{T/2}^{T} \hat{F}_i^{(1)}(t)dt$, where $\mathbf{t}_{21}$ are photo-event time statistics and $\hat{F}_i^{(1)}(t)$ are channel sate estimators under H_1.

6.5.2 Error Performance

In this section, the error performance of the optimal receiver is analyzed. In particular, a lower bound and an upper bound are derived for the exponentially tight Chernoff Bound of the symbol error probability.

As illustrated in Section 6.3, the symbol error probability of the genie-aided receiver is the 'genie-aided' limit of the proposed architecture within the class of structured receivers. For a sense of how well the optimal receiver performs, one can use the Chernoff Bound of the genie-aided receiver as a lower bound for Chernoff Bound of the optimal receiver because the Chernoff Bound is exponentially tight [61]. This suggests that the following lower bound for the Chernoff error bound of the optimal receiver,

$$PB_{opt} \geq \left[f + (1-f)e^{-\psi(N_s,N_n,M)}\right]^M \triangleq PB_{opt}^{LB}, \tag{6.38}$$

where $\psi(N_s, N_n, M) = (\sqrt{N_s/M} - \sqrt{N_n})^2$, $N_s = \lambda_s T/2$ is the average number of signal-driven photo-events per bit, $N_n = \lambda_n T/2$ is the average number of noise-driven photo-events per half a bit, and PB_{opt} is the error bound of the optimal receiver.

On the other hand, the optimal receiver must perform better than any suboptimal receiver within the class of structured receivers [35]. It follows that one can use the Chernoff Bound of any suboptimal receiver as an upper bound for the performance of the optimal receiver. In particular, one can choose a suboptimal receiver that uses the following non-causal estimator,

$$\tilde{F} = \begin{cases} 0, & \text{if} \quad \hat{F}(T) \leq 0.5 \\ 1, & \text{if} \quad \hat{F}(T) > 0.5 \end{cases} \tag{6.39}$$

where $\hat{F}(T)$ is the MMSE causal estimate of the channel state at time $t = T$ and is the estimated lightpath state. If $\tilde{F} = 0$, the receiver estimates the lightpath to be DOWN and thus discards the received signal over that lightpath. Otherwise, the receiver estimates the lightpath to be UP and thus uses the received optical signal over that lightpath for optimal combining and symbol decisions.

As derived in Appendix C.4, the upper bound for the Chernoff error bound of the optimal receiver, which is also the Chernoff error bound of the suboptimal receiver, is given by

$$PB_{opt} \leq \left[g + (1-g)e^{-\psi(N_s,N_n,M)}\right]^M \triangleq PB_{opt}^{UP}. \tag{6.40}$$

Here, the probability that the lightpath is estimated to be DOWN, $g = \Pr(\hat{F}(T) \leq 1/2)$, is given by

$$g = f\sum_{k=0}^{N_{TH}}\left[\frac{(N_n)^k}{k!}e^{-N_n}\right] + (1-f)\sum_{k=0}^{N_{TH}}\left[\frac{(N_s/M+N_n)^k}{k!}e^{-(\frac{N_s}{M}+N_n)}\right], \tag{6.41}$$

where

$$N_{TH} = \frac{\frac{N_s}{M} + \ln(\frac{f}{1-f})}{\ln(1+\frac{N_s}{MN_n})} \tag{6.42}$$

is the number of photons per bit beyond which the lightpath is estimated to be UP. In (6.42), N_s/M is the average number of photons per lightpath per bit, $\ln[f/(1-f)]$ is the additional number of photons needed to declare that the lightpath is UP, and both numbers must be adjusted by the term $\ln(1 + N_s/MN_n)$, which is the scaling factor in (30), to obtain the actual number of photons. If $0 < f < 1/2$, then $\ln[f/(1-f)] < 0$. This means that the actual number of photons needed is reduced since the probability of the lightpath being UP is higher and fewer photons per lightpath are needed for the estimator to declare that the lightpath is UP. If $1/2 < f < 1$, then $\ln[f/(1-f)] > 0$. This means that the actual number of photons needed is increased since the probability of the lightpath being DOWN is higher and more photons per lightpath are needed for the estimator to declare that the lightpath is UP.

Note that the lower bound (6.38) and the upper bound (6.40) have the same form, except that the prior lightpath failure probability f in (6.38) is replaced by the estimated lightpath failure probability g in (6.40). This implies that the tightness of the lower bound and the upper bound highly depends on the difference between the estimated lightpath failure probability and the prior lightpath failure probability. To explore this, the estimated lightpath failure probability g is compared with the

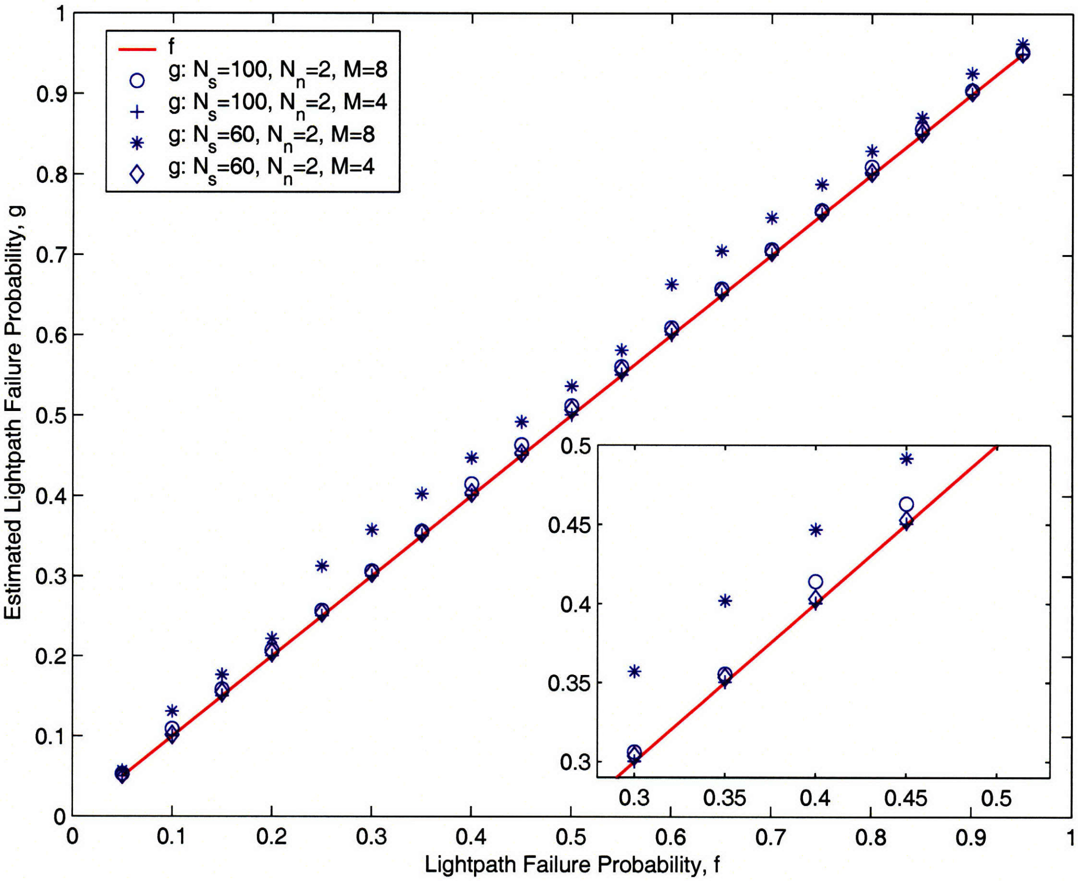


Figure 6-9: Estimated lightpath failure probability g is compared with the prior failure probability f under different signal-to-noise photon rate ratios.

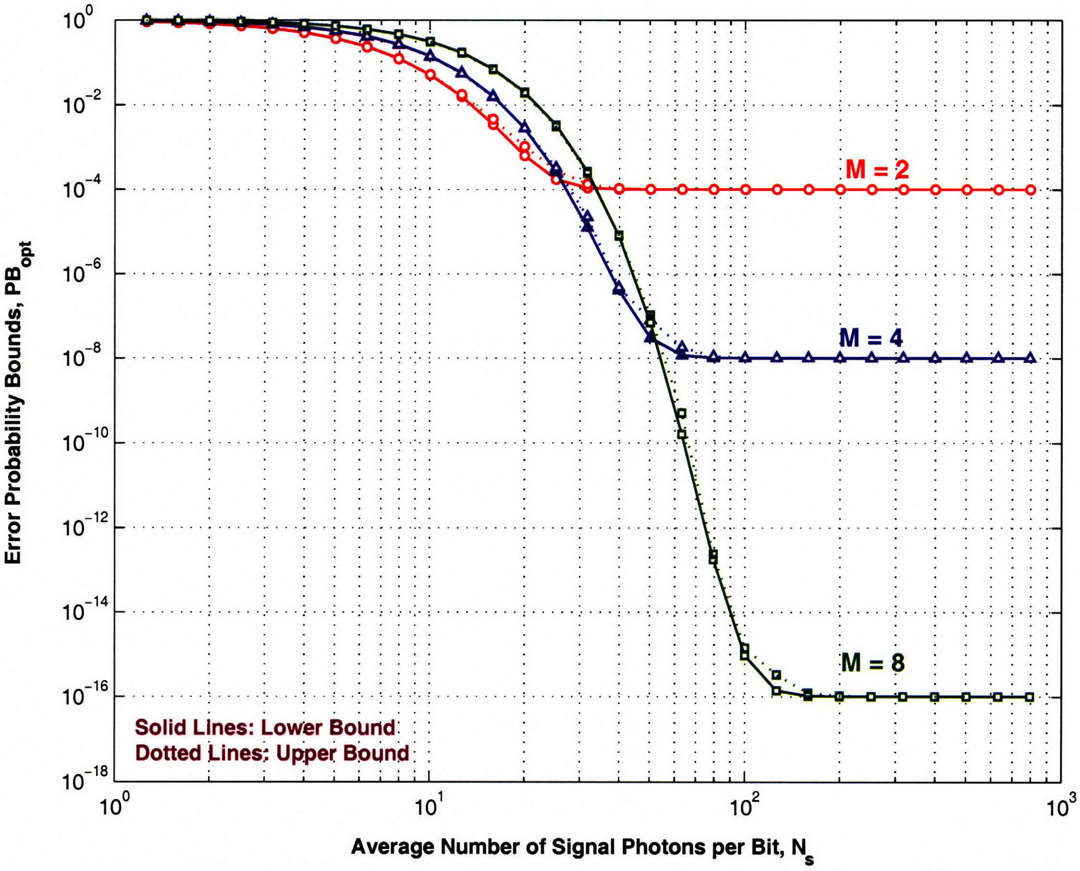


Figure 6-10: Lower bound and upper bound for the Chernoff bound of the optimal receiver. $f = 0.01$ and $N_n = 2$.

prior failure probability f in Fig. 6-9. Note that the difference between these two probabilities is negligible when the signal-to-noise photon rate ratio is high enough. It follows that the lower bound and the upper bound are close to each other, and thus both are very tight. This is verified in Fig. 6-10 where the lower bound and the upper bound are plotted against the average number of signal photons per bit. Moreover, these tight bounds suggest that the optimal receiver exhibits the same error characteristics in different signal-to-noise photon rate ratio regimes as the 'genie-aided' receiver, as shown in Fig. 6-10. In the super-high signal-to-noise photon rate ratio regime, the error bound converges to an error floor f^M, the probability with which the source-destination pair is disconnected. This suggests that network topologies with small probability of disconnection should be considered for ultra-high reliable optical networks. In the lower signal-to-noise photon rate ratio regime, the error probability increases with more lightpaths. It indicates that lightpath diversity actually hurts in this regime and be of no engineering interest. In the medium-to-high signal-to-noise photon rate ratio regime, the error probability depends on both the number of lightpaths and the signal-to-noise photon rate ratio. These two factors, however, are competing with each other for a given amount of optical energy. Hence, one needs to balance this trade-off to achieve better energy efficiency. This, along with the fact that the optimal receiver performs close to the 'genie-aided' receiver limit, suggests that system parameters optimized for the genie-aided receiver, such as the optimum number of lightpaths derived for different objective functions, also apply for the optimal receiver in the medium-to-high signal-to-noise photon rate ratio regime.

6.6 Equal-Gain-Combining Receivers

Although the optimal receiver has the lowest symbol error probability, it involves complicated processing by estimating the individual lightpath state throughout the symbol duration. In this section, we will develop one particular suboptimal receiver, the equal-gain-combining receiver, which not only approaches the optimal receiver in the symbol error probability under most scenarios, but also has the advantage of a simpler architecture.

6.6.1 Receiver Architecture

In the equal-gain-combining receiver, rather than estimating lightpath states, one assumes all the lightpaths to be UP and use the Maximum Likelihood decision rule to do symbol detection. Mathematically, the equal-gain-combining receiver employs the following decision rule

$$\sum_{i=1}^{M} k_{i1} \underset{\hat{H}=H_1}{\overset{\hat{H}=H_0}{\gtrless}} \sum_{i=1}^{M} k_{i2}, \tag{6.43}$$

to make symbol-to-symbol decision based only on the photo-event counts. Decision rule (6.43) is much simpler than decision rule (6.36) in that only one photon-counting receiver is needed. This indicates that the equal-gain-combining receiver offer a significant reduction in implementation complexity compared to the optimal receiver, at the expense of a degraded error performance, as shown in the following subsection.

6.6.2 Performance Analysis

Let us start with the calculation of the error bound for the equal-gain-combining receiver. Given the lightpath state vector $\mathbf{F}$, the conditional error probability is defined by

$$\begin{aligned}\Pr(\varepsilon|F\mathbf{m}) &= p_0 \Pr[\sum_{i=1}^{M} k_{i1} \leq \sum_{i=1}^{M} k_{i2}|H_0, \mathbf{F}] + p_1 \Pr[\sum_{i=1}^{m} k_{i1} \geq \sum_{i=1}^{M} k_{i2}|H_1, \mathbf{F}] \\ &= \Pr[\sum_{i=1}^{M} k_{i1} \leq \sum_{i=1}^{M} k_{i2}|H_0, \mathbf{F}] \end{aligned} \tag{6.44}$$

where p_0, p_1 are probabilities of sending the "ZERO" or "ONE" bit, and the second equality is due to the symmetry of binary pulse-position modulation and $p_0 = p_1 = 1/2$ for equiprobable digital source.

Let $K_1 = \sum_{i=1}^{M} k_{i1}$ be the total photo-event count recorded over $[0, T/2]$, and $K_2 = \sum_{i=1}^{M} k_{i2}$ be the total photo-event count recorded over $[T/2, T]$. Note that, given hypothesisH_0 and the lightpath state vector $\mathbf{F}$, $K_1$is a Poisson random variable with mean $mN_s/M + MN_n$, where $m = \sum_{i=1}^{M} F_i$ is the number of UP lightpaths for a given lightpath state vector $\mathbf{F}$, and $K_2$is a Poisson random variable with mean MN_n.

Using the Chernoff Bound, the conditional error probability is bounded by

$$\Pr(\varepsilon|\mathbf{F}) = \exp\{-(\sqrt{m(\frac{N_s}{M}) + MN_n} - \sqrt{MN_n})^2\}. \tag{6.45}$$

Notice that m is a binomial random variable with a distribution function of $\Pr(m = k) = \binom{M}{k}(1-f)^k f^{M-k}, k = 0, 1, \ldots, M$. Averaging (6.45) over all possible lightpath state vectors $\mathbf{F} \in \{0, 1\}^M$, we obtain the error bound for the equal-gain-combining receiver as

$$\Pr(\varepsilon) = \sum_{k=0}^{M} \frac{M!}{k!(M-k)!}(1-f)^k f^{M-k} e^{-(\sqrt{k(N_s/M)+MN_n} - \sqrt{MN_n})^2}. \tag{6.46}$$

Using (6.46), one can compare the error bound of the equal-gain-combining receiver with the 'genie-aided' receiver limit in Fig. 6-11. In the low signal-to-noise photon rate ratio regime, the error probability is inherently high and of no engineering interest. In the medium-to-high signal-to-noise photon rate ratio regime, the gap between error bounds of the equal-gain-combining receiver and the 'genie-aided' limit is larger than the gap between error bounds of the optimal receiver and the 'genie-aided' limit. With the equal-gain-combining receiver, noise from DOWN lightpaths will degrade the average signal-to-noise photon rate ratio and thus increases the error probability since the error probability in the medium-to-high signal-to-noise photon rate ratio regime is sensitive to the signal-to-noise photon rate ratio. However, in the high signal-to-noise photon rate ratio regime, the equal-gain-combining receiver has

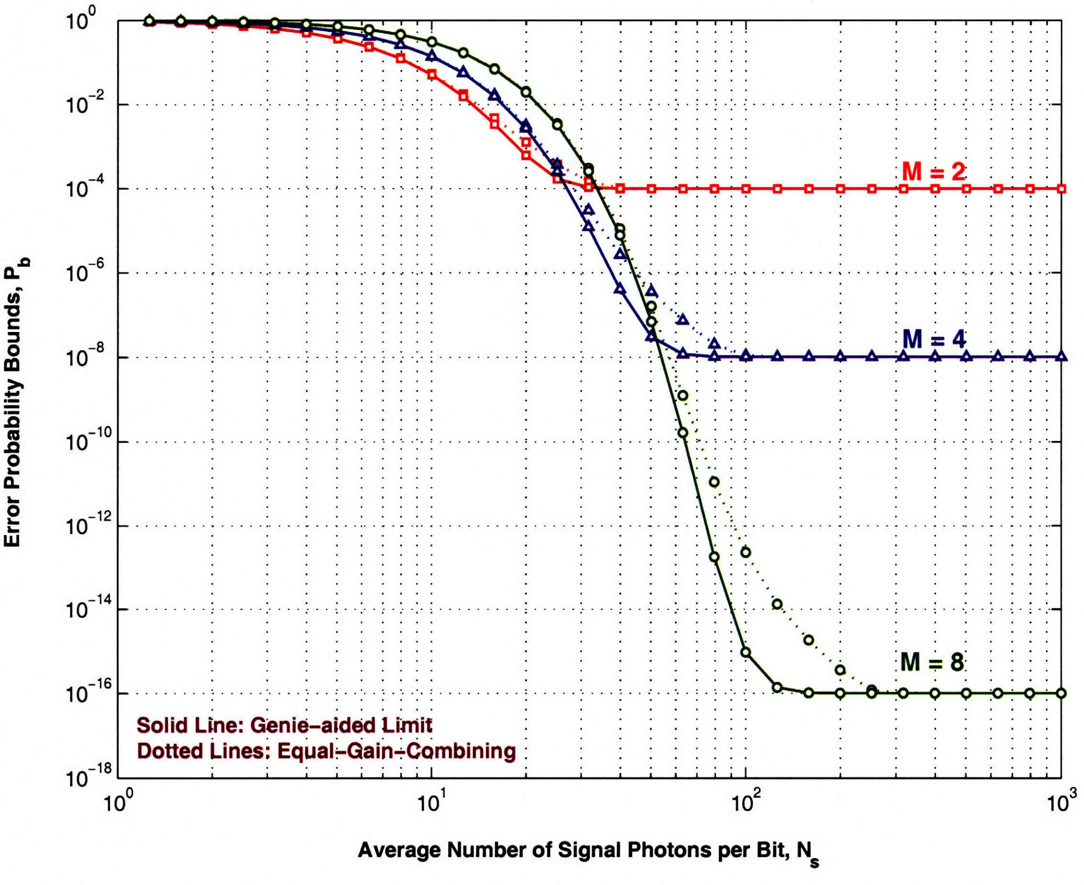


Figure 6-11: Error bounds of the EGC receiver are compared with the genie-aided receiver limit under different lightpath numbers. $f = 0.01$ and $N_n = 2$.

an error bound close to the 'genie-aided' receiver limit. This indicates that the equal-gain-combining receiver is preferable to the optimal receiver in the high signal-to-noise photon rate ratio regime due to its simplicity. In fact, the equal-gain-combining receiver approaches asymptotically the optimal receiver when the noise is negligible, as to be shown next.

6.6.3 Power Penalty

Since the error probability of the equal-gain-combining receiver is higher than that of the 'genie-aided' receiver, one needs to transmit more optical energy in order for the equal-gain-combining receiver to achieve the same target error probability as the 'genie-aided' receiver does in the medium-to-high signal-to-noise photon rate ratio regime. In this subsection, we determine this amount of additional power for the equal-gain-combining receiver to achieve a target error probability bound compared to the genie-aided receiver. For a target error probability bound of P_b , the power penalty of the equal-gain-combining receiver over the 'genie-aided' receiver is defined as

$$\delta = 10\log_{10}\left[\frac{N_s^*(P_b, f, N_n; EGC)}{N_s^*(P_b, f, N_n; GA)}\right], \tag{6.47}$$

where $N_s^*(P_b, f, N_n; GA)$ and $N_s^*(P_b, f, N_n; EGC)$ are the minimum amounts of optical power (in terms of average number of signal photons per bit) for the genie-aided receiver and the equal-gain-combining receiver respectively to achieve a target error probability P_b.

Using numerical results by exhaustive searching, the optimal number of lightpaths and the minimum transmitted optical energy are plotted in Fig. 6-12 (a) and (b). To achieve the same error probability bound, the equal-gain-combining receiver requires more lightpaths and more optical energy. This suggests that a more densely-connected network topology is needed to provide enough independent lightpaths for the equal-gain-combining receiver. The power penalty is plotted in Fig. 6-12 (c) and (d). From plot (c), the power penalty is asymptotically independent of the target error probability. This is due to two reasons. First, the error bound of the equal-gain-

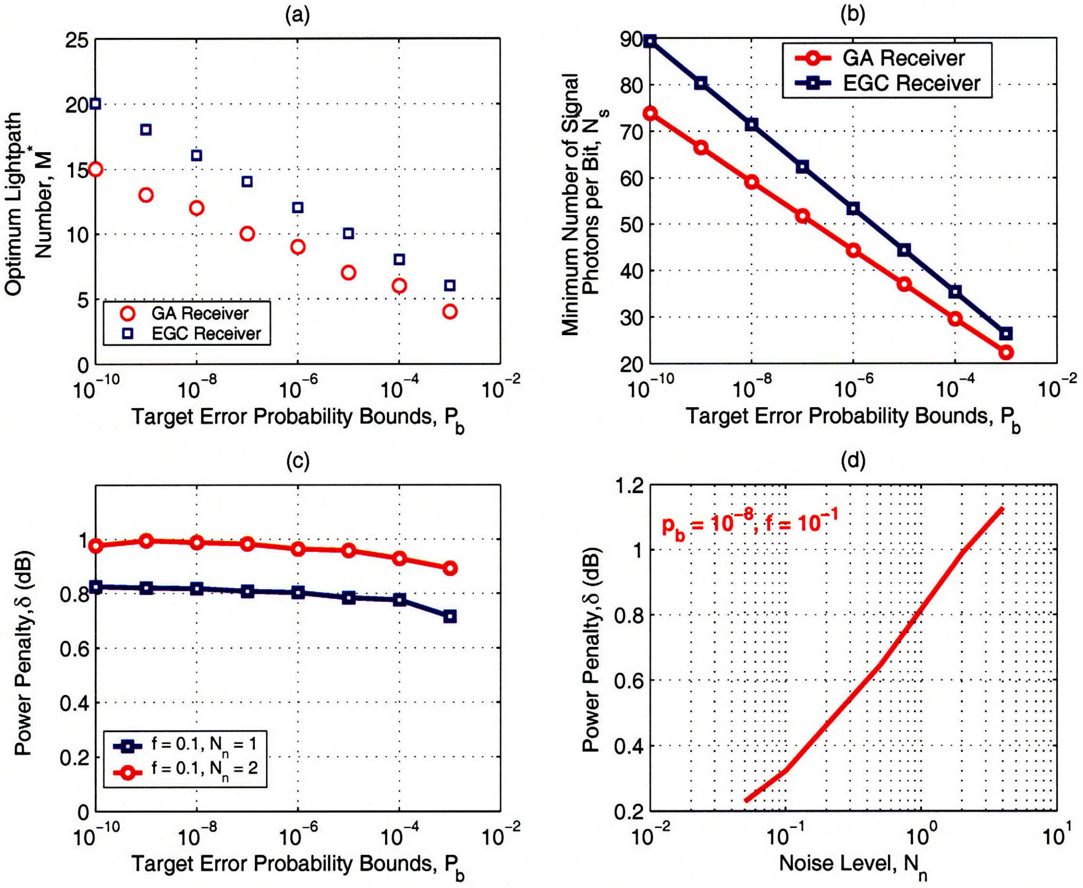


Figure 6-12: (a)Optimal number of lightpaths to minimize the total optical energy is plotted against different target error probability bounds. (b) The minimum number of signal photons per bit is plotted against different target error probability bounds for the genie-aided receiver and the EGC receiver. In (a) and (b), we set $f = 0.01$ and $N_n = 2$. GA: genie-aided receiver; EGC: equal-gain-combining receiver. (c) Power penalty of the EGC receiver is plotted under different target error probability bounds. (d) Power penalty of the EGC receiver is plotted under different noise levels.

combining receiver is close to that of the genie-aided receiver with optimized system parameters. Second, the minimum transmitted power is linear with the error exponent given by (6.24) in Section 6.4. It follows that, at the optimum operating points, both error bounds are parallel to each other in a log-log plot. The power penalty is approximately determined by the ratio between the slopes of the error exponents of the 'genie-aided' receiver and the equal-gain-combining receiver at the respective optimum operating points. Therefore, the power penalty is independent of the target error probability bounds. On the other hand, the power penalty increases with higher noise levels as shown in plot (d), and approaches zero when the noise level goes to zero. This demonstrates that the equal-gain-combining receiver is generally suboptimal and approaches the optimal receiver when the noise level decreases. In particular, if there is no noise, the equal-gain-combining receiver would be optimal because the receiver would not receive any noise from DOWN lightpaths to degrade the error performance. Moreover, for the practically interesting parameters, the power penalty is around 1-dB. In practical system design, if this 1-dB penalty is acceptable, the equal-gain-combing receiver is preferable over the optimum receiver due to its simplicity.

6.7 Conclusion

In this chapter, the use of lightpath-diversity was proposed to achieve ultra-reliable end-to-end communication with low delay requirements in all-optical networks. For a network with dense connections, arbitrary reliability can be achieved if enough independent lightpaths are used. Since this approach is implemented entirely at the Physical Layer without the use of higher layer protocols such as ARQ's (i.e., Automatic Repeat reQuest), the response is fast enough for applications with super-high date rates and/or critical time deadlines.

From a theoretical perspective, we have characterized the proposed lightpath-diversity system with a Doubly-Stochastic Point Process model. The limit on the error probability of the scheme has been obtained via a 'genie-aided' receiver. This 'genie-aided' receiver limit serves as a benchmark for practical receiver architectures. Under typical operating scenarios, we have optimized the system performance by choosing an optimal number of lightpaths to utilize the limited optical power efficiently. Analytical proof showed that, *at the optimum operating point, each lightpath requires an optimum number of signal photons to bias itself at the effective error probability of* $2f$, where f is the lightpath failure probability. This optimum average number of photons per lightpath is fully determined by the lightpath parameters, including the lightpath failure probability and the noise level.

From an engineering perspective, we have investigated the class of structured receivers for the multiple-lightpath transmission architecture. Using the Doubly Stochastic Point Process model, we have developed the architecture of the optimal receiver, and have bounded its error performance with a lower bound (the genie-aided receiver) and an upper bound (non-causal estimator). The tightness of the lower bound and the upper bound indicates that the optimal receiver approaches the genie-aided limit of structured receivers, and thus system parameters optimized for the genie-aided receiver apply to the optimal receiver in the medium-to-high signal-to-noise photon rate ratio regime. However, the optimal receiver needs to estimate lightpath states throughout the symbol time, which is complicated. To balance error probability performance and implementation complexity, we suggested the use of a suboptimal equal-gain-combining receiver with lower complexity, and have characterized its error performance. Performance comparison between the equal-gain-combing receiver and the 'genie-aided' receiver limit of structured receiver showed that the power penalty of the equal-gain-combining receiver decreases with decreasing noise level. These results suggest that *the equal-gain-combing receiver is preferable to the optimal receiver in the high signal-to-noise photon rate ratio regime, and the optimal receiver is needed for good performance in the low signal-to-noise photon rate ratio regime at the expense of increased complexity.* For practical system design, if the 1-dB penalty (for typical system parameters)is acceptable, the equal-gain-combing receiver is always the preferable due to its simplicity.

Chapter 7

Thesis Contributions

Rapid progress in the field of communication systems and networks during the last decade has yielded increasingly faster, more intelligent and almost ubiquitous network services. Quality of service (e.g., fast, reliable, robust and secure) is of vital importance in current network design. Achieving this objective at an affordable cost, however, has become more challenging mainly due to network dynamics. Examples of network dynamics include varying channel conditions in wireless networks and transient effects enabled by agile lightpath reconfigurations in optical networks. In this thesis, we are particularly interested in two research areas: i) monitoring state information about network dynamics and ii) overcoming detrimental effects resulting from network dynamics. Both areas are crucial for network carriers to maintain promised quality of service. This thesis focuses on characterizing major tradeoffs among various performance metrics, with an objective of providing engineering insights for practical network designs.

7.1 Network Diagnosis via a Framework of Group Testing over Graphs

In order to deliver promised quality of service, network carriers run sophisticated network management systems (NMS) to manage and control network operations. NMS relies on network state information provided by network monitoring schemes that normally have multiple design objectives, such as low overhead, small delay, and high accuracy. Network-monitoring schemes for different applications have been tailored previously to their unique design requirements, for example, small delay for fault detection and high accuracy for network tomography. Nevertheless, different network-monitoring functions share some commonalities (e.g., using random variables to model parameters under monitoring). Therefore, we have been motivated to develop: i) a common mathematical framework that models key issues in network monitoring, and ii) an information-theoretic approach that maps the network-diagnosis problem into the source-coding problem by viewing network states as source alphabets and diagnosis algorithms as source codes. This mapping allows us to exploit well-established information-theoretic results to characterize the trade-offs among different design metrics and develop optimal diagnostic algorithms.

As an example, we have investigated the fault detection and localization problem for dynamic optical networks. Main results obtained in this research are as follows.

- We have developed a group-testing-over-graphs framework to model the fault-diagnosis problem. The network is abstracted as a graph in which the failure status of each node/link is modeled as a Bernoulli random variable. Probing signals are sent along a set of lightpaths and their measurements are used to infer the network state of health. This framework can be extended to model many other network monitoring applications by choosing appropriate state variables.

- We have identified and characterized the trade-off between the number of lightpath probes (as a metric of the diagnostic effort) and the number of probing steps (as a metric of the diagnostic delay). This trade-off can be balanced by scheduling lightpath probes in different fashions: i) adaptive diagnosis, where individual probes are sent sequentially, ii) non-adaptive diagnosis, where probes are sent in parallel, and iii) multi-stage diagnosis, where probes are sent sequentially in batches.

- We have initiated an information-theoretic approach to minimizing the number of lightpath probes for adaptive diagnosis schemes, by mapping the fault-diagnosis problem into the source-coding problem. This mapping leads to an entropy lower bound on the number of probes and an approach to translating efficient source codes (e.g., the run-length code) into scalable fault-diagnosis schemes.

- We have started to characterize the trade-off between the hardware cost and the probability of successful diagnosis. Preliminary results indicate that the hardware cost can be reduced significantly by accepting some uncertainty in assessing the network state.

The immediate plan to continue this research focuses on exploiting the theoretical frontier in the group-testing-over-graphs framework:

Diagnosis for different performance parameters Our previous research assumes that the network performance parameter being monitored is the node/link failure status, modeled as a Bernoulli random variable. Choosing appropriate state random variables, we would like to extend this framework to design scalable diagnosis schemes for other network performance parameters (e.g., noise level, packet delay, packet drop ratio, etc).

Diagnosis with probabilistic measurements Our previous research assumes that the probe syndrome is a deterministic function of all the probed node/link states. In general, probe syndromes could be probabilistic, due to noisy or unreliable measurements. In this case, the key design objective is to characterize the trade-off between the diagnostic effort and the estimate error.

Diagnosis with acceptable uncertainty Our previous research has indicated that the hardware diagnostic cost (e.g., transmitters/receivers) is prohibitively high for 100% diagnostic confidence. It is of practical interest to investigate how this capital cost scales with some diagnostic uncertainty.

7.2 Lightpath Diversity for Ultra-reliable Communications

In the data plane, quality of service can be improved by incorporating redundancy into network design in order to overcome the detrimental effects engendered by network dynamics. The idea of adding redundancy, such as error-correction codes for noisy channels, has been successfully implemented in point-to-point communications. In networked communications, the richly-connected network fabric, inherent in emerging mesh networks, provides additional venues for redundancy via multipath connections. In Chapter 6, we have explored a physical-layer lightpath diversity scheme that transmits the same signals along multiple lightpaths in optical networks. In particular, we have developed an optimum signaling and detection scheme to optimize system performance and provide reliable end-to-end data delivery in the presence of network failures.

Specifically, we have obtained the following results:

- We have derived an upper bound (i.e., the Chernoff bound) on the bit error probability for the proposed lightpath-diversity system with a Doubly-Stochastic Point Process model. The limit on the error probability of the scheme has been obtained via a 'genie-aided' receiver. This 'genie-aided' receiver limit serves as a benchmark for practical receiver architectures.

- We have optimized the system performance by choosing an optimal number of lightpaths to utilize the limited optical power efficiently. Analytical proof showed that each lightpath requires an optimum number of signal photons to bias itself at the effective error probability of $2f$ where f is the failure probability of each lightpath. This optimum average number of photons per lightpath is fully determined by the lightpath parameters, including the lightpath failure probability and the noise level.

- We have developed the architecture of the optimal receiver(causal estimator), and have bounded its error performance with a lower bound (the genie-aided receiver) and an upper bound (non-causal estimator). The tightness of the lower bound and the upper bound indicates that the optimal receiver approaches the genie-aided limit of structured receivers, and thus system parameters optimized for the genie-aided receiver apply to the optimal receiver in the medium-to-high signal-to-noise photon rate ratio regime.

- We have also developed a suboptimal equal-gain-combining receiver with lower complexity, and have characterized its error performance. Performance comparison between the equal-gain-combing receiver and the 'genie-aided' receiver limit of structured receiver showed that the power penalty of the equal-gain-combining receiver decreases with decreasing noise level. These results suggest that the equal-gain-combing receiver is preferable to the optimal receiver in the high signal-to-noise photon rate ratio regime, and the optimal receiver is needed for good performance in the low signal-to-noise photon rate ratio regime at the expense of increased complexity. For practical system design, if the marginal 1-dB penalty is acceptable, the equal-gain-combing receiver is always the preferable due to its simplicity.

Appendix A

Adaptive Fault Diagnosis Schemes

A.1 Optimality of the Link-Wise Probing Scheme

In this section, we will establish the optimality condition under which the link-wise probing scheme is optimal, as summarized in the following theorem:

Theorem A.1. *For any non-trivial network topology with a connected component of more than one edge, the edge-wise probing scheme is optimal if and only if the edge failure probability is larger than* $\frac{3-\sqrt{5}}{2}$ *(the golden ratio).*

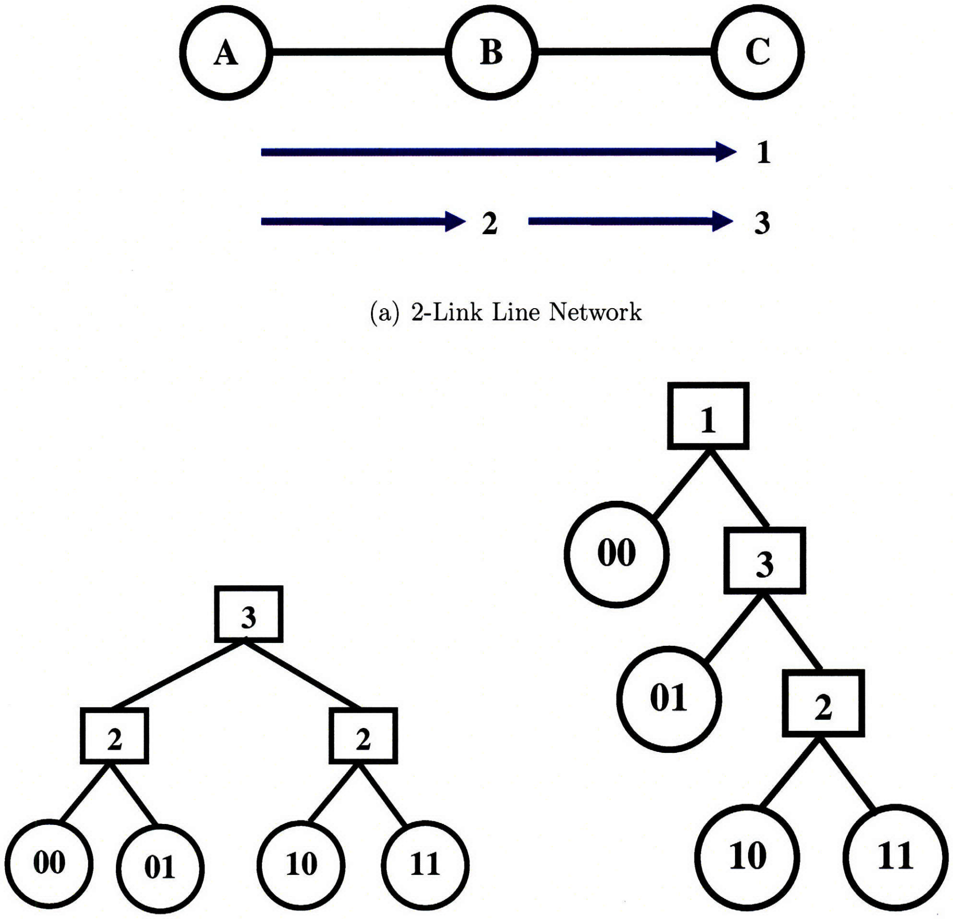
Proof. First of all, if the number of edges in any connected component is less than or equal to one, the optimal probing scheme is always to probe each individual edge in the network. In the following, we focus on network topologies with a connected component of at least two edges.

Let $\bar{\mathcal{L}}^*(m,p)$ be the minimum average number of probes for any network with m links, where p is the failure probability of each individual link. It is easy to see that $\bar{\mathcal{L}}^*(m,p) \leq m$, because one can always probe each individual links.

First, let us look at the case of $m = 2$, as illustrated in Fig. A-1(a). There are only two possible probing schemes in this case, as shown in Fig. A-1(b) and A-1(c).

For the probing scheme Υ_1, the average number of probes required is

$$\bar{\mathcal{L}}_{\Upsilon_1} = 2. \tag{A.1}$$



(a) 2-Link Line Network

(b) Probing Scheme One

(c) Probing Scheme Two

Figure A-1: Case study for a 2-line line network: (a) a line network with 2 edge;(b) probing scheme one (Υ_1); (c) probing scheme two (Υ_2).

For the probing scheme Υ_2, the average number of probes required is

$$\bar{\mathcal{L}}_{\Upsilon_2} = 1 \times (1-p)^2 + 2 \times (1-p)p + 3 \times p = -p^2 + 3p + 1. \tag{A.2}$$

Solving the inequality of $\bar{\mathcal{L}}_{\Upsilon_1} \leq \bar{\mathcal{L}}_{\Upsilon_2}$, we obtain

$$\frac{3-\sqrt{5}}{2} \leq p \leq 1. \tag{A.3}$$

This suggests that link-wise probing scheme for this case is optimal when $p \geq \frac{3-\sqrt{5}}{2}$.

For the case of $m > 2$, we split the network into two subgraphs: one network with k links and the other network with $m-k$ link. These two networks are probed separately. It can be seen that

$$\bar{\mathcal{L}}^*(m,p) \leq \bar{\mathcal{L}}^*(k,p) + \bar{\mathcal{L}}^*(m-k,p). \tag{A.4}$$

Using the upper bound of $\bar{\mathcal{L}}^*(m,p) \leq m$ and the fact that $\bar{\mathcal{L}}^*(m,p) < 2$ if $0 < p < \frac{3-\sqrt{5}}{2}$, we can obtain

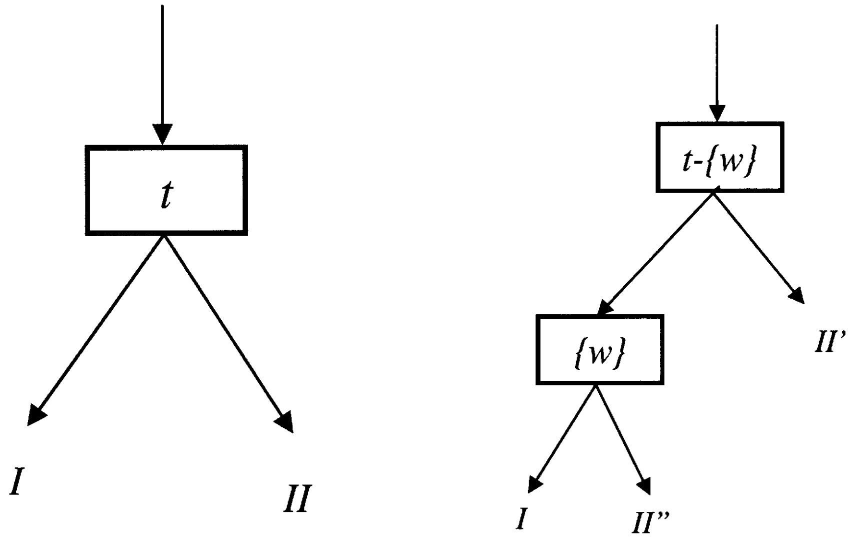
$$\bar{\mathcal{L}}^*(m,p) \leq \bar{\mathcal{L}}^*(m-2,p) + \bar{\mathcal{L}}^*(2,p) \leq m - 2 + \bar{\mathcal{L}}^*(2,p) < m. \tag{A.5}$$

This result indicates that, for $0 < p < \frac{3-\sqrt{5}}{2}$ and $m \geq 2$, the link-wise probing scheme is not optimal.

The part of the proof is to show that the link-wise probing scheme is optimal for $\frac{3-\sqrt{5}}{2} \leq p \leq 1$ and $m \geq 2$. Without loss of optimality, we assume that every good probing scheme (equivalently, probing decision trees) has the following properties:

1. The same probe will not occur more than once on the same path from the root to any of network states, although it may occur in many inner nodes of the probing decision tree.

2. Let t be a probe at an inner node ς. In the left subtree T^l_ς (corresponding to the probe syndrome $r_t = 0$), none of probes will be constituted by a subset of links from probe t and one more link that is not from probe t. In the right subtree



(a) Original Probing Scheme, Υ

(b) New Probing Scheme, Υ'

Figure A-2: Reducing an original probing scheme into a link-wise probing scheme: (a) the original probing scheme and (b) the new probing scheme.

T_ς^r (corresponding to the probe syndrome $r_t = 1$), none of probes will contain all the link from probe t.

3. A probe will not be performed if its syndrome can be inferred from previous probe syndromes.

Now consider an arbitrary probing scheme which satisfies the above properties and contains a probe whose length is more than 1. We will reduce it into the link-wise probing scheme so that the average number of probes for the link-wise probing scheme is less than that of the original probing scheme under the condition of $p \geq \frac{3-\sqrt{5}}{2}$.

Let ς be the inner node on the probing decision tree Υ such that the probe t at ς has a length of $l_t \geq 2$ and all other probes at the subtree T_ς has a length of 1. As shown in Fig. A-2(a), we denote the left subtree for $r_t = 0$ as I and the right subtree for $r_t = 1$ as II.

Let w denote one of the end links in probe t and the state of link w is unknown (otherwise, there is no reason to include link w in probe t). The new probing scheme Υ' is constructed as follows. Instead of performing probe t at the node ς, we perform the probe $t - \{w\}$. If the syndrome of probe $t - \{w\}$ is ONE, we know that the syndrome of probe t would be also ONE and we can continue the probing tree in the same manner as when when then syndrome of probe t is ONE in the old probing decision tree Υ. The additional information may enable us to infer the result of some probes; these will of course not be performed in the new probing scheme. This part of the subtree is denoted at II'.

If the syndrome of probe $t - \{w\}$ is ZERO, we perform probe $\{w\}$. If the syndrome of probe $\{w\}$. If the syndrome of probe $\{w\}$ is ZERO, we have exactly the same information as when the syndrome of probe t is ZERO in the old probing scheme, and we continue the probing tree in the same manner as in Υ. If the syndrome of probe $\{w\}$ is ONE, we again proceed as in the case of $r_t = 1$ in the old probing scheme, except that we skip all the tests whose syndromes can be inferred.

The reminder of the probing decision tree, i.e., everything that is not in the subtree T_ς, is left unchanged.

Next, we will calculate the difference between the average number of probes in the original probing scheme Υ and the new probing scheme Υ'. For any network state in the subtree T_ς, its probability is the product of three components: the probability $\Pr(A)$ of individual link states that have been determined before node ς, the probability $\Pr(w)$ of unknown link states of probe t, and the probability $\Pr(B)$ of individual link states that are yet to be determined after node ς except for links in probe t. Furthermore, these network states can be classified into 2^{l_t} subsets, and each subset has the same link states in probe t and different link states for other links.

From Fig. A-2, we have the following observations:

1. The probing depth of network states in subtree I is increased by 1, since we probe $t - \{w\}$ and $\{w\}$ in the new probing scheme, but we only probe t in the original probing scheme.

2. The probing depth of network states in subtree II" is reduced by $l_t - 2$. In the original probing scheme, we need $l_t - 1$ individual probes after node ς to identify the states of links in t, i.e., to probe the $l_t - 1$ fault-free links implies

that the last link fails. In the new probing scheme, only 1 probe is needed after node ς.

3. The probing depth of at least one subset of network states in subtree II' is reduced by 1. For example, the first $l_t - 2$ individual links of probe $t - \{w\}$ are fault-free implies that the last link fails. The probing depth of this subset of network states is reduced by 1.

In the following, we calculate the reduction of the average number of probes under two scenarios.

Case 1: $l_t = 2$

Without loss of generality, we assume that $t = w_1 w_2$. Using the above observations, we obtain

$$\begin{aligned} \bar{\mathcal{L}}_{\Upsilon} - \bar{\mathcal{L}}_{\Upsilon'} &= -\Pr(A)\Pr(B)(1-p)^2 + \Pr(A)\Pr(B)p(1-p) + \Pr(A)\Pr(B)p^2 \\ &= \Pr(A)\Pr(B)(-p^2 + 3p - 1). \end{aligned} \tag{A.6}$$

In order to make the reduction of the average number of probes positive, we have

$$-p^2 + 3p - 1 \geq 0 \Rightarrow \frac{3 - \sqrt{5}}{2} \leq p \leq 1. \tag{A.7}$$

Case 2: $l_t > 2$

In this case, the reduction of the average number of probes in subtree I is given by

$$\Delta\bar{\mathcal{L}}_I = -(1-p)^{l_t}\Pr(A)\Pr(B). \tag{A.8}$$

The reduction of the average number of probes in subtree II" is given by

$$\Delta\bar{\mathcal{L}}_{II"} = (1-p)^{l_t - 1} p \Pr(A)\Pr(B). \tag{A.9}$$

The reduction of the average number of probes in subtree II' is at least

$$\Delta\bar{\mathcal{L}}_{II'} = (1-p)^{l_t - 2} p^2 \Pr(A)\Pr(B). \tag{A.10}$$

Combining all these results, we obtain

$$\begin{aligned}\bar{\mathcal{L}}_{\Upsilon} - \bar{\mathcal{L}}_{\Upsilon'} &= \Delta\bar{\mathcal{L}}_{I} + \Delta\bar{\mathcal{L}}_{II'} + \Delta\bar{\mathcal{L}}_{II''} \\ &\geq (1-p)^{l_t-2}\Pr(A)\Pr(B)(-p^2+3p-1). \qquad (A.11)\end{aligned}$$

For $\frac{3-\sqrt{5}}{2} \leq p \leq 1$, we have $\bar{\mathcal{L}}_{\Upsilon} - \bar{\mathcal{L}}_{\Upsilon'} \geq 0$. Therefore, the average number of probes decreases by the modification.

The same procedure can continue until we end up with the link-wise probing scheme without increasing the average number of probes. Therefore, the link-wise probing scheme is optimal if $\frac{3-\sqrt{5}}{2} \leq p \leq 1$.

□

A.2 Proof of Theorem 2.1: Performance of Run-Length Probing Schemes

In this section, we prove the performance of run-length probing schemes by resorting their information-theoretic interpretations.

Lemma A.1. *For a large Eulerian network whose link failures are modeled as identical and independent Bernoulli random variables with parameter p, the average number of probes per edge required by the run-length probing scheme to fully identify the network state, denoted as $\bar{\mathcal{L}}_{RLPA}$, can be approximated by the code rate of its corresponding run-length code, i.e.,*

$$\bar{\mathcal{L}}_{RLPA} \approx p \cdot \left(\lfloor \log_2 K \rfloor + 1 + \frac{(1-p)^k}{1-(1-p)^K} \right) \triangleq \mathcal{R}(p), \qquad (A.12)$$

where $K = \lceil -\log_{1-p}(2-p) \rceil$ and $k = 2^{\lfloor \log_2 K \rfloor + 1} - K$.

Proof. For any prefix code, the average codeword length is defined as

$$\bar{L}_c = \sum_{z \in Z} \Pr(z) l_z, \qquad (A.13)$$

where Z is the symbol set of $\{0^i1\}_{i=0}^{\infty}$, and l_z is the codeword length for any prefix code. Gallager proved that the run-length code minimizes the average codeword length, i.e.,

$$\bar{L}_c^* = \sum_{z \in Z} \Pr(z) l_z^* = \min_{c \in C} \Pr(z) l_i, \tag{A.14}$$

where $\bar{L}_c^*$ is the average length of the run-length code, l_z^* is the codeword length for source alphabet z, and C is the set of all prefix codes for the geometrically distributed source.

For an Euler trail with m links, the average number of probes required by the run-length probing scheme is given by

$$\bar{\mathcal{L}}_{RLPA} = \sum_{z \in Z} l_z^* n_z, \tag{A.15}$$

where l_z^* is the run-length code for symbol z, and n_z is the number of occurrence of the network sub-state $z$in a typical network state s.

Note that, when $m \to \infty$, using the law of large number, we obtain

$$n_z \approx \Pr(z) \cdot \frac{m}{\bar{n}_s}, \tag{A.16}$$

where $\Pr(z)$ is the probability of substate z and $\bar{n}_s = \frac{1}{1-p}$ is the average number of pattern bits per source alphabet.

Substituting (A.16) into (A.15), we obtain the average number of probes per edge as

$$\lim_{m \to \infty} \bar{\mathcal{L}}_{RLPA} = \lim_{m \to \infty} \frac{\bar{\mathcal{L}}_{RLPA}}{m} \approx \frac{1}{\bar{n}_s} \sum_{z \in Z} l_z^* \Pr(z) = \frac{\bar{L}_c^*}{\bar{n}_s} = \mathcal{R}(p). \tag{A.17}$$

□

Appendix B

Non-Adaptive Fault Diagnosis Schemes

In this appendix, we present the proof of the correctness of testing algorithms for various networks[1].

B.1 Correctness of Testing Algorithm for 2-D Grid Networks

The correctness of Algorithm 3 can be established as follows.

- Suppose that the edge failure happens in column 1. This fact will be uncovered in Step 1a. The edges in all other columns and in all rows are intact, and therefore it is valid to use them for routing in Step 1b. It follows that Step 1b correctly performs the LTP on the edges of column 1 and identifies the edge failure.

- Suppose that the edge failure happens in row 1. A similar argument shows that Step 2 identifies the edge failure.

[1] The content in this chapter is based on the joint work with Nicholas J.A. Harvey, Mihai Patrascu and Sergey Yekhanin at CSAIL MIT., and has been published in [30].

- Suppose that the edge failure happens on the i^{th} edge in row $j \geq 2$. All column edges are intact, and can be used to route probes in Step 3a. It follows that Step 3a correctly performs the LTP on all rows and identifies the row containing the edge failure. The edges of row 1 are intact, and can be used for routing probes in Step 3b to identify the edge failure.

- Suppose that the edge failure happens on the ith edge in column $j \geq 2$. A similar argument shows that Step 4 identifies the edge.

B.2 Proof of Theorem 3.6 for Tree Topologies

For the proof, we fix an arbitrary root.

First consider the lower bound. The $\Omega(\log n)$ bound is from the CGT lower bound of (3.1).

We now show the upper bound of $\mathbf{O}(D \cdot \log n)$. The strategy works as follows. Starting from $d = 1$ and increasing d from 1 to D, we do the following two types of probes:

1. Probe the sub-tree containing the root and all nodes up to depth d. This constitute one probe.

2. Assuming that the failed edge is at level $d + 1$, use the sub-tree of depth d as a hub to test nodes at depth $d + 1$. The number of Type 2 probes for each d is up to $\log n$.

The diagnosis algorithm first looks at probes of Type 1, and determines the level at which the failure occurred. Then, it uses the probes of Type 2 made at the relevant level to identify the edge failure.

The number of probes for each d is $\mathbf{O}(\log n)$, and thus the total number of probe is $\mathbf{O}(D \cdot \log n)$.

Appendix C

Lightpath Diversity

C.1 Optimum Power Allocation Algorithm for Homogenous Lightpaths

For a M-connected source-destination pair, the power allocation vector is

$$\mathbf{P} = (P_1, P_2, \ldots, P_M)^T \in \mathbb{R}^+, \tag{C.1}$$

and the state vector is

$$\mathbf{F} = (F_1, F_2, \ldots, F_M)^T \tag{C.2}$$

with a probability distribution

$$\Pr(\mathbf{F}) = f^{\sum_{i=1}^{M} F_i}(1-f)^{M-\sum_{i=1}^{M} F_i}. \tag{C.3}$$

For the 'genie-aided' receiver, the overall error probability upper-bound is given by

$$PB_{GA} = \sum_{\mathbf{F}\in\{0,1\}^M} \Pr(\mathbf{F}) e^{-(\sqrt{\mathbf{F}^T(\mathbf{P}+\mathbf{N}_n)}-\sqrt{\mathbf{F}^T\mathbf{N}_n})^2}, \tag{C.4}$$

where $\mathbf{N}_n = (N_n, N_n, \ldots, N_n)^T$ is the noise power vector and $\{0,1\}^M$is the M-dimensional vector space over the $\{0,1\}$ field. To minimize the error probability, one can solve the following nonlinear programming problem,

$$\begin{aligned} \min \quad & \mathcal{H}(\mathbf{P}) = \sum_{\mathbf{F}\in\{0,1\}^M} \Pr(\mathbf{F}) e^{-(\sqrt{\mathbf{F}^T(\mathbf{P}+\mathbf{N}_n)}-\sqrt{\mathbf{F}^T\mathbf{N}_n})^2} \\ s.t. \quad & \mathbf{P}^T\mathbf{1} \le P_s, \end{aligned} \tag{C.5}$$

where $\mathbf{1} = (1, 1, \ldots, 1)^T$.

From the fact that, for each $\mathbf{F} = (F_1, F_2, \ldots, F_M)^T \in \{0,1\}^M$, the function

$$h(\mathbf{P}) = e^{-(\sqrt{\mathbf{F}^T(\mathbf{P}+\mathbf{N}_n)}-\sqrt{\mathbf{F}^T\mathbf{N}_n})^2} \tag{C.6}$$

is a convex function defined over a compact convex set

$$\{(P_1, P_2, \ldots, P_M) \in \mathbb{R}^+ : \sum_{i=1}^{M} P_i \le P_s\}, \tag{C.7}$$

one can conclude that the objective function $\mathcal{H}(\mathbf{P})$ is convex over the compact convex set. It follows that the minimization problem (C.5) has a unique solution due to the convex property.

From the Karush-Kuhn-Tuck Conditions [7], one can have

$$\nabla_{\mathbf{P}} L(\mathbf{P}, \mu) = 0, \tag{C.8}$$

where the Lagrange function is given by

$$L(\mathbf{P}, \mu) = \mathcal{H}(\mathbf{P}) - \mu(\mathbf{P}^T\mathbf{1} - P_s), \tag{C.9}$$

and μ is a Lagrange multiplier. It can be verified that the following power allocation vector

$$\mathbf{P} = (\frac{P_s}{M}, \frac{P_s}{M}, \ldots, \frac{P_s}{M})^T \tag{C.10}$$

satisfies the necessary condition of (C.8). It follows that (C.10) must be the unique minimizer of the objective function. This result indicates that the uniform power allocation algorithm is optimal under the assumption of homogenous lightpaths.

C.2 Optimum Number of Lightpaths Used for a Limited Amount of Transmitted Optical Energy

In this subsection, I will solve the nonlinear programming problem given by

$$\min_{M>0} G(M) = [f + (1-f)\exp\{-\psi(N_s, N_n, M)\}]^M. \tag{C.11}$$

From the Implicit Function Theorem [7], there exists a function $M^* = \varphi(N_s, N_n, f)$ such that $G(M)$ is minimized over the convex set $M : M \in \mathbb{R}^+$. One can find an approximation of the function $M^* = \varphi(N_s, N_n, f)$ as follows.

Let $a = f$ and $b(M) = (1-f)\exp\{-\psi(N_s, N_n, M)\}$. Note that $0 \leq a, b(M) \leq 1$ and $G(M) = (a + b(M))^M$. To a first order approximation, in the medium signal-to-noise photon rate ratio regime, the optimum number M^* of lightpaths can be approximated by the value of M where the curve a^M and the curve b^M meet, i.e.,

$$a^M = b(M)^M. \tag{C.12}$$

If $0 < f < 0.5$, (C.12) has a unique solution given by

$$M^* = \frac{n_s}{\ln(\frac{1}{f} - 1) + 2\sqrt{N_n}\sqrt{\ln(\frac{1}{f} - 1)}}. \tag{C.13}$$

This approximation is found to be very accurate when compared to a numerical search for M^*.

Intuitively, this derivation can be understood as follows. For each individual lightpath channel, there are two detrimental factors that degrade the error performance.

One is the noise, the other is the lightpath failure. If the lightpath works in high signal-to-noise photon rate ratio regime, the error due to the noise is dominated by the error due to the lightpath failure such that the error probability is floored by the failure probability f. The energy efficiency in this regime is very low but error probability is also low. On the other hand, if the lightpath works in low signal-to-noise photon rate ratio regime, the error due to the noise dominates the error due to the lightpath failure such that the error probability is on the order of 1. In this regime, the energy efficiency is high but the error probability is also high. As a trade-off between the energy efficiency and the error probability, the optimal operation point should be the point where both noise and failure contribute equally to the error probability, i.e., $f = (1-f)\exp\{-\psi(N_s, N_n, M)\}$. The optimal number of signal photons per lightpath follows from this observation.

C.3 MMSE Lightpath State Estimator for Optimum Receivers

In designing the optimal receiver, one needs to find the MMSE causal estimator of lightpath states. I will start by incorporating the following lemma in [56], which is crucial to the derivation of the MMSE causal lightpath state estimator.

Lemma C.1. *Estimation of random variables in Doubly Stochastic Point Processes For a Doubly Stochastic Point-Process $N(t) : t > t_0$ with a random arrival rate $\lambda(t, \mathbf{x},$ where $\mathbf{x}$ is a time-independent random vector, let $\mathbf{a}_t(\mathbf{x})$ be a time-dependent vector-value function of the random vector $\mathbf{x}$ and such that $E(|\mathbf{a_t}(\mathbf{x})^2) < \infty$. Then, for a recorded time statistic $\mathbf{t} = (t_1, t_2, \ldots, t_n)$, the MMSE causal estimate of the function $\mathbf{a}_t(\mathbf{x})$ of $\mathbf{x}$ is the conditional mean $\hat{\mathbf{a}}_t$, given by*

$$\hat{\mathbf{a}}_t = E[\mathbf{a}_t(\mathbf{x})] = \frac{E[\mathbf{a}_t(\mathbf{x})\exp\{\mathbf{A}_t(\mathbf{x})\}]}{E[\exp\{\mathbf{A}_t(\mathbf{x})\}]}, \tag{C.14}$$

where $\mathbf{A}_t(\mathbf{x}) = -\int_{t_0}^{t} \lambda(\tau, \mathbf{x})d\tau + \int_{t_0}^{t} \ln \lambda(\tau, \mathbf{x})dN_\tau$.

For simplicity, the subscript i is suppressed in the following derivation. Due to the random channel model, the arrival rate of the photo-event process at the output of each detector, $\lambda(t, F) = F\lambda(t) + \lambda_n$, is a random variable. In particular, F is a Bernoulli random variable with the probability density function $p_F(x) = f\delta(x) + (1 - f)\delta(x-1)$.

Using (C.14), the MMSE causal estimator of the channel state F is given by

$$\hat{F}(t) = E[F|\mathbf{t}] = \frac{E[F\exp\{A_t(F)\}]}{E[\exp\{A_t(F)\}]}, \tag{C.15}$$

where $A_t(F) = -\int_{t_0}^{t} \lambda(\tau, F)d\tau + \int_{t_0}^{t} \ln \lambda(\tau, F)dN_\tau$.

For hypothesis H_0, the photo-event rate is

$$\lambda_i^{(0)}(t) = \begin{cases} \frac{F_i\lambda_s}{M} + \lambda_n, & 0 \le t \le \frac{T}{2} \\ \lambda_n, & \frac{T}{2} \le t \le T \end{cases} \tag{C.16}$$

Substituting (C.16) into (C.15), the MMSE causal estimator of the channel state F turns out to be

$$\hat{F}^{(0)}(t) = \frac{1}{1 + \frac{f}{1-f}\exp(\frac{\lambda_s}{M}t)(1+\Omega)^{-N_t}}, t \in [0, \frac{T}{2}], \tag{C.17}$$

where N_t is the number of photo-events over $[0, t]$.

For hypothesis H_1, the photo-event rate is

$$\lambda_i^{(1)}(t) = \begin{cases} \lambda_n, & 0 \le t \le \frac{T}{2} \\ \frac{F_i\lambda_s}{M} + \lambda_n, & \frac{T}{2} \le t \le T \end{cases} \tag{C.18}$$

Substituting (C.18) into (C.15), the MMSE causal estimator of the channel state F turns out to be

$$\hat{F}^{(1)}(t) = \frac{1}{1 + \frac{f}{1-f}e^{\frac{\lambda_s}{M}(t-\frac{T}{2})}(1+\Omega)^{-(N_t - N_{T/2})}}, t \in [\frac{T}{2}, T], \tag{C.19}$$

where N_t is the photo-event count over $[0, t]$, and $N_{T/2}$ is the number photo-event count over $[0, T/2]$ of the same sample function of photo-event process.

C.4 Chernoff Bound of the Symbol Error Probability for the Receiver with Non-causal Lightpath State Estimator

The suboptimal receiver makes hard-decisions on estimated lightpath states from causal state estimators at time $t = T$. The non-causal hard-decision rule is given by

$$\tilde{F} = \begin{cases} 0, & \text{if} \quad \hat{F}(T) \leq 0.5 \\ 1, & \text{if} \quad \hat{F}(T) > 0.5 \end{cases} \tag{C.20}$$

where $\hat{F}(T)$ is the MMSE causal estimate of the lightpath state at time $t = T$. If $\tilde{F} = 0$, the receiver estimates the lightpath to be DOWN and thus discards the received signal over that lightpath. Otherwise, the receiver estimates the lightpath to be UP and thus uses the received optical signal over that lightpath for optimal combining and symbol decisions.

With hard-decision lightpath states, the symbol decision rule is given by

$$\sum_{i=1}^{m} k_{i1} \underset{\hat{H} = H_1}{\overset{\hat{H} = H_0}{\gtrless}} \sum_{i=1}^{m} k_{i2}, \tag{C.21}$$

where m is the number of lightpaths that are estimated to be UP during the symbol transmission. Note that m is a binomial random variable with a probability distribution function, $\Pr(m) = \binom{M}{m}(1-g)^m g^{M-m}$, where $g = \Pr(\hat{F}(T) \leq 1/2)$ is the probability with which the lightpath is estimated to be DOWN during the symbol transmission. For both hypotheses, the channel state estimator has the form,

$$\hat{F} = [1 + \frac{f}{1-f} \exp(\frac{N_s}{M})(1+\Omega)^{-N}]^{-1}. \tag{C.22}$$

The probability distribution function of the photon count is

$$\Pr(N=k) = f\frac{(N_n)^k}{k!}e^{-N_n} + (1-f)\frac{(\frac{N_s}{M}+N_n)^k}{k!}e^{-(\frac{N_s}{M}+N_n)}. \tag{C.23}$$

Combining (C.22) and (C.23), the probability with which the lightpath is estimated to be DOWN is given by

$$\begin{aligned} g &= \Pr(\hat{F} \leq 0.5) \\ &= \Pr(N \leq \frac{\frac{N_s}{M} + \ln(f) - \ln(1-f)}{\ln(1+\Omega)} \triangleq N_{TH}) \\ &= \sum_{k=0}^{N_{TH}} \Pr(N=k). \end{aligned} \tag{C.24}$$

To calculate the error bound, one can first calculate the error probability conditioned on the number of lightpaths estimated to be UP during the symbol time. For given m, the conditional error probability is defined as

$$\begin{aligned} \Pr(\varepsilon|m) &= p_0 \Pr[\sum_{i=1}^{M} k_{i1} \leq \sum_{i=1}^{M} k_{i2}|H_0, m] + p_1 \Pr[\sum_{i=1}^{m} k_{i1} \geq \sum_{i=1}^{M} k_{i2}|H_1, m] \\ &= \Pr[\sum_{i=1}^{M} k_{i1} \leq \sum_{i=1}^{M} k_{i2}|H_0, m] \end{aligned} \tag{C.25}$$

where the second equality is due to the symmetry of BPPM. Using the Chernoff Bound, the right hand side of (C.25) is bounded by

$$\begin{aligned} \Pr[\sum_{i=1}^{M} k_{i1} \leq \sum_{i=1}^{M} k_{i2}|H_0, m] &\leq \min_{s>0}\{e^{mN_n(e^s-1)}e^{m(\frac{N_s}{M}+N_n)(e^{-s}-1)}\} \\ &= \exp\{-m\psi(N_s, N_n, M)\} \end{aligned} \tag{C.26}$$

where $\psi(N_s, N_n, M) = (\sqrt{\frac{N_s}{M} + N_n} - \sqrt{N_n})^2$.

Using (C.24) and (C.26), the error bound of the hard-decision receiver is obtained by averaging (C.26) over all possible m , that is,

$$\begin{aligned}
\Pr(\varepsilon) &= \sum_{m=0}^{M} \Pr(\varepsilon|m)\Pr m \\
&\leq \sum_{m=0}^{M} e^{-m\psi(N_s,N_n,M)} \binom{M}{m} (1-g)^m g^{M-m} \\
&= \left[g + (1-g)e^{-\psi(N_s,N_n,M)}\right]^M
\end{aligned} \tag{C.27}$$

Note that (C.27) is also an upper bound for the optimal receiver.

Bibliography

[1] R. Ahlswede and I. Wgener. *Search Problems.* New York: Wiley, 1987.

[2] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows.* Prentice Hall, 1993.

[3] E. Athanasopoulou and C. N. Hadjicostis. Realistic approaches to fault detection in networked discrete event systems. *IEEE Transactioins on Neural Networks*, 16(5):1042–1052, 2005.

[4] I. Bar-David. Communication under the poisson regime. *IEEE Transactions on Information Theory*, IT-15(1):31–37, 1969.

[5] A. Bar-Noy, F. K. Hwang, I. Kessler, and S. Kutten. A new competitive algorithm for group testing. *Discrete Applied Mathematics*, 52(1):29–38, 1994.

[6] M. Barborak, M. Malek, and A. Dahbura. The consensus problem in fault-tolerant computing. *ACM Computing Surveys*, 25(2):171–220, 1993.

[7] D. P. Bertsekas. *Nonlinear Programming, 2nd edition.* Belmond, MA: Athena Scientific, 1999.

[8] D. P. Bertsekas. *Dynamic Programming and Optimal Control.* Belmond, MA: Athena Scientific, 2000.

[9] D. P. Bertsekas and J. N. Tsitsiklis. *Introduction to Probability.* Belmond, MA: Athena Scientific, 2002.

[10] B. Bollobas. *Moder Graph Theory.* New York: Springer-Verlag, 1998.

[11] V. W. S. Chan. Coding for the turbulent atmospheric optical channels. *IEEE Transactions on communications*, COM-30(1):269–275, 1982.

[12] V. W. S. Chan. All-optical networks. *Scientific American*, 273(3):56–59, 1995.

[13] V. W. S. Chan and A. H. Chan. Reliable message delivery via unreliable networks. In *Proc. IEEE International Symposium on Information Theory (ISIT)*, 1997.

[14] Vincent W. S. Chan. Robust architectures for multi-service, multi-level-reliability, multi-level-security and multi-priority wdm local area networks, 2003.

[15] C. J. Chang-Hasnian. Tunable vscel. *IEEE Journal on Selected Topics in Quantum Electronics*, 6(6):978–987, November/December 2000.

[16] S. Chaudhuri, G. Hjalmtysson, and J. Yates. Contole of lightpaths in an optical network. In *Optical Internetworking Forum OIF 2000.04, IETF Internet Draft*, 2000.

[17] C. J. Colbourn, J. H. Dinitz, and D. R. Stinson. Applications of combinatorial designs to communications, cryptography and networking. In *Surveys in Combinatorics, 1993, Walker(Ed.), London Mathematical Society Lecture Note Series 187, Cambridge University Press*, 1999.

[18] T. M. Cover and J. A. Thomas. *Elements of Information Theory.* New York: Wiley, 1991.

[19] E. B. Desurvire. Capacity demand and technology challenges for lightwave systems in the next two decades. *IEEE Journal of Lightwave Technology*, 24(12):4697–4710, 2006.

[20] D. Du and F. Hwang. *Combinatorial Group Testing and Its Applications.* World Scientific, 2000.

[21] A. G. D'yachkov and V. V. Rykov. Bounds for the length of disjunctive codes. *Problems of Information Transmission*, 18(3):7–13, 1982.

[22] A. G. D'yachkov and V. V. Rykov. A survey of superimposed code theory. *Problems of Control and Information Theory*, 12(4):229–242, 1983.

[23] A. G. D'yachkov, V. V. Rykov, and A. M. Rashad. Superimposed distance codes. *Problems of Control and Information Theory*, 18(4):237–250, 1989.

[24] D. C. Kilper et al. Optical performance monitoring. *IEEE Jounal of Lightwave Technology*, 22(1):294–304, 2004.

[25] V. W.S. Chan et al. A precompetitive consortium on wide-band all-optical networks. *IEEE Journal of Lightwave Technology*, 11(5/6):714–735, 1993.

[26] ITU-T Recommendation G.874. Management aspects of the optical transport network element, 2001.

[27] R. Gallager and D. Van Voorhis. Optimal source codes for geometrically distributed integer alphabets. *IEEE Transactions on Information Theory*, IT-21(2):228–230, 1975.

[28] S. W. Golomb. Run-length encodings. *IEEE Transactions on Information Theory*, 1966.

[29] C. K. Guan and V. W. S. Chan. Topology design of oxc-switched wdm networks. *IEEE Journal of Selected Areas in Communications*, 23(8):1670–1686, 2005.

[30] N. Harvey, M. Patrascu, Y. G. Wen, S. Yekhanin, and V. W. S. Chan. Non-adaptive fault diagnosis for all-optical networks via combinatorial group testing on graphs. In *Proc. IEEE Conference on Computer Communications (INFOCOM)*, 2007.

[31] C. W. Helstrom and R. S. Kennedy. Noncommuting observables in quantum detection and estimation theory. *IEEE Transactions on Information Theory*, IT-20(1):16–24, 1974.

[32] T. Ho, B. Leong, Y. Chang, Y. Wen, and R. Koetter. monitoring in multicast networks using network coding. In *Proceedings of International Symposium on Information Theory (ISIT)*, 2005.

[33] T. Ho, M. Medard, and R. Koetter. An information-theoretic view of network management. In *Proc. 22nd Annual IEEE Conference on Computer Communications (INFOCOM)*, 2003.

[34] ITU. *Architecture of Optical Transport Networks*, itu-t recommendation g.872 edition, 1999.

[35] S. Karp and J. R. Clark. Photon counting: A problem in classical noise theory. *IEEE Transactions on Information Theory*, IT-16(11):672–680, 1970.

[36] W. H. Kautz and R. C. Singleton. Nonrandom binary superimposed codes. *IEEE Transaction on Information Theory*, 4(10):363–377, 1964.

[37] W. J. Lai, P. Shum, and L. N. Binh. Nolm-nalm fiber ring laser. *IEEE Journal of Quantum Electronics*, 41(7):986–993, 2005.

[38] S. Lee and K. G. Shun. Probabilistic diagnosis of multiprocessor systems. *ACM Computing Surveys*, 26(1):121–139, 1994.

[39] F. T. Leighton. *Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes.* Morgan-Kaufman, 1991.

[40] L. Lewis. *Service Level Management for Enterprise Networks.* Norwood, MA: Artech, 1999.

[41] C. S. Li and R. Ramaswami. Automatic fault detection, isolation, and recovery in tranparent all-optical networks. *Journal of Lightwave Technology*, 15(10):1784–1793, 1997.

[42] C. St. J. A. Nash-Williams. Edge-disjoint spanning trees of finite graphs. *Journal of the London Mathematical Society*, 36:445–450, 1961.

[43] H.Q. Ngo and D-Z Du. A survey on combinatorial group testing algorithms with applications to dna library screening. In P.M. D-Z. Du, P.M. Pardalos and J. J. Wang, editors, *Discrete Mathematical Problems with Medical Applications*, number 55 in DIMACS Series. American Mathematical Society, 2000.

[44] M. J. O'Mahony, C. Politi, D. Klonidis, R. Nejabati, and D. Simenidou. Future optical networks. *IEEE Journal of Lightwave Technology*, 24(12):4684–4696, 2006.

[45] S. Paikh. On the use of erasure codes in unreliable data network. M.sc. thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, 2001.

[46] K. R. Pattipati and M. G. Alexandridis. Application of heuristic search and information theorty to sequential fault diagnosis. *IEEE Transactions on Systems, Man and Cybernet*, 20(4):872–886, 1990.

[47] H. J. Payne and W. S. Meisel. An algorithm for constructing optimal binary decision trees. *IEEE Transactions on Computers*, C-26(9):905–916, 1977.

[48] F.P. Preparata, G. Metze, and R. T. Chien. On the connection assignment problem of diagnosable systems. *IEEE Transactions on Electronics Computation*, 16(6):848–854, 1967.

[49] V. Raghavan and A. Tripathi. Sequential diagnosability is co-np complete. *IEEE Transactions on Computers*, 40(5):584–595, May 1991.

[50] R. Ramaswami and K. N. Sivarajan. *Optical Networks: A Practical Perspective.* CA: Morgan Kaufmann, 2002.

[51] C. Rodriguez, S. Rementeria, J. I. Martin, A. Lafuente, J. Muguerza, and J. Perez. A modular neural network approach to fault diagnosis. *IEEE Transactioins on Neural Networks*, 7(2):326–340, 1996.

[52] G. Rossi, T. E. Dimmick, and D. J. Blumenthal. Optical performance monitoring in reconfigurable wdm optical networks using subcarrier multiplexing. *IEEE Jounal of Lightwave Technology*, 18(12):1639–1648, 2000.

[53] Jr. S. J. Dolinar. *A Class of Optical Receivers using Optical Feedback.* PhD dissertation, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, 1976.

[54] S. R. Safavian and D. Landgerebe. A survey of decision tree classifier methodology. *IEEE Transactions on Systems, Man and Cybernet*, 21(3):660–674, 1991.

[55] A. Saleh. Dynamic multi-terabit core optical networks: architecture, protocols, control and management (coronet), 2006.

[56] D. L. Snyder. *Random Point Processes.* New York: Wiley, 1975.

[57] A. K. Somani, V. K. Agarwal, and D. Avis. A generalized theory for system-level diagnosis. *IEEE Transactions on Computers*, C-36(5):538–546, May 1987.

[58] Internet World Statistics. http://www.internetworldstats.com/.

[59] M. Subramanian. *Network Management: Principles and Practice, 1st edition.* Addison-Welsey Lognman, 2000.

[60] H. Tanaka and A. Leon-Garcia. Efficient run-length encodings. *IEEE Transactions on Information Theory*, IT-28(6):880–890, 1982.

[61] H. L. Van Trees. *Detection, Estimation and Modulation Theory: Part I.* New York: Wiley, 1968.

[62] W. T. Tutte. On the problem of decomposing a graph into n connected factors. *Journal of the London Mathematical Society*, 36:221–330, 1961.

[63] P. Ungar. The cut-off point for group testing. *Communications of Pure and Applied mathematics*, 13:49–54, 1960.

[64] G. E. Weichenberg. High-reliability architectures for networks under stress. M.sc. thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, 2003.

[65] G. E. Weichenberg, V. W. S. Chan, and M. Medard. A reliable architecture for networks under stress. In *Proc. 4th Interntional Workshop on Design of Reliable Communication Networks*, 2003.

[66] Y. G. Wen and V. W. S. Chan. Ultra-reliable communication over vulerable all-optical networks via lightpath diversity. *IEEE Journal of Selected Areas in Communications*, 23(8):1572–1587, 2005.

[67] Y. G. Wen, V. W. S. Chan, and E. Swanson. Cost-efficient transmitter/receiver deployment for proactive fault diagnosis in all-optical networks. In *Submit to IEEE International Conference on Communications (ICC)*, 2008.

[68] Y. G. Wen, V. W. S. Chan, and L. Z. Zheng. Efficient fault diagnosis algorithms for all-optical wdm networks with probabilistic link failures (invited paper). *IEEE Journal of Lightwave Technology*, 23(10):3358–3371, 2005.

[69] Y. G. Wen, V. W. S. Chan, and L. Z. Zheng. Efficient fault detection and localization for all-optical networks. In *Proc. IEEE Global Communications Conference (GLOBECOM)*, 2006.

[70] Y. G. Wen, V. W. S. Chan, and L. Z. Zheng. Efficient fault diagnosis algorithms for all-optical networks: An information theorectical approach. In *Proc. IEEE International Symposium on Information Theory (ISIT)*, 2006.

[71] D. B. West. *Introduction to Graph Theory.* Pearson Education, second edition, 2001.

[72] J. K. Wolf. Born again group testing: Multi-access communications. *IEEE Transactions on Information Theory*, IT-31(2):185–191, 1985.

[73] E. Wong. A linear search problem. *SIAM Reviews*, 6(2):168–174, 1964.

[74] W. G. Yang. Sensitivity issues of optical performance monitoring. *IEEE Photonics Technology Letters*, 14(1):107–109, 2002.

[75] R. W. Yeung. *A First Course in Information Theory.* New York: Kluwer/Plenum, 2002.

[76] D.Y. Zhou and S. Subramaniam. Survivability in optical networks. *IEEE Network*, 2000.