Title
Survey of unified approaches to integrated-service networks

Authors
Bradley, Tracy T.
Suda, Tatsuya

Publication Date
1989

Peer reviewed

SURVEY OF UNIFIED APPROACHES TO INTEGRATED-SERVICE NETWORKS

Technical Report No. 89-16

Tracy T. Bradley
Tatsuya Suda

Department of Information and Computer Science
University of California, Irvine
Irvine, CA 92717
Phone: (714) 856-5474
Fax: (714) 856-4056

*This material is based upon work supported by the National Science Foundation under Grant No. DCI-8602052. This research is also in part supported by University of California MICRO program.

Abstract

The increasing demand for communication services, coupled with recent technological advances in communication media and switching techniques, has resulted in a proliferation of new and expanded services. Currently, networks are needed which can transmit voice, data, and video services in an application–independent fashion. Unified approaches employ a single switching technique across the entire network bandwidth, thus, allowing services to be switched in an application–independent manner. This paper presents a taxonomy of integrated-service networks including a look at N-ISDN, while focusing on unified approaches to integrated–service networks.

The two most promising unified approaches are burst and fast packet switching. Burst switching is a circuit switching–based approach which allocates channel bandwidth to a connection only during the transmission of "bursts" of information. Fast packet switching is a packet switching–based approach which can be characterized by very high transmission rates on network links and simple, hardwired protocols which match the rapid channel speed of the network. Both approaches are being proposed as possible implementations for integrated–service networks. We survey these two approaches, and also examine the key performance issues found in fast packet switching. We then present the results of a simulation study of a fast packet switching network.

1. Introduction

The arrival of the Information Age has intensified the demand for communication services of all kinds. Applications for voice and data, as well as video, are rapidly expanding. The current communication network approach has been to develop application–specific services. It is not uncommon to find a single user subscribing to separate networks for telephony, cable television, and computer data communication. Although this redundancy in connections has been reasonably successful in satisfying customer demands, the expected future proliferation of new services will make this approach financially and practically unfeasible. Thus, an integrated communications system which does not require redundant resources is needed. Future communication techniques must be able to facilitate a wide variety of diversified services in a practical and easily expandable fashion.

Recently, a great deal of attention has focused on the narrowband integrated services digital network (N-ISDN) as a solution to the service integration problem. The design of N-ISDN has resulted in a standard for a digital line interface composed of two 64 Kbit/s circuit switched channels for voice and bulk data traffic, and a 16 Kbit/s packet switched channel for data and control information [STA85,TUR86]. While this design provides transmission capabilities for a variety of currently used services, not all services supported are switched in the same fashion. Thus, only at the level of physical packaging does any true integration exist, and, even at this level, two separate classes of channels are specified. The proposed N-ISDN is simply a new way of packaging current network technologies. Separate switching technologies are still required for the circuit switched and packet switched channels, and little thought has been given to the problem of providing higher rate services such as video [TUR86].

Most proposals for the implementation of integrated-service networks use some type of hybrid switching scheme. Varying applications are accommodated through different switching schemes which divide up the available bandwidth. For example, voice applications may be transmitted via circuit switching, while data applications use packet switching. Hybrid networks of current application–specific technologies will not be satisfactory in meeting growing communication demands. Their application–specific design limits easy expansion to accommodate new services. They may also increase the cost of network administration, design, and transmission by limiting the ability to share network resources. The proposed N-ISDN standard, as well as other hybrid schemes, may provide a temporary solution, but its implementation will be only a postponement of the true problem: application–independent integration of services. Through application–independent networks, information transport can be provided as a basic network service without regard to how the service is used [KUL84]. New applications can then be easily facilitated by using the network as simply an information transport service. Thus, the next generation of communication systems must go beyond a simple repackaging of current communication technologies to application–independence.

Unified approaches to the integrated service switching problem employ a single (as opposed to hybrid) switching technique across the entire network bandwidth. As a result, the network services provided can be implemented in an application–independent manner. Proposals for unified integrated service networks seek to facilitate a broad range of services while remaining easily expandable. Since all applications are switched in the same fashion, issues such as the division of bandwidth between various switching techniques and the implementation of new switching techniques for additional services are not a concern. Instead a new set of issues is raised. To evaluate a proposed unified approach, whether the approach can easily and effectively handle a broad range of services (including voice, data, video, and other services only in the proposal stages) and can adapt to different transmission technologies must be considered.

Unified approaches to integrated service networks can be categorized into two broad groups: circuit switching-based and packet switching-based approaches (Figure 1.1). Traditional circuit switching, as in the current telephone network, is limited because it is not capable of satisfying the wide range of bandwidth sizes required to support varying services such as spoken voice and video. It is also inefficient for switching bursty, non-continuous traffic since circuit switching establishes a reserved communication path for the duration of the connection reguardless of whether information is actually being transmitted. A relatively new circuit switching-based approach called *burst switching* has been proposed to overcome the disadvantages of traditional circuit switching. As in circuit switching, users in burst switching request a fixed channel capacity which is a multiple of some basic rate. But unlike circuit switching, burst switching does not allocate the requested channel bandwidth to a connection until data information is ready for transmission; bandwidth is allocated only for a burst of information. When the channel bandwidth is no longer immediately required, the channel capacity is released for use by other connections. Thus, voice spurts or data bursts are circuit switched.

This technique is also referred to as *fast circuit switching.*

In traditional packet switching, the complicated, node-to-node protocols required to ensure error-free packet delivery can result in a large delay and variance in the packet transmission time incurred across a network. For some services, notably telephony, large delays can be intolerable, causing unacceptable degradation of the service quality. To compensate for these limitations in the traditional packet switching approach to the integrated service problem, *fast packet switching* (or *Asynchronous Transfer Mode* (ATM)) has been suggested. Fast packet switching is characterized by very high transmission rates on network links, and a simple, hardwired node-to-node protocol which matches the rapid channel speed of the network. Short packets of information are carried through the network to their destinations, with routing information stored in the header of each packet. Packet delays across the network are reduced due to the simplified protocol, and retransmission of erroneous packets is handled by the higher level, end-to-end protocol.

In this paper we categorize and discuss the major unified approaches to integrated service networks. The evolution of the N-ISDN concept, in addition to the emerging view of N-ISDN, is discussed in Section 2. Section 3 of the paper discusses circuit switching-based proposals with a primary focus on the burst switching approach. In Section 4 packet switching-based proposals are discussed, and fast packet switching is examined in detail. Section 5 discusses key performance issues for the two most promising unified approaches: burst and fast packet switching. In section 6, simulation results for the performance of a fast packet switching network are presented.

2. Narrowband Integrated Service Digital Networks (N-ISDN)

By the end of the 1990's, it is expected that a multiplicity of communication services will be offered by the narrowband integrated services digital network (N-ISDN). This network will provide services through the use of digital connections, and will support a wide range of telecommunication media such as voice, data, and video. N-ISDN planning and research is taking place at an international level with the aim of achieving a worldwide public telecommunications network. The implementation of N-ISDNs may differ from country-to-country and network-to-network, but through standardized interfaces users will conceptually have access to a single international N-ISDN [STA85]. The first company in the United States to begin N-ISDN service, Illinois Bell, began its new N-ISDN service began on a limited basis in the third quarter of 1988. Thus, the immense scale, scope, and timeliness of this project, as well as its expected impacts on network implementors and users, makes N-ISDN design of primary importance in the telecommunications field.

Present proposals for N-ISDN networks are tied to current network technologies and applications. In this section, we will examine the advantages and limitations of such approaches, with the aim of providing motivation for the alternative unified approach advocated by this paper.

2.1 The N-ISDN Concept

The recent implementation of digital networks for telephony and data transmission has narrowed the gap between computer and voice communications. With the advent of digital transmission and switching technologies, the same digital techniques can be used for different media such as data, voice, and video. Thus, it is possible to integrate telephony and digital data communication into the same network. Currently, telephone networks are incorporating these technologies to evolve into integrated digital networks (IDNs). In IDNs, digital transmission and switching technologies are used. These networks are based on PCM (64 Kbit/s) encoding and digital switching at the same rate. The change to IDNs, in addition to the increased demands of users for communication services, has spurred the concept of a single network, supporting digitally transmitted services of all kinds, the N-ISDN.

The N-ISDN will support a variety of user services. These can be divided into four main groups: voice, digital data, text, and image. Voice services consist of telephone, voice information retrieval, and music, and are characterized by strict real-time delay requirements while tolerating some amount of loss. Digital data communication requires error-free transmission with delay requirements determined by application. Typically, digital data is not as sensitive to delays as voice service. Examples of digital data communication include interactive terminal-to-computer communication, telemetry, electronic mail, and high-speed computer communication. Telex, teletex, information retrieval, and some electronic mail systems fall into the text category. Text communication, like digital data communication, requires error-free transmission, but it can tolerate even greater delays than digital data services. Image services—including facsimile, video conferencing, videophone, and cable TV distribution—do not require error-free transmission. The range of delays they allow depends on the particular service specified. For example, facsimile can tolerate large delays while videophone service requires real-time delivery. Most of the services falling into these four groups can be transmitted at a rate of less than 64 Kbits/s with satisfactory performance; however, some services such as music, high-speed computer communication, video conferencing, videophone, and cable TV will require significantly larger bandwidths. [STA85]

One of the key components of the N-ISDN will be its user interface. Transparent transmission of services is desired so that users can view the N-ISDN as a "digital pipe"[STA85]. Through this user interface, users can gain access to the pipe, and telecommunications services—each perhaps requiring different bandwidths—will be supported. Standards for the design and implementation of the N-ISDN must be provided to ensure an effective user interface. The controlling body for these standards is the International Telecommunications Union's (ITU) International Consultive Committee for Telegraph and Telephone (CCITT). Standards are currently being developed for signaling, user interface, and network protocols.

2.2 The Emerging N-ISDN—Hybrid Approaches

A primary design goal, established by the CCITT, is the evolution of the N-ISDN from existing telephone networks and IDNs. Thus, the CCITT has specified a narrowband N-ISDN with 64 Kbit/s switched digital connections based on the currently evolving IDN. The basic user interface structure for N-ISDN is a combination of two different types of channels: B and D channels. This user interface will provide $2B + D$ service. The first component of the basic user interface will be two full-duplex 64 Kbit/s B channels. The B channels can carry PCM encoded voice, digital data for circuit or bulk data switching, or multiple lower-rate traffic streams going to the same endpoint. B channels are considered the primary user channels. The second component of the basic user interface is a full-duplex 16 Kbits/s D channel. The D channel will support lower-speed digital traffic, as well as control information between the user and the network. The B channels will be circuit switched, using the D channel to transmit control information. The primary function of the B channels is to support PCM encoded voice, but they will also be used to carry bulk data traffic. Traditional packet switching will be performed on the B channel for lower-speed interactive data. In addition, 48 Kbit/s are added to the basic user interface structure for framing, synchronization, and other overhead, giving a total channel rate of 192 Kbits/s [STA85, LUE86].

The CCITT also defines two primary rate user interfaces: 1544 Kbits/s and 2048 Kbits/s. These interfaces are designed to satisfy the needs of users with larger bandwidth requirements. Both types of these primary user interfaces are divided up into channels.

B and D channels, as well as H channels for circuit or packet switched data transmission, are used. The H channels can be further defined as 384 ($H0$), 1536 ($H11$), or 1920 ($H12$) Kbits/s channels. The CCITT recommendations define the way these channels can be packaged to form primary rate interfaces. Channel combinations such as $23B + D$, $30B + D$, $5H0 + D$, $2H0 + 11B +$ D, and $H11$ are specified [DEC86].

The necessary network architectures required to provide these proposed user interface services are not specified by the CCITT, but, due to the nature of the proposed user interfaces, the ability to handle both circuit-switched voice, and circuit and packet switched data is implicitly assumed. Thus, network architectures which meet the CCITT N-ISDN proposal can be termed "hybrid" networks since they must include both circuit switched and packet switched channels. For example, the N-ISDN implementation approach being taken in Southern Bell's BellSouth region of the United States is to implement three N-ISDN switching nodes supporting both basic and primary-rate user interface services. These nodes will use an AT&T 5ESS digital switch and two Northern Telecom DMS-100 digital switches to provide circuit switching. Packet-switching service will be provided for the N-ISDN network via a Northern Telecom SL-10 packet switch [MIT86]. Another example of the hybrid design of an N-ISDN network is the RENAN project, an experimental N-ISDN project implemented in France. The RENAN project will be connected to the current French telephone network via 64 Kbit/s circuits to support circuit switched voice. In addition, the Telecom 1 digital satellite will be available for digital connections, and through existing gateways or dedicated internetworking units, packet-switching services will be provided by the Transpac packet-switching network [TRO86]. Both of these typical N-ISDN implementations—the BellSouth region and RENAN—meet the recommendations of the CCITT by creating a hybrid network with separate channels for circuit and packet switching.

Although it is the currently accepted approach, designing an N-ISDN in a hybrid fashion has a number of limitations. First, a division of the 192 Kbit/s basic user interface bandwidth was made to create circuit switched and packet switched channels. The two circuit switched channels were chosen to be 64 Kbit/s each, based on the PCM voice encoding scheme.[STA85] However, since that decision was made, advances in technology have provided acceptable voice transmission schemes of 32 Kbit/s or less. Thus, the N-ISDN proposal is locked into a primary channel rate based on an obsolete voice encoding speed; the proposal is based on an application-specific technology. A second limitation of the hybrid N-ISDN proposal is the complexity it requires of the user network interface. In order to allow users to view the N-ISDN as a digital pipe, the user interface must be intelligent enough to recognize the service desired by the user and to choose the transmission scheme best suited to that service. For example, if the user sends digitized voice to the digital pipe, the user interface mechanism must use one of the 64 Kbit/s circuit switched B channels for transmission. Similarly, if the user is sending low-speed interactive data to the digital pipe, the information should be packet switched via the 16 Kbit/s D channel. Another possible limitation resulting from the hybrid proposal is the difficulty in expanding a hybrid switching scheme which is application-dependent to handle new higher-rate user services. Services based on a 64 Kbit/s channel can be easily accommodated by the CCITT N-ISDN proposal, but services requiring higher bandwidth, such as cable TV or video conferencing, will require more complex switching arrangements.

Unified, or non-hybrid, approaches to N-ISDN design can overcome these application-specific limitations. In unified approaches, a single switching scheme is employed across the entire network, thus all services are switched in the same fashion. This eliminates any division of the available bandwidth by switching techniques, as well as reducing the complexity the user interface required. Another advantage of unified approaches is that expansion to accommodate higher rate services depends only upon the bandwidth available to the network. Thus, the remainder of this paper concentrates on proposed unified approaches to application-independent integrated service networks.

3. Circuit Switching Approaches

Many telecommunication services rely on circuit switching techniques. Telephone networks worldwide utilize circuit switching for the transmission of voice information, and data information is also commonly transmitted via circuit switching [KUM80]. However, despite its widespread usage, circuit switching is inadequate to satisfy the varying demands of an integrated services digital network. Circuit switching is not easily able to support services of widely varying bit rates, nor can it adapt to the burstiness characterizing many forms of information.

3.1 Multi-Rate Circuit Switching (MRCS)

In circuit switching, before information transfer begins, a physical circuit is established between source and destination users by sending a control packet. The circuit is established, end-to-end, for the duration of the call, and the reserved bandwidth cannot be shared with other calls. Usually, a single transmission rate exists for all circuit switched calls. This fixed rate severely limits the types of services that can be supported by circuit switching. To facilitate a greater range of services, Multi-Rate Circuit Switching (MRCS) has been suggested [KUL84]. In MRCS systems, bandwidth can be allocated to calls in integer multiples of some basic rate (for example, 8 Kbits/sec or 64 Kbits/sec). The circuit bandwidth required for a call is established by the customer at call set-up.

Unfortunately, employing the MRCS approach to implement N-ISDNs has a number of inherent limitations. Selection of an appropriate basic transmission rate for use in MRCS is a difficult engineering decision. High basic rates result in wasted bandwidth when services do not require the full bandwidth they are allocated. Similarly, small basic rates may cause extra overhead for higher rate services due to the need to allocate a large number of low rate services. An additional problem with MRCS is that the reassembly of multiple streams comprising a single call must be possible at the destination user. This synchronization requires close coordination of switching and transmission across networks so that variable delays between traffic streams can be avoided [KUL84].

The most critical limitation of circuit switching and MRCS is its inability to efficiently adapt to bursty traffic sources. A fixed channel rate is maintained throughout the duration of a call reguardless of the actual bandwidth being utilized. This forces users to either select a channel rate which matches their peak transmission needs (thus wasting bandwidth during non-peak transmission periods), or to select a channel rate which matches their average bandwidth usage (thus being more efficient, but risking the loss of information at peak transmission periods). Neither approach is acceptable for all applications since both approaches force the user to chose between network efficiency and performance.

3.2 Burst Switching

The investment of the telephone industry in transmission medium is immense (in fact, the replacement value of this plant is greater than the replacement value of current switching and terminal equipment). Motivated by this investment and the limitations of the MRCS approach, the telephone industry is exploring ways to extend the usefulness of physical transmission medium through improved switching techniques. One promising proposal is the burst switching method recently developed at GTE laboratories. Burst switching is expected to allow more, and a wider variety of, services to be derived from the existing transmission medium by moving much of the switching function onto the medium, and thus closer to users [AMS83,MOR85].

Burst switching can be classified as a unified integrated switching technique because data is switched through the same circuits, and in the same fashion, as digitized voice. This integrated switching of voice and data allows the network great flexibility in adapting to changes in traffic mixes. However, burst switching does differentiate between voice and data traffic in one manner: it switches voice samples at a higher priority than data samples. This prioritization is performed to preserve the time sensitive nature of real-time voice samples. Voice samples must arrive at their destination within some time-bound or they will be considered useless. Data samples, on the other hand, are much more tolerant of network delays.

A. Bursts

Since speakers in an ordinary voice conversation are only active for approximately 35-40% of their total calling time, conversation can be broken up into segments of voice activity called "talk spurts" [AMS83,DES85]. Some form of silence (or speech) detection can be employed to determine the duration of these talk spurts. If only voice traffic is being transmitted via burst switching, the 35-40% speaker activity can result in a utilization of network resources with almost three times the efficiency of traditional circuit switching. Similarly, data transmission can be broken-up into message units which contain information to transmit. In burst switching, *bursts* are defined to be either talk spurts or data messages. Burst switching assigns channel bandwidth exclusively to bursts; thus, resources in burst switching are dedicated only when there is actual information to send.

Burst switching is sometimes referred to as *fast circuit switching* because it employs the basic connection establishment concept found in circuit switching. Burst switching establishes end-to-end circuits through the network for the transmission of information as in traditional circuit switching, but, rather than maintaining the circuit for the duration of a call, the circuits are maintained only for the period of a burst. To make this approach feasible, circuit establishment and termination must be rapid and require little overhead. GTE's proposed architecture is designed to achieve this goal.

B. Burst Switches

Burst switching can be characterized as having many small, highly dispersed switches with distributed control. The switch in Figure 3.1 is a *burst switch.* It corresponds to a central office or PBX of approximately 65,000,000 lines. In the switch, many small *link switches*, each of which are capable of handling approximately 16 standard telephone lines, are connected together via *links*. The ports where outside lines access the network occur at these link switches. There are also 0, 1, or more high capacity switching centers called *hub switches* in the network. Note, the switching function has been brought closer to the users by moving link switches onto the copper plant. It is expected that these switches will be small in size (well under 1 ft^3), so that they can be easily located in neighborhoods or businesses [AMS83].

The path a burst will take through the network is calculated by the information stored in its header. Figure 3.2 shows the basic format of a burst. The beginning of the burst contains a 4 byte header. The first two bytes of header contain the network address of the destination. This information is used to route the burst through the network. The third byte contains control information specifying whether the burst is voice, data, or a command. The fourth byte contains a header checksum. If, using the checksum, it is determined that an error has occurred in the header, transmission of the burst is aborted. The next field of the burst is of variable length and contains voice or data information. The final field of the burst specifies the end of the burst. In the GTE approach, the final field is a one byte termination character called a FLAG. The FLAG pattern is also transmitted when the channel is idle. When a binary bit stream equivalent to the FLAG occurs naturally in the voice or data information, a special escape character is placed in front of the bit sequence, thus differentiating it from an end-of-burst FLAG [AMS83,HAU84].

C. Link Switches

The design of the link switch is presented in Figure 3.3. The link switches provide input and output switching service for links and ports. The switch consists of specialized processors surrounding a central memory. The processors are fast switching processors which manipulate both input character streams and storage buffers. The link switch central memory is a high-speed memory with circuitry capable of providing exclusive access to one processor at a time. The processors communicate only through this memory.

Most of the link switch central memory is made up of dynamic buffers. The buffers are circular in design and are assigned to active link or port channels. The circular design of the buffers allows concurrent input and output of characters. When a burst is switched through the link switch in the normal fashion (i.e., the burst flows through the link switch with output to a destination link occurring concurrently with input from the source link), only a few characters of the burst must be stored in the buffer. These characters correspond to the length of time it takes to determine the proper output link for the burst from the burst header information. In a link switch that is congested with multiple traffic bursts, a particular data burst may be temporarily blocked from output due to the unavailability of an appropriate output link. In that case, the burst is stored in the buffer assigned to its input link or port. If the size of the burst exceeds the allocated buffer space, another buffer is granted and the two are stored as a linked-list. The dynamic buffers can also be used to store characters arriving from a low speed data line, for later transmission at the maximum character/second channel rate allowed.

Also implemented in the central memory are queues associated with output links and ports. Thus, when a burst is ready for output, it is placed on the queue corresponding to its appropriate output link or port to await transmission. Three separate queues—high, normal, and low priority—are associated with each output link or port. The high priority queue is used for command bursts since they typically require fast transmission and occupy the channel for very brief periods. The normal priority queue is used for voice bursts, while the low priority queue is used for data bursts. A link or port is considered unavailable by a burst when its appropriate queue is allocated to another burst transmission.

To minimize delay through the network, and since the path of a burst may include many link switches, it is essential that the input and the output of the burst through the link switches occur concurrently. This is facilitated by matching the rates of the voice sources and destinations so that no buffering due to speed mismatches is required at the link switches. In practical terms, if PCM character generation is used for voice coding, the rates within the channels must match the character generation rate of 8000 characters/second (i.e., 64 Kbit/s). In GTE's implementation, the burst switches are connected by T1-rate (1.544 Mbit/s) transmission spans, with information transmitted between switches in the TDM channels of the spans. The character rate within these channels is 64 Kbit/s matching the PCM rate. Thus, at the link switches, burst output can begin as soon as the header information is processed for routing. The delay through a link switch can be as small as the channel time required to transmit two bytes (i.e., $10\mu s$).

In a link switch, bytes of information from a previously idle channel arrive successively. As each byte is received, it is stored in a buffer and processed if necessary. As soon as adequate routing information has been obtained, the buffered information is placed on the appropriate link output queue. The first two bytes (see Fig.3.2) of an incoming burst will always provide sufficient routing information if the burst is destined for another link switch on the network. The burst is then forwarded to the next switch. If the destination of the burst is the current link switch, the third byte of the header is used to determine the address of the appropriate port on the current link switch. The fourth byte is received and stored, and the header checksum calculated. If the checksum indicates an error, the burst is aborted, and subsequent bytes before the end-of-burst FLAG are discarded. If the checksum is good, characters up to the end-of-burst FLAG are stored in the buffer for subsequent output. Once the output queue or port has been determined and the checksum verified, input and output can continue simultaneously.

When more bursts exist in the queues of a link than idle channels for transmission, channel congestion occurs and input continues to be stored into a buffer. If the burst contains voice information and a predetermined amount of voice samples (e.g., 2 msec in GTE's scheme) have been accumulated without the initiation of output, the samples in the buffer are discarded. This is known as front-end clipping. If the burst contains data information, the buffer continues to accumulate the data characters, acquiring additional buffers if necessary, until a channel becomes available.

D. Hub Switches

In portions of the network where heavy traffic occurs, high-capacity burst switches called *hub switches* are installed. Figure 3.4 shows a hub switch design. The switch consists of three main elements: *Interface Link Switches*, *Hub Switch Elements* (HSE), and *hubs*. The interface link switches provide the buffering and routing procedures necessary to switch between links and hubs. The HSE is a high-speed switch between the interfacing link-switch memory and the high-speed parallel transfer paths (hubs) between adjacent HSE's. Depending on the transfer rate desired, the hubs are 1-4 bytes wide, plus control bits. The hubs are connected into a ring form, called a "hub ring". There are two hub rings in each hub switch to provide redundancy and extra available channels.

In the GTE implementation, there are 32 channels on the hub during each T1 TDM frame time. In the course of each channel there are 256 clock ticks with each clock tick advancing a 8-bit word on the hub ring from one HSE to the next HSE. Thus, within a channel, an HSE can send a hub word to any other HSE. In total, 256 origin HSE's can send 256 hub words to 256 destination HSE's within one channel.

3.3 Advantages of Burst Switching

GTE's burst switching approach has a number of advantages when compared with previous circuit switched-based approaches for integrated service digital networks. The most important advantages of the burst switching approach follow:

1) Local loops from the user to the switching site are reduced, because burst switching moves link switches onto the transmission medium, and thus closer to the users. This may allow the extraction of higher bandwidths from the existing transmission medium, since it is expected that a short loop—of length in the hundreds of yards—will be better able to support higher bandwidths than a long—perhaps miles long—two-wire pair. In new or replacement installations, the shortened loop length means significantly less copper is needed than for centralized switches. For example, one GTE Laboratories study shows that burst switches installed in a rural area of about 2000 lines would have required only 15% of the outside copper plant that the present centralized system requires [AMS83].

2) The unified nature of burst switching allows the network to adapt to changes in its traffic mix more easily than hybrid systems. Both voice and data bursts are transmitted in the same fashion—with the exception of prioritizing voice bursts—so that changing traffic mixes do not require reconfiguration of the network resources. The resources can be redistributed equitably.

3) Higher calling rates may be achieved, because burst switching has many distributed processors available for switching. It is possible for portions of the burst switch to operate successfully in isolation, thus letting the switch handle a more calls without the limitations imposed by the capacity of a central switching office. This also makes burst switches more fault tolerant than centralized schemes. Additional reliability is achieved by designing at least two routes between any two link switches, and implementing two hub rings in each hub switch.

4) Dispersed control also allows the burst switch to grow gracefully to meet increased calling rates. In a very natural way, the number of control processors on a burst network will be increased as the number of ports increases. Additional link switches can be added to the network without the complexity or down-time involved in adding a new multiprocessor to a central processor.

5) Compared to traditional circuit switching, the use of silence (or voice) detection improves the bandwidth utilization of the system by eliminating the transmission of "useless" information. The actual improvement achieved depends on the silence (or speech) detection mechanism used.

6) Output can occur concurrently with input, because the link-channel and hub-channel character rates are matched to the voice character generation rate. Output can begin a very short time after input begins. This allows a burst to be of any length, compared to the typical packet switching (including fast packet switching) case where a whole packet must be input and stored in a buffer before output begins. Thus, the header overhead for a burst is very small as the burst size increases.

3.4 Disadvantages of Burst Switching

Although there are many advantages to burst switching, at least two major disadvantages to the scheme exist as well.

1) The delay through a burst switch is greater than the delay through a traditional circuit switch. This is true because once a circuit is established in a circuit switched network, no routing delay is incurred by information on the circuit. In a burst network, each burst incurs some—perhaps minimal—routing delay, while the header information is used to calculate the proper outgoing link or port for the burst.

2) The dispersed nature of burst switching makes system testing and diagnostics quite difficult. No longer is there a loop from each user to a central office, rather there may be a number of link switches between the user and the hub switch. When service problems arise, some form of automated testing will be needed to pinpoint the source of the problem.

4. Packet Switching Approaches

For more than twenty years, packet switching has been used for computer-to-computer communication [LUD87, KUL84]. It is characterized by the division of transmission streams into small units, called *packets*, each with complete destination information. In packet switching, transmission links carry the packets of multiple users through the network via a series of switching nodes. Packet switching exploits the bursty nature of most services by statistically multiplexing the available network bandwidth, thus allowing arbitrary data rates to be sent by individual sources. Users transmit packets only when they have actual information to send. Specific time slots are not assigned to users, instead asynchronous time division multiplexing is performed. For bursty services, fast packet switching utilizes network bandwidth much more efficiently than synchronous, fixed bandwidth approaches like circuit switching.

Figure 4.1 shows the decoupling of the service transmission rate and the network transmission resources that results during packet switching. Digital information is sent from the source user to the packet assembler according to the clock and the natural bit rate of the service being transmitted. When a sufficient amount of information is accumulated in the packet assembler, a packet (including a header with control information) is produced and stored in the access queue. For example, when digitized voice is being transmitted, speech or silence detection mechanisms can be used to ensure that packets are created only when there is actual voice information to send. From the access queue, packets enter the network and are switched via their header information according to the network clock and protocol. When packets reach their destination, they are extracted from the network and queued in the packet disassembler. This queueing allows a smoothing of the network delays accumulated by the packets. The destination user can then read the data according to its own clock and service requirements.

4.1 Traditional Packet Switching

To provide error free communications, packet switching uses complex protocols. Most traditional packet switching networks have protocols based, at least loosely, on the ISO OSI (International Organization for Standardization's Open Systems Interconnection) model [TAN81, ZIM80]. Figure 4.2 shows the four lower layers of this protocol design. Layers 1 through 3 are applied link-by-link across the network providing the basic network transfer mechanisms. The physical layer (layer 1) controls the transmission characteristics of the transmission medium. The data link layer (layer 2) provides an error free channel between adjacent nodes through error detection and retransmission. The data link layer may also handle flow control on a link-by-link basis. Routing and congestion control are handled by the network layer (layer 3). The transport layer (Layer 4) is applied on an end-by-end basis to provide reliable host-to-host communication.

Unfortunately, the overhead entailed by these packet switching protocols is substantial. For example, at each node in the network, packets are manipulated and processed to identify transmission errors. Erroneous packets are retransmitted from the preceding node until they arrive without error. At each node, packets are stored awaiting initial transmission and subsequent acknowledgement of successful reception at the next node along their path through the network. Often this algorithm is implemented using a *sliding window protocol.* Typically, network flow and congestion problems are also handled on a link-by-link basis. "Choke" packets can be used to inhibit the rate of packet transmission at sending nodes or buffer reservation schemes can be employed. The transmission of control information and retransmission of erred packets on a link-by-link basis results in poor network performance and packet delays and variance across networks have proved prohibitive to real-time applications. Thus, packet switching has been implemented almost exclusively for the transmission of non-real-time information.

4.2 Fast Packet Switching

Recently, the requirements for creating integrated-services networks, along with new technological advances, have prompted researchers to re-examine packet switching. The network/services decoupling inherent in the packet switching approach, and the efficiency with which packet switching handles bursty traffic, suggests packet switching's intuitive appeal as a scheme for implementing application–independent networks. Now, with recent technological advances like fiber optics, realistic solutions to improving packet switching's performance difficulties for ISDN applications are being proposed.

The main packet switching proposal is called *fast packet switching* or *ATM* (Asynchronous Transfer Mode). This proposal is characterized by very high link transmission rates, and simple, hardwired, node-to-node protocols which match the rapid channel speed of the network. Fiber optic lines can now provide transmission rates of Gbits/sec, while VLSI technology provides the means to implement the switching function, including simplified protocols, entirely in hardware. As a result, the skeletal structure exists for the creation of packet switching networks with acceptable packet delays and variance for real-time communications. The central performance issue is whether protocols and switching architectures exist which can provide reliable communications while minimizing the delay and variance incurred by packets traversing the network.

4.3 Fast Packet Switching Protocols

In fast packet networks, as in traditional packet switching, packets of information, each containing a header field with complete destination information, traverse the network, traveling from node-to-node following a layered protocol. However, the protocol architecture used in fast packet networks differs greatly from the one used in traditional packet switching. A variety of fast packet protocols have been proposed, with most advocating at least two major modifications to the ISO OSI 7-layer protocol model [HOB83]:

1) *Simplified Protocols.* The complicated link-to-link layer protocols are eliminated and "pushed out" to the higher, edge-to-edge level. Specifically affected are the error correction techniques which require link-to-link retransmission of packets. Instead of relying on sliding window protocols implemented at each link, erroneous packets are retransmitted only at the edge-to-edge level.

2) *Virtual Circuit Service.* Packets are transmitted through the network on a virtual circuit basis to preserve packet ordering and reduce processing delays at both intermediate switching nodes and destination users. All packets in a given message are sequenced and follow a single path or virtual circuit through the network.

The aim of these two protocol modifications is to reduce the overhead of traditional packet switching, thus capitalizing on newly available technologies. Very high-speed transmission media increase the ratio of processing time to transmission time, moving the network bottleneck from the transmission links to the switching nodes. Thus, the processing cost of performing packet error detection and retransmission on a link-by-link basis increases relative to the actual transmission cost. Simplification of the link-to-link protocols, as mentioned in (1), is performed to minimize these processing costs. The feasibility of this approach is also supported by the very high-quality, low error-rate transmission media available. (We will discuss this issue in more detail in Section 5.) Exclusive use of virtual circuit service, as in (2), also serves to help reduce the processing delay incurred by packets traversing the network since no routing is necessary at intermediate switching nodes. Backlogs at destination users due to the reordering of packets are eliminated since virtual circuit service guarantees that packets will arrive in order.

4.4 Fast Packet Switching Architectures

A number of designs have been proposed for implementing fast packet switching. Recent conference proceedings have been filled with proposals of fast packet switch and network designs. To be competitive with circuit switching approaches, fast packet architectures must support the low delay transmission of packets across the network and, since each packet in fast packet switching incurs a processing delay within the network, the time spent in network switches is of primary importance. Thus, VSLI technology is used at the switching nodes of the network to implement the switching protocols in hardware rather than software, while the network links consist of very high-speed transmission media (e.g., fiber optic cable). Under consideration, then, are architectures which can be implemented using VLSI.

While many differences exist among the proposals for fast packet switching architectures, they can be broadly classified by their switch implementation into two main categories: ATD (Asynchronous Time Division) and Spaced Switched architectures.

A. ATD (Asynchronous Time Division) or Single Path Architectures

ATD architectures provide a single, multiplexed path through the fast packet switch for all packets. Typically, a bus [DEP87, COU87, THO84] or ring [ADA84, GAL87] is used, although the single path could also be implemented by multi-port memory. Control of the switch is usually handled in a distributed fashion with intelligent components controlling the receive and transmit ports, and the protocol processing. Using this distributed control, the fast packet switches can achieve much greater switching rates than traditional packet switches which rely on a single processor to interpret protocol control and handle the switching function [KUL84]. These new designs are also characterized as being non-blocking (that is, if sufficient capacity exists on the single path to handle an additional connection, the connection can be established).

As an example of this approach, Figure 4.3 shows the basic structure of the broadband switching element (BSE) proposed in [DEP87]. In this figure, eight receive ports are connected to eight transmit ports by a TDM bus. Each receive port is allocated a fixed time slot on the TDM bus, and the bus is designed to operate at a speed equal to the sum of the incoming bit rates at its receive ports. The packet sizes are fixed, and determined so that one packet can be transmitted in a TDM slot. Thus, during one TDM cycle the eight receive ports can transfer eight packets to the eight transmit ports. This ensures that no contention exists on the bus and, thus, the switch is non-blocking.

Congestion in the switch occurs when packets in different receive ports wish to access the same transmit port at the same time. To handle this problem, the BSE's have an output queue in each transmit port. These queues can store packets at a rate equal to the speed of the TDM bus when congestion occurs. Of course, with buffers of a finite size, loss due to queue overflow is possible.

In the BSE, as in other ATD designs, the throughput of the fast packet switch is determined by a single, multiplexed path. The switches usually have n input and n output lines, where n is much greater than 2; thus, the multiplexed path must run at a substantially faster rate than the transmission links. The total throughput of the ATD fast packet switches is bounded by current device logic. Most proposals operate at rates in the hundreds of Mbit/s [LEB86, DEP87], while throughput of 1 Gbit/s may be achievable with advances in the area of very high-speed logic [KUL84]. To eliminate the single path limitation and increase total throughput, spaced switched fast packet architectures implement multiple paths through the switching nodes. We discuss space switched architectures in the next subsection.

B. Space Switched or Multi-Path Architectures

Space switched architectures are based on the routing of packets via sophisticated switching fabrics. These fabrics are composed of cross-point switches which are linked together to provide multiple paths through each switching fabric. The cross-point switches are usually small (2×2), and routing through these cross-point switches may be accomplished through a bus, ring, or shared memory design. Rather than being multiplexed on a single path, packets are space switched through the fabric. The total throughput of these space switched architectures is based on the size of the switching fabric. Thus, space switching has a greater possible throughput than ATD approaches when the same device technology is used.

The choice of a switching fabric for designing a packet switch is of primary importance. A number of fabric designs (such as, cross-bar, Clos, Banyan, and Batcher-Banyan networks) have been proposed for this task [UEM88, HUI87, SIE88]. Besides the usual considerations of the packet delay through the switch and the cost of implementation, at least three significant switching fabric characteristics must also be considered: packet routing, blocking, and congestion. Since each packet in packet switching carries its own destination information and must be routed individually, it is desirable to make routing decisions as quickly and simply as possible. One approach is to use a self-routing switching fabric. In self-routing fabrics, cross-point components make the routing decisions for packets, performing the routing function in a distributed fashion.

Figure 4.4 displays a self-routing Banyan network [OBA88]. The switch fabric is composed of thirty-two 2×2 cross-point elements assembled into four stages. From any one of the sixteen input ports, it is possible to reach all of the sixteen output ports. One desirable characteristic of the Banyan network is that it is self-routing. Since each cross-point switch has only two output lines, only one bit is required to specify the correct output path. Very simply, if the desired output address of a packet is stored in the packet header in binary code, routing decisions for the packet can be made at each cross-point switch by examining the appropriate bit of the destination address. A simple way to handle the address manipulation is, at every cross-point switch, to read the first bit of the header and then perform a shift left before transmitting the packet on the selected output line.

Many switch fabric designs are not non-blocking. For example the Banyan network of Figure 4.4, is blocking. It is possible for an available inport port to be prevented from sending a packet to an available output port while there is available capacity in the switch. This can occur when an internal link of the fabric becomes overloaded. Some switch designs, such as $n \times n$ crossbar switches, are non-blocking. Unfortunately, the n^2 cross-point switches that they require for implementation makes them difficult and expensive to build. Clos and Benes networks are non-blocking and require fewer components than crossbar switches, unfortunately the algorithms they require to avoid blocking are complex and no self-routing algorithms exist for these fabrics.

Switch congestion is inherent in the packet switching approach; all switching fabrics can experience it. Even in non-blocking switching fabrics where internal congestion is eliminated, congestion occurs when multiple packets contend for the same output port. To solve this congestion problem, packet buffers are used [ANI88, OBA88, KIM88]. Buffers may be placed at the input, and/or output ports of the switch. In input buffering, multiple packets destined for the same output port are buffered and sequenced on the input links. In output buffering, packets contending for the same output port are buffered and passed to the output port one at a time.

The buffering required in a switching fabric is determined by the speed of the switching fabric and the speed of the network transmission lines. For example, consider a $n \times n$ non-blocking switch with all input and output ports operating at the same speed, say m packets per second. If the switching fabric operates at a speed greater or equal to $n \times m$ packets per second, buffering is not necessary at the input ports since the switch can handle the maximum number of packets the input lines can submit. Output buffering is still necessary in this case to resolve packet contention at the the output ports. If the switching fabric operates at a speed of m or less packets per second, input buffering is needed to accommodate the possible the incoming traffic load; however, no output buffering is necessary in this case, since the maximum throughput of the switch can be supported by each output port. If the speed of the switch is $l \times m$ where $1 < l < n$, then buffering becomes necessary at both input and output ports.

To satisfy packet routing, blocking, and congestion control requirements, many space switching fabrics have complex designs. For example, Figure 4.5 shows the design of a Broadcast Packet Switch [TURN86]. This switch fabric is composed of a series of major components: a Copy Network, Broadcast and Group Translators (BGT), a Distribution Network, and a Routing Network. The routing network is self-routing, binary switching network (Banyan network) with buffers at each input port capable of holding two complete packets. Blocking on the Routing Network is reduced by the Distribution Network. The Distribution Network evenly distributes all packets it receives across all its outputs breaking up any "communities of interest" that may exist. The Copy Network and Broadcast and Group Translators are included to accommodate multi-point connections throughout the network.

4.5 Advantages of Fast Packet Switching

Several advantages to fast packet switching are discussed in this section. The first two—unified switching methods and silence/voice detection—are also advantages of the burst switching approach.

1) Fast packet switching provides complete unified switching of voice and data packets; voice and data packets are transported across the network in an indistinguishable fashion. This means that the network can easily adjust to changes in the mix of data and voice traffic without reconfiguration. Differentiation between the traffic types does occur, but only at the higher end-to-end layers of the protocol.

2) User services in packet switching are transmitted through the network at their natural bit rate. For example, the formation of voice into packets involves the detection of silence (or voice) intervals. Packets are formed from the active segments of voice input at their natural bit rate, and do not include the silence intervals found between sentences, words, and letters. Silence detection is important in maximizing the network throughput.

3) By moving application-specific requirements (such as, error detection and recovery, and flow control) into an end-to-end, higher-level protocol, the internal link-to-link protocols of the fast packet switch are greatly simplified. This provides faster transportation for packets through the network.

4) Fast packet switching basically uses the transmission facility as a "digital pipe" for carrying short packets of information one after another. Packets from all users are multiplexed on the network. Thus, connections of arbitrary bandwidth can be accommodated in a simple fashion [TURN86].

4.6 Disadvantages of Fast Packet Switching

Two main disadvantages of fast packet switching exist. The first disadvantage, header overhead, is an inherent limitation of all packet switching approaches. The second disadvantage, dependency on transmission facilities, is intensified in the fast packet switching approach.

1) Fast packet switching requires more header overhead than either circuit switching or burst switching. Routing information must be stored in the header of each packet in packet switching, at the beginning of each conversation in circuit switching, and at the beginning of each burst in burst switching. Packets are typically smaller than average burst sizes because they must be buffered at the source and intermediate nodes of the network before they are transmitted. Thus, the large number of packets used in packet switching results in a larger header overhead.

   Buffering of packets is necessary, because packets are typically switched using the full bandwidth of the link. This means that packets must be accumulated at the source rate and then transmitted at a higher rate. Since this accumulation process adds to the delay of packets, the size of packets must be limited [AMS83].

2) A major assumption in designing fast packet switched protocols, is that high-speed, high-quality digital transmission facilities can be used to provide very reliable service for the packet network. This assumption allows the flow control and error correction facilities of the protocol to be pushed to a higher end-to-end level without damaging the performance of the network. Thus, if problems occur which reduce the speed or quality of the network transmission facilities, severe performance degradation may result.

5. Performance Issues

Because burst and fast packet switching use unified approaches to solve the integrated service problem, they are being advocated as better technologies for the backbone ISDN network than the presently defined hybrid scheme which uses separate switching techniques for circuit-switched voice and packet-switched data. While the two approaches have few similarities to the current view of ISDN, they share many common features. Both burst and fast packet switching transmit and switch voice and data in an unified fashion, and both schemes typically employ speech/silence detection. Burst and fast packet switching also have simplified internal (i.e., link-to-link) protocols with many functions pushed toward the edge-to-edge, network level. In contrast to the ISDN network, burst and fast packet switching transmit their control messages in the same way they transmit voice and data. On the negative side, speech loss (either front-end clipping or voice packet loss) occurs in both approaches.

While both burst and fast packet switching are discussed, a survey of recent conference proceedings and journals shows a definite trend toward the use of fast packet switching technologies for integrated service networks. The primary reason for this trend seems to be the high degree of flexibility found in fast packet switching. This flexibility can be seen in the intuitively appealing way fast packet switching handles two main issues: (1) incorporating new services into the network, and (2) utilizing new technologies for network implementation.

First, new services can be easily supported by fast packet switching since there are no basic service rate limitations imposed on the network bandwidth. Unlike burst switching where users must specify a service rate that is a multiple of some basic rate, in fast packet switching, services can be submitted to the network at their natural bit rates. For example, voice services can be transmitted through the network at the rate of the encoding scheme they utilize. If PCM encoding is used, voice can be transmitted at a rate of 64 Kbit/s, while the use of Adaptive Differential PCM (ADPCM) allows voice to be transmitted at a rate of 32 Kbit/s. This support of services at any bit rate provides a high potential for technological improvement from the service end of the network. Network designers are not limited by network specified basic bandwidth rates; rather, improved coding schemes for current services and newly developed network services can be implemented with regard only to natural bit-rate characteristics.

The second area in which fast packet switching has more flexibility than burst switching is in its ability to exploit new innovations in switching hardware and transmission media. Fast packet switching, and packet switching in general, supports a decoupling of the user service transmission rate and the network transmission resources (recall Fig. 4.1). This decoupling shields user services from the internal implementation of the network. Thus, technological advances in packet switching architectures can be incorporated into network designs without any adverse effects on the user services supported. This two-fold flexibility—service and technological flexibility—clearly gives fast packet switching a strong appeal as a solution to the integrated service problem.

Although fast packet switching is still in its initial development stages, researchers have explored two of the essential performance issues raised by fast packet switching advocates. The feasibility of fast packet switching is tightly linked to these performance issues, and, thus, we discuss the principle research contributions addressing these issues in sections 5.1 and 5.2. First, we examine research which supports the feasibility of the protocol simplifications proposed for fast packet switching. Specifically, we discuss the issue of moving the internal, link-to-link error control protocols to the outer, edge-to-edge level. Second, we explore the effects of the packet header overhead incurred in fast packet switching on network performance. We examine research which claims that fast packet switching can achieve performance results comparable to that of burst switching.

5.1 Protocol Issues in Fast Packet Switching

Most proposals for fast packet switching networks rely on high-quality, high speed digital transmission facilities to provide very fast and reliable communication services, and advocate simplified link-to-link layer protocols to reduce the processing bottleneck which can occur at the internal switching nodes. Thus, network performance is highly dependent on an effective implementation of these simplified protocol schemes. Particularly important is the feasibility of pushing the error recovery protocols of the lower link-to-link layers to the higher edge-to-edge layers.

In Suda and Watanabe [SUDA88], a mathematical analysis of the effects of protocol processing overhead on the performance of a fast packet switching network are presented. The fast packet switching model developed is based on the lower three layers of OSI ISO 7-layer protocol model. The lower three layers of the system provide packet transmission and routing through the network, while allowing the possibility of packet loss. The protocols in these layers are simplified with no error recovery performed between adjacent nodes. The fourth protocol layer functions between source and destination nodes and performs edge-to-edge error recovery, thus assuring reliable data communication throughout the network.

The fast packet switching network is modeled as a tandem queueing network with feedback loops between adjacent nodes, and the source and destination nodes. Each queue in the model represents a protocol layer rather than a whole switching node. The fast packet model was analyzed under some of the most common error control protocols: link-to-link, edge-to-edge, and a combined link/edge scheme. From a mathematical analysis of the system, the authors conclude:

- The edge-to-edge error recovery scheme can provide a high quality (i.e., low error probability) transport pipe when a sufficiently large number of retransmissions are allowed between the source and destination nodes. For example, the edge-to-edge scheme can achieve a cross-network error probability of 10^{-8} bits/s in a 4-hop network with link error rates of 10^{-3} bits/s, when three or more retransmissions are allowed on the edge-to-edge layer.

- When network traffic is light, the edge-to-edge scheme shows a shorter delay than the link-to-link scheme.

- The edge-to-edge scheme achieves a shorter delay when packet error rates on the links are small ($< 10^{-3}$ bits/s) and there are a small number of processing nodes (≤ 4).

In summary, Suda and Watanabe have determined that in realistic network situations (i.e., in high-speed, low error rate networks), the end-to-end error control scheme performs better than the link-to-link or combined link/edge scheme. As an example, consider a 1 Gbit/s network with 4-hops using the go-back(n) packet retransmission scheme. In the range where link error rates are of the order expected on a typical fast packet switching network (i.e., $< 10^{-4}$ bits/s), the optimal error recovery scheme for traffic intensities below 80% is shown to be edge-to-edge.

Similar results have been obtained through the tandem queueing network model developed by Bhargava *et al.* [BHAR88]. In their approach, each switching node along a virtual circuit is modeled by a single queue. At each node, the performance of separate, as well as shared, buffer pools for each virtual circuit are examined. Explicitly considered are the effects of message propagation delays, finite buffer capacities (blocking), channel errors, and the use of a timeout mechanism in the error control scheme. The analytic models used in this paper are validated by simulation of a 4-hop virtual circuit with a link propagation delay time equal to 25 times the packet length and a transmission rate of 100 Mbit/s. The authors conclude that even under assumptions that strongly favor the link-to-link approach, such as using analytic techniques which overestimate the delay for end-to-end error control, the end-to-end approach to error control in high-speed networks achieves superior performance and requires fewer network resources (e.g., buffers and processing time).

In Brady [BRAD88], an experimental approach is presented which supports the theoretical results presented above. A small, three switch packet network was simulated by coding the complete X.25 link protocol in detail on each link, and coding an edge-to-edge protocol for each virtual circuit. For the study, the link error control protocol could be turned on or off. By examining the round-trip response times a user experiences from transmission to the receipt of an acknowledgement, Brady concludes that the edge-to-edge protocol provides good performance if its timeout threshold is high enough to account for delays due to packet queueing. Additionally, the study shows that adding a link-to-link layer error control protocol to a network with an edge-to-edge protocol will significantly improve network performance only if (1) a window of sufficient size is provided in the link-to-link layer to prevent blocking due to acknowledgements, and (2) the network is operating under a light load and unfavorably high error conditions.

5.2 The Effects of Overhead

O'Reilly [ORE86] compares two networks, one implemented using burst switching and one implemented using fast packet switching, with the same capacity and TASI advantage. The TASI advantage for a burst switching system is defined to be the ratio of S, the number of active voice sources, to v, the maximum number of channels the system can support. For fast packet switching, the value of v is equal to the link capacity, or ratio of the link transmission rate, to the voice encoding rate. The number of voice users each switching scheme can support is significantly influenced by the the amount of overhead necessary for voice transmission. In both schemes, overhead results from the transmission of burst or packet headers across the network. In burst switching, destination information is assigned to each burst. In fast packet switching, destination information must be included in each packet. Since talkspurts are typically much longer than packets, fast packet switching generates many packets per talkspurt and requires more overhead than burst switching. Note, typically voice packet lengths in fast packet switching are kept small to minimize the packetization delay at the source.

In burst switching, voice performance is degraded when all of the output links of a switch are busy. Since voice bursts are not buffered, when such a situation occurs, the front of delayed voice bursts are clipped. The result is cutout, or freezeout, of voice. In contrast in fast packet networks, when voice packets are delayed more than some specified time at a node, the packets are dropped. This is done to preserve the integrity of conversations on the network. In contrast to the bursty loss resulting from front-end clipping, the discarded voice packets are usually spread across the complete talkspurt, and are dropped from a number of active talkspurts not just one or two. In fast packet switching, there are also ways of covering up packet losses during voice playback. For example, previous packets can be duplicated and played, or approximate values for a lost packet can be determined from the packets adjacent to it. These differences lead to the result presented by Tucker [TUC84], that, on the average, 0.5% front-end clipping is equivalent to about 2% packet loss. Thus, the reconstructed voice quality of burst switching is more sensitive to front-end clipping than the reconstructed voice quality of fast packet switching is to packet loss. Tucker also reports that, using some advanced encoding techniques [JAY80], satisfactory voice quality can be achieved with up to 5% packet loss.

O'Reilly first compares the performance of voice traffic in burst and fast packet systems. Following Tucker's guidelines, the average freezeout in burst switching is compared with corresponding packet loss and packet delay in fast packet switching for the same TASI advantage. Four sample cases, ranging from a very small, eight channel network to the asymptotic behavior of a network with an extremely large link capacity, are explored. O'Reilly concludes that fast packet switching can provide comparable voice performance to that of burst switching, although the effect of the extra overhead required in fast packet switching becomes most pronounced in systems of large capacity. Additionally, the residual capacity remaining for data traffic is significantly less in the fast packet switching case due to the much greater overhead.

In the remainder of the paper, O'Reilly compares data performance in the burst and fast packet networks using Tucker's finding that, on the average, 0.5% front-end clipping is equivalent to 2% packet loss. He concludes that in the case of 64 Kbit/s encoded voice, data performance for burst and fast packet switching is not significantly different for the range of realistic network capacities. At low data loads, burst switching provides better performance since a data burst will typically find an available outgoing link for transmission. For a significant portion of higher data loads, fast packet switching demonstrated marginally better performance. In fact, if the more optimistic case of 5% loss of voice packets is considered equivalent to 0.5% freezeout [TUC84], the performance of fast packet switching is superior to burst switching for most of the data load range.

6. Fast Packet Switching Simulation Study

The performance of fast packet switching networks is closely tied to the feasibility of pushing protocol functions out from the internal nodes of the network to the network edges (see Section 5.1). Of particular interest in this section is the issue of error control, and whether satisfactory network performance can be achieved if the network error control functions are shifted from a link-to-link to an end-to-end basis. To evaluate this issue, a model of a fast packet switching network was developed and extensive simulations were performed. Network performance was evaluated under two of the most popular error control techniques: go-back(n) and selective repeat.

6.1 Network Model

Three possible error control schemes are examined in the simulation study: (1) the end-to-end scheme where retransmission of erred packets takes place only at the network edges, (2) the link-to-link scheme where retransmission of erred packets takes place on a hop-by-hop basis between adjacent nodes, and (3) a combined link/edge scheme where both end-to-end and link-to-link retransmissions are allowed. In all three schemes, limitations may be placed on the number times a particular packet will be retransmitted. Note, the end-to-end scheme corresponds to the protocol simplifications proposed for fast packet networks, while the other two schemes are similar to those found in traditional packet switching.

In our model of a fast packet switching network (Figure 6.1), the lower three layers of the OSI-ISO 7-layer protocol model are represented. Layer 1 (the physical layer) and Layer 2 (the data link layer) operate on a node-to-node basis, each communicating only with their adjacent peer layer protocols. Layer 3 can be subdivided into Layer $3a$ (the packet network sublayer) and Layer $3b$ (the packet transport sub-layer). Layer $3a$ operates on a node-to-node basis, while Layer $3b$ is implemented end-to-end across the network from source to destination nodes. Layers 2, $3a$, and $3b$ are modeled as queues with service rates μ_x representing the protocol processing times required at each layer, where x is the layer number.

Fast packet switching networks typically employ virtual circuit service, thus our simulation study examines single paths through the network. All packets follow a predetermined path from source node to destination node. The length of the network path can be varied from a simple two-node, source-destination network to any specified path length. For our simulation study, we use a 4-hop network (i.e., a source node, a destination node, and three internal nodes).[1] Feedback loops are included in our model to handle the retransmission of erred packets. The status of transmitted packets (i.e., acknowledgement (ACK) or negative acknowledgement (NAK)) is returned via these loops to the sending nodes. Feedback from the source to the destination node occurs at Layer 3*b*, while feedback between adjacent link-to-link nodes occurs at Layer 2.

We assume packet errors may occur in the network at all transmission links. For the link-to-link scheme, these errors may be detected on a link-by-link basis at Layer 2, while for the end-to-end scheme, errors are detected only at the destination node. Thus, the processing time is increased at Layer 2 when the link-to-link scheme is used, and at Layer 3*b* when the end-to-end scheme is used. We assume that packet errors occur on a given link with probability p_1. We consider only a constant value of p_1. Note, in our simulation, we assume a packet error occurs when the contents of the packet are corrupted during transmission.

Two popular error control techniques are considered in our simulation: go-back(n) and selective repeat. Using the go-back(n) technique, an erred packet causes all packets following it in sequence number to be retransmitted. Retransmission occurs regardless of whether the succeeding packets are also erred. In contrast, the selective repeat technique requires retransmission of packets only when they are erred or do not fall into the current window of packet sequence numbers. The sliding window of acceptable packet sequence numbers is initially allocated and updated as packets are successfully received. Throughout our simulation, whenever selective repeat is used, the window size is assumed to be 100 which is large enough to prevent window sequence errors.

6.2 Simulation Results

The figures in this section show the results of our simulation. In all of the figures, the following parameters are fixed: the network length is four hops, the realistic value of 1 Gbit/s is used for the link transmission speed, packet lengths are exponentially distributed with average length 10 Kbits, and a maximum of three retransmissions are allowed at any feedback loop. Packet lengths are exponentially distributed with average length 10 kbits, and the packet processing rate at the protocol layers involved in error control is assumed to be 10^{-5} bits/s.

Figure 6.2 shows the average end-to-end packet transmission delay versus traffic intensity of a network using the go-back(n) error control technique. In this figure,

[1] Using a 4-hop network seems quite realistic given current ARPANET configurations.

packet errors occur at the network links with probability 0.01 per packet. For comparison purposes, the bottom curve in this figure shows a network operating with no error control. For the cases we consider, this curve represents the average minimum end-to-end delay through the network. Of the error control schemes we examine, the end-to-end scheme provides the smallest average transmission delay when traffic is light, but the average delay rapidly increases as traffic intensity rises and the network quickly saturates. The link-to-link scheme is shown to be better than the end-to-end scheme for traffic intensities above 0.13, and slightly better than the link/edge scheme, which requires processing time at both Layer 2 and 3*b*, for all traffic intensities. This is true because the end-to-end scheme reduces the internal processing time required at the link nodes by eliminating hop-by-hop error control. This reduction in processing time occurs at the additional expense of retransmitting erred packets end-to-end through the network from the source to the destination node. In contrast, the link-to-link approach retransmits packets and corrects errors at each network link. As seen in this figure, at low traffic intensities the number of erred packets is small, and, thus, the processing time saved at the internal nodes in the end-to-end scheme outweighs the additional end-to-end retransmission cost. Unfortunately, as the traffic intensity and thus the number of erred packets increases, the longer time required in the end-to-end scheme for packet feedback and retransmission causes increased network delays, eventually resulting in network saturation.

The curves in Figure 6.3 also show the average end-to-end packet transmission delay versus traffic intensity of a network using the go-back(n) error control technique; however, the probability of an error occurring at a link is reduced to 0.001. Note, the general placement of the curves is similar to that in Figure 6.2. In this figure, the end-to-end scheme provides lower delays for a wider range of traffic intensities. In Figure 6.2, the end-to-end scheme was superior only for traffic intensities less than 0.13, while with the lower error probability of Figure 6.3, the end-to-end scheme is superior through traffic intensities of 0.35. Comparing these two figures, we can see that the end-to-end scheme performs well in low link error rate environments, since there are few erred packets that must be retransmitted through the entire network from source to destination. Thus, as the error probability is reduced, the end-to-end scheme provides lower delays for a wider range of traffic intensities.

In Figure 6.4, the average end-to-end packet transmission delay versus traffic intensity for a network using the selective repeat error control technique is shown. The error probability of this network is 0.01. Comparing this figure with Figure 6.2 shows the difference in end-to-end delay caused by modifying the error control from go-back(n) to selective repeat. Again, the general placement of the curves are similar in both figures. However, in Figure 6.4, the delay values for all retransmission schemes—end-to-end, link-to-link, and link/edge—are lower. For example, at a traffic intensity of 0.2, the end-to-end scheme with the go-back(n) error control technique has a 0.269 ms average packet transmission delay, while using the selective repeat error control technique results in an average transmission delay of 0.158 ms. The lower delay stems from the fact that selective repeat requires only one retransmission per occurrence of an error. Fewer retransmissions in the selective repeat technique also causes saturation to occur at higher traffic intensities in the end-to-end scheme. As a result, the end-to-end scheme is clearly superior for an even wider range of traffic intensities when the selective repeat technique is used.

Similarly, Figure 6.5 shows the average end-to-end packet transmission delay versus traffic intensity for a network using the selective repeat error control technique. In this figure, the error probability of the network is reduced to 0.001. As a result of fewer erred packets, the end-to-end error control scheme provides the smallest average end-to-end packet delays for all traffic intensities. At these parameters, there is no traffic intensity at which the additional cost of end-to-end packet retransmissions outweighs the savings resulting from the reduced hop-by-hop error control processing.

In Figure 6.6, the optimal error recovery schemes are summarized for a network using the go-back(n) error control technique. Given that the packet error rate and the traffic intensity of the network, the error control scheme (i.e., either end-to-end or link-to-link) which provides the smallest average end-to-end delay is given. The top curve in the figure represents the values at which the network reaches saturation. Above this curve, both error control schemes result in network overload with neither scheme providing acceptable network delays. The bottom curve in the figure represents the points at which the end-to-end scheme performs equivalently to the link-to-link scheme. For values between these two curves, the link-to-link scheme is optimal. Below the lower curve, the end-to-end scheme achieves the lowest delays. As the figure displays, at high error rates (i.e., $\geq 10^{-1}$) the link-to-link scheme provides the best performance for all traffic intensities up to the saturation level. However, as the network error rates begin to decrease, the end-to-end scheme becomes optimal for small traffic intensities. As error rates decrease further, the end-to-end scheme is superior to the link-to-link scheme for a wider and wider range of traffic intensities. Considering the error rates anticipated in real-world implementations of fast packet switching networks (i.e., extremely low error rates below 10^{-7}), the results of our simulation suggest that the end-to-end error control scheme will provide better performance than the link-to-link scheme for almost all traffic intensities.

7. Conclusion

This paper surveys a variety of proposals for future integrated-service networks. It focuses on unified approaches which employ a single switching technique across the entire network bandwidth. Unified approaches are of particular interest because they can facilitate a broad range of user services while remaining easily expandable. Current hybrid approaches to N-ISDN networks are also discussed.

Burst and fast packet switching are the two most promising unified approaches to integrated-service networks. Of the two, fast packet switching is receiving significant attention because it is extremely flexible in supporting new and improved user services, as well as, in incorporating technological advances in switching hardware and transmission media.

Two of the key performance issues in determining the ultimate effectiveness of fast packet switching are: (1) the feasibility of simplifying the error control protocols found in traditional packet switching networks, and (2) the effects of the packet header overhead incurred in fast packet switching on network performance. Both of these issues are discussed in the current fast packet switching literature, and research results suggest that fast packet switching can meet the demands of real-time integrated service networks, and can perform favorably when compared to burst switching.

To verify that satisfactory network performance can be achieved by a fast packet switching network when error control functions are shifted from a link-to-link to an end-to-end basis, an extensive simulation study was performed. In the study, three error control schemes were examined: (1) end-to-end, (2) link-to-link, and (3) a combined link/edge scheme. Results of the study show that for real-world networks which utilize the high-quality (i.e., extremely low error rates below 10^{-7}) transmission media available, the end-to-end error control scheme provides lower network delays than either the link-to-link or the combined link/edge scheme for almost all traffic intensities.

REFERENCES

[ADA84] J. L. Adams and R. M. Falconer, "'Orwell': A Protocol for Carrying Integrated Services on a Digital Communications Ring," *Electronics Letters*, vol. 20, No. 23, Nov. 1984.

[AMS83] S. R. Amstutz, "Burst switching—An introduction," *IEEE Communications Magazine*, vol. 21, pp. 36-42, Nov. 1983.

[ANI88] G. J. Anido and A. W. Seeto, "Multipath Interconnection: A Technique for Reducing Congestion Within Fast Packet Switching Networks," *IEEE J. on Select. Areas in Commun.*, vol. SAC-6, No. 9, Dec. 1988.

[BHAR88] A. Bhargava, J. F. Kurose, D. Towsley, and G. VanLeemput, "Performance Comparison of Error Control Schemes in High Speed Computer Communication Networks," *IEEE INFOCOM '86*, New Orleans, Louisiana, March 1988.

[BRAD88] P. T. Brady, "Performance of an Edge-to-Edge Protocol in a Simulated X.25/X.75 Packet Network," *IEEE J. on Select. Areas in Commun.*, vol. SAC-6, No. 1, Jan. 1988.

[COU87] J-P Coudreuse and M. Servel, "PRELUDE: An Asynchronous Time-Division Switched Network," *ICC '87*, pp. 769- 773.

[DEC86] M. Decina and E. L. Scace, "CCITT Recommendations on the ISDN: A Review," *IEEE J. on Select. Areas in Commun.*, vol. SAC-4, No. 3, May 1986.

[DEP87] M. De Prycker and M. De Somer, "Performance of a Service Independent Switching Network with Distributed Control," *IEEE J. on Select. Areas in Commun.*, vol. SAC-5, No. 8, Oct. 1987.

[DES85] A. Descloux, "Models for switching networks with integrated voice and data traffic," ITC 11, ed. Minoru Akiyama, Elsevier Science Publishers B. V. (North-Holland), 1985.

[GAL87] I. D. Gallagher, "A Multi-Service Network Based on the Orwell Protocol," *ISS '87*, pp. 979-983.

[HAU84] J. F. Haughney, "Application of burst-switching technology to the defense communications system," *IEEE Commun. Magazine*, vol. 22, pp. 15-21, Oct. 1984.

[HOB83] W. L. Hoberecht, "A layered network protocol for packet voice and data integration," *IEEE J. on Select. Areas in Commun.*, vol. SAC-1, Dec. 1983.

[HUI87] J. Y. Hui and E. Arthurs, "A Broadband Packet Switch for Integrated Transport," *IEEE J. on Select. Areas in Commun.*, vol. SAC-5, No. 8, Oct. 1987.

[JAY80] N. S. Jayant, "Effect of Packet Losses in Waveform-Coded Speech," *ICC '80*, Atlanta, GA, Oct. 1980.

[KIM88] A. S. Kim and A. Leon-Garcia, "Performance of Buffered Banyan Networks Under Nonuniform Traffic Patterns," *IEEE INFOCOM '88*, New Orleans, Louisiana, March 1988.

[KUL84] J. J. Kulzer and W. A. Montgomery, "Statistical Switching Architectures for Future Services," *Proc. of the Int'l Switching Symposium*, May 1984.

[KUM80] K. Kummede and H. Rudin, "Packet and Circuit Switching: Cost/Performance Boundaries," Computer Networks Magazine, No. 2, 1980.

[LEB86] H. Le Bris and M. Servel, "Integrated Wideband Using Asynchronous Time Division Techniques," *Globecom '86*, pp. 1720-1724.

[LUD87] G. W. R. Luderer, J. J. Mansell, E.J. Messerli, R. E. Staehler, and A. K. Vaidya, "Wideband Packet Technology for Switching Systems," *ISS '87*, 1987.

[LUE86] J. C. Luetchford, "CCITT Recommendations—Network Aspects of the ISDN," *IEEE J. on Select. Areas in Commun.*, vol. SAC-4, No. 3, May 1986.

[MIT86] O. Mitchell and M. Mracek, "Implementing ISDN in the United States," *IEEE J. on Select. Areas in Commun.*, vol. SAC-4, No. 3, May 1986.

[MOR85] J. D. Morse, "Performance evaluation of burst-switched integrated voice-data networks," ITC 11, ed. Minoru Akiyama, Elsevier Science Publishers B. V. (North-Holland), 1985.

[OBA88] H. Obara and T. Yasushi, "High Speed Transport Processor for Broad-Band Burst Transport System," *ICC '88.*

[ORE86] P. O'Reilly, "Burst and fast-packet switching: performance comparisons," *IEEE INFOCOM '86*, Miami, Florida, pp. 653-666, April 1986.

[SIE88] H. J. Siegel and W. T-Y Hsu, "Interconnection Networks," *Computer Architecture: Concepts and Systems*, ed. V. M. Mulutinovic, Elsevier Science Publishers B. V. (North-Holland), 1988.

[STA85] W. Stallings, "Tutorial: Integrated Services Digital Networks (ISDN)," IEEE Computer Society Press, Washington D.C., 1985.

[SUDA88] T. Suda and N. Watanabe, "Evaluation of Error Recovery Schemes for a High-Speed Packet Switched Network: Link-by-Link Versus Edge-To-Edge Schemes," *IEEE INFOCOM '86*, New Orleans, Louisiana, March 1988.

[TAN81] A. S. Tanenbaum, "Network Protocols," *Computing Surveys*, Vol. 13, No. 4, 1981.

[THO84] A. Thomas, J-P Coudreuse, and M. Servel, "Asynchronous Time-Division Techniques: An Experimental Packet Network Integrating Videocommunication," *ICC '84*.

[TRO86] M. Trouvat, "RENAN—An Experimental ISDN Project in France," *IEEE J. on Select. Areas in Commun.*, vol. SAC-4, No. 3, May 1986.

[TUC84] R. C. F. Tucker, "Transmission of speech in packet-switched networks," Ph.D. diss., Dept. of Electrical & Electronic Eng., Univ. of Aston in Birmingham, Aug. 1984.

[TUR86] J. S. Turner, "New directions in communications (or which way to the information age?)," *IEEE Commun. Magazine*, vol. 24, pp. 8-15, Oct. 1986.

[TURN86] J. S. Turner, "Design of a Broadcast Packet Network," *IEEE INFOCOM '86*, Miami, Florida, pp. 667-675, April 1986.

[UEM88] H. Uematsu and R. Watanabe, "Architecture of a Packet Switch Based on a Banyan Switching Network with Feedback Loops," *IEEE J. on Select. Areas in Commun.*, vol. SAC-6, No. 9, Dec. 1988.

[ZIM80] H. Zimmermann, "OSI Reference Model—The ISO Model of Architecture for Open Systems Interconnection," *IEEE Trans. on Commun.,* COM-28, April 1980.

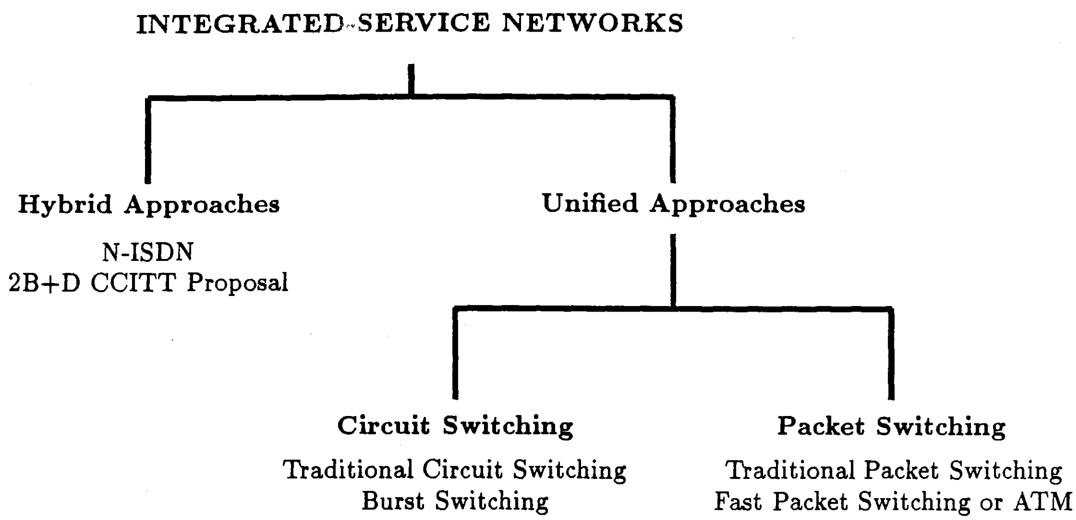

Figure 1.1 Classification of Integrated Service Networks

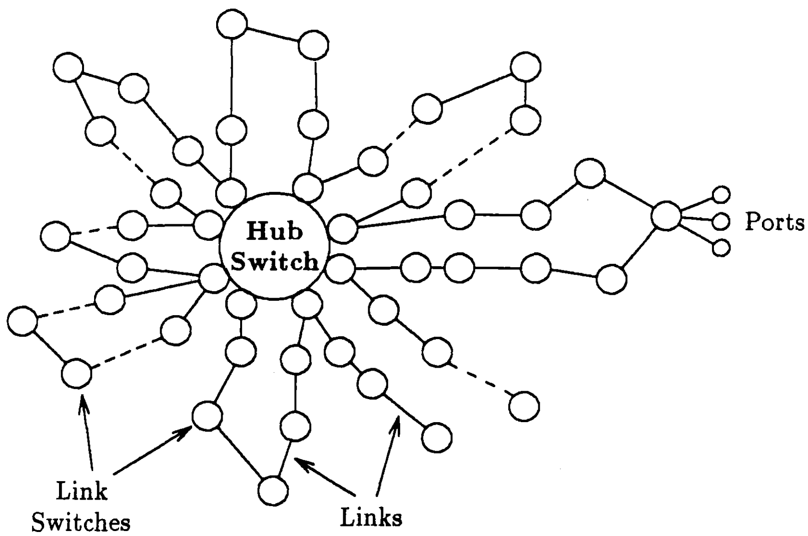

Figure 3.1 Burst Switch

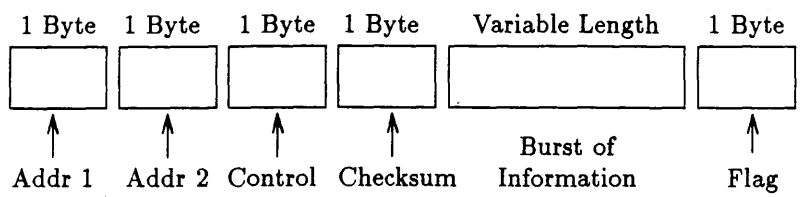

Figure 3.2 Format of a Burst

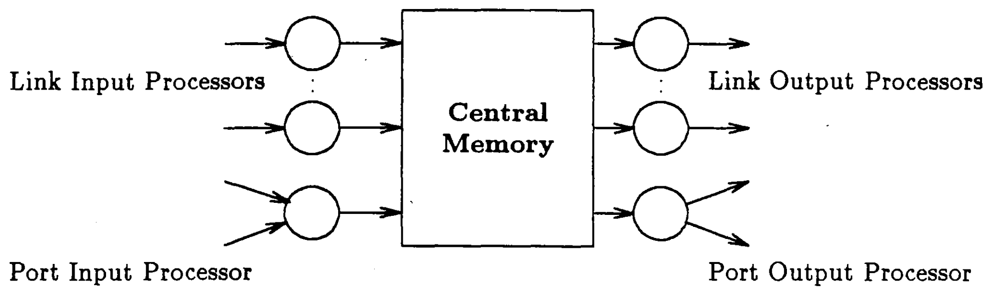

Figure 3.3 Link Switch

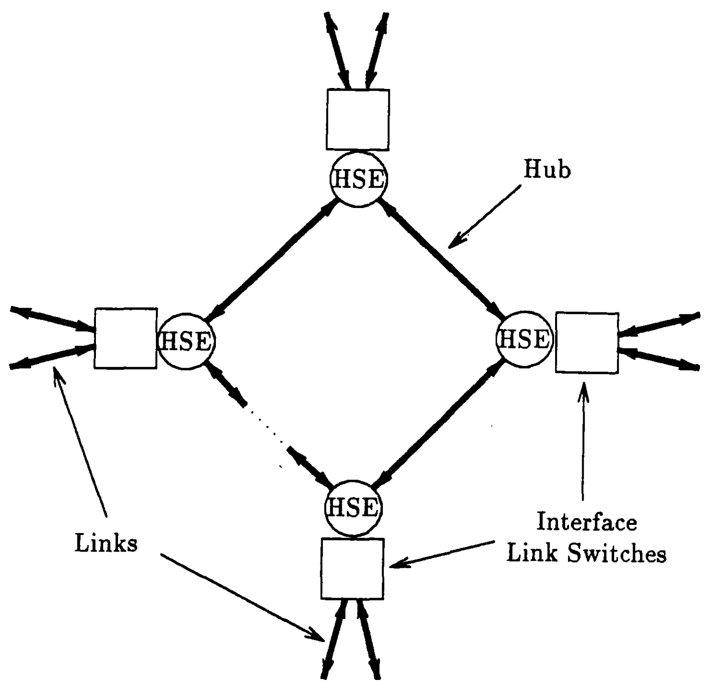

Figure 3.4 Hub Switch

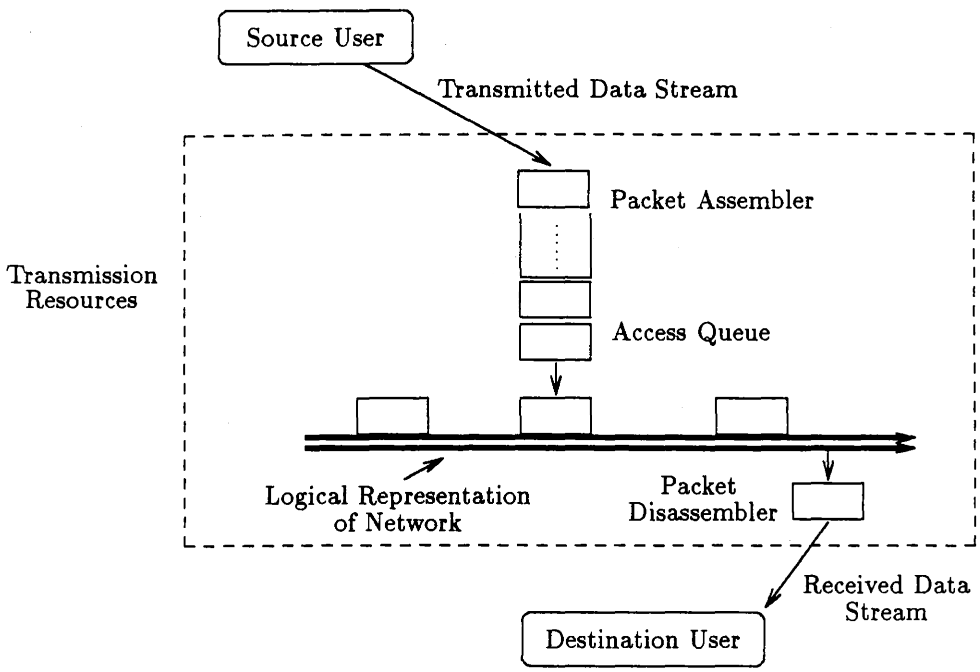

Figure 4.1 Decoupling of Service Transmission Rates and Network Transmission Resources

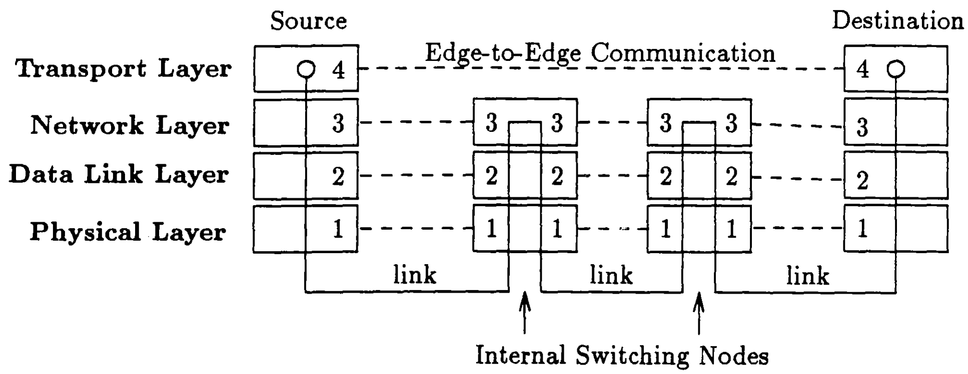

Figure 4.2 ISO OSI Protocol Model (Layers 1–4)

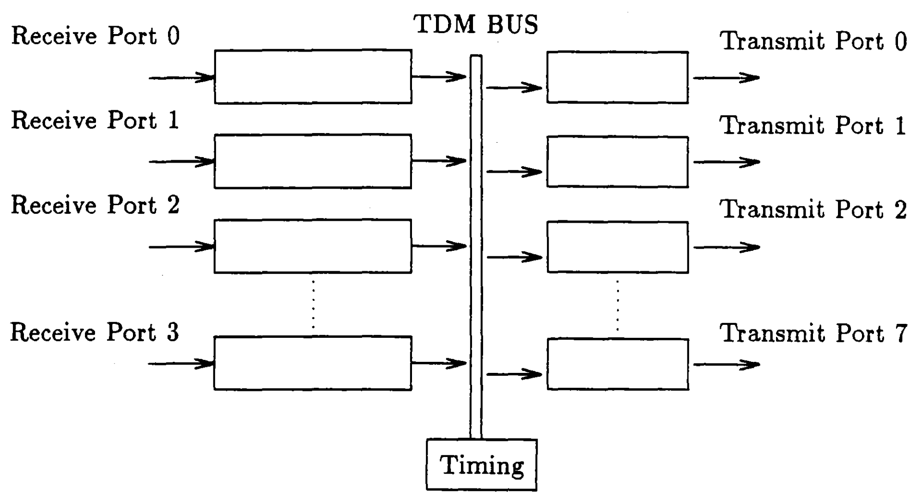

Figure 4.3 Broadband Switching Element (BSE)

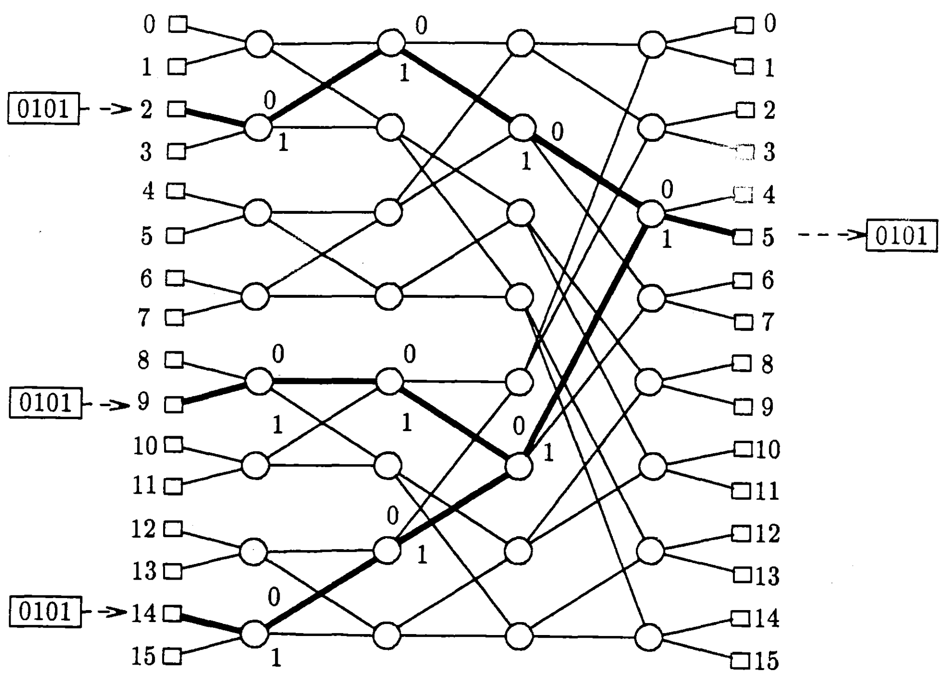

Figure 4.4 A 16×16 Banyan Switch

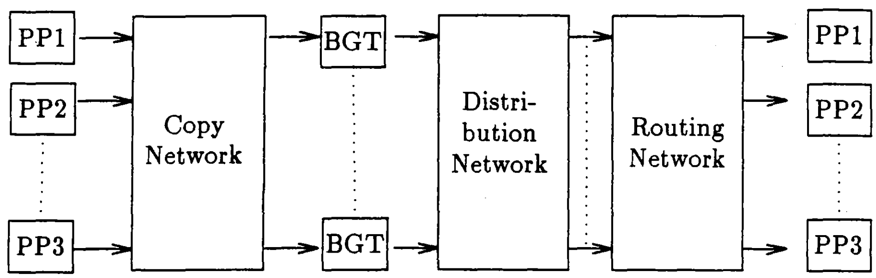

Figure 4.5 Broadcast Packet Switch Module

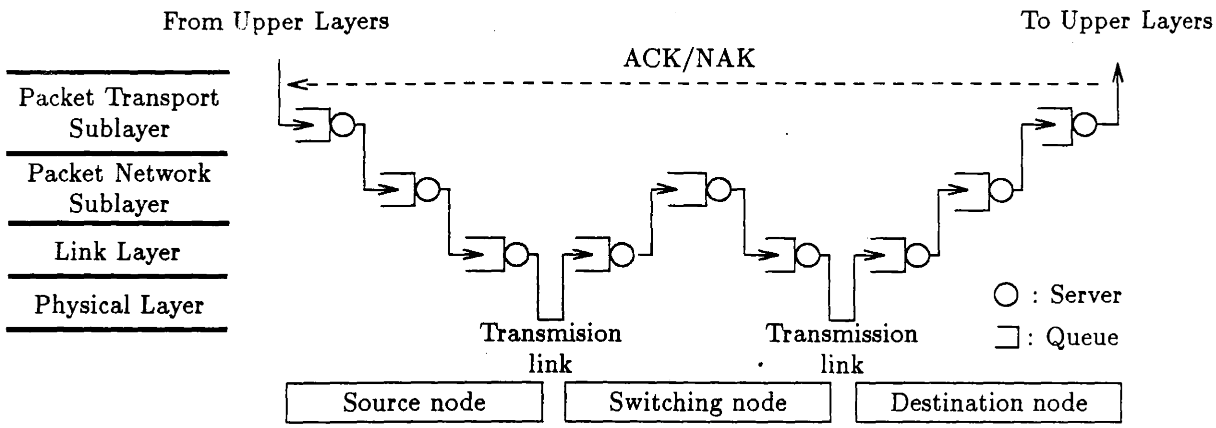

Figure 6.1 Simulation Network Model

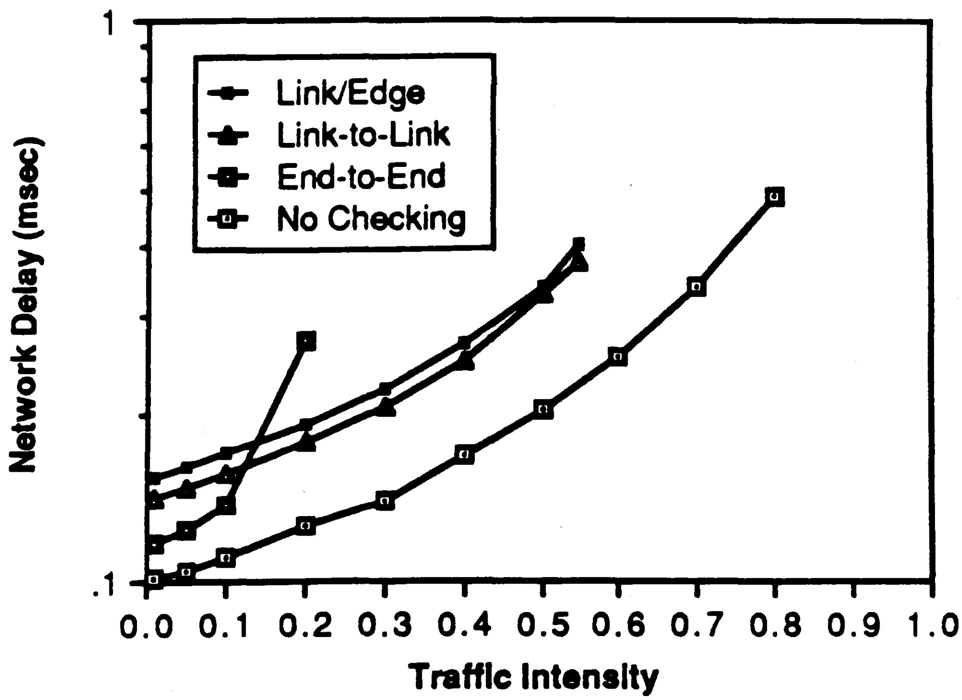

Figure 6.2 Average End-to-End Packet Transmission Delay
(go-back(n), $\rho_1 = 0.01$)

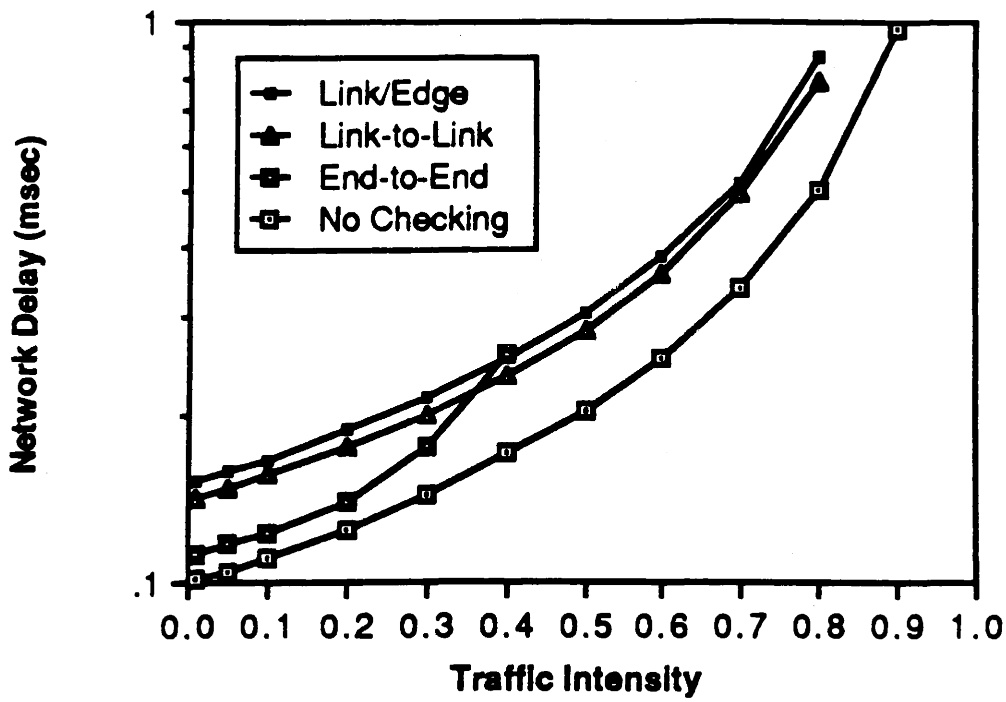

Figure 6.3 Average End-to-End Packet Transmission Delay (go-back(n), $\rho_1 = 0.001$)

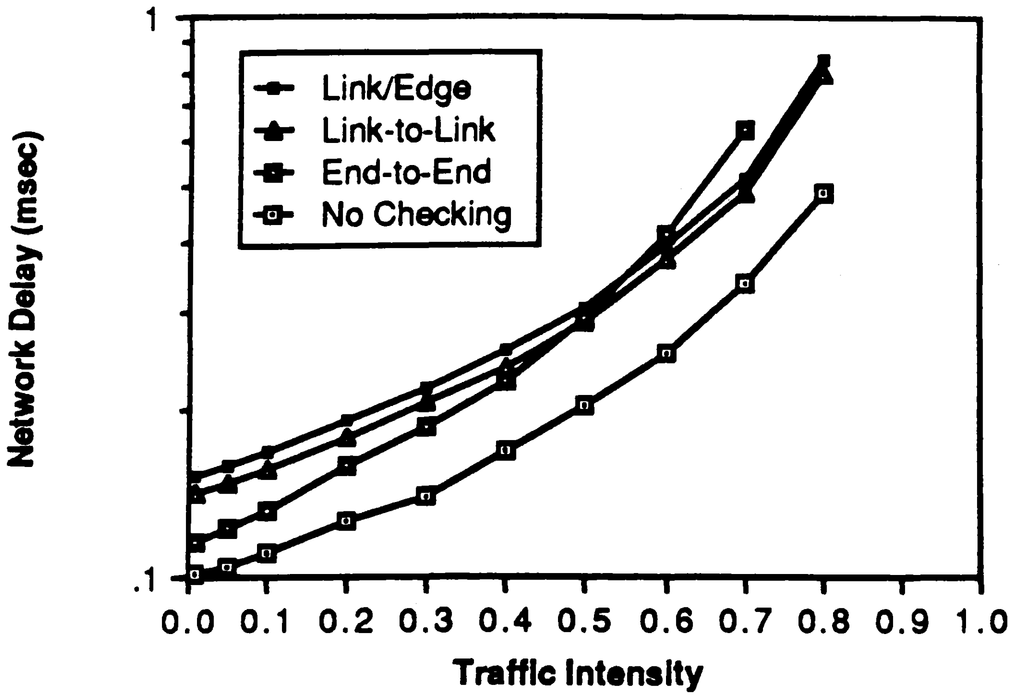

Figure 6.4 Average End-to-End Packet Transmission Delay
(selective repeat, $\rho_1 = 0.01$)

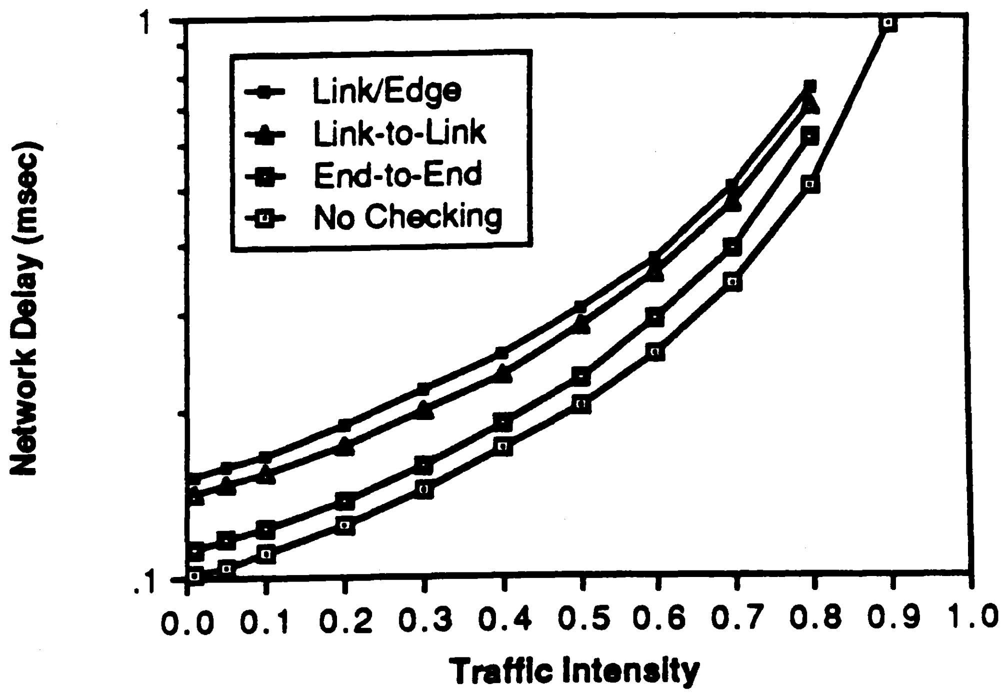

Figure 6.5 Average End-to-End Packet Transmission Delay (selective repeat, $\rho_1 = 0.001$)

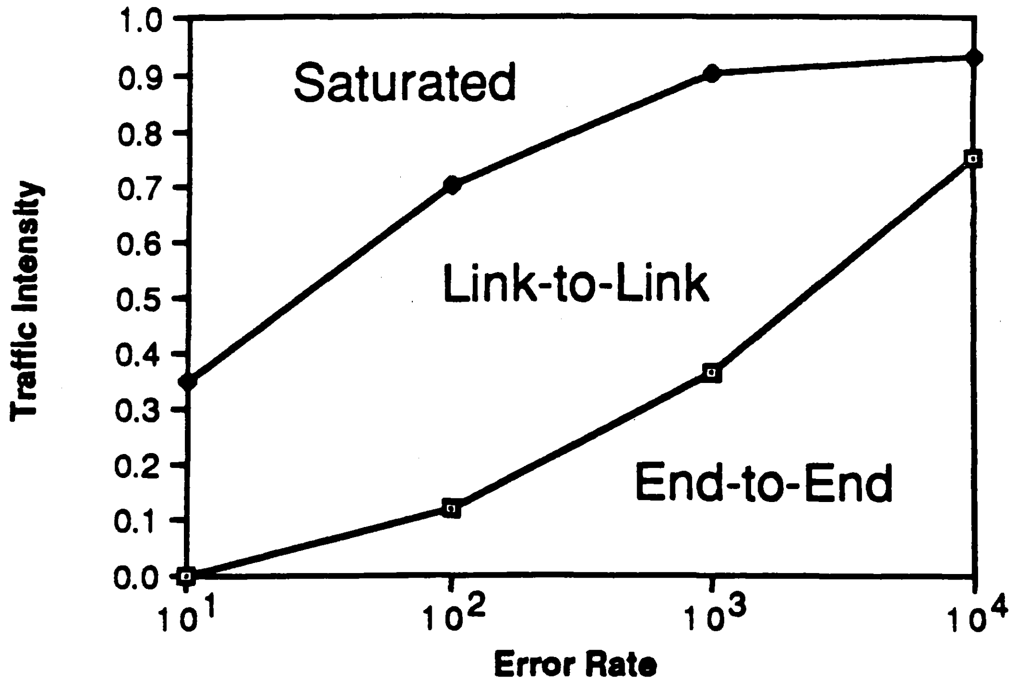

Figure 6.6 Optimal Error Recovery Schemes
(go-back(n))