Windows in the sky—Flow control in SNA networks with satellite links

by G. A. Grover
K. Bharath-Kumar

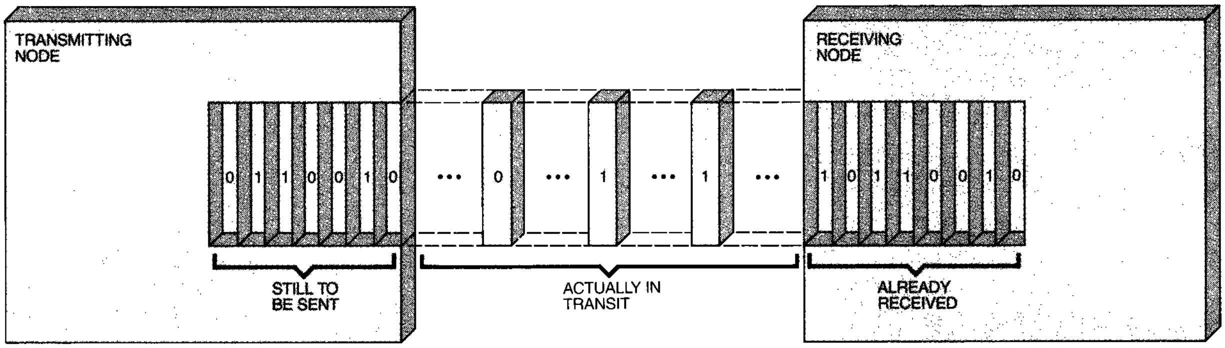
Geosynchronous communications satellites provide a unique means of high-speed computer-to-computer transmission of large volumes of data over long distances. The physical distances involved in transmitting to and from the satellites cause relatively long propagation delays for messages. In order that the high bandwidth be used effectively, large quantities of data have to be transmitted before pausing for an acknowledgment. This condition creates a potential for some new and unique types of traffic jams. This paper discusses these situations in the context of Systems Network Architecture (SNA) networks. In particular, the issues related to SNA's flow control and traffic management facilities in the presence of satellite links are discussed, along with potential solutions to ensure efficient network operation.

In science fiction, it is quite clear that the cardinal issue of communication into deep space is that of message transit time: Typically, intergalactic signaling implies propagation delays that are measured in hundreds of thousands of light years. In contrast, the reality of computer-to-computer communication via geosynchronous, or geostationary, satellites (satellites that remain in a fixed position relative to the point on the earth's surface directly beneath them) is a comparatively modest affair, involving communication into rather shallow space—roughly 22 300 miles beyond the surface of the earth. Given the speed of light, the propagation delays amount to mere fractions of a second rather than millennia.

Even so, transit times are already much, much larger than any experienced in terrestrial networks. An instantaneous snapshot of a message being transmitted between two computers over a terrestrial link would typically show some of the leading bits of the message in the memory of the receiving node (entities such as computers, terminals, and the like that send and receive messages in a communications network are referred to as the *nodes* of the network), some of the remaining bits still untransmitted in the memory of the sending node, and a few tens of bits actually flowing in the communication media "in between" the two nodes, as shown in Figure 1. While transmitting via a satellite link, however, the propagation delay is such that normally the two connecting nodes have many distinct messages between them; literally a multitude of messages are "in the air."

Message transmission protocols on communications links typically involve the sender and receiver agreeing on a certain number of messages that the sender can transmit before pausing for an acknowledgment from the receiver. This limit is established to protect the receiver buffer from overruns and to facilitate recovery from transmission errors. To sustain efficient utilization of the satellite link, it is necessary for the sender to be authorized to transmit a large group of messages at one time. It is perhaps not completely obvious why such a group needs to be

Figure 1 Terrestrial transmission



large. The essential reason is that it will take the sender a relatively long period of time to ask for and to obtain an answer from the receiver as to whether it has permission to send the *next* group of messages. If continuous transmission is to occur during this waiting period, the sender completes the transmission of messages from the *current* group just as the permission is received to send the next group (or at least not earlier than such permission is received). Such authorization groups are called *windows*.

The design of proper window schemes is important for achieving efficient message traffic control in networks, since it can impact link utilization and nodal buffer pool requirements. It can also affect the throughput and the end user's perceived response time of the network. These considerations take on additional importance in the presence of satellite links. An examination of these issues in Systems Network Architecture (SNA) networks[1] is the main subject of this paper.

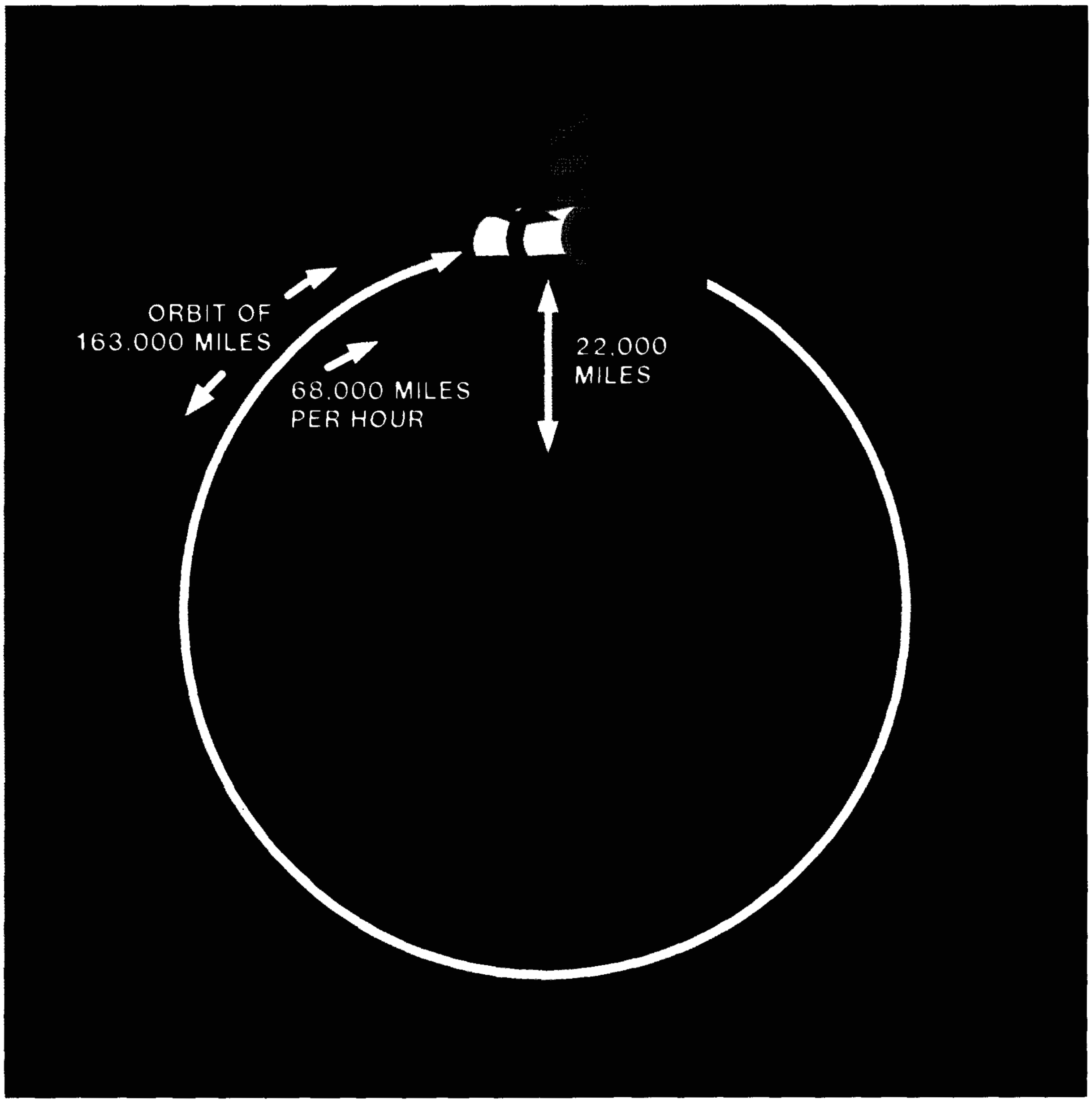
In the course of this discussion, we focus on several aspects of this subject. We begin with a brief description of geosynchronous communication satellites and how their orbits define the parameters of the data flow problem in a way that is nearly identical for all such satellites. Next we focus on window protocols, the general arrangements by which sender and receiver agree on the timing and sizes of the windows of data that are to be transmitted from the one to the other. This aspect is followed by a very brief description of two levels of message flow control in SNA networks—the link level control and virtual route level control—and the way in which windowing procedures relate to each. We then present a quantitative discussion of the impact of satellites on network resource utilization, along with some alternative solutions for eliminating or mitigating the more severe effects. Finally, we conclude by pointing out that at least in the near term one of the solutions is realizable in SNA by the application of a newly announced product function which basically had been developed for other purposes.

Satellites with stationary orbits

In considering the impact of satellites on network flow control in more detail, we need to consider the nature of geosynchronous satellites and the physical paths to and from the satellites. The propagation time for all such communication satellites is very similar because the distance over which carrier waves must travel is similar in all cases. This situation follows from the nature of the orbits of geosynchronous satellites. They are positioned at 22 300 miles above the earth in an equatorial orbit, and their speed is approximately 6800 miles per hour. At this speed the satellite travels 163 000 miles in a 24-hour day, completing exactly one orbit in perfect synchronism with the earth's own rotation. Thus we have the phenomenon of the satellite appearing to be, and, indeed, in fact being stationary relative to the point on earth that lies directly beneath it (Figure 2).

The parameter that is our main concern is the total propagation delay introduced by the satellite link. The speed of light (186 000 miles per second) and the distance of the satellite from earth of 22 300 miles implies that the signal propagation delay must be at least 247 milliseconds in traversing the link. Actually, since the hosts are ordinarily not directly

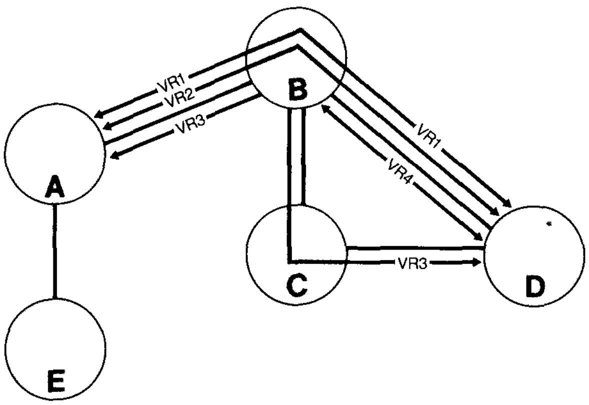
Figure 2 Stationary satellite orbit



beneath the satellite on the equator, the distances to be traversed may be somewhat longer. A delay of 270 milliseconds is normally used to take this factor into account.

If the satellite bandwidth is shared by means of Time Division Multiple Access (TDMA), an additional 30 milliseconds is added to account for buffering delays at the earth station. Of course, there are also some delays and handling times involved in going from the computer host facility to the actual satellite earth stations. To account for these additional delays, a worst-case figure of 50 milliseconds is added into the delay. Thus, a total of 350 milliseconds, or 0.35 second, for the overall propagation delay is used in the subsequent analysis.

Figure 3 Virtual routes in SNA networks



Window protocols

A common protocol among communicating entities in packet-switched computer networks is the window protocol. In general, a window protocol is employed to limit a transmitting process in the network to sending a fixed number of messages (a window) to its receiving process. The transmitting process then waits for a "go-ahead" message from the receiver before attempting to send more.

There are two popular implementations of the window protocol. One is the *sliding window* scheme where the transmitter can send up to W (window size) messages without receiving a go-ahead from the receiver. The receiver returns one go-ahead per message received. The transmitter, in turn, can send one additional new message per go-ahead. Thus, the window of messages advances or "slides" one message at a time.

The second implementation is referred to as the (W, i)-*window* scheme. Here, the transmitter maintains a permit counter (PC) that contains the number of messages that the transmitter has permission to send at any instant. The transmitter makes an explicit request for a go-ahead message once every W messages. This is done by setting a bit in every ith message of a new window, $0 < i < W + 1$. Upon receiving this tagged message, the receiver can send the go-ahead. The arrival of the go-ahead causes the PC at the transmitter to be incremented by W.

In both of these schemes, the go-ahead signal can in general either be "piggy-backed" on the messages being transmitted in the opposite direction or be sent as a separate message. The window protocols are also commonly referred to as pacing protocols, and the go-aheads are referred to as pacing responses.

Window protocols in SNA. In SNA, the link-level protocols—known as the Synchronous Data Link Control (SDLC) protocols—represent the ground floor on which the rest of the network structure is constructed. This ground floor, which in itself consists of simple protocols for exchanging messages between two nodes across a single communication link, employs flow control of the sliding window type. If the link window is $W + 1$, the sender is initially authorized to send messages 1 through W. When the sender receives an acknowledgment that message 1 has been received, it may send up to and including message $W + 1$; when message 2 is acknowledged, it may send up to and including $W + 2$, etc. For a detailed description of the protocol, see Reference 2.

VRs are routes that are logical end-to-end "pipes."

A second use of a windowing procedure is in the network end-to-end transport facility, known as the path control layer in SNA, where it is used for network flow control. At the path control layer the messages are exchanged between nodes by means of virtual routes (VRs).

VRs are routes that are logical end-to-end "pipes" that carry traffic across the network through multiple node and link hops. They can connect any two "subarea" nodes (hosts or communications controllers) in a network, as shown in Figure 3.

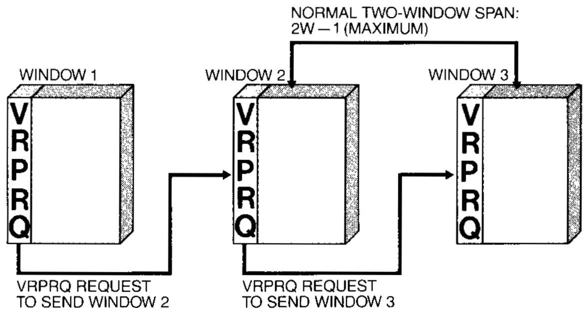
Note in this illustration that a given link can be spanned by many VRs:

- Multiple virtual routes can flow over the same physical route (particular sequence of nodes and links), e.g., VR1 and VR2. Different priorities could be one reason for having such redundant routes, in which case VR1 and VR2 are flowing at different rates.

- VRs may overlap on some links and be disjoint on others: VR4 carrying traffic between B and D overlaps VR1 and VR2; VR3, an alternate path between A and D, overlaps VR1 and VR2 on the link connecting A and B, but is disjoint from them in its path between B and D.

Virtual routes use the (W, i)-windowing scheme with $i = 1$. That is, the first message of each window carries a request for permission to send the next window. The authorization chain thereby created is illustrated in Figure 4.

Figure 4 VR flow control windows and authorization chain



Several points are worth noticing in this figure:

1. A Virtual Route Pacing Request (VRPRQ) is contained in the header of the first message of each block of messages (each window). This pacing request asks permission from the receiving node to send the next window and forms the authorization chain depicted in the illustration.
2. When answered by a Virtual Route Pacing Response (VRPRS) (not shown in the illustration; it only shows the authorization chain but not the actual transmitter-receiver data flows), the transmitter is authorized to send the next window as well as the remainder of the current window.
3. When the request to send Window 3 has been answered affirmatively, the transmitter is authorized to send all of Window 3 and all of Window 2 which has not already been sent; if the second message of Window 2 has not been sent, $2W - 1$ are now authorized, the maximum that can occur.

In this situation, it therefore follows that the receiver and also any of the intermediate nodes along the way are committed by this protocol to receive a maximum of $2W - 1$ messages at any time. When, for reasons of congestion, a node is temporarily committed to receive messages but is unable to pass them on, the quantity $2W - 1$ accordingly represents the "worst case," per VR, with which the node must deal from a storage point of view. This worst case will be used in later computations of buffering requirements.

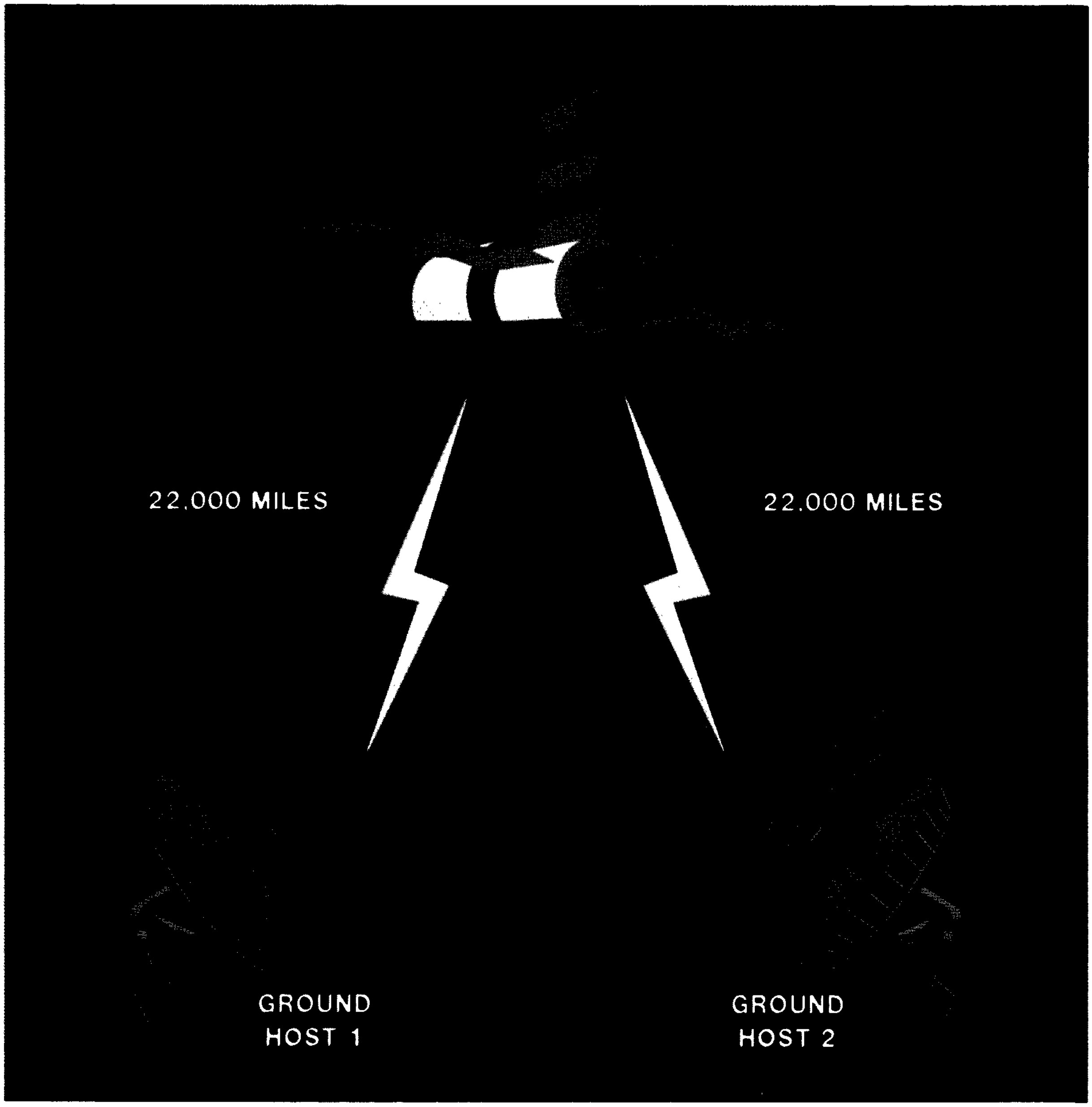
The need for controlling traffic flow in networks is well-known. The main focus of most window protocols involves matching the speeds of the transmitting and receiving processes. This function can also be described as end node buffer "protection," i.e., prevention of end node buffers from being overrun by messages. Some window protocols address congestion control at the intermediate network nodes as well. For example, the SNA VR window protocol includes the ability for an intermediate node to send a request asking the origin node of the VR to reduce the window size. For additional details on SNA flow control schemes, see Reference 3.

In this paper, we are interested in both the link and network transport-level window protocols, but we focus more on the latter. Specifically, we consider the design of window protocols to control the flow of message traffic on VRs in the presence of satellite links. But first we briefly characterize link-level transmission quantitatively.

Link-level window size considerations

Assuming error-free operation of the link, we find it is simple to calculate the minimum size for a window to enable efficient utilization of a stationary satellite link. Let us consider the first message leaving ground host 1 in Figure 5 to be transmitted through a satellite link to ground host 2. The window size defines the number of messages that ground host 1 is permitted to send, including this first message, until it receives an acknowledgment back from ground host 2 that the first message has been received. The window size therefore equals the maximum number of messages which may be, from the viewpoint of ground host 1, outstanding, i.e., known to have been sent and not known to have been received. In terms of Figure 5, the minimum time that it takes for message 1 to go from ground host 1

Figure 5 Satellite link distance



to ground host 2 and for its acknowledgment to then return from ground host 2 to ground host 1 is the time required for one round trip.

If near 100 percent utilization of the link is to be achieved, the window must be sufficiently large that ground host 1 can be sending messages throughout this round trip. Thus, the round trip duration provides the basis directly for calculating an efficient minimum window.

Assuming equal-length messages and piggy-backing of acknowledgments, we can see that this round trip time is twice the sum of the satellite propagation time (0.35 second) and the message transmission time T_r. Ideally, for 100 percent link utilization, the number of bits the transmitter would have sent, L, is simply the product of the round trip time multiplied by the bandwidth, B, at which transmission is occurring:

$$L = (0.7 + 2T_r)B \text{ bits.}$$

If the message size is M bits, T_r is given by M/B, and hence the formula can be rewritten as

$$L = (0.7B + 2M) \text{ bits.}$$

The window size in messages, W, needed to allow L bits to be transmitted is now simply L/M and thus

$$W = (0.7B/M + 2).$$

For a 230-kbit/s (kilobits per second) satellite link (the maximum size currently supported by IBM 3705/3725 communications controllers) this is

$$W = (161\,000/M + 2).$$

The SDLC protocol used today implements a window of seven *messages*. For the 230-kbit/s satellite link this rate implies a message size of

$$M = 32\,500 \text{ bits} = 4063 \text{ bytes.}$$

Since this message size is unusually large, the messages arriving at the satellite transmitter may have to be blocked to attain appropriate sizes for large messages.

For a 1.5-mbit/s (megabits per second) satellite link (offered by some satellite vendors today, with still much larger bandwidths likely in the near future), the blocking gets worse, since the message size needed is

$$M = 224\,000 \text{ bits} = 28\,000 \text{ bytes.}$$

To make it possible to utilize satellite links efficiently without having to always create artifically large messages by blocking groups of smaller messages, the Network Control Program, which runs on IBM's communications controllers, has recently announced support for a maximum window size of 127.

Even so, it can be seen that for the second example above, where the satellite bandwidth was 1.5 megabytes, a maximum of 127 messages would still imply an average message size of

$$M = 8960 \text{ bits} = 1120 \text{ bytes,}$$

so that blocking or other measures might still be required for large bandwidths. For additional consideration of the performance of data link control protocols, see References 4, 5, and 6.

VR window size considerations

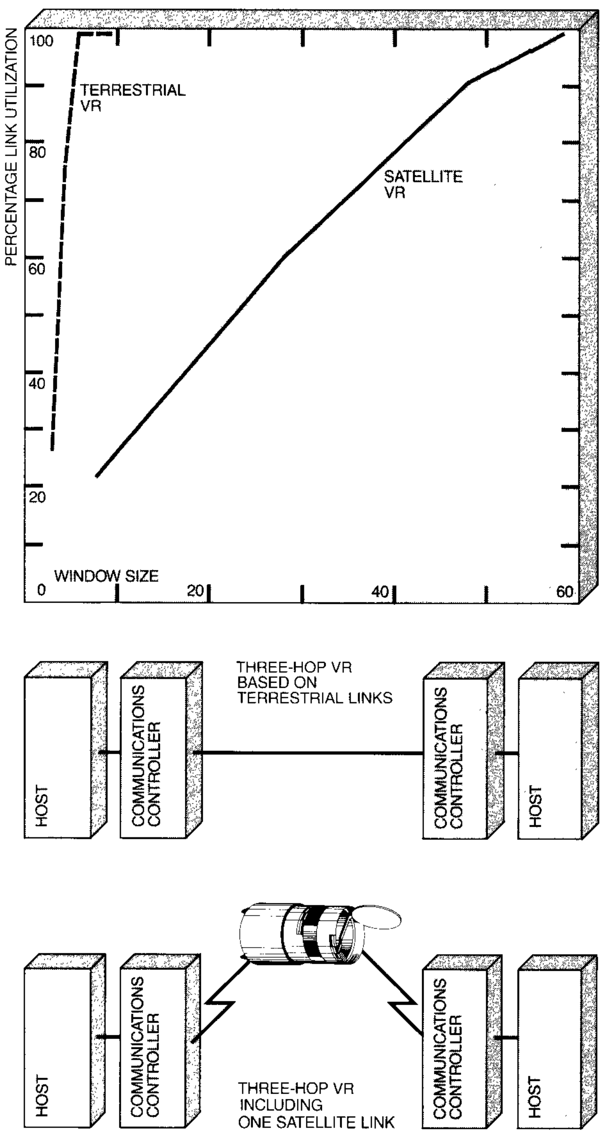
Arguments similar to the one that led to a large window requirement for link protocols also apply to the VRs of SNA networks. Since the VR window is variable, we are seeking to compute the maximum window size. It should ordinarily be large enough to permit a single VR operating alone with this window to fully utilize the links of the route. Even for terrestrial routes this window size is a function of several quantities such as number of hops, number of links per transmission group, and their speeds at each hop, etc. Currently, in SNA the recommended maximum window size is three times the number of hops for terrestrial links; this size appears to be reasonable in most situations with respect to the terrestrial portion of the network.[7] Thus, in considering VRs that use satellite links, we will only consider the excess window size requirement because of the impact of the propagation delay of the satellite link.

The window sizes can get very large for satellite VRs.

Because of the complex nature of the VR window scheme and its interactions with the SDLC Normal Response Mode (NRM) protocol, we have resorted to simulation to analyze the impact of satellites. Consider the following example consisting of a "three-hop" VR. (A three-hop VR is one with three connecting links stringing together a sequence of four nodes, e.g., VR3 in Figure 3.) All the links are 50-kbit/s links. The message sizes are exponentially distributed with average size equal to one kbit.

Figure 6 shows the link utilization as a function of the window size used. The case where the VR uses only terrestrial links and the case where one of the links is a satellite link are shown. The general

Figure 6 Window size — satellite versus terrestrial



observation that can be drawn from the figure is that in order to ensure the same level of link utilization, the satellite-based VR requires a much larger window size than the all-terrestrial link VR. Specifically, with window size equal to nine (which is three times the number of hops, the default maximum window size currently assigned by SNA products), the terrestrial VR utilizes 100 percent of the link capacity. However, this same window size results in only about 22 percent utilization of the links in the satellite case; a window size of 59 in this case is required to increase the utilization to 100 percent.

As a result of a number of simulations of data flowing over VRs that span satellite links, a simple formula for calculating the excess window for the VR, W_x, is given by

$$W_x = 1.0B/M,$$

where B is the bandwidth of the satellite link in bits per second and M is the average message size.

This value is slightly larger than the comparable size for a link window which was given earlier, and it effectively covers that amount of information generated during one second of continuous transmission. This increase to one second (the comparable figure was 0.7 second at the link level) is due to an additional delay sometimes experienced by the VR pacing response when it becomes stuck at the secondary station of the link waiting for a link-level permission (poll) to arrive from the primary. This link procedure is part of the SDLC Normal Response Mode (NRM) protocol. The value of one second was arrived at by considering several typical simulation runs.

Thus, window sizes can get very large for satellite VRs. For example, for a 230-kbit/s link and a 125-byte average message size, the excess window size is 230 messages. Compared to this amount, the terrestrial portion of the window size, which equals three times the number of terrestrial hops that should be added to this excess, becomes insignificant.

Buffer implications of intermediate nodes

Earlier it was shown that when for reasons of congestion a node is temporarily committed to receive messages but unable to pass them on, the quantity of $2W - 1$ messages represents the "worst case," per VR, with which it must deal. As window size W becomes large, the expression $2W - 1$ can be approximated by $2W$, and the extra storage requirement, worst case, for *each* node along the VR beyond the transmitter node is given by

$$S = 2MW_x = 2B \text{ bits.}$$

Moreover, since the VR is a two-way route, the worst-case implication for an intermediate node is clearly

$$I = 2S = 4B \text{ bits.}$$

Some intermediate node values for interesting bandwidths are given in Table 1.

IBM 3705/3725 Communications Controllers do not currently support telecommunication attachments in excess of 230.4 kbit/s. The bandwidths in Table 1 reflect satellite vendors' currently offered capabilities and future likely technical capabilities and are therefore indicative of the potential for explosive growth of these requirements in the future.

The problem posed by such large quantities is to find some way to prevent surges of traffic and congestion from overwhelming storage in arbitrary nodes in the VR paths that include satellite links. The entry of a large new window of data can be likened to the entry of a large group of automobiles in rapid succession on a highway. If traffic is smooth, all will go well, but if a congestion point is encountered anywhere along the way, it will be aggravated by the new arrival of a more or less continuous block of vehicles.

Existing flow control protocols contain some provision for reducing VR window sizes upon request from intermediate nodes.

Moreover, such surges and bottlenecks can easily occur not just randomly but in "steady-state" conditions. For example, the satellite-based VR depicted in the bottom of Figure 5 was simulated on the basis of all of the underlying links, terrestrial as well as satellite, operating at 50 kbit/s. More realistic would be a configuration in which the channel connecting the host to a communications controller would in fact operate at speeds of several megabits per second, so that the first hop would have one or two orders of magnitude more capacity than the second. In this situation the opening of each new window would cause most of the messages to accumulate in the communications controller at the end of the first hop, and be queued in front of the 50-kbit/s satellite link, reflecting a steady-state bottleneck situation.

Table 1 Sample bandwidths and required sizes

Bandwidth	Window Size (W_x converted to bytes)	Intermediate Node Buffer Requirement per VR
56 kbit/s	7 000 bytes	28 000 bytes
256 kbit/s	32 000 bytes	128 000 bytes
1.5 mbit/s	187 500 bytes	750 000 bytes
6.3 mbit/s	787 500 bytes	3 150 000 bytes

Finally, the preceding buffer requirement calculations are all based on the requirements imposed by a single VR. In practice, many VRs will typically cross a satellite link. (See Figure 3 again for an illustration of different ways in which VRs may overlap.) This overlap is particularly likely for a satellite link, which by its nature will often serve as a major high-capacity artery interconnecting geographically dispersed sections of the network.

Existing flow control protocols contain some provisions for reducing VR window sizes upon request from intermediate nodes. The net effect of these contention management provisions is that worst-case buffer requirements should ordinarily be much better than a linear extrapolation of the above per single VR worst cases, but still will reflect a substantial increase over the case of a single VR.

As a result of all of the foregoing considerations, it is desirable to find ways of lessening and also perhaps more clearly bounding the buffering exposures that are introduced to a significant degree throughout the entire network by the introduction of satellite links.

Solution approaches

Notions for dealing with these new flow control problems have tended to fall into two categories:

1. The potential of steady-state surges was first detected in simulations and was one of the first concrete indications of the general problem. Since these surges are associated with the opening of a VR window and since all of the messages in the window may be released onto the VR in a rapid burst, one set of approaches involves ways

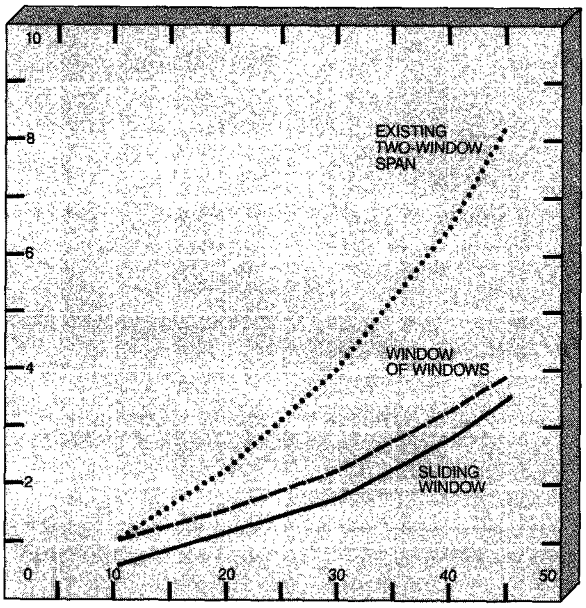


Figure 7 Variance of buffer occupancy in a VR intermediate node

in which the release of messages onto the VR might be metered so as to occur in a more gradual and even way.

2. A second approach is suggested by a consideration of the physical situation that gives rise to the problem—the long propagation delay in traversing a satellite link. It is inherent in this time delay that the receiving node directly adjacent to the satellite link will need buffers to handle large quantities of information, since at any one instant a large number of messages are "in the air" en route to this node. The receiver is committed to accept these messages even though congestion may temporarily prevent it from passing them on.

It is by no means inherent, however, that buffering requirements for nodes other than those adjacent to the satellite link will be affected. In light of this, it seems worth investigating whether there are ways of partitioning the VR, at least from the point of view of the flow control protocols, so as to limit the span of large satellite-sized windows.

Metering approaches. Several possibilities for controlling the rate at which messages are released onto a VR have been considered:

1. Time metering of the release of messages is one possibility. If messages were released at fixed time intervals, an evenness of initial release would at least be ensured. This method could be made more sophisticated by getting feedback dynamically on the degree of actual congestion on the route and adjusting the time interval.
2. A sliding window scheme, similar to the SDLC protocol, could be implemented at the VR level as well. This scheme would require the messages to carry sequence numbers. The response sent back to the VR transmitter to confirm that the message with sequence number M has arrived correctly then authorizes the sending of message $M + W$. Once W messages are outstanding on the VR, this scheme has the effect of moderating the release rate for new messages in accordance with the arrival rate for previous messages, so that a steady state without surges is achieved.
3. A difficulty with the sliding window approach is that it entails a large number of response messages—in the simplest implementation, one per message. In practice this can be reduced somewhat by "piggy-backing" responses on traffic going in the opposite direction, but there will still be common situations where an undesirably large amount of response traffic is introduced by the mechanism. A generalization of the sliding window method meets this objection by reducing the amount of response message traffic, and has been called the "window of windows." In this scheme the flow control window can be thought of as sliding forward on units that are not individual messages but rather subwindows. With such a mechanism, the pacing response confirming the arrival of the Mth *group*, or subwindow, authorizes the release of subwindow $M + W$.

The window of windows and the sliding window approaches achieved substantial buffer savings.

Both the sliding window and the window of windows approaches achieve a substantially more uniform usage of buffers than the current SNA scheme involving two fixed windows.

Figure 7 shows a comparison of alternate methods in terms of the variance of buffer occupancy in one of the interior nodes of a VR simulation configuration. Variance is the appropriate measure to use, since we are interested in quantifying the surges and ebbs in traffic that are introduced by the window control. The measure provides an indication of the extent of deviation around the average buffer occupancy—the lower the variance, the more effective the control.

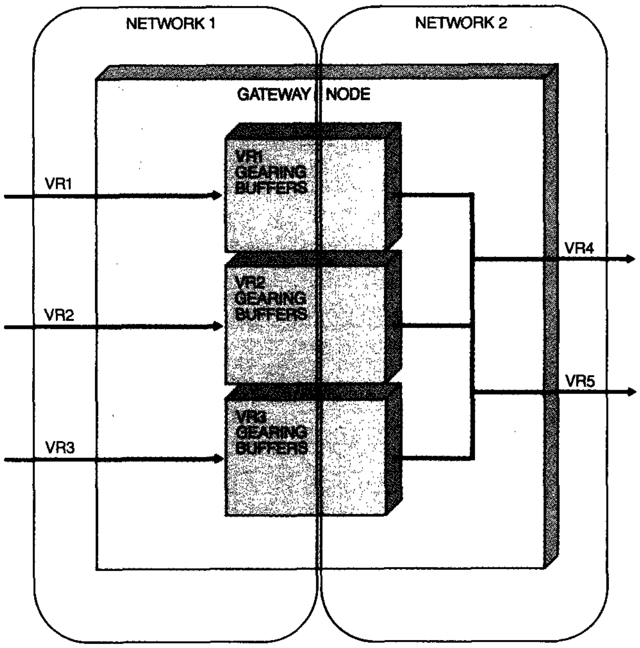
At a throughput of 40 kbit/s (all underlying links in this particular simulation case study were 50 kbit/s), the variance in buffer occupancy of the window of windows mechanism is about 50 percent better than that of the existing SNA-implemented two-window mechanism and only 10 percent worse than the sliding window. Thus, both the window of windows and the sliding window approaches achieved substantial buffer savings, and the window of windows does almost as well as the sliding window without incurring the penalty of requiring heavy response traffic in the opposing direction.

Partitioning approaches. Although metering approaches achieve a significant buffer savings *on the average*, they share the common difficulty of exposure to a worst case: No matter how evenly released, large numbers of messages (large windows) can become authorized and can accumulate at any congested node along the path.

The notion of partitioning a VR into a sequence of coupled segments, each with its own indepenent window, overcomes this problem as follows. It allows a large window to be operative over the satellite segment and the usual small window over terrestrial segments. The obvious beneficial consequence of this partitioning is to isolate and completely restrict the buffering impact of satellite links to the nodes that immediately adjoin such links. It eliminates from the other nodes in the terrestrial portion of the route the burden of providing for extra storage because of potential impact from satellite-sized windows.

It turns out that a recently announced SNA function, the SNA Network Interconnection release,[8] can play an important role in the solution. The SNA Network

Figure 8 Gearing buffer facility



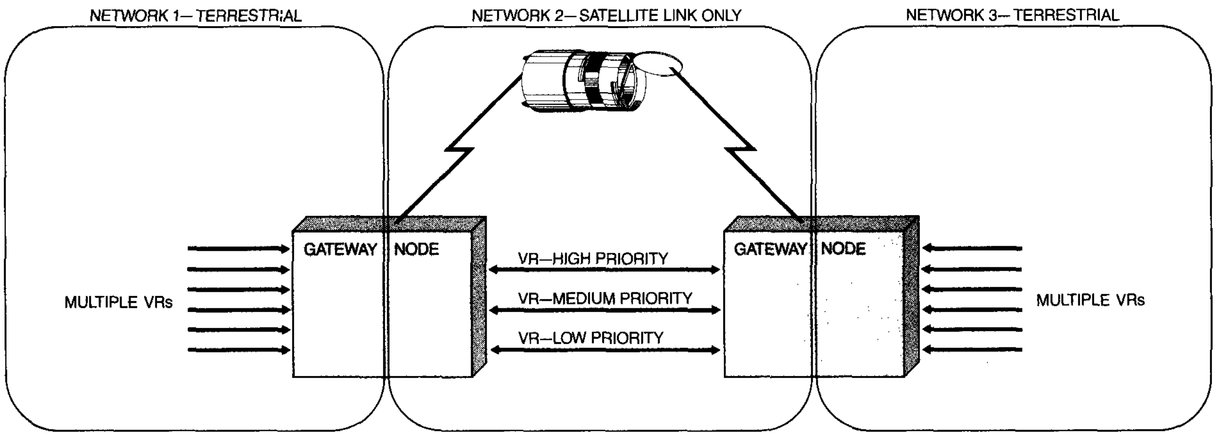
Interconnection function, or the "gateway," has nothing to do with the support of satellites *per se*, but upon analysis can be seen to provide a route segmentation solution for most satellite configurations where the ripple of excessive buffer requirements through the terrestrial portions of the network is a problem.

The consideration of the partitioning approach therefore needs to begin with a discussion of this gateway function and its associated set of supporting product enhancements, which by felicitous force of circumstance can be used to solve the satellite problem.

Network Interconnection function

For some time, there has been a growing requirement among SNA users to allow independent SNA networks to operate in concert as though they were a single network, while preserving their independence in terms of separate name spaces, address spaces, and the ability for any one of them to change without impacting the others. Some users simply wish to tie existing networks together without having to redefine them. A few users have very large networks, so large that they are in danger of running

Figure 9 **Use of SNA Network Interconnection mechanism**



out of address space (i.e., network addresses); here the ability to interconnect independent networks provides the ability to operate a composite network larger than any single network could be.

The latter case is especially interesting because it involves segmenting of VRs. Namely, in the SNA Network Interconnection product offering, when networks are interconnected, the VRs of each network cannot pass through the boundary of the network, but must be terminated there. The boundary nodes in which data enter from one network and leave to go into another (thus, "gateway" nodes) have a facility for joining together the truncated VRs from the two networks. This facility is illustrated in Figure 8.

Consider as an example VR1 in Figure 8, which is inputting data into a gateway node. The data are transmitted into a kind of special gearing buffer for VR1. This gearing buffer consists of a multiple number of VR1 windows. Data can flow out of this buffer into Network 2 on either VR4 or VR5. If the flow of data into the buffer from Network 1 is faster than the flow out into Network 2 so that the buffer reaches a preset maximum capacity, flow control pacing responses are withheld from VR1 until the buffer can accommodate another window's worth. The gearing buffer has the effect of smoothly adjusting for different rates of flow *and at the same time adjusting for the difference in window size* between incoming and outgoing VRs. It is just this property of adjusting for different window sizes that can be used to advantage in the case of satellites.

Figure 9 shows how the SNA Network Interconnection mechanism can be used to completely isolate the terrestrial portions of a network from the large VR windows needed to utilize the satellite link efficiently. In the figure, the satellite link is in effect defined as a separate network unto itself (Network 2), with VRs from the terrestrial routes on each side running into gateway nodes. Three VRs are shown crossing the satellite link to carry traffic between Networks 1 and 3 at the three allowable SNA VR transmission priority categories (High, Medium, Low). If there is no difference in priority among the traffic over the satellite, a single VR would be sufficient to join together the many VRs from Networks 1 and 3. This is an important additional point, because the large storage effects have now been isolated to only the two gateway nodes, and even in these nodes, they are strictly minimized by minimizing the number of VRs that cross the link.

Conclusions

This paper has briefly discussed the requirements imposed by satellites for large windows at the link level. Only very large blocking by the application will ordinarily be sufficient for satellite links of high bandwidth within the context of the presently supported maximum window size of seven.

Beyond and somewhat independently of these link considerations, we have extensively examined the need for large windows at the virtual route level of networks, along with ways to control and mitigate their implications.

The possibilities for further development of algorithms to deal with satellite VR windowing effects is still open for future environments as both the number and variety of satellite user configurations increase. Such possibilities remain because the use of SNA Network Interconnection carries some overhead in planning and execution—the definition of separate networks and the processing associated with crossing network boundaries. As satellite usage becomes more and more common, it may be desirable to eliminate some of this overhead.

Moreover, there are configurations where the method of partitioning the network in order to partition routes (in essence what is involved in the SNA Network Interconnection or gateway approach) becomes unwieldy. For example, if a large number of nodes are to be mutually interconnected over a single satellite link, as many as one separate network *per node* could be implied.

Notwithstanding the openness of future possibilities, for the present it is clear that the SNA Network Interconnection mechanism will provide most users sufficient ability to reduce the impact of satellite links on their network-wide storage constraints, when and where that impact is severe enough to necessitate its reduction.

Cited references

1. *Systems Network Architecture, Concepts and Products*, GC30-3072-0, IBM Corporation; available through IBM branch offices.
2. *IBM Synchronous Data Link Control, General Information*, GA27-3093, IBM Corporation; available through IBM branch offices.
3. F. D. George and G. E. Young, "SNA flow control: Architecture and implementation," *IBM Systems Journal* **21**, No. 2, 179–210 (1982).
4. W. Bux and K. Kummerle, *Data Link Control Performance: Results Comparing HDLC Operational Modes*, Research Report RZ 1027, IBM Research Laboratory, Rüschlikon, Switzerland (1980).
5. G. Deaton and D. Franse, "Analyzing IBM 3270 performance over satellite links," *Data Communications* **9** (October 1980).
6. K. C. Traynham and R. F. Steen, "SDLC and BSC on satellite links: a performance comparison," *ACM Communications Review* **7** (1977).
7. V. Ahuja, "Routing and flow control in Systems Network Architecture," *IBM Systems Journal* **18**, No. 2, 298–314 (1979).
8. J. H. Benjamin, M. L. Hess, R. A. Weingarten, and W. R. Wheeler, "Interconnecting SNA networks," *IBM Systems Journal* **22**, No. 4, 344–366 (1983, this issue).

Reprint Order No. G321-5205.

George A. Grover *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598.* Mr. Grover is a member of the Network Architecture and Protocols Group at the Research Center. From 1979 to mid-1982 he was a member of the Communication Programming System Design Group in IBM's Communication Products Division in Kingston, New York. In both of these capacities he has worked extensively in the design of SNA networking functions. Previously, he participated in assembler, compiler, and operating system design and development activities in conjunction with System/360, the Stretch computer, and the 7950 — a special-purpose extension of the Stretch computer; and in technical planning activities relating to advanced technology and to security and privacy. Mr. Grover received a B.A. from Amherst College in 1951 and joined IBM in 1954.

Kadaba Bharath-Kumar *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598.* Dr. Bharath-Kumar has been with IBM since February 1979. He is a Research Staff Member at the Research Center, working in the area of Communications Systems. His main current interest is in the area of analysis and design of computer communications network protocols. Dr. Bharath-Kumar received his Bachelor's degree in electrical engineering in 1972 and a Master's degree in automation in 1974, both from the Indian Institute of Science in Bangalore, India. He then joined the Electrical Engineering Department at the University of Hawaii, where he obtained his Ph.D. in 1979.