MIRAGE: A Model for Ultra-High-Speed Protocol Analysis and Design

Joseph D. Touch
University of Pennsylvania

David J. Farber
University of Pennsylvania

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-89-79.

MIRAGE: A Model for Ultra-High-Speed Protocol Analysis and Design

Abstract

Current protocols are expected to become inefficient if used at speeds in excess of 1 Gigabit per second. While this premise is widely accepted, no model exists to explain the phenomenon. We define a model for understanding protocols which is aimed at explaining why such a barrier exists, and indicates alternate designs which do not have this limit.

Existing protocols are akin to classical mechanics; 1 Gigabit/second is the speed near which relativistic effects emerge. In order to account for these effects, we need to express knowledge at a distance, latent measurement, and uncertainty as real entities, not negligible estimates. The result is a model which expresses not only existing protocols, and may contribute to a better understanding of the Gigabit communications domain.

Comments

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-89-79.

MIRAGE:
A Model For Ultra-High-Speed Protocol Analysis And Design

MS-CIS-89-79
DISTRIBUTED SYSTEMS LAB 1

Joseph D. Touch
David J. Farber

Department of Computer and Information Science
School of Engineering and Applied Science
University of Pennsylvania
Philadelphia, PA 19104-6389

December 1989

Acknowledgements:
This work is supported, in part, by the Industrial Affiliates Fund of the Distribued Systems Laboratory

MIRAGE: A MODEL FOR ULTRA-HIGH-SPEED PROTOCOL ANALYSIS AND DESIGN*

Joseph D. Touch and David J. Farber

University of Pennsylvania
Department of Computer and Information Science
Philadelphia, PA 19104-6389
touch@cis.upenn.edu / farber@cis.upenn.edu

Current protocols are expected to become inefficient if used at speeds in excess of 1 Gigabit per second. While this premise is widely accepted, no model exists to explain the phenomenon. We define a model for understanding protocols which is aimed at explaining why such a barrier exists, and indicates alternate designs which do not have this limit.

Existing protocols are akin to classical mechanics; 1 Gigabit/second is the speed near which relativistic effects emerge. In order to account for these effects, we need to express knowledge at a distance, latent measurement, and uncertainty as real entities, not negligible estimates. The result is a model which expresses not only existing protocols, and may contribute to a better understanding of the Gigabit communications domain.

1. INTRODUCTION

There is significant consensus that advancing the technology of future national networks in the United States will require a fundamental reexamination of protocol design [Gr87, Ra87, Po88]. A modification of existing protocol techniques will not suffice; a "clean-sheet approach," involving a revolution, rather than an evolution, in designs is needed [Fa87, Ra87].

At the high communications rates (1 Gigabit/second) suggested for future phases of the national research network, most implementations are inadequate, because their efficiency deteriorates markedly. Current protocols may have "design features that prevent them from providing the service needed at higher speeds or in large systems" [Po88].

This rate has become a barrier to protocol design, commonly circumnavigated by the use of application-specific optimizations. Omission of node processing and restricting the protocol to a subset specific to a particular domain, topology, or architecture are the more common means used to adapt existing protocols to up to 500 Megabits/second, in very restricted cases [Ja88]).

The problem with these optimizations is not only their domain specificity, but also the unrealistic environments in which these speeds have been obtained. Usually, only two-node transfers of trivial information over unrealistically small distances is tested. There is no basis for assuming that these rates can be attained in a full-scale national network with real data and delays or heterogeneous nodes and data rates.

The proposed Mirage model indicates that the barrier is due to a fundamental characteristic of raw data communication with a high bandwidth-delay product. **Mirage** denotes the difficulty with high-speed protocols, in that by the time requested information arrives, it may no longer be accurate. Nodes in a high speed network never really "see" each other; rather, they work with (and around) the mirages which time delay and high bandwidth conjure before them.

The model also suggests extensions to existing communication techniques which operate under these conditions and minimize the bandwidth that must be devoted to overhead, rather than to real information transfer. These trade-off indications are needed if the model is to be useful not only in analyzing existing or proposed protocol designs, but also in indicating future designs as well.

Note that the Mirage model is not intended to be implemented directly - as a model, it is most useful in design and analysis. It may be possible, however, to implement sub-structures such that they are effectively computable. This may be one way to design new protocols - as various sub-structures of the complete model.

---

* This work is supported, in part, by the Industrial Affiliates Fund of the Distributed Systems Laboratory.

From the Proceeedings of the IFIP WG 6.1/WG 6.4 Workshop on Protocols for High-Speed Networks, Zurich, Switzerland, 9-11 May 1989

2. PROBLEMS WITH EXISTING PROTOCOLS AND MODELS

There are several problems with existing protocol designs and models. Neither take into account high bandwidth-delay products or elapsed time. They rely on complete, non-degrading knowledge of the state of remote nodes, and assume that variations from this norm are of negligible consequence.

Most current protocols also deal specifically with only file transfers, for which optimal techniques exist. We believe that file transfers will be a small part of future high speed communications, and that communication protocols will have to incorporate more features of distributed operating systems and databases, pipelining, information redundancy (as contrasted with data redundancy), and virtual communication (as contrasted to real direct and real indirect). These issues will be discussed later here; they are presented to hint at the shortcomings of conventional, file transfer based communication.

3. HOW DOES THIS AFFECT PROTOCOL MODELS AND DESIGN

In current protocols, this local information is considered accurate and precise - it is correct with respect to its equivalent at the remote node, and it does not vary with time. The only current exceptions to this are timers maintained in the local node, which attempt to account for overall time delay, without its effect on data, except for media loss.

Another factor which impedes the use of existing protocols is the absence of *time* as it affects the communication. All protocols rely on some partial representation of remote nodes in the local nodes. Information such as the status of the data links, amount of buffer space, and last acknowledged message receipt comprise this representation, which can be considered a subset of the total information at the remote node.

While these are not new concepts in general, they have not been applied in a uniform consistent manner to protocol models. Portions of models used for distributed data and control systems are appropriate here, as is a uniform extension of the concepts of windowed flow control protocols.

4. DESIRED MODEL CHARACTERISTICS

We want to define a model that predicts and explains the widely-acknowledged speed barrier, and thus possibly indicates alternate designs which do not have these limits. This model, Mirage, should explain the deficiencies of existing protocols, and incorporate as substructures those speed-increasing optimizations which are incorporated into other proposed general protocol models. Further, the model should explain these optimizations in terms of general principles of high-speed/latency communication, and not be particular to a domain, network topology, or node architecture.

A complete model should account for the aging of its locally-kept network status, whether or not new data were received to revise this approximation. As time progresses, the accuracy of the representation should also diminish. For example, if at time T a value $V = 7$, and we know that it ages as +2 or -1 for each time-step, then at time T+2, we know only that $5 \leq V \leq 11$, and in fact V is in the set {5, 8, 11}, and also that V has twice the probability of being 8 as being either 5 or 11 (since there are two possible permutations which yield 8).

5. THE MIRAGE MODEL:

Mirage uses as complete a representation of remote nodes as is believed required to maintain communication at these rates and delays. This model allows an analysis of the effect of high bandwidth-delay product on networks whose nodes require tightly coupled distributed information, and a determination of how tightly coupled they may be given a particular bandwidth-delay.

Coupling of distributed information is defined as the dependence of two or mode nodes in a network on the same information. In this case, the information represents a subset of the state space of the nodes themselves; information which is necessary for any protocol to operate. A tightly coupled network relies on a large amount of shared information. It is assumed that shared information is identical at each node, or that some other definition of consistency will be devised according to the type and use of the particular data.

There are two aspects of the model: an analog of communication in terms of relativity theory, and an expression of uncertainty via a subspace representation of remote nodes. The model was developed via analogies to concepts from relativity, and the representation of the analogies has similarities to Feynman paths and string theory as well.

5.1. Types of communication information

Existing protocol and communications models make use of two primary forms of communication: direct, and indirect. Direct communication refers to the real transfer of data between two nodes in a system. If a node has real information about a neighbor, it is because that neighbor reported it directly.

$$\mathrm{DIRECT}(N_j) = \mathrm{DATA}(N_j)$$

Indirect communication refers to inferred knowledge, conclusions made about a neighbor's status, based on information from nodes other than that neighbor. This type of communication requires knowledge of some global system constraints. If we know that at most six of eleven nodes have a data item, and we have polled six nodes with that data item, we can infer that the remaining nodes, which we have not communicated with directly, do not have that item. Thus indirect communication involves the combination of direct communication with other nodes, and global constraints. The distributed system concept of read and write quorums is one example of such communication.

$$\mathrm{INDIRECT}(N_j) = \bigcup_{k \neq j} \mathrm{DATA}(N_k) + \text{global constraints}$$

Mirage additionally makes use of a third type, virtual communication. This is information derived about a node, based on no direct communication with any other nodes. The local node makes use of local constraints, known about the node whose status is being inferred. Virtual communication makes use of the absence of communication, the lapse of measured time, and local constraints about the node being inferred to derive that node's status.

$$\mathrm{VIRTUAL}(N_j) = \bigcup_{k} \neg\mathrm{DATA}(N_k) + \text{local constraints}$$

5.2. Physics analogies

When examining the problems of existing protocols, a rich field in which to explore is that of relativity. Existing protocols are akin to classical mechanics; 1 Gigabit/second is the speed at which relativistic effects emerge. These protocols have difficulty operating at these rates precisely because they attempt to ignore the effects of the Heisenberg Uncertainty Principle.

This principle states that an object's speed and location can never both be known precisely. In order to measure location precisely, we must assume the object has zero velocity; in order to measure its speed, we must measure the time it takes to traverse some distance, in which case we don't know quite where in that span the object is.

In protocols, the effect emerges as the latency increases with respect to the transmission rate. As the transmission rate increases, more information is in transit among nodes at any given instant. That information will alter its destination node; as a result, it embodies the uncertainty we try to avoid. At Gigabit rates, the amount of information in transit exceeds the capacity of the node.

In that case, our representation of the node has no consequence, since information in transit - which we know nothing about - can completely overwrite the node by the time we interact with it. In effect, we never know what message to send, since the node we think we're communicating with could change completely by the time the message arrives. Any assumptions we make about the capacity of free buffers in the destination node, the current active data links, etc., must be assumed to be in error. This severely disturbs the assumptions of existing protocol models.

The effect of the Heisenberg principle is to invalidate the local representation of the remote nodes to such an extent that the variations between the local representation and the actual remote node is so great that communication breaks down. It has already been noted that existing protocols have difficulty with the notion of uncertainty, especially that "the receiver has to cope with much more uncertainty than does the transmitter."[Fr88]

There are other analogies in physics to communications. In particular, quantum field theory is a direct analog to communication. Packets are quanta, nodes are particles, and nodes affect other nodes by the exchange of quanta. There is a time delay in the information exchange, and the effect of one node on another is related to the size of the nodes, the time delay in the exchange of quanta, and the size of the quanta themselves.

There are several conclusions derived from field theory, which may be applicable to this model. These include the notion that small nodes at large distances simply cannot effectively interact (field strength and dissipation of field effect over distance), and the size of the quanta is directly proportional to the strength of the interaction of the particles. This may be an indication that some types of strong interactions, of tightly coupled information, cannot be maintained using large bandwidth systems, given the constant speed of light.

The Mirage model, specifically the modelling of time as a parameter, and denoting the variation of all variables with respect to each other as well as time, is similar to the model presented by Feynman path theory. This theory attempts to explain the interaction of conventional particles, such as electrons, in so far as they behave as waves.

For example, electrons passing through a double-slit interfere, just as electromagnetic waves do. The problem in understanding this interference is that a single electron stream, when scanned across the slits, exhibits the same interference pattern as would a diffuse point source. For this phenomenon to occur, an electron would, in effect, have to pass through both slits simultaneously, and interfere with itself, to cancel out in the appropriate locations. Feynman path theory explains this by suggesting that particles do not travel through a precise path in space; rather they travel through all possible paths in space, each with an associated probability. In effect, as the electron travels, its essence spreads out to a probabilistic wave, and in fact the electron is a conventional point entity only when detected.

Feynman path theory represents the uncertainty of travelling through various paths as a probabilistic wave. In Mirage, we represent this uncertainty as the volume of the subspaces (to be explained forthwith), and consider the entity modelled to be somewhere within that volume, rather than any specific point in particular. By assuming only that the entity is within the volume, and reacting to the entity as if it were anywhere within, we are assured of affecting it in the proper way.

5.3. Expanding state-space model

First we define the following terminology:

Information is a finite data value, which may be static or dynamic. A **system** is an entity which controls information. **Control** involves storage, retrieval, and transformation of that information. A **protocol** is a method of interaction among components of a system. A **node** is an indivisible component of a system. Nodes are annotated as:

$$N_{1 \ldots m}$$

The **state** of a node is a finite set of data values. The collection of all the data values of all the nodes determines the information of the system. States are annotated as a function on nodes, consisting of a finite set of values (to be described later):

$$S(N_i) = \{v_1, \ldots, v_k\}$$

A node's **perception** of another node is some subset of the other node's state. Perceptions are annotated as a function on nodes:

$$P(N_i) \subset S(N_i)$$

A node's **view** is its own state and its perception of all external nodes. Views are annotated as a function on nodes:

$$V(N_i) = S(N_i) \cup \bigcup_{j \neq i} P(N_j)$$

Data reception causes the state of a node to be updated, or transformed. We annotate data reception as a function on states. By extension, this function can also apply to perceptions of that nodes state:

$$P'(N_i) = DATA(P(N_i))$$

5.3.1. Current protocols - notation

In current protocols, the data items stored in each node are integral values, e.g. elements of the integers (actually some finite subset of the integers):

$$v \in I$$

The set of values which comprise a node's state thus represent a **point** in state space. When a node either receives or transmits data, it transforms its view into a new view. This transformation is equivalent to moving that point to a new point; that new point is composed of values from the previous point and values from the data involved. Note that transmitting and receiving data are represented by the same operation, and time is not modelled.

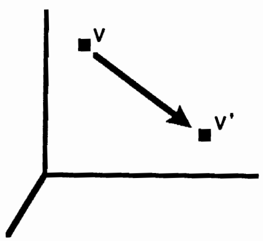
Remember that a view is:

$$V(N_i) = S(N_i) \cup \bigcup_{j \neq i} P(N_j)$$

A data transformed view, data being received at node i from node j, is a function of the previous state and the transformed values of the state of the node being reported:

$$V(N_i) = S(N_i) \cup DATA(P(N_j)) \cup \bigcup_{k \neq i,\ k \neq j} P(N_k)$$

Note that both transmitting and receiving data are modelled as point transformations in state space:



5.3.2. The Mirage model - notation

In Mirage, the data items store in each node are subsets of integral values:

$$v \subset I$$

The set of values which comprise a node's state represent a **subspace** in state space. Transmitting and receiving data, and time are each modelled uniquely.

Transmitted data causes some other node's state to potentially change to a new state. If we send data from node i to node j, then the view of node i must accommodate the potential receipt of that message by node j. That node's view becomes the union of two views - one as if the message were lost, the other as if it were received. Other nodes in the system are ignorant to this transfer, unless informed otherwise. Thus data sent from i to j causes the view of i to change. The result is the union of the state of node i, the *union* of the perception of node j and its data-transformed perception, and the union of the perceptions of the remainder of the nodes:

$$V_{Trans}(N_i, DATA, N_j) = S(N_i) \cup [DATA(P(N_j)) \cup P(N_j)] \cup \bigcup_{k \neq i,\ k \neq j} P(N_k)$$

Note two things about this formula. First, the subspace of node i expands, by being unioned with a subspace transformed by data reception. This implies that the transmission of data necessarily disturbs a node's view of the system, and makes that view *less precise* .

Receiving data at node i from node j causes node i's view to be made *more precise* , by contrast. The data is more recent information about the view of another node, so node i's perception of that node is updated. The result is the union of the state of node i, the *intersection* of the perception of node j and its data-transformed perception, and the union of the perceptions of the remainder of the node:

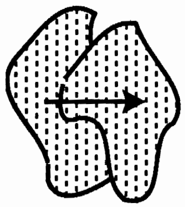
$$V_{Recv}(N_i, DATA, N_j) = S(N_i) \cup [DATA(P(N_j)) \cap P(N_j)] \cup \bigcup_{k \neq i,\ k \neq j} P(N_k)$$

In Mirage, the perception of a node incorporates subsets of all its potential states. The intersection taken at the reception of a message implies that the state indicated by the message must be a subset of the perception of that node.

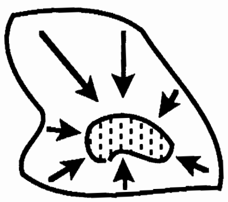
Time is represented as a function on states of nodes, and, by extension, on their perceptions as well. The action of time is to expand the set of possible states which the node may occupy. The effect of time on the view of a system is calculated by the effect of time on its components:

$$V_{Time}(N_i) = TIME(S(N_i)) \cup \bigcup_{j \neq i} TIME(P(N_i))$$

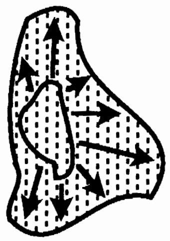
With respect to the operations being performed on the subspace of the views, the operations of transmitting data, receiving data, and lapsing of time are thus represented as follows. The shaded areas are the subspace after the operation, and the arrows indicate transformations which occur.



Transmit



Receive



Time

5.3.3. Discussion

Current protocols maintain information about remote nodes as points in a state space. For example, each node knows the precise active links numbers, amount of buffer space available (i.e. window information, in the sliding window protocols), and last acknowledged transmission. While the sizes of the send and receive windows is an attempt to account for information in transit, there is no attempt to account for the aging of this information.

Let us assume that information is received at some time, which denotes the representation of the remote node, as it reports itself to the local node. We also assume that some guarantees exist, also self-reported, as to the variability of the information presented as a function of time [Me76a]. Then the local node can create a representation of that remote node as a point in state space denoted by the reported information. It can begin with a point in the state space of the representation corresponding to this information.

Since the information was received some time after it was sent, that time must be used to update the state space representation. Since we know only that time has elapsed, but not how it has specifically affected the remote node, the former point in state space is expanded to a subspace. We now use the notion of subspaces to denote the information in remote nodes. Each node, in order to participate in a protocol, must maintain information about remote nodes; this information must be considered a finite subspace in the possible state space of the node representation, rather than a precise point in that space. Each dimension of the subspace is determined by the time-variant guarantees reported, over the time interval elapsed. That time interval can be computed either from some form of global clock (assuming such exists), or some reasonable estimate of the transit time of the messages.

The 1 Gigabit/sec barrier is the rate at which remote nodes cannot be effectively approximated by points in state space. The amount of information in round-trip transit between nodes is of the order of magnitude of the size of the node itself, so common feedback techniques no longer suffice. Some sort of anticipatory, or constrained initiative feedback is required, and is modelled here by the use of the subspaces.

When a subspace is expected to expand (due to time) into a region of invalid operation, messages are sent which correct the subspace of the remote node, in an anticipatory fashion.

Message transfer is modelled as a transformation on that space, feedback as collapsing a large possible subspace to a smaller one (about which location is known to a greater certainty), and time as the expansion of the subspace in the directions of possible computation.

In the conventional protocol models, remote nodes are modelled as points in state space. In that case, each action of the protocol effects the location of that point. Sending and receiving data both move the point to a new location (i.e. to represent a lost or gained buffer at the remote node, respectively). Time is not modelled in these protocols as part of the state space representation; timers and the like represent information about the communications link, not the remote node and its delay in responding.

In the Mirage model, incoming and outgoing data affect the subspace representation of the remote node differently, which is a more intuitive result. When a packet is sent, there are two possible resultant subspaces; one represents the node before it receives the packet (i.e. the subspace before the packet was sent), the other represents successful receipt of the packet. The result is the union of these two possible subspaces. This is done to maintain an invariant that the actual state of the remote node, the point in state space where it lies, should always lie in the subspace of the representation.

The receipt of a packet from the remote node indicates information about that node. As such, it refines the subspace to a sub-region thereof. Incoming packets thus constrict the subspace, collapsing it to a smaller region. Note that this region is also a subspace, not a point, accounting for the travel time of the packet from the remote node to the local node.

It is the responsibility of the protocol both to maintain the subspace representations, and to ensure that the subspace remains within the defined regions. When a subspace expands beyond the boundaries of the state space which denote error conditions (i.e. overflow/underflow of buffer space, etc), messages are sent which correct the condition. In the best possible protocol, the expansion of the subspace to within some warning region would trigger this activity; the dimension of the warning region would be calculated such that the estimated round trip for the correcting/warning message and the receipt of the reply would prevent the subspace from expanding into an invalid region during the transit. Again, we rely on a reasonable estimation of round trip time in order to compute these warning regions.

6. COMPARISONS TO OTHER MODELS

Some existing protocols or protocol variations incorporate some aspects of the Mirage model, but none is as complete. Mirage attempts to unify aspects of several types of models, from distributed operating systems, partitioned databases, and general feedback and control systems. The following is a preliminary analysis of the Mirage components in relation to existing work.

6.1. Feedback

In particular, the issue of transmitter/receiver feedback as anticipating state space expansion has been used successfully in windowed flow control schemes, albeit in a discrete and restricted fashion, with respect to the level proposed here. The concept of expansion of state space upon data transmission, and collapsing upon message receipt, has been examined in the design of buffer "barriers", a modified flow control which attempts to equalize the uncertainty of communication among transmitter and receiver [Fr88].

6.2. Time

The incorporation of time into protocols has thus far been limited to two methods, where time is modelled either by boundaries or by finite time-steps. In the former, time is denoted by "$T_1 \leq T \leq T_2$", for some T_1, T_2. Actions occur when these boundaries are exceeded, as in time Petri Nets, temporal logic, or time-out timers in more conventional models [Sc82]. In the latter, actions occur at some time, where "$T = T_1$". In Mirage, time is a fully parametric value. The values of all other entities may change over time, and no barriers or finite points exist. Time is considered a continuous entity, over which other entities vary.

The modelling of time as an aging variable [Sh82] is not as general a concept. These time markers age, but other entities do not vary with time; aging is considered a static process. The aging of an entity reduces the precision to which the value of that entity may be known. The aging of other variables may affect the aging of a variable, and so the dependency of the variables on time is arbitrarily functional and interdependent. In the Mirage model, time is allowed to transform the subspaces arbitrarily (the transformation is known at the time it is invoked, but the model does not restrict the transformation a-priori).

There exist models of time in protocols which are more closely related to that of the Mirage model. Incorporating time as a valid period for each state of a protocol machine is akin to denoting the interval over which the expansion of the subspace is well-defined [Ag83]. The extension of this method, which presents hold times of protocol states as cumulative distribution functions, is very much similar to a time extrapolation of this state space, Feynman paths incorporated into strings theory. The paths, over time, are represented an extrusions; the node is considered to occupy all states, rather than any one individually.

6.3. Subspace restrictions

The notion of restricting a machine to operate only within the valid subspace is an extension of distributed/replicated database techniques, most notably read/write quorum strategies. In addition, there have existed designs for external entities, which maintain the operation of a system to within some desired constraints [Fa76]. The use of these environments or supplemental programs to warn of dead-ends, maintain locality, and restrict other programs to within some valid subspace is similar to the methods used here. Mirage differs in that it appears that these notions are central to the operation of the protocol, not external, supplemental constraining devices.

6.4. Equivalent substructures

As noted, the Mirage model is not suitable for direct computation, but equivalent substructures of the model may be. This is accomplished by a more coarse-grained partitioning of the state space, via equivalence relations. The subspaces become partitioned as a result of the imposed homomorphism of the relation, and particular location in the subspaces becomes less important than the traversal of these boundaries. This work has been examined as "protocol projections" [La82, Sh82], and is a substructure of the Mirage model.

7. OBSERVATIONS

There are some points of interest, with respect to this model. In particular, there is no assumption that the dimensions of the state space representation are strictly orthogonal. The functional interdependence of the dimensions of the state space of the nodes has been examined with respect to local and global buffer space [Ak88]; other interdependencies also exist, especially with respect to the dependence of variables on time.

Note that the model has two uses. First, as an analysis technique, and second (if directly implemented) as a possible solution. The latter will prove difficult, since we believe a direct implementation would be intractable (most likely exponential, rather than NP). This does not preclude the possibility of partial direct implementations, of discrete subsets of subspaces of interest, or of substructures of this model which are effective for reasonable systems.

7.1. High bandwidth-delay product as a cause of the problem

One of the results of viewing high-speed protocols in the Mirage model is to see the effects of high bandwidth-delay product on the communication possible. Since the amount of data in transit is large with respect to the size of the node, the number of transmitted messages, causing state space expansion, is large. These messages have not yet been received, thus no corresponding collapsing of the state space occurs.

It seems apparent then that the problems which high-speed protocols should specifically address are related to this phenomenon. In local or metropolitan area networks, the latency is insufficient to effect the system in this way.

Again, this is a relativistic type of effect. Latency is the only constant in the system. If the effects of latency are negligible - over a small area, for example - then there is no difference between a high-speed protocol and a low-speed protocol running with an accelerated local clock, over faster media. Latency indicates the speed of the network to the protocol; where it is large, the protocol must be redesigned.

7.2. Information vs. data communication

Further, there is a fundamental problem with unrestricted communication in media with high latency. Raw Gigabit rates lose meaning in a situation where the data cannot be regulated. Links must maintain self-reported constraints, such that the subspaces do not grow beyond the valid state space boundaries before a message can travel between two nodes. This indicates that the link must be configured to the traffic which it carries, rather than any generic raw data. As such, aggregate information flow between any two nodes may total a Gigabit, without exceeding the constraints of the model.

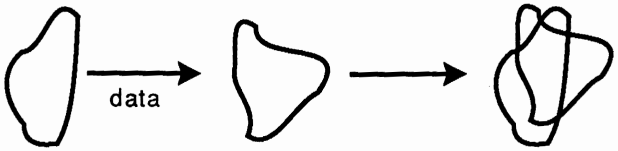
The notion that there exist constraints on communication links, and that they have optimal values has been examined with respect to the determination of optimal window sizes in windowed flow control schemes. The effect of the constraints is especially apparent when the interdependence between local link windows and global (overall network input/output) network windows is examined [Lu88]. Mirage attempts to extend this search for optimal interdependency values to multi-dimensional systems.

Note that these constraints pose a radical change in the view that the data stream is transparent at the lower levels of the protocol. The requirement of consistency constraints, and their monitoring, requires the lower levels of the protocol to access the data directly, rather than treat it as a sealed, private entity.

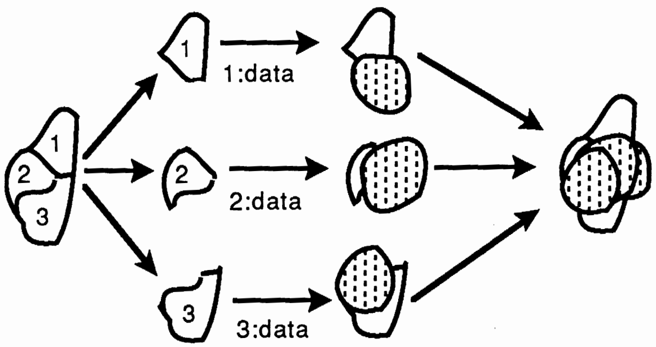
7.3. Guarded messages - an indicated potential solution

One of the more interesting research areas indicated by the Mirage model is that of guarded messages, a communications extension of Dijkstra's guarded commands. In a guarded message, the data is preceded logically by a guard condition. If the receiver satisfies the conditions of the guard, the message is accepted, otherwise it is ignored. Note that simple analog of this concept occurs in windowed flow control, since the receive window acts as a guard function on the message sequence numbers.

In guarded messages, we use the guard to restrict the effects of the data message on only a portion of the subspace of the receiving node. This reduces the possible state space of that node, in its perception at the transmitting node.



Without guarded messages, entire space is affected.



Guarded messages provided selective reception.

Further, it becomes possible to send information redundant messages. In current protocols, it is common to send identical copies of data, to reduce loss due to corruption. In this model, there is an indication that two messages with null-intersecting guards are information redundant. Messages can be constructed to have equivalent results on the receiving node. They are data distinct, but thus information redundant. The data stream may cease to be hidden to the lower levels of the protocol, in these situations.

There are two advantages to the use of guarded messages. First, the expansion of the state space of a transmitting node can be constrained sufficiently to remain tractable. Second, the information redundancy used by these replicate communication paths provides a use of the additional bandwidth provided by the high speeds of the problem domain. In effect, we then use the phenomenon which causes the effect - the high bandwidth product - to cure it, by using the latent transmission to store replicates of the same information content. It is this latter approach that we believe holds the most promise, in its effect on the utilization of the extraordinary bandwidths to be provided by proposed networks.

REFERENCES

[Ag83] Aggarwal, S., and Kurshan, R. P., Modelling Elapsed Time Protocol Specification, in: Rudin, H., and West, C. H. (ed Protocol Specification, Testing, and Verification, III (1983) 51-62.

[Ak88] Akyildiz, I. F., Performance Analysis of Computer Communicat Networks with Local and Global Window Flow Control, Proceedings IEEE Infocom (1988) pp. 401-410.

[Fa76] Farber, David J., and Pickens, John R., The Overseer: A Powe Communications Attribute for Debugging and Security in Tl Wire Connected Control Structures, in: Proceedings of IE International Communications Conference (1976)

[Fa87] Farber, David J., et. al., notes from meetings of the Ultra-Hi Speed Information Networks groups at the University Pennsylvania (uhsin@dsl.upenn.edu, 1987)

[Fr88] Fraser, A. G., Current Research on Local Area and Wide A Networks, in: Report on the Advanced Computer Communicat Workshop Mar. 30-31, 1987 (Information Sciences Institute, (1988) pp. 27-29.

[Gr87] Graham, William R., A Research and Development Strategy High Performance Computing, in: Report of the Executive Office the President (Office of Science and Technology Policy, N 1987)

[Ja88] Jacobson, Van, Congestion Avoidance and Control, in: A Computer Communication Review (Oct. 1988) pp. 314-329.

[La82] Lam, Simon S., and Shankar, A. Udaya, An Illustration of Protocol Projections, in: Sunshine, C. (ed.), Protocol Specification, Testing, and Verification (North-Holland, 1982) pp. 343-360.

[Lu88] Luan, Daniel T. D., and Lucantoni, David M., Throughput Analysis of a Window-Based Flow Control Subject to Bandwidth Management, in: Proceedings IEEE Infocom 1988, pp. 411-416.

[Me76a] Merlin, Phillip M., and Farber, David J., On the Recoverability of Communication Protocols, in: Proceedings of IEEE International Communications Conference (1976) pp. 21-26.

[Me76b] Merlin, Phillip M., and Farber, David J., Recoverability of Communication Protocols- Implications of a Theoretical Study, in: IEEE Transactions on Communications (Sept. 1976) pp. 1036-1043.

[Po88] Postel, Jon, Summary of National Communication Initiative Issues, in: Report on the Advanced Computer Communication Workshop Mar. 30-31, 1987 (Information Sciences Institute, CA, 1988) pp. 43-44.

[Ra87] Raveche, Harold J. (ed.), A National Computing Initiative, in: SIAM Workshop Report, (Phila, PA, 1987)

[Sc82] Schwartz, Richard L., and Melliar-Smith, P. M., From State Machines to Temporal Logic: Specification Methods for Protocol Standards, in: Sunshine, C. (ed.), Protocol Specification, Testing, and Verification (North-Holland, 1982) pp. 3-19.

[Sh82] Shankar, A. Udaya, and Lam, Simon S., On Time-Dependent Communication Protocols and Their Projections, in: Sunshine, C. (ed.), Protocol Specification, Testing, and Verification (North-Holland, 1982) pp. 215-233.

[Wu85] Wu, William W, Elements of Digital Satellite Communications, v2 (Computer Science Press, MD, 1985) pp. 375-555.