Distributed Fiber Optics Systems for Commercial Aircraft

Final Report

Principal Investigator: Dr. David Game

Department of Physics and Computer Science

Period: July 18, 1991 through December 26, 1991

CHRISTOPHER NEWPORT UNIVERSITY
NEWPORT NEWS, VIRGINIA 23606-2998

NASA Research Grant NUMBER NAG-1-1309

July 13, 1994

Prospective Communications Research to Support Fly by Light / Power by Wire

David Game
Department of Applied Physics
and Computer Science
Christopher Newport University
Newport News, Virginia 23606

July 7, 1994

Abstract

A NASA Research Grant NAG-1-1309 - Distributed Fiber Optic Systems for Commercial Aircraft was awarded to David Game during July 1991 effectively providing partial release time from his duties as a professor of Computer Science at Christopher Newport University for the purposes stated in the grant proposal. As a result of miscommunication between the investigator and NASA contact, the investigator did not submit a report at the end of the time-frame funded by the grant. This report primarily constitutes a summary of findings of the original background research done at that time.

1 Background

NASA is embarking on a research project to design the next generation of commercial aircraft, Fly by Light / Power by Wire. The objectives of this effort are to improve commercial aircraft design by

- reducing the weight of the aircraft to improve efficiency and
- improving the fault-tolerance and safety of the aircraft by

  enhancing current systems with new technologies, or

  introducing new systems into the aircraft.

1.1 Fly by Light

The Fly by Light component of this research effort addresses the use of optical sensors and fiber optics in conjunction with electronic computers to perform control and other support functions in the aircraft. Today's electrical components are susceptible to interference and other types of electromagnetic forces, whereas the fiber optical components are not. Therefore, replacement of the electrical components with optical components provides an opportunity to reduce significantly the probability of failure due to these types of errors. In addition to the improved immunity to factors in the operating environment, optical systems have bandwidth improvements on the order of ten to one hundred times that of current digital electrical systems and this factor may grow significantly higher depending on the results of current research in coherent transmission and reception. The utilization of such large bandwidths in aircraft systems has not been investigated and it is anticipated that it will allow for both new systems and for fault tolerance in the communication systems. The design of the optical sensors will be headed by NASA Lewis, the sister site for the research on the project, and the design of the architecture of the computer system (including the communications system) is under the leadership of NASA Langley.

1.2 Power by Wire

Power by Wire considers the replacement of the current hydraulics systems of the aircraft with systems which are electrical, resulting in a significant weight reduction in the aircraft. This grant does not encompass the consideration of Power by Wire.

1.3 Evolution of Integration

Although digital systems have been integrated into, or replaced many electronic devices in existence today and have also replaced many mechanical devices, areas

such as avionics have been slower than many technological areas to embrace digital devices due to the strict requirements for flight critical systems. The initial use of digital systems was to function as a backup to mechanical systems or as a primary system with electro-mechanical backup systems available in the event of a system failure.

Fly by wire flight control systems have been utilized to enhance stability and control systems but were not themselves flight critical. Similar claims could be made about systems supporting navigation and communication. In the last decade there has been a significant move towards the integration of digital avionics into flight/life-critical systems of the aircraft. The fact that digital systems typically require less power, less space, and weigh less than a corresponding mechanical or electro-mechanical system having numerous consequences including the following:

- the crowded flight panel could be reengineered to include more manageable digital displays and more efficient use of the cockpit space,
- the total weight of the aircraft could be reduced resulting in both fuel savings and maneuverability or that the payload of the aircraft could increased,
- less stable aircraft requiring greater manueverability could be flown by control mechanisms with better reaction time than is manually possible, and
- more intelligent interpretation of the information could be performed by digital systems.

As mentioned in the Statement of Work in the original proposal, this grant will concentrate in the communications system design. Communications system alternatives will be identified and a suggested list of specific research initiatives are included. NASA Lewis work is not incorporated in this report.

2 Reliability

The Achilles heel of most digital systems has long been the inability to prove correctness of either hardware or software performance. Hardware devices suffer from the unknowns of environmental factors including temperature, humidity, electromagnetic fields, lightning, etc. One can simulate performance of such devices and systems, but it is impossible to predict exactly those influences which might affect its performance given the extremely large numbers of combinations of those effects. Isolation and testing of one factor is difficult, but testing combinational effects can only be performed within limits.

Development of reliable software has long been the target of extensive effort. Perhaps the best description of the obtainable objective for the software industry at large is damage control as opposed to elimination of errors. Nonetheless, a specific

problem domain adhering to specific practices could potentially lower the probability of error significantly.

2.1 Hardware

The problems of verification of hardware are numerous as are the levels at which reliability must be examined. At the *gate/device* level, there are questions concerning the effects of environmental factors mentioned previously as well as the effects of timing constraints. The infinite combinations of these factors and their combinational effects represent an intimidating problem.

At the *instruction* level there is the question of correctness, whether or not the instruction set functions as specified without any unanticipated side effects. Proving correctness of the instruction set is even more difficult than proving correctness of a specific program. In this case it is necessary to show that every combination of instructions which can be written will function properly. On the other hand, hardware systems are typically slowly evolved systems with the same fundamental architecture of previous generation devices.

[3] contains a synopsis of a presentation made by Brock and Hunt on behalf of Computational Logic, Inc. which summarizes a body of work constituting a formal methodology for design of correct circuits. The methodology entails writing and verifying circuit generator programs and then using the generators to produce circuits which can be proven to be correct. This process was used to produce an ALU generator and it was anticipated that it would be used to create a FM8502 microprocessor. Work in areas such as this should provide the basis for creating more verifiable hardware systems.

At the *device/board* level there are questions of interface and interaction. In order to assure the correct interaction of devices, there exists a body of research in the area of communications which could be applied. If devices are specified in terms of state machines, the device interactions could be analyzed using the same type of tools to verify protocol correctness, deadlock avoidance, etc.

In addition, the hardware industry is such a rapidly evolving field that one must strive to develop rigid standards for any new devices so as to assure their safe integration into the existing systems. Extremely rigid testing procedures exist for any flight/life-critical systems on the aircraft, but testing alone can not be expected to achieve the reliability levels expected in these systems. Reliability is such a critical issue with flight systems that the industry appears to be doomed to using technology which is a decade old in order to use devices with proven effectiveness. It is an interesting paradox that the industry is on the cutting edge of many technological fields but appears to be constrained to use it until the technology becomes effectively outdated.

2.2 Software

Similar and perhaps more difficult problems exist in the area of software. Although formal verification techniques for software exists, the application of these techniques to large systems is far from mature. Some might argue that the potential for these verification methods to large systems does not look promising. It is more likely that effective software systems will evolve in the same ways that effective hardware systems do, by starting with smaller working systems and adding to them incrementally, relying on the core elements from earlier stages as building blocks. By relying on these core elements as building blocks, one also gains the advantage of additional testing and insight, and increased reliability in those building blocks.

Most of the computing industry has embraced object-oriented methodologies because of the ability of the methodology to build in such an incremental framework. Similarly, many applications are moving towards the client/server model of design. This has a number of significant advantages.

1. A functional interface can be developed to provide generic services. As more efficient means of implementing the service evolve, the component(server) can be replaced without affecting the rest of the system.

2. Association of logical functions with physical devices can be made transparent through the use of networks. Networks have been in sufficient use for more that a decade with a strong body of research and practice defining the strengths and weaknesses of alternatives to support such activities. The use of object-oriented design will allow for easier integration of new functionality while maintaining prior functionality. It is incumbent on the aircraft industry to begin designing the functional software services it needs and the interface specifications for these systems to allow for competitive development of products within the industry.

3. Many of the services required are also likely the same type of services required in other software environments (databases, synchronization, virtual device management, etc.) carrying with them the vast testing already associated with its use in industry.

Although there have been continued developments in the area of design and development of reliable software systems, it remains difficult to show reliability at a quantifiable level such as those required for life-critical systems (probability of 10^{-9} or less of failure per flight hour). The added complexity of building systems with greater functional expectation brings with it the problem of increased difficulty of building an equally reliable system.

Viewing the state of design of avionics systems from the perspective of a computer scientist, I find most of the approaches to solving the problem to be the results of systems evolved incrementally from prior working engineering solutions as

opposed to those which view the problem from an abstract design perspective employing accepted and proven design methodologies. As avionics embarks on the use of computer-based systems to control more systems within the aircraft, the problems are either now, or will soon grow, beyond the ability of the engineers to control their correctness and reliability without the employment of more formal tools and methods.

[3] contains some revealing comments which imply a severe lack of understanding of the depth of complexity of software development along with an acknowledgment that these problems can only be addressed with the employment of some formal methods. Examples:

- "The FMS of the A320 'was still revealing software bugs until mid-January', according to Gerard Guyot (Airbus test and development director). There was no particular type of bug in any particular function, he says. 'We just had to do a lot of flying in order to check it all out. Then suddenly it was working,' he says with a grin" (Flight International, 27 Feb 1989)".

  The director of testing implies that software bugs were arising for an extended period of time, but 'not in any particular function'. They just disappeared magically after testing a long enough period of time. There are some serious ambiguities here accompanied by a belief that the bugs were eliminated by flying long enough to test all possible things that could go wrong. The uncovering of such problems at actual flight test time indicates to me that there are some significant design oversights, not simply an uncovered bug. With good testing strategies and rigid software interface/performance specifications, the type of bugs uncovered during flight testing should have been problems due to timing interaction and problems from of unanticipated sources. The latter indicates a need for a reevaluation of the design. The former implies a need for a more rigorous investigation of the timing requirements of the system, similar to the problem with the next example.

- Dale Mackall, NASA Engineer if the AFTI F16 Flight test states that nearly all failure indications were not due to actual hardware failures, but to design oversights concerning asynchronous computer operation. This represents an important insight. As hardware becomes more and more reliable, our more significant concerns are not that of component failure rather it is due to inconsistency of interface expectation within system components and timing problems on the system level.

Work in the area of estimating software reliability also has limited applicability. The results of such related research [5] appears to have validity to closed system components, but it is difficult to characterize the entire system as having errors occur with characteristics such as exponential interarrival times. Systems which are less deterministic are more difficult to characterize with the types of confidence

intervals (probability of success rates) required in avionics. For example, [5] uses reliability rates of 90%, far below that of the requirements of this arena.

System specifications and interactions do appear to be in need of the inclusion of some formal methods and likely the development of new research to reach overall system reliability expected is required. The use of some of the formal tools for evaluation of communications protocols could provide a basis for the development of similar tools for specification of device and component interaction. Reliability tools such as described in [4] provide another approach for developing an understanding of this problem. However, the technique as presented in [4] does not have the ability to represent time dependencies, nor does it suggest a mechanism for exchange of pertinent state variables describing the state of all relevant system components.

Adoption of SEI level requirements of its own software development teams and contractors might be a reasonable first step in solving this problem. Lastly, the specification and verification tools should evolve to some common methodology for both hardware and software systems so that the systems can be analyzed as a unit rather than separate components.

3 Data Communications

The communications system for an aircraft must meet the same requirements for reliability as for other flight/life-critical systems, and in addition it would be desirable to maintain interoperability. Contrary to other digital system components the communications systems already have a significant amount of research and practice with design methodologies which support interoperability. The OSI model for design of communication systems or a similar architecture would provide a framework with which to design a system which can evolve incrementally as more reliable hardware evolves or as protocol stacks are designed which are either more efficient or provide support for other forms of traffic.

3.1 Error Detection and Correction

Error detection and correction has always been an important area of digital communication systems. Bit error rates of current fiber optic standards are on the order of 2.5×10^{-10}[8]. For

n, Bits in a frame,

Pf_{bit}, Probability of a bit in error, and

Pf_{frame}, Probability of a frame in error,

$Pf_{frame} = (1 - Pf_{bit})^n$

Probability of an error in a frame is:

Frame Size (Bytes)	Frame Error
10	2 x 10^(-8)
1000	2 x 10^(-6)

One can see that the probability of a frame error is high relative to ultrareliable levels previously stated, but what is important is not whether an error occurs, rather will the error be detected. Error detection mechanisms such as CRC are adept at detecting most errors and burst errors of a sufficiently small nature, however, there do exists certain errors which CRC will not detect. Error correction is handled by protocols which manage retransmissions, buffering etc. The limitation of the effectiveness of current communication strategies is determined by the accuracy of detection of errors by CRC. With sufficient bandwidth, transmission multiple times or on multiple links can produce sufficiently low error rates. Sending on separate links assures independence of probability of error, with the probability of two messages in error being the product of the individual frame probabilities of error. If a frame with transmission error rate of 10^{-5} is transmitted on two identical links, the probability of both frames having an error would be $10^{-5} \times 10^{-5} = 10^{-10}$. By employing redundancy with proper frame size, a communications physical link with unacceptable error detection rates could be made sufficiently reliable, raising the probability of detection.

3.2 Reliability

The kinds of reliability problems one intends to solve here are very different than the typical kinds of reliability issues addressed in most data communications systems. Line breaks and chattering nodes are the most frequent sources of errors and are discussed later.

Aircraft reliability can and should incorporate past history of physical damage to the aircraft of assign probability of damage to a physical area and design the links and processing centers so that the probability of incapacitating the system with an explosion is minimized.

4 Utilization of Higher Bandwidth

Higher bandwidths provide an opportunity to pass more state information and assure that system reliability constraints are met. Increases in processor performance and communication bandwidth provide opportunities to check system variables for consistency/reliability.

Current bandwidth requirements of aircraft systems are on the order of 10 Mbps. MIL-STD-1553 defines the standards for a digital bus on a modern military aircraft and Mark 33, 429 on a civil aircraft. Fiber optic networks operate in the 100Mbps-1Gbps range and are more than sufficient to meet these minimum requirements.

The question here appears to be the degree to which the network and the speed of corresponding memory and processors exceed the minimum requirements. With sufficiently fast devices, a design can be created to allow time for redundancy checks and recalculation, reconfiguration or rollback depending on the situation. The major factors to be determined are what constitutes the calculation requirements of the entire system. One should be optimistic that this will be attainable as processors and communication devices become faster as the basic requirements to fly an aircraft remain stable. Once the basic requirements can be met in a fault tolerant manner, additional bandwidth can be used to incrementally add optional functionality to assist the pilot.

4.1 Flight Control

The potential applications of optical fiber are constantly evolving. Stress and strain gauges functions are being investigated by examining the manner in which the light characteristics change as the fiber is stretched or distorted. By embedding these fibers into the structures of the aircraft, the pilot can be alerted as the aircraft experiences critical stress. As more sophisticated weather systems evolve, the networks can provide ample bandwidth for high-resolution imaging of the systems.

As more and more of the components in the aircraft become digital, networks will comprise an essential function to support integration of these devices. Intelligent computing systems to support the pilot require input from as many of these devices as possible, and place a requirement of high reliability on the network operation. Current transmission rates of fiber optic systems is sufficiently high to meet the basic data rate requirements of existing devices. As mentioned previously, one use of the bandwidth could be to use the incorporate redundant transmission for higher reliability.

4.2 Passenger Utility and Service

One of the most exciting aspects of fiber communication is the opportunity that such a leap in potential bandwidth provides to realize systems to make commercial flight more enjoyable and productive for the passengers. Fundamentally there are two types data supported by a digital network, synchronous and asynchronous. Synchronous traffic would typically include video, voice and real-time devices with time constraints on the interaction. Flat panel displays and densely interchangeable storage devices such as cd-rom combine to provide entertainment and education opportunities so that passengers make more efficient use of the flight time. Each seat could have its own display and choose from numerous entertainment/education options. Information could be available to allow the passenger to view other flight schedules for connections, project arrival times, learn about the geography over which the plane is currently traversing or will cross at some time during the flight, or

even provide travelogues for cities and towns in the vicinity of the flight destination. Advertisement from local merchants could be used to generate income. With the large bandwidths available, device traffic which is not flight-critical could accompany this traffic.

Voice traffic could also be digitized and used to provide communication with other passengers on the plane or conceivably to provide external communication to passengers on the ground or in other aircraft. The proliferation of ISDN phones in the near future will make digital voice at commonplace.

As the United States and the rest of the world becomes more committed to an infrastructure to support a completely connected world, computing services to include standard software packages or perhaps standard network functions such as email, internet, and information services will be in higher demand. Fiber optics are capable of providing sufficient for the anticipated network traffic.

4.3 Alternative Coding Techniques

Assuming the required data rates are in the tens of Mbps range, an interesting alternative use of the bandwidth is increase the network tolerance of error in its optical receivers and transmitters. An example of the use of this bandwidth employing a combination of frequency-division and code-division multiplexing in optical networks is contained in [11].

5 Fault Tolerance

5.1 Requirements

The need for fault tolerance in aircraft, spacecraft and many other areas in which NASA has interests is evident. [9] indicates that the reliability requirements for SIFT and FTMP computers expected to be used in ultrareliable systems limit failure probability not to exceed 10^{-9} for a 10 hour flight. Such a stringent requirement impacts every component of design. One could always slowly integrate more sophisticated systems, gradually increasing the reliability and critical applicability of the systems, but achieving the kinds of reliability cited above can not be achieved by exhaustive testing. More formal evaluation is demanded. This statement is not contradicting a prior statement concerning the need to build reliable systems out of simpler working systems. In this context the concept indicates that as we learn to formally evaluate the simpler systems, better tools and methods should evolve for evaluating more complex systems.

For completeness it necessary to evaluate the reliability of all components of the system:

1. gates,

2. chips,

3. signaling and timing,

4. storage devices,

5. operating system,

6. communications, and

7. applications.

However, the primary purpose of this report focuses on the communications support, so emphasis will be placed only on those aspects which directly impact the communications architecture.

5.2 Byzantine Resilience

There exist a number of potential architectures which provide varying degrees of fault-tolerance depending on system requirements [7]. Previous research into viable fault-tolerant architectures for aircraft and spacecraft indicates that architectures which are "Byzantine resilient" are being given the greatest attention.

The architecture proposed by CDSL [6] is a medium to coarse grain architecture which appears to allow voting on functional results such as either control decisions or coarse grained intermediate steps in the control decision process. Although this may not be the final architecture chosen for the aircraft versus the long space mission application of the referenced literature, it is assumed so for the purposes of this report. Limitations of some alternatives will also be discussed.

The Byzantine resilient architectures require highly interconnected components. Each fault-containment region must be connected on a separate communications path with corresponding fault-containment regions. Distinct fault-containment regions could be organized on the basis of numerous factors to include performance, physical position, and/or function. For example, at a very high level, concentration of the decision making devices or intercommunication paths into one physical area of the aircraft could render the craft inoperable if a structural failure such as a break in the wing, tail section or damage to the control panel occurs. Placement of redundant elements should incorporate both functional and other considerations.

5.3 Fault Tolerance in Communications Systems

Highly interconnected devices have been the subject of considerable research, but the results of these investigations have limited applicability to this problem area. Such networks are not utilized to a significant degree in practice and consequently do not

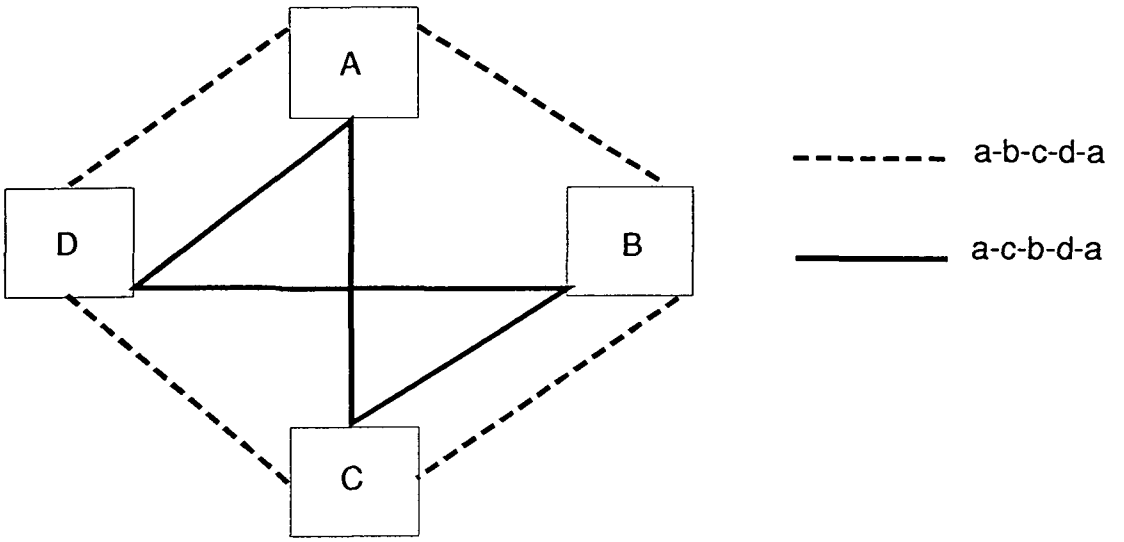

Figure 1: A 4-node Byzantine Resilient Structure

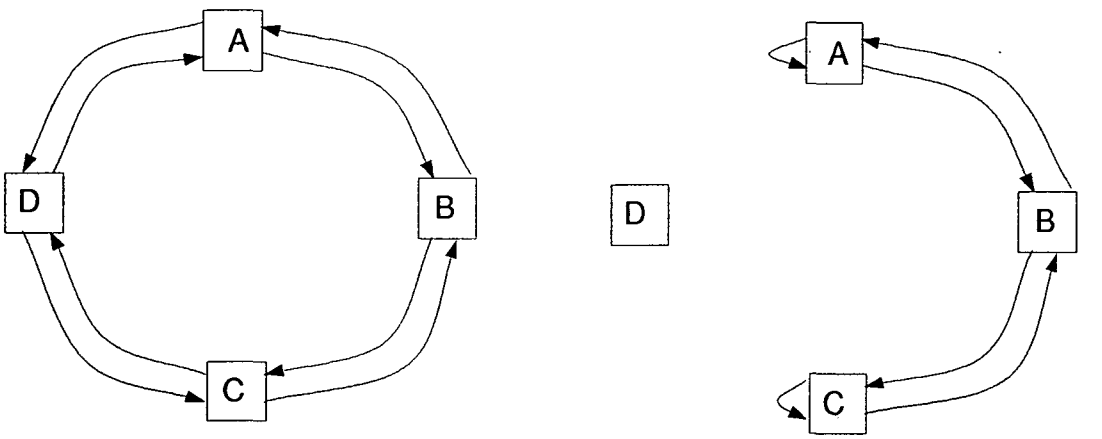

Figure 2: FDDI Healing

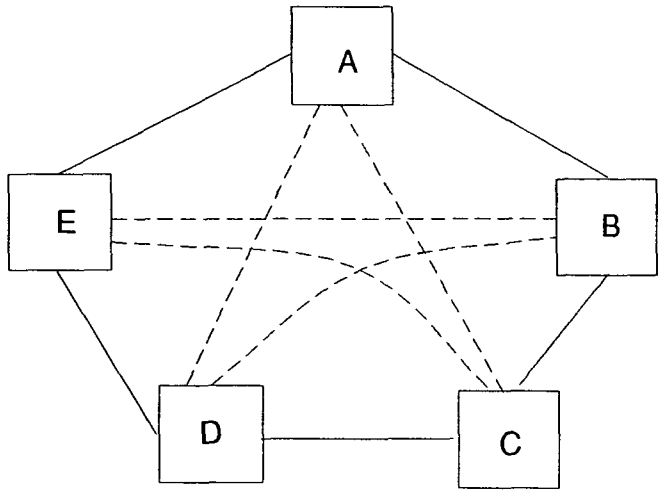

Figure 3: Braided Ring

have an associated set of experiences and the insights which this experience creates. Fault-tolerance in communications networks of today focus more on issues such as how to deal with cable breaks and chattering nodes, and down bridges/routers rather than how to support complex detection and reconfiguration algorithms as is used in a Byzantine resilient design. For example, consider how a dual counter-rotating token ring (FDDI) handles healing in Figure 2. In the first case observe a fully functional ring. If a node were to begin chattering or a break in the cable occurred, the network could be reconfigured into a single ring as indicated. This approach would work fine for a single failure, but a second failure would not allow for exclusion of the individual node, but would segment the network, disabling communication between vital elements.

The interconnectivity is usually accomplished by employing existing architectures (token rings, CSMA/CD and token bus)in conjunction with bridges, routers and gateways and higher level protocol stacks to manage routing functions. Figure 3 illustrates a highly interconnected design. The network is created by braiding multiple rings.

This approach is not applicable to a typical FTPP cluster. Figure 1 shows one implemented with ring architectures. Two rings a-b-c-d-a and a-c-b-d-a could cover the internode connections, but would violate fundamental Byzantine rules for the same reason. The application to specific problem areas such as this are in need of investigation, but the fundamental Byzantine strategy should be upheld. The

expense for implementation will be high, but the approach has a proven reliability.

Designs such as this fail to satisfy the Byzantine requirements and could negatively compromise the reliability of the design. Each connection must function independently in the expectation that a failed connection would not affect any other connection. A token ring is typically implemented as a series of point to point transmissions with a one or more bit delays at each node. Despite this characteristic a failed link can impair the transmission between any pair of nodes of the ring containing the failed link. If these types of designs are being considered, it is important that this problem be investigated further.

6 Protocols

It is not clear that the advantages of inefficient protocol stacks such as the ISO model advocates can be afforded. The specific nature and the load requirements associated with the design will influence the overhead affordable. Nonetheless, it appears that one of the significant contributions which has yet to be made in support of this are is the design of protocol alternatives which support concepts such as

1. replication and voting,
2. recalculation,
3. confirmation, and
4. advancement to next calculation.

Current protocol error detection entails using CRC-based error detection and request for retransmission. Support for fault tolerance requires detection on the basis of comparison of results from duplicated processes. In addition there are naming problems associated with the proper identification of the task being executed and which redundant process is sending the result. Proposals for these protocols and the associated overhead and verification will be an important component of the design process.

7 Survey of Architectures

The study of communications topologies with respect to fault tolerance has received a great deal of recent attention [1,2,10]. [1] contains a summary of the fault tolerant characteristics of a series of topologies. [7] makes some compelling observations concerning the applicability of architectures such as hyper-cubes to Byzantine resilient design and concludes that they can not be rapidly reconfigured. In order

to be consistent with the ultra-reliable levels of fault tolerance, point to point interconnectivity appears to be a must. Using a regular design such as FTPP and electro-optical interfaces should lead to a design which is modular, amenable to VLSI design, and easily reconfigured as faults occur.

If the architecture supports sufficient reconfiguration capabilities so as to meet the reliability requirements, it should be possible to consider the network as an abstract entity and design a protocol to support a transparent healing when necessary. A device would simply transmit a solution and receive a response that its ok to proceed or recalculate, assuming the system is still functional. Last, it is crucial that consideration is only given to networks which are deterministic in nature to the degree that a minimum performance can be guaranteed. Token rings can be guaranteed a minimal access whereas CSMA/CD buses can not.

8 Conclusions

The network architecture which will evolve will probably be a series of networks, not necessarily, but likely, interconnected. The most serious mistake which can be made is to try to accomodate current networks such as token rings into a fault tolerant architecture. A fault tolerant design should have a network architecture which is consistent with the philosophy of the overall design. An architecture of highly interconnected, point-to-point links appears to be the logical direction for the degree of fault tolerance expected in this area.

My suggestions for related research are as follows.

1. If Byzantine resiliency is determined to be a necessary condition for the Fly by Light / Power by Wire generation of aircraft, then significant consideration need to be given to a physical layer communications interface which will support point to point connections. Point to point connection is a minimal requirement or the requirement in order to provide independent paths. Other alternatives exist but they violate the design of the rest of a Byzantine system such as FTPP.

2. In addition a protocol should be developed to provide support for

   (a) replication and voting,

   (b) recalculation,

   (c) confirmation, and

   (d) advancement to next calculation.

3. A study of software interface specifications and the potential adaptation of protocol verification to this problem could bring some formal tools to the process.

4. If the communications infrastructure for a Byzantine resilient architecture is to be build from rings, the effect of a failed link should be investigated.

5. A study should be made to analyze the probability of physical damage and otherwise to certain areas of the aircraft and to locate redundant computing centers and network links throughout the aircraft in order to minimize the likelihood of total system failures. This study should incorporate to whatever degree possible the history available of past incidents and any predictions for anticipated problem areas.

References

[1] G.B. Adams, D.P. Agrawal, and H.J. Siegel. Fault-tolerant multistage interconnection networks. *Computer*, 14–27, June 1987.

[2] L. Bhuyan. Interconnection networks for parallel and distributed processing. *Computer*, 9–12, June 1987.

[3] R.W. Butler. Nasa formal methods workshop 1990 - august 20-30, 1990. *NASA Conference Publication 10052*, November 1990.

[4] G.C. Cohen and C.M. McCann. Reliability model generator specifications. *NASA Contractor Report 182005*, March 1990.

[5] K.J. Dotson. Development of confidence limits by pivotal functions for estimating software reliability. *NASA Technical Paper 2709*, June 1987.

[6] R.E. Harper. Critical issues in ultra-reliable parallel processing. *CSDL-T-944*, June 1987.

[7] S.C. Johnson. Evaluation of fault-tolerant parallel-processor architectures over long space missions. *NASA Technical Memorandom 4123*, August 1989.

[8] G. Kajos. Link error monitoring action item. *SMT Working Group*, December 1988.

[9] D.K. Pradhan. *Fault-Tolerant Computing, Theory and Techniques, Vol II.* Prentice Hall, 1986.

[10] T.G. Robertazzi. Toroidal networks. *IEEE Communications*, 45–50, June 1988.

[11] G. Vannucci. Combining fdm and cdm in a high-capacity optical network. *IEEE Network*, 21–30, March 1989.