Networking for Computer-Integrated Manufacturing*

Asok Ray

A major challenge in Computer-Integrated Manufacturing (CIM) is the integration of computer-controlled complexes of robots, automated machine tools, material handling devices and guided vehicles on the shop floor with engineering design and plant management databases and inter-person information processing (for example, electronic mail) across local and remote stations [1-4]. The architecture and performance of the network that interconnects these diverse functions are of paramount importance to the efficient and reliable operation of integrated design and manufacturing processes. Lack of interoperability, reliability and flexibility in the network design can cause chain reactions of delay and congestion resulting in loss of control of safety and productivity.

Although a large body of analytical research in the general area of network control and communications protocol development has focused on modelling and performance analysis as well as on designing flexible systems for accommodating future growth [5-11], the specific challenges presented in CIM have not yet been fully addressed. The future research in this area needs to utilize the existing knowledge in computer networking by identifying, modifying, and validating the theories, that are relevant to CIM.

The performance (for example, delay and throughput) of a multiplexed data communication system could be significantly influenced by the intensity and distribution of traffic in the network. The data transmission between a pair of interacting machines in a factory environment is subject to time-varying delays due to the latency of messages in the network as well as to mis-synchronization between the system components (for example, sensor and controller computers within a feedback control loop) [12,13]. This delay occurs in addition to the sampling time delay that is inherent in digital control systems. The potential instability resulting from the network-induced delays is evident in very fast processes like flight control systems in tactical aircraft [14]. But their effects on the dynamic performance of feedback control systems are often ignored in relatively slow processes which are prevalent in manufacturing environment. However, as the number of users in the network increases, the augmented traffic causes a larger data latency to a point when its impact on the performance of some of the control loops (sharing the network) can no longer be ignored.

The detrimental effects of network-induced delays on the dynamic performance of real-time distributed control systems are further aggravated by loss of messages resulting from saturation of buffers at the terminals and data corruption by noise in the network medium. For a network which is shared by processes with different response times, an appropriate traffic load distribution is critical for stability of the control systems. A few examples of integrated manufacturing processes where network-induced delays could be detrimental are presented below.

Within an integrated manufacturing environment, networking provides flexibility for coordinated control of inter-cell equipment. For example, in an intelligent welding system [15,16] where a positioning table and a robot may not be hardwired to the same computer, the table position coordinates could be relayed to the robot controller via the network. The robot controller, in turn, may transmit back signals for a more convenient table position. This requires timely arrivals of the data at both machines. Another example is the coordinated control of two robots in a master-slave configuration while handling a bulky workpiece together. If these two robots do not belong to the same supervisory computer, they will communicate via the network so that the slave robot follows a prescribed trajectory. The timeliness of the transmitted data is essential because a delay could damage the workpiece or the robots' wrists and arms. Another example is a machine tool transfer line where several machines are assisted by robots for loading, unloading and handling of materials and parts. The timeliness of interrupt signals arriving at individual machines is critical for successful operations.

Although ample research papers in modelling and simulation of communication protocols have been published [3,10,11], significance of network-induced delays relative to stability of feedback control systems has apparently been addressed in only a few cases [12,13,14]. Since network-induced delays are time-varying and possibly stochastic, the delayed control systems have so far been evaluated by combined discrete-event and continuous-time simulation. (Further research is needed to develop an anlytical methodology for designing control systems that are subject to network-induced delays.)

Since standard factory and office communication architectures like MAP, MAP/EPA [17], and TOP [18] do not

This work was supported in part by National Science Foundation under Research Grant No. DMC-8707648 and Research Equipment Grant No. DMC-8604412.

adequately provide for real-time operations of CIM processes, it is timely to develop a network design concept to deal with this critical problem. In this perspective the major objectives of the article are:

- to introduce the concept of a CIM networking architecture for integration of factory and office communications;
- to elucidate some of the potential problems in CIM networking and possible approaches for their solution; and
- to present a simulation methodology for evaluating the performance of integrated networks with different protocols and architectures and under diverse operating conditions.

CIM Networking Architecture

Integration of various design and production control strategies requires a number of computing machines which may range from main frame and large minicomputers in the office environment to microcomputer workstations at the cell control level and microprocessors and programmable logic controllers at the factory floor. The mainframe computers at the design office are usually responsible for executing the Manufacturing Requirements Planning (MRP), Tool Management (TM) and CAD/CAM functions, and interface with the cell control workstations.

A common practice in computer-integrated manufacturing (CIM) is to interconnect individual cell computers to one or more mainframe computers in a point-to-point cluster as shown in Fig. 1. In this configuration, any information exchange between the workstations at the cell control level has to be routed through a mainframe computer, and each cell control workstation is, in turn, connected to its own Data Control Point (DCP) serving a number of robots and machine tools. An alternative CIM configuration is to serve individual (rigid) work cells via a hierarchically structured network as shown in Fig. 2. The carrier band networks in Figure 2 directly interconnect intra-cell devices but inter-cell components must communicate via the broadband backbone network. Although this approach does not need the services of the main frame computers as switching nodes, the bridges in the inter-cell communication path are a source of additional delay and potential congestion.

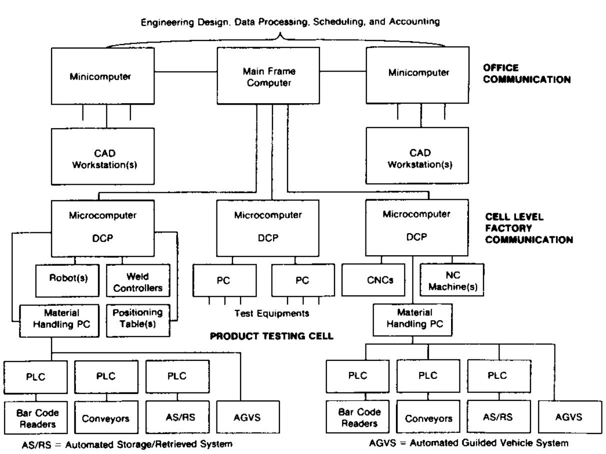

Fig. 1. Conventional CIM Configuration.

A high degree of flexibility and modularity in manufacturing automation can be achieved by partitioning the shop level facilities into several virtual cells in which dynamic production control structures permit time sharing of workstation level processing systems [4]. In the dynamic environment of flexible manufacturing systems (FMS), the same robot(s) or machine tool(s) may be assigned to different virtual cells from time to time. The concept of virtual cells can be realized by interconnecting the common equipment by a reliable high-speed network which facilitates dynamic reallocation of resources, i.e., machine tools, robots, and intelligent terminals in a flexible manufacturing environment. A network architecture for integration of design and manufacturing systems via a single high-speed medium is illustrated in Fig. 3. This architecture differs from those in Figs. 1 and 2 in the sense that devices in different cells are allowed to directly communicate with each other via a common network medium (without any bridges or routers) to share the common resources.

Given that the network is highly reliable and designed to have fault-accommodating properties (for example, redundant network media and head-end remodulators with automatic switch-over and cable plant status monitoring capabilities), the advantages of the networking concept in Fig. 3 relative to the conventional CIM approaches in Figs. 1 and 2 are as follows.

- Traffic due to information exchange between cell control workstations will not influence the performance of the mainframe computers.
- Since a workstation is allowed to serve more than one virtual cell, the efficiency of manufacturing operations could be enhanced in the following sense.
  - improved flexibility of operations in which a common DCP can be used for different control areas,
  - greater utilization of individual workstations and shop floor equipment,
  - reduced downtime due to the failure and scheduled maintenance of computing and production equipment, and
  - reduced network-induced delays and increased data throughput.

The above concept is similar to what is encountered in integrated control systems of advanced aircraft [14] where a single reliable high-speed network allows communications between diverse but inter-related functions ranging from fly-by-wire (or fly-by-light) active control to management support and information display. The flight and propulsion control systems in advanced aircraft could be considered to be analogous to real-time multi-robot or multi-machine control processes in a factory environment, and the flight, mission and weapons management to engineering design and manufacturing plant management.

Using the network architecture illustrated in Fig. 3, the CIM networking problems can be approached as follows:

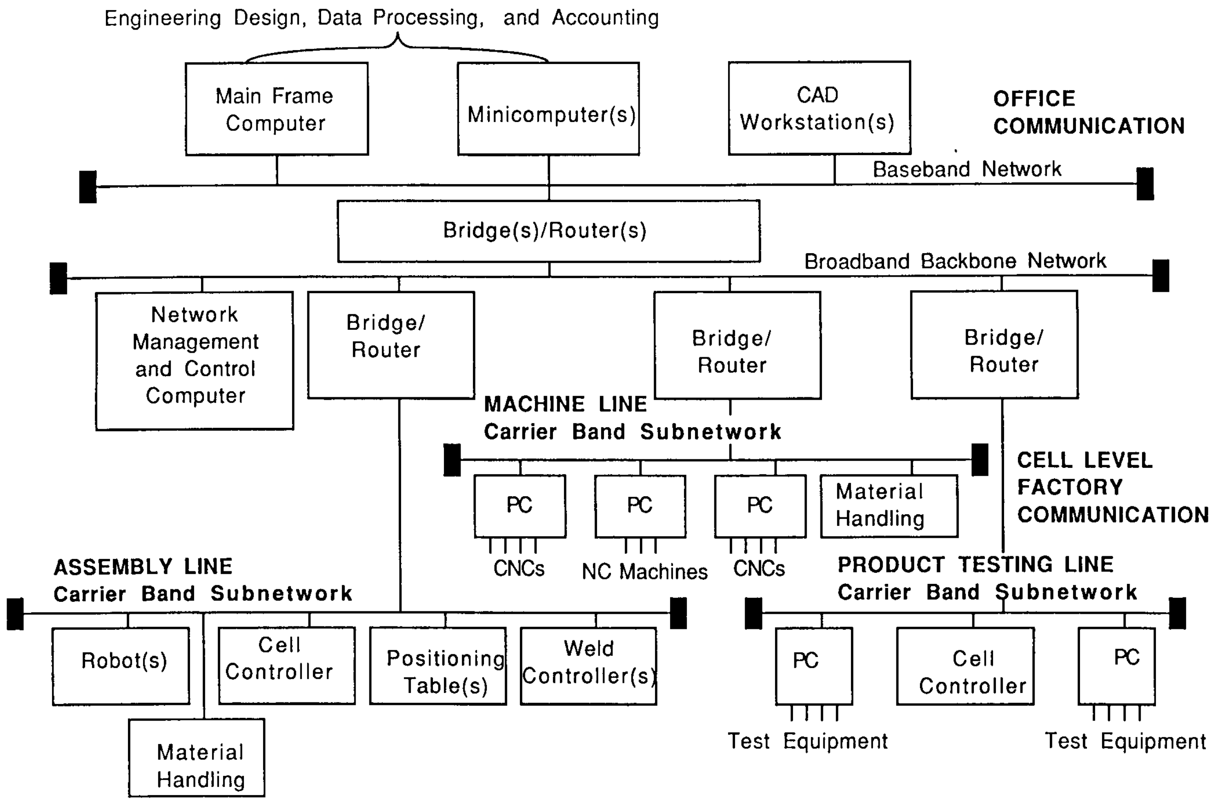

Fig. 2. Conventional Networking for CIM.

Interfacing of Heterogeneous Protocols

The computerized equipment for autonomous manufacturing processes, engineering design and office management may have their own individual control languages, data structures, and operating systems, which are not likely to be mutually compatible. The problems of incompatibility can be handled by adopting a layered network architecture whereby individual computers can communicate to peer level counterparts in their own languages. Examples are MAP [17] and TOP [18] which are under consideration as standards for factory and office communication networks, respectively.

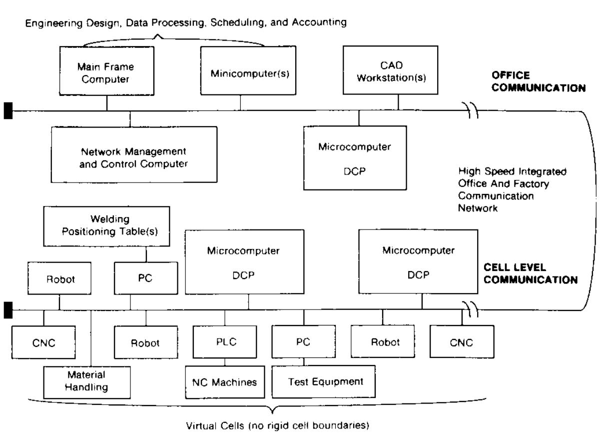

Fig. 3. Integrated Networking Concept for CIM.

There are, however, potential problems in interconnecting TOP (in office environment) with MAP (in factory environment). The Medium Access control (MAC) protocol in TOP is IEEE 802.3 CSMA/CD which is appropriate for office communications with bursty traffic. The corresponding protocol in MAP is IEEE 802.4 linear token passing which is essentially a controlled access protocol and assures an upper bound for data latency as it is required for real-time control of robots and automated machine tools in factory communications. Interfacing of MAP and TOP can be accomplished by one or more bridges (or routers) which have IEEE 802.3 on one side and IEEE 802.4 on the other side interconnected by a common IEEE 802.2 Logical Link Control (LLC) protocol. Although this arrangement is conceptually simple, the potential difficulty lies in designing the two sides of the bridge without complete knowledge of the exact implementation of the two MAC protocols.

One reasonable approach to solve the problems of incompatibility between factory and office communication protocols (viz., MAP and TOP) is to seek an alternative Medium Access Control (MAC) protocol which would satisfy the requirement of the diverse traffic. Hybrid protocols like CSMA/CD-DP (Kiesel and Kuehn in [9]), CSMA/CD-DCR (Takagi et al. in [9]) and Buzz-Net (Gerla et al. in [11]) which combine many features and advantages of CSMA/CD and token passing are reaching technological maturity. Although certain features of these protocols (for example, Unilink protocol of Applitek Inc.) have been commercially available, they may not qualify as standards because of their proprietory architecture and lack of proven reliability. There is a need for a hybrid fault-accommodating link-layer protocol which does not require any bridges for interconnecting factory and office communication networks and would serve as a standard. The research in this category will enhance integration of MAP and TOP with other standard (such as SNA and DECNET) and non-standard architectures.

Mixing of Real-Time and Non-Real-Time Data Traffic

Within a manufacturing cell environment, the majority of the traffic accounts for communications between control devices such as programmable controllers, NC machines, and guided vehicles [4]. The control devices usually generate short messages and communicate on a periodic basis as the sampling interval T of a control system is predetermined according to the dynamic characteristics of the controlled process. For example, T could be very small, viz., on the order of 20 ms for communications between two robots working on the same transfer station, and significantly larger for other processes (on the same network) like updating of table positions. The situation is similar to that in avionic networks where the flight control system could share the same network medium with flight management and mission systems which have slower dynamic responses [14].

Occasional losses of sensor and control signals in the feedback loop of a manufacturing process control system can be tolerated whereas the additional delay due to retransmission of erroneous signals may degrade its dynamic performance. Therefore the real-time data may not use the acknowledgement option; packets with detected errors would be simply discarded. On the other hand the design and administrative information such as production orders, part drawings, and status reports is transmitted nonperiodically and much less frequently. These messages are usually long, and do not need to be processed within the constraints of a real-time environment. Therefore the acknowlegement option should be adopted for such non-real-time data. Whereas data latency is critical for real-time signals, non-real-time messages need the assurance of accurate delivery via the shared communication medium. Thus the real-time signals, under certain circumstances, can be allowed to interrupt the non-real-time data packets. This interruption may delay the delivery of the non-real-time data packets but will not compromise their data integrity because of the built-in error recovery or packet retransmission capabilities of the network protocol. The concept of mixing of real-time unacknowledged data and delayable acknowledged data is somewhat similar to that of packetized voice and data communication [19] if the high speed real-time data packets are replaced by voice packets.

The existing factory communication protocols such as MAP and MAP/EPA [3,17] apparently do not address the above problem of integration of real-time and non-real-time data. The logical link control (LLC) sublayers in MAP/EPA and MAP use the IEEE 802.2 Type 3 (acknowledged connectionless) and Type 1 (unacknowledged connectionless) options of IEEE 802.2, respectively. Since the MAP/EPA normally uses 5 Mb/s carrier-band transmission media as opposed to 10 Mb/s broadband in the backbone MAP, they cannot directly communicate with each other over the same transmission medium—a bridge or a router is required for their interconnection resulting in additional data latency and potential congestion.

Timely delivery of real-time data can be assured by a Logical Link Control (LLC) sublayer protocol with the following features:

- Acknowledgment and waiver of acknowledgement options.
- Right of privileged terminals to transmit a message by interrupting other terminals.
- Selective admission of non-privileged terminals into the logical ring.

This LLC protocol should operate in conjunction with the hybrid MAC sublayer protocol, discussed earlier, and the resulting link layer protocol suite might as well be an augmentation of the IEEE 802 family.

Analysis of Distributed Data Communication Networks

The key parameters for performance evaluation of communication network protocols in a real-time distributed control environment are 1) data latency (the time delay encountered in accessing the network medium as well as for serial transmission of message bits) and data throughput, 2) data frame error rates (detected and undetected), and 3) network reliability and system availability.

Reliability and availability are largely hardware-dependent and are usually analyzed during the hardware design phase when detailed specifications become available. Data frame error rate depends on the raw bit error rate in the medium and the error detection algorithm provided in the protocol (for example, the cyclic redundancy check). Raw bit error rates in (coaxial cable) network media in a typical industrial environment range from 10^{-12} to 10^{-8} [20]; corresponding values for optical fiber media are significantly lower. Although message retransmissions or rejections (as a result of detected frame errors) apparently have no significant bearing on data latency and throughput under these circumstances, it is the undetected frame errors that degrade control system reliability. Therefore, the issue of undetected frame error rate should be dealt along with reliability.

Data latency and throughput are dependent on traffic pattern which, in turn, is governed by message arrival rate, time sequence of message arrival and transmission on the network medium, message length, buffer size at individual terminal's queues, and number of terminals in the network. On this basis, the network performance should be evaluated by both simulation and analytical techniques [12,13,14].

The analytical modelling techniques [21-24] such as those based on queueing theory, routing algorithms, and graph theory could provide compact, closed form solutions if the governing equations for the modelled system are mathematically tractable. Simplifying assumptions, such as Poisson arrival of network traffic and existence of infinitely many subscribers, that are usually applied to arrive at analytical models [24] may introduce unacceptable inaccuracies in the performance evaluation of protocols and integrated network functions. In these cases the simulation approach should be adopted for evaluating the network performance under various operational scenarios. Examples are evaluation of network control and management strategies and protocols under initialization and error recovery, and investigation of potential failure modes and operational malfunctions which can degrade the network system performance. These simulation programs are expected to incorporate analytical models of some of the network functions and statistical computations as subprogram modules. The simulation approach allows the stochastic elements of the network as well as the real-time discrete and continuous processes to be modelled with fewer simplifying assumptions. Simulation also plays an important role in the verification of analytical models and vice versa.

Protocols belonging to the class of distributed controlled access (viz., token passing bus in factory communication), random access (viz., CMSA/CD in office communication) as well as hybrid protocols are suitable for CIM network simulation. The fundamental theorems for network performance analysis are given in earlier publications [12-14] and are not repeated here. However, pertinent definitions of network parameters are given in Appendix A.

CIM Network Simulation

One of the major objectives of the CIM network system simulation is the comparative evaluation of different architectures using diverse traffic scenarios and real-time models of the manufacturing process control systems. This is achieved by decomposing the network system into two subsystems as described below.

The network system model consists of two subsystem models: 1) Discrete-event model of the network, and 2) Continuous-time model of process dynamics and discrete-time model of the process controller. The program structure is modular, that is, any protocol model (for example, linear token bus or token ring or random access) can be inserted in the simulation program while operating on the same control system model and vice versa. Discrete-event models are suitable for representing the time-ordered sequences of operations that are encountered in network protocols. On the other hand continuous-time models are essential for solving the initial-value problems associated with the plant and controller dynamics. The coupling of two types of simulation permits the evaluation of the network as a dynamical element of the process control system and thus allows for monitoring the effects of network-induced delays. A schematic diagram for the combined discrete-event and continuous-time model structure of the integrated network system is shown in Fig. 4.

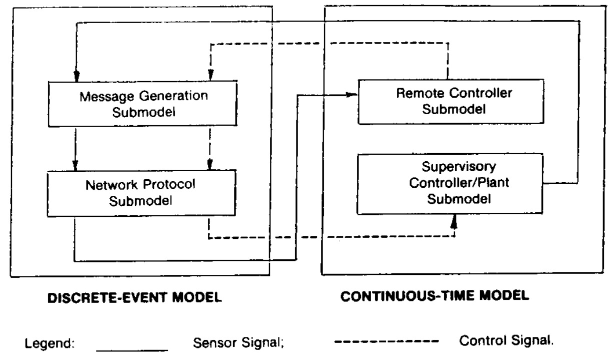

Fig. 4. Simulation Program Structure for the DDCCS Network.

The network subsystem model consists of two independent but interacting submodels: 1) Message generation submodel; and 2) Protocol submodel.

The message generation submodel has an identical structure for all types of protocols and is driven by an external pool of messages that arrive at the network system either periodically or at random intervals of time (Poisson arrival for example). Similarly, the message lengths can be either constants or randomly distributed (exponential for example). When a new message arrives at the system from the external message generator, the message attributes are defined to establish the message identity in the following ways:

- Time of arrival—this is the instant at which the arrival of a message at the transmitter queue is recorded.
- The message information length (overhead not included).
- The source terminal, that is, the terminal from which the message is generated.
- The destination terminal—this could be any terminal on the network other than the source terminal.
- The message priority if applicable.

The protocol submodel essentially represents the algorithm of the network access protocol under consideration. Different submodels, each of which identically enters the simulation program as a subroutine, should be developed for a number of protocols. Although internal algorithms of the individual protocol submodels are different, their interactions with the message generation submodel and the plant and controller model are identical. For example, the attributes of the generated messages are captured by the protocol submodel which, in turn, regulates the time delays for exchange of message signals between the plant controller terminals.

The plant and controller models are formulated using the standard continuous-time and discrete-time state-variable approaches, respectively. The ordinary differential equations describing the plant model can be solved by integration routines provided in the standard IMSL package. The solution of the difference equation describing the digital controller is rather straight-forward. The interactions between the plant and controller models

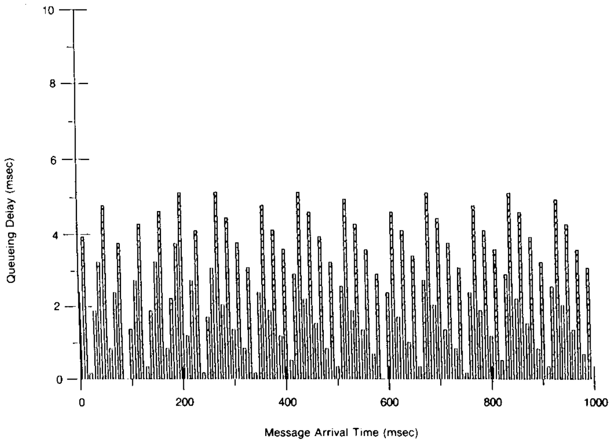

Fig. 5. Steady State Profile of Queueing Delay for SAE Linear Token Bus Protocol.

involve exchange of sensor and control signals which undergo time-varying delays introduced by scheduled events in the network model.

Standard network performance parameters such as the expected value, standard deviation and confidence intervals of delays and throughput do not provide sufficient information for analyzing the delayed control system dynamics. A time history of network-induced delays is needed to accurately analyze the dynamic performance. As an example the steady-state profile of the queueing delay at a terminal is shown in Fig. 5 for a linear token passing bus protocol under periodic traffic with constant message length and offered traffic of 0.5. (See Definition 5 in Appendix A.) The abscissa indicates the time instants when the message arrives at the transmitter queue and the ordinate indicates the time varying queueing delay for these messages. For example, the message that arrives at the instant 200 ms in Fig. 5 waits in the queue for 5.25 ms. On the basis of Remarks 1, 5, 7 and 8 in Appendix A, it follows that the control systems are always time-varying even under steady-state and deterministic traffic conditions.

Further results on simulation and experimentation for manufacturing system networks are given in previous publications [12,13] to demonstrate how the network-induced delays could degrade the dynamic performance of real-time manufacturing processes.

Simulation Model Coding and Implementation

Key considerations in the choice of language for the integrated network system simulation are: 1) Combined discrete-event and continuous-time simulation capability; 2) Programming flexibility and software portability; 3) Verification and debugging capability; 4) Built-in statistical testing capability; and 5) Automatic ordering of scheduled events. A comparison of different simulation languages which offer these key features for coding and implementation of network models is given in [25]. A specific simulation language is described in [26].

Summary and Conclusions

While networking significantly enhances the flexibility and modularity of manufacturing operations and management, it introduces several new problems which include interfacing of communication protocols, and mixing of real-time and non-real-time data traffic leading to additional delays. The concept of a CIM network architecture has been introduced to address these problems. Furthermore, since existing factory and office communication architectures like the MAP, MAP/EPA, and TOP do not adequately provide for real-time operations of integrated manufacturing processes, the CIM network should be designed to accommodate the following features.

- Reliable and efficient interfacing of office and factory communication services for computer-integrated design and manufacturing.
- Sharing of a common network by subscribers with heterogeneous traffic and quality of service.
- Efficient utilization of the network channel capacity for a wide range of offered traffic.
- Enhanced interoperability of software and hardware between different manufacturers' equipment.

Analysis and design of an integrated network for real-time distributed manufacturing processes require interactions between the disciplines of communication systems and control systems engineering. To this effect the general structure of a combined continuous-time and discrete-event simulation methodology has been presented. Since the network-induced delays are time-varying and possibly stochastic, conventional frequency-domain techniques may not be applicable and therefore advanced time-domain techniques [27-30] need to be developed for control systems analysis and design.

Acknowledgment

The author acknowledges the contributions of Messrs. S. Lee, S. H. Hong, A. Ayyagari and L. W. Liou, and Dr. Y. Halevi in this work.

References

[1] N. G. Odrey, "Integrating the automated factory," *Data Processing Management,* Auerbach Publishers Inc., 1985.

[2] B. A. Loyer, "LANs: the coming revolution in factory control," *Machine Design,* pp. 127-130, Nov. 22, 1984.

[3] *Workshop on Factory Communications,* R. Rosenthal, ed., National Bureau of Standards Special Publication No. NBSIR 87-3516, March 1987.

[4] A. T. Jones and C. R. McLean, "A proposed hierarchical control model for automated manufacturing systems," *Journal of Manufacturing Systems,* vol. 5, no. 1, pp. 15-25, Jan. 1986.

[5] A. Segal, "Distributed network protocols," *IEEE Trans. on Information Theory,* vol. IT-29, pp. 23-55, Jan. 1983.

[6] W. Bux, "Local-area subnetworks: a performance comparison," *IEEE Trans. on Communications,* vol. COM-24, no. 10, pp. 1465-1473, Oct. 1981.

[7] F. Vakil, M. T. Hsiao and A. A. Lazar, "Flow control in integrated local area networks," *Global Telecommunications*

Conference, San Diego, CA, pp. 11.5.1-11.5.5, Nov.-Dec. 1983.
[8] *Workshop on Analytic and Simulation Modeling of IEEE 802.4 Token Bus Local Area Networks*, R. Rosenthal, ed., National Bureau of Standards Special Publication No. 500-127, June 1985.
[9] *IEEE Journal on Selected Areas in Communications, Special Issue on Local Area Networks*, ed. K. Kuemmerle, B. W. Stuck and F. A. Tobagi, vol. SAC-1, no. 5, Nov. 1983.
[10] *IEEE Journal on Selected Areas in Communications, Special Issue on Network Performance Evaluation*, ed. A. Leon-Gracia, vol. SAC-4, no. 5, Sept. 1986.
[11] *IEEE Journal on Selected Areas in Communications, Special Issue on Multiple-Access Networks*, ed. V.O.K. Li, vol. SAC-4, no. 5, July 1987.
[12] A. Ray, "Distributed data communication networks for real-time process control," *Chemical Engineering Communications*, Vol. 65, March 1988, pp. 139-154.
[13] A. Ray and A. Ayyagari, "Networking for real-time control of integrated manufacturing processes," *IEEE International Conference on Robotics and Automation*, Philadelphia, PA, April 1988.
[14] A. Ray, "Performance evaluation of medium access control protocols for distributed digital avionics," *Journal of Dynamic Systems. Meas. and Control. Trans. ASME*, vol. 109, pp. 370-377, Dec. 1987.
[15] D. R. Thompson, A. Ray and S. Kumara, "A hierarchically structured knowledge-based system for welding automation and control," *Journal of Engineering for Industry, Trans. ASME*, Feb. 1988, pp. 71-76.
[16] N. Nayak, A. Ray and A. N. Vavreck, "Adaptive real-time intelligent seam tracking system," *Journal of Manufacturing Systems*, vol. 6, no. 3, pp. 241-245, Sept. 1987.
[17] Manufacturing Automation Protocol Reference Specification 2.1A and 2.2, *Society of Manufacturing Engineers*, Dearborn, MI, Aug. 1986.
[18] Technical and Office Protocols Specification Version 1.0, *Boeing Company*, Seattle, WA, Nov. 1985.
[19] W. L. Hoberecht, "A Layered Network Protocol for Packet Voice and Data Integration," *IEEE Journal on Selected Areas in Communications*, vol. SAC-1, no. 6, pp. 1006-1013, Dec. 1983.
[20] W. Stallings, *Data and Computer Communications*, Macmillan Publishing Company, New York, 1985.
[21] C. H. Sauer and E. A. MacNair, *Simulation of Computer Communication Systems*, Prentice-Hall, Englewood Cliffs, 1983.
[22] L. Kleinrock, *Queueing Systems, Volume II: Computer Applications*, John Wiley, New York, 1976.
[23] D. Bertsekas and R. Gallagher, *Data Networks*, Prentice-Hall, Englewood Cliffs, NJ.
[24] J. L. Hammond and P. J. P. O'Reilly, *Performance Analysis of Local Computer Networks*, Addison Wesley, Reading, MA, 1986.
[25] J. Banks and J. S. Carson, "Process interaction simulation languages," *SIMULATION*, pp. 225-235, May 1985.
[26] C. D. Pegden, *Introduction to SIMAN*, Systems Modeling Corporation, State College, PA, 1985.
[27] K. Hirai and Y. Satoh, "Stability of a system with variable time delay," *IEEE Trans. on Automatic Control*, vol. AC-25, no. 3, pp. 552-554, June 1980.
[28] M. Ikeda and T. Ashida, "Stabilization of linear system with time-varying delay," *IEEE Trans. on Automatic Control*, vol. AC-24, no. 2, pp. 369-370, April 1979.
[29] A. P. Belle Isle, "Stability of systems with nonlinear feedback through randomly time-varying delays," *IEEE Trans. on Automatic Control*, pp. 67-75, Feb. 1975.
[30] A. P. Belle Isle and F. Kozin, "On the almost-sure sample stability of systems with randomly time-varying delays," *Automatica*, pp. 755-763, Nov. 1972.

Appendix A—Network Performance Parameters

Pertinent concepts and definitions of network performance parameters for a controlled access protocol are introduced below.

Definition 1. For a message, the word count Wc is defined as

$$Wc(L) = \begin{cases} Int(L/Wd) \text{ if } Rem(L,Wd) = 0 \\ Int(L/Wd) + 1 \text{ if } Rem(L,Wd) > 0, \end{cases}$$

where L = length (in bits) of the data part of a message excluding the bits due to formatting and overhead,
Wd = number of data bits per word,
Int(•) indicates the integer part of •, and
Rem(a,b) is equal to a − [Int(a/b)]b.

Definition 2. Frame length (bits) of a message is defined as

$$= Wf\ Wc(L) + \Omega$$

where Wf = Length (bits) of a formatted word, and
Ω = Overhead (bits) associated with a message.

Definition 3. Queueing delay δq of a successfully transmitted message is the difference between the instant of arrival of the message at the transmitter queue of the source terminal and the instant of transmission of its first bit on the medium.

Definition 4. Data latency δ of a successfully transmitted message is defined as the difference between the instant of arrival of the message at the transmitter queue of the source terminal and the instant of reception of its last bit at the destination terminal.

Remark 1. Queueing delay and data latency are approximately related as

$$\delta = \delta q + \ell / R$$

where R = data rate in bits/unit time.

Defintion 5. Offered traffic G is defined as

$$G = \sum_{i=1}^{N} (E[Li])/(R\,E[T])$$

where N = number of active terminals in the network,
R = Medium bandwidth (bits/sec),
T = Message interarrival time at a terminal,
E[•] denotes the expected value of • with the assumption that the generation and transmission of messages are ergodic processes, i.e., the time average is equal to the ensemble average, and the subscript i corresponds to the terminal i.

Definition 6. Cycle time T is defined as

$$\tau = \sum_{i=1}^{N} [\ell i/R] + N\sigma$$

where σ = Average bus idle time prior to the beginning of a message transmission, and the subscript i corresponds to the terminal #i.

Remark 2. For a given traffic, the cycle time may be interpreted as the total time required to complete the transmission of a waiting message, if there is one, from each of the N terminals.

Definition 7. For a given traffic, normalized cycle time, G′ is defined as the ratio of the expected values of the cycle time and the message interarrival time, that is, $G' = E[\tau]/E[T]$.

Remark 3. For a given G, individual protocols may load the medium to different levels and thus influence the performance of the integrated network system to different degrees. Therefore, G is used as a parameter for selection of network access protocols instead of G′. A limit of G above which a given protocol is expected to overload the medium, resulting in message rejection, needs to be specified.

Definition 8. The critical offered traffic Gcr, for a protocol, is defined as the largest offered traffic for which no message frame is rejected under steady states due to queue saturation for a periodic traffic with constant message lengths.

Remark 4. $G' \leq 1$ if $G = Gcr$.

Definition 9. Throughput S is defined as the ratio of the average rate of the data bits transmitted through the medium under steady state and the medium bandwidth.

In a feedback control loop, messages are transmitted via the network medium from the sensor terminal to the controller terminal and from the controller terminal to the actuator terminal. Thus the control system is subject to time-varying delays due to data latency, and its performance is dependent on the traffic in the network. Additional definitions of pertinent parameters are introduced in this regard.

Definition 10. Sensor-controller delay Θsc is defined as the interval between the instant of arrival of the sensor data at the transmitter queue of the sensor terminal and the instant when the controller starts processing the same data or would have done so if not replaced by any fresh sensor data.

Remark 5. If the sensor and controller have the same sampling time T, then Θsc can be expressed in terms of the sensor-to-controller data latency δ and the time skew $(0 \leq \Delta s < T)$ between the sensor and controller terminal sampling instants as

$$\Theta sc(\delta, \Delta s) = kT + \Delta s \text{ for } (k-1)\,T + \Delta s \leq \delta < kT + \Delta s$$

where k is a non-negative integer.

Definition 11. Controller-actuator delay Θca is defined as the sum of the processing delay at the controller and the data latency between the controller and actuator terminals.

Remark 6. The waiting time of the control signal at the actuator terminal is negligible, that is, the control signal is assumed to act upon the plant as soon as it arrives at the actuator terminal.

Remark 7. Θsc and Θca are time-varying and possibly stochastic and, therefore, may not be lumped together for stability analysis of control systems.

Remark 8. Sources of the network-induced delay in a feedback control loop are:

Remark 8. Sources of the network-induced delay in a feedback control loop are:

- Delays associated with Θsc and Θca.
- Additional delay contributed by rejected messages due to queue saturation and detected frame errors.

Asok Ray holds a Ph.D. degree in Mechanical Engineering and Master degrees in Electrical Engineering, Computer Science, and Mathematics. Dr. Ray has about fifteen years of research and management experience at GTE Strategic Systems Division, Charles Stark Draper Laboratory, and MITRE Corporation. He has also held research and academic positions at Carnegie-Mellon University and Massachussetts Institute of Technology. Dr. Ray joined the faculty of Mechanical Engineering at the Pennsylvania State University as an Associate Professor in July 1985.

Current research interests of Dr. Ray include distributed control systems and applied stochastic processes, and computer networking for aeronautical and manufacturing systems.

Dr. Ray has authored or co-authored over one hundred research publications, and is a Senior Member of IEEE and a member of ASME.