Real time communications in manufacturing

Luís Almeida, Miguel Leon-Chavez, José Alberto Fonseca, Jean-Pierre Thomesse

Real Time Communications In Manufacturing

L. Almeida[1], M.L. Chavez[2], J.A. Fonseca[1], J.P. Thomesse[2]

[1] Universidade de Aveiro, Dept. Electrónica e Telecomunicações, U.I. INESC, 3810 Aveiro, Portugal
Tel +351 34 370984, fax +351 34 381128, e-mail {lda,jaf}@ua.pt

[2] LORIA-INPL, 2 avenue de la forêt de haye, F54516 Vandœuvre lès Nancy, France
Tel +33 3 83 59 55 76, fax +33 3 83 44 07 63, e-mail {mleon,thomesse}@loria.fr

ABSTRACT

Manufacturing applications are today distributed and integrated, relying on different types of communication networks to interconnect the different levels of the systems' architecture. This paper discusses the main topics on manufacturing communications from the applications requirements to the most common solutions derived to fulfill them.

Temporal and spatial properties of the data to be exchanged are analyzed. Data is classified as periodic, aperiodic and sporadic and, for these types, the communication system performance requirements are identified with emphasis on the real-time aspects. A similar discussion is carried on in what concerns the size of data to be transferred per transaction. The use of multimedia and of wide-area communication is also briefly explored.

Solutions for different levels of the CIM (Computer Integrated Manufacturing) architecture are identified and an overview of their characteristics is presented. This includes a short description of their organization under the OSI Reference Model and a more detailed discussion on techniques for traffic scheduling. This discussion is mainly focussed on the techniques to control the access to the communications medium and on the use of static or dynamic scheduling, taking into account the data temporal constraints. A short overview of the possibilities opened by the use of Internet in these applications is also included.

Keywords: Industrial communications, real-time systems, real-time scheduling.

1. INTRODUCTION

Manufacturing applications are more and more distributed and integrated, not only inside the factory itself but also between remote sites of production, of management, of product development and of product distribution. These applications are then composed of a lot of software pieces which have to cooperate, to be coordinated in order to provide the quality of service required by the different end-users. Various communication networks are then the support of this cooperation and become now the backbone of the manufacturing applications.

Communication systems provide different services according to the different needs, and an important point of comparison is their capability to meet the constraints associated to the traffics induced by the software processes and essentially the real time constraints.

This paper deals with these topics. What is the real time communication and the non-real time one? Which are the needs? Which are the possible solutions? Is a future common solution possible for current very different networks?

To treat these aspects, the paper will be organized as follows. In a first part, a top-down approach will be considered to analyze the end-users requirements and to identify the different traffics in relation with the current architectures defined for such applications as well inside a factory, around local area networks, as in remote cooperation based on wide area networks.

In a second part, the main solutions will be studied considering the real-time communication networks which are essentially known as fieldbuses. The different message scheduling and the resource sharing methods used will be summarized and classified for having a global view on the real-time communication management. These methods will also be analyzed as possible solutions leading to integration, not only in the factory but also in wide area cooperation.

2. APPLICATIONS REQUIREMENTS

The application requirements within a manufacturing system are directly deduced from those of the end-users. In manufacturing applications, the end users are the production machines controllers, the transport systems, the people who are in charge of the control, of the maintenance, of the production scheduling, of the technical management and so on.

The communication requirements of the distributed manufacturing applications found in the automotive industry, textile machinery, factory automation, semiconductor fabrication (chip-manufacturing), electronics manufacturing, food and beverage, chemical processing, and so on, can be essentially characterized by the types of the data that need to be exchanged, either in what concerns their temporal properties as well as the size of the respective transactions.

Temporal requirements

In these applications time-constrained data can be commonly found, i.e., data that have a limited time validity. Hence, all the processing of such data, including both computing and transmission, must be carried out in a bounded time. This time window, or deadline, must be sufficiently small in order to allow efficient monitoring and control of the manufacturing process operation and consequently of the quality of the products. How short such deadline must be depends on the particular physical process under consideration. Examples of such data are the feedback and control signals of closed loop control subsystems such as the axis position control in a robot arm or the temperature control in a chemical reaction.

These data are called *critical data* which can be further characterized as *soft* or *hard.* In the former case, a few violations of the processing deadline can be tolerated. On the other hand, in the latter case, the applications cannot tolerate any delay in the respective processing beyond the deadline. This is the case when missing a single deadline may jeopardize the whole system operation and even endanger equipment, people or environment.

Besides critical data, some applications found in manufacturing systems make also use of non-time-constrained data. In this case, a delay on the respective processing has no effect on the proper operation of the manufacturing process nor on the quality of the products. This is called *non-critical data* and is normally handled by background application tasks.

Another time-related aspect that concerns the data in a distributed manufacturing system is the regularity and frequency at which it is produced and processed. Typically, the manufacturing applications know the current status of the manufacturing process by means of the data produced and transmitted either periodically or randomly by the sensors or higher level devices. In the same way, the applications control the manufacturing process by means of the control data transmitted to the actuators or control devices. In any case, when the transmission of such data is carried out with a constant frequency then it is named *periodic* and it can be critical or not.

On the other hand, critical data resulting from random emergency situations is normally transmitted aperiodically and thus it is named *aperiodic*. Notice, however, that there might be non-critical data requested aperiodically, too, e.g. by a high-level supervision task.

In some cases it is possible to bound the minimum separation time between two consecutive aperiodic transactions of the same source. This particular sort of aperiodic data is normally known as *sporadic*.

<table>
<tr><td rowspan="2">DATA</td><td colspan="2">Critical</td><td rowspan="2">Non-critical</td></tr>
<tr><td>Hard</td><td>Soft</td></tr>
<tr><td>Periodic</td><td colspan="2" rowspan="3">Real-time</td><td rowspan="3">Non-real-time</td></tr>
<tr><td>Aperiodic</td></tr>
<tr><td>Sporadic</td></tr>
</table>

Figure 1. Temporal properties of data in distributed manufacturing systems.

The previous classification of the data according to both its criticality and its transmission frequency is depicted in figure 1. The shadow area represents the critical data, also known as *real-time data*, which can be under two different time-constraints: on one hand, the deadline constraint (associated to its criticality), on the other hand, the transmission frequency constraint (associated to its transmission regularity). Nevertheless, both real-time and non-real-time data can be of a periodic, aperiodic or sporadic nature.

In brief, with regard to these types of data, the communication requirements of the manufacturing applications can be summarized as follows:

- Distinction between real-time and non-real-time traffic is essential [1]. The communication system should convey the data from source to destination in either type of traffic.
- It should also be possible to distinguish the relative urgency of data, basically between periodic and aperiodic hard-critical data.
- The communication system should guarantee that all hard-critical data transfers will be completed within a bounded time interval.
- The communication system should try its best effort to transfer the soft-real-time data, since minimizing the number of delayed or lost transfers will result in a proper operation of the manufacturing process in the long term, for example.
- Non-real-time data transfers shall not jeopardize the delivery of the real-time data.

Size of data per transaction

With regard to the size of data to be transferred per transaction, some manufacturing applications exchange bits, some of them exchange words of data, some others exchange streams of data and long streams of data.

The bits of data are typically associated with the status of very simple devices, usually restricted to on/off status. Applications dealing with such sort of devices require transfers of a few bits of I/O data per node, only. However, some devices, either sensors or actuators, are more elaborated and use more information such as status, diagnostic and control. In this case, the associated applications require the transfer of words of I/O data.

Going up in the device complexity hierarchy there are the higher level devices such as CNC machines, robots, and other equipment. The operation of such devices requires a stream of several words of data associated with status, configuration, diagnostic, calibration and control. The applications that manage these devices require, then, the exchange of streams of I/O data. At even higher levels in the manufacturing system there is the need to transfer files, to access databases, to gather the data associated to the whole manufacturing system, to allow remote access to the system, etc. These operations commonly require the exchange of long streams of data per transaction.

It is interesting to note that, in general, as the size of the data per transaction increases, the associated time-constraints, either deadline or frequency, become more relaxed. In fact, transactions carrying bits or words of I/O data are normally associated to sensors and actuators which are directly involved in the inner control loops of a system or subsystem, thus requiring relatively high communication rates (periods or deadlines in the order of 1-10 ms). Streams of I/O data, normally associated with higher level devices, are involved in outer control loops, thus requiring relatively lower communication rates (periods or deadlines in the order of 100 ms). At the manufacturing system higher levels, the applications typically require long streams of data associated to the overall control and supervision of the whole plant and the feedback actions are usually performed by human intervention. Thus, the respective temporal requirements are even more relaxed and response times in the order of seconds are admissible.

In any case, a time constraints analysis must always be considered associating attributes such as response times, deadlines and frequencies to the tasks or to the messages and using a common method to specify them (temporal logic [2], timed automata [3], ...).

Multimedia applications

Advances in the semiconductor industry led to the availability of network interfaces with very large bandwidths. This opened the way to the integration of different human communication media (sound and vision) in digital communication systems including manufacturing communication systems. Multimedia applications have the potential of facilitating the communication with humans through the use of images, sound, video, animation, simulation, etc [4]. This is of great interest in manufacturing systems where multimedia can be used to support the presentation of status information of devices, higher level equipments, cells, etc. to the people who are in charge of the control, maintenance, production scheduling, technical management, etc.

The use of such communication media resulted in new types of data which are now more and more used by manufacturing applications. Such new types can be divided in two groups: *continuous-media*, such as audio and video, and *discrete-media*, such as text, graphics and images. While continuous-media are characterized by substantial volumes of data composed of time dependent sequences of relatively small information units, involving temporal requirements such as synchronization and bounded transmission times, the discrete media are often time independent information composed of a small number of relatively large units.

The applications that process these types of data are called *multimedia applications*. The respective requirements, particularly for continuous-media data, involve the ability to manipulate high data volumes at high data rates with time dependent data values. Moreover, these applications require sophisticated support for synchronization both to co-ordinate multimedia presentations and to maintain timing relationships between a number of real-time streams [5].

For multimedia applications, the Internet community, the ATM community, and the Tenet Group have proposed different service models [6]. The models differ in many ways but share some of the fundamental characteristics. For example, there is agreement about the necessity that an integrated services network provides at least the two extreme services: non-real-time on one side, and deterministically guaranteed-performance service on the other. Also, because of the low utilization of resources expected to characterize the latter, the need is felt for at least one intermediate, less expensive service type that caters to tolerant real-time applications.

Wide-area communication

Manufacturing applications are more and more distributed and integrated, not only inside the factory itself but also between remote sites of production, of management, of product development and of product distribution. Hence, there is not only the need for local communication within the factory but also for remote communication which has to be carried out over wide-area networks.

The communication requirements of these wide-area distributed applications are similar to those of the higher level applications within the factory plant. Exchanges of long streams of data as well as of multimedia streams are typically part of these applications. However, the wide-area dimension introduces new technical problems and strongly restricts the available solutions.

3. SOLUTIONS OVERVIEW

Existing networks for the CIM architecture

There are presently many different communication systems, protocols and profiles available off-the-shelf, each with its capabilities [7]. When considering manufacturing systems, the applications' communication requirements identified in the previous section have to be taken into account in order to choose the most appropriate solution. However, within a single manufacturing system, such needs vary either in terms of the associated time constraints or in terms of the amount and type of the information that must be exchanged. For example, the communication needs of a machine-tool controller are different from those of a cell controller and also different from those of plant control.

In order to satisfy these different needs, several communication systems coexist in the manufacturing plant and are hierarchically organized according to levels in what is known as the Computer Integrated Manufacturing (CIM) architecture [8].

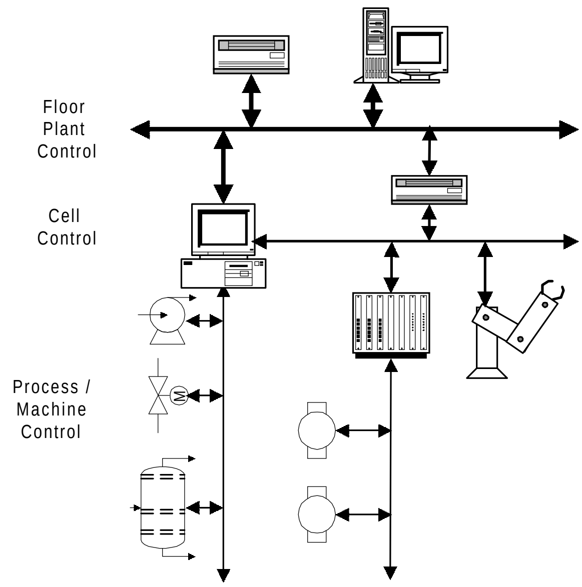

Figure 2. The Computer Integrated Manufacturing architecture with three main levels.

Three main levels are usually considered (fig. 2): floor plant control, cell control and process/machine control. For each of these levels there are adequate off-the-shelf solutions for the respective communication needs.

The higher level, floor plant control, promotes the integration of all cells within the factory. It supports the global factory management and monitoring. The long streams of data shared by applications at this level are normally conveyed by *general-purpose networks* such as Ethernet, IBM Token Ring, FDDI, MAP, ATM. Some of them already have limited real-time capabilities. Others, such as Ethernet, have a good average performance up to a certain level of traffic load but, above that, the performance degrades substantially [9]. The support of distributed multimedia applications is also carried out by these networks.

The simultaneous control of several processes or machines is carried out at the cell control level. Due to the temporal and size properties of the data transactions at this level, general-purpose networks cannot be used. Instead, special purpose networks known as *fieldbuses* are used which rely on simpler protocols with reduced communication and processing overhead. These protocols make use of scheduling techniques particularly adapted to support time constrained communication services. Examples of such field networks are Profibus/FMS [10, 11], WorldFIP [12,13], P-Net [14], Interbus-S [15], IEC Fieldbus [16] and Mini-MAP which is a reduced profile of the higher level network MAP.

At the lowest level, close to the field devices, i.e. sensors, actuators and controllers, is the process/machine control level which includes, for example, the position control of axis in a robot arm or in a CNC machine, or the temperature and pressure control in a chemical reaction. The requirements of the applications at this level are even more demanding in what concerns timing constraints. Also, the data exchanged per transaction is short. In these cases, standard fieldbuses as those referred above may not cope with the application requirements. In such case, a particular type of fieldbuses is used which are known as *device networks*. Examples are Profibus/DP and Device WorldFIP (DWF) which are reduced profiles of the respective fieldbus standards, DeviceNet and Smart Distributed System (SDS), which are based on the Controller Area Network (CAN) [17, 18].

At this low level there is also another type of networks which use very simple protocol solutions to provide fast interconnection of simple sensors and actuators. They are known as *sensor networks* or also as *sub-fieldbuses* due to their simplicity. Examples of these are AS-Interface (AS-I) and Seriplex.

Above all these levels, the wide-area distribution of large sectors of the manufacturing system such as production, management and product development should be considered as it is bringing a new dimension to the factory upper layer. The interconnection of the different sites is carried out with wide-area networks (WANs) which rely upon services provided by telecommunications operators. At this level either dedicated leased lines or ATM services are typically used so that some real-time performance can be achieved.

The protocol stack - OSI compliance

Most of the network systems referred above are based on a layered structure known as the OSI Reference Model. Such model organizes the protocols used and the services provided by a general communication system in a stack of layers [19]. The process of transmitting a message from one node to another in the network, requires the user application in the transmitting node to call upon the services of the top layer, which is the application layer. Then, the services of each layer call upon the services of the next layer down to the bottom of the stack, which is the physical layer. There, the effective physical transmission is carried out and the message is transferred from the transmitting node to the receiving one. On this node, the message is forwarded in the opposite direction, from the physical layer up to the application layer where it is delivered to the receiving user application (figure 3-a).

The OSI reference model uses seven layers which are known as: *application* which deals with the application interface, *presentation* for data interpretation which allows for interoperability among heterogeneous equipments, *session* concerning the execution of remote actions, *transport* which deals with end-to-end communication control, *network* concerning logical addresses of nodes and routing, *datalink* which is responsible for the access to the communication medium and for the logical transfer of the data, and the *physical* layer that concerns the way the communication is done physically. Communication systems based on these seven layers have a high level of flexibility and interoperability. However, the seven layers also impose a considerable processing and communication overhead.

After the MAP (Manufacturing Automation Protocol) project in the early 1980s it became commonly accepted that the implementation of the OSI seven layer capability at the device level is too complex, too slow and too expensive. Therefore, in order to respond adequately to the requirement of sending short messages at frequent intervals the application layer was modified to allow direct access to the datalink. The resulting networking systems, the fieldbuses, have since then been described as based on a 3 layers structure which includes the physical, datalink and application layers, only (figure 3-b). Some of the functionalities of the missing layers are still present in some fieldbuses but are merged in the application layer, e.g. support for interoperability, end-to-end control, fragmentation and reassembling of messages, routing among several fieldbuses.

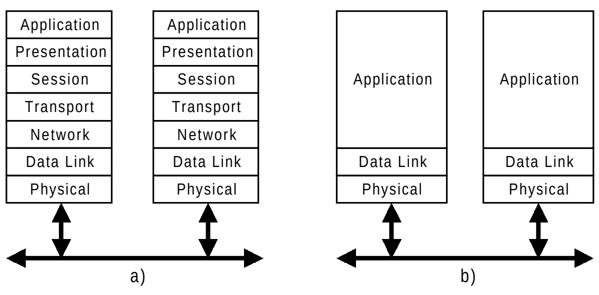

Figure 3. The OSI reference model, a) full 7 layers stack; b) reduced 3 layers stack.

Traffic scheduling: a key to handle temporal constraints

Although many of the options taken in the protocols of each layer may influence the ability of the networking system to handle time constrained communication, the Medium Access Control (MAC) which is part of the datalink layer (DLL) is the one that has the utmost importance in this matter because it is responsible for the scheduling of the network trafic [20, 21]. This scheduling establishes the order by which communicating nodes access the transmission medium. Therefore, it directly influences the response time of the communication system to the requests issued by the nodes.

Controlled versus uncontrolled access: One major distinction that can be done concerning the MAC sublayer is between protocols based on controlled and uncontrolled bus access [22]. Most real-time communication systems use controlled access as a way to avoid collisions on the bus. In this case there are two approaches that can be followed, centralized and decentralized. In the former one, a special node called master or arbitrator explicitly gives each node, one at a time, the right to access the bus and send some information. This is normally known as the *Master-Slave* model and is used, for example, by Profibus between each master and the respective slaves and also by AS-I. There is a variant of the Master-Slave which is known as *Producer-Distributor-Consumer* model. In this case, the master does not explicitly address one given node, instead, it addresses a data entity, e.g. a process variable (*source addressing*). The node responsible for the production of such entity, whichever node it is, responds to the master call and broadcasts the respective value. This is the approach followed by WorldFIP and by IEC Fieldbus (between the Link Active Scheduler and the Link Elements). Centralized access control supports tight synchronization and thus it is particularly suited to handle periodic transactions with low jitter at fast rates. It is, then, suited to the lowest levels within the CIM architecture.

Controlled access protocols can also be distributed. In this case, several masters coexist at the same logical level. One possible technique to allow such coexistence is to partition global time in widows during which only one master can access the network (TDMA - time division multiple access). The right time window for a given node to access the bus is identified by a token that is circulated among the nodes. Timely behavior is achieved limiting the time that a node can hold the token continuously before sending it to the next node. Profibus uses this sort of token passing among bus masters as well as IEC Fieldbus among special nodes called Link Masters. Besides, also FDDI and MAP use a timed-token protocol. Another decentralized technique for allowing several nodes to access the bus is to use a predefined order of access. This is known as virtual token passing and is used in P-Net. In general, token passing techniques are not well suited to handle periodic traffic. Besides, for large networks with many nodes, the token takes a long time before it returns to the same node thus severely limiting the highest rate at which periodic traffic can be transmitted. Therefore, this sort of access control is more suited to the middle and upper levels within the CIM architecture.

The alternative to controlled access is uncontrolled access. These MAC protocols are generally known as CSMA (carrier sense multiple access) and they are fully distributed. All nodes are considered at the same logical level which grants the communication system a high level of modularity and flexibility. Besides, the communication overhead is low since there are no extra messages such those issued by master nodes or those required to carry tokens. However, in this case it is possible that several nodes start transmitting at the same time leading to access collisions.

Three main different approaches are usually followed in what concerns the way collisions are handled. One is to avoid collisions by the use of synchronization and timers (CSMA-CA). This is used, for example, in the ARINC fieldbus that is based on the TTP protocol [23]. Another possibility is to use a way of resolving collisions deterministically (CSMA-DCR) [24]. There are several methods to achieve such deterministic resolution. One example is CAN. In this case, a communication model known as *Producer-Consumer* is used which relies in source addressing such as in WorldFIP, each data entity must have a unique identifier which also sets the priority of the respective transactions. When several nodes access the bus simultaneously, the transaction with the highest priority proceeds while all other nodes quit transmitting and retry at the end of the current transmission. Networks based on CAN are well suited to the lowest levels within the CIM architecture.

Perhaps the most well known CSMA protocol is the one used by Ethernet (CSMA-CD). In this case, whenever two or more nodes start transmitting simultaneously, the conflict is detected and all nodes stop the transmission and try again later, after a random period of time. This randomness may result in unbounded access delay and therefore, this sort of access protocols are not well suited to real-time applications. In fact, for traffic loads above 60% the performance of the bus starts degrading exponentially resulting in long and unpredictable access delays. However, it is still possible to use Ethernet in real-time applications if a controlled access scheme is forced at the application level, thus avoiding the pernicious effects of access collisions. For example, it is possible to implement a Master-Slave scheme over Ethernet so that any node transmits only when specifically addressed by a particular master node which must be unique in the system.

Static versus dynamic scheduling: Another issue of great relevance to the real-time performance of a communication system is whether the traffic scheduling is carried out statically or dynamically. In the former case, all the communication entities as well as their communication needs are known *a priori* and will not change during system operation. Hence, a schedulability analysis can be performed off-line, at pre-run--time, which will give information on whether the communication time-constraints will always be met. This feature can be useful for periodic traffic and even for sporadic traffic. However, aperiodic traffic normally associated to transient overload or emergency situations is not well handled by this approach due to the unpredictability of the respective arrival instants. In brief, *static scheduling* allows for timely guaranteed scheduling in systems with a fixed set of requirements.

There are basically two distinct forms of static scheduling: *static table-based* and *static priorities-based*. In the former case, the scheduler operates off-line and builds a schedule table which is then used on-line by a dispatcher, running on a master node, to timely initiate the respective transactions. This approach, used to manage the periodic traffic in WorldFIP, results in very low processing overhead at run-time and thus it is suited to be used with high speed networks, supporting high rate periodic transactions. Nevertheless, no on-line adaptations are allowed and thus, the flexibility of the communication system is rather low. On the other hand, when static priorities-based scheduling is used, just the schedulability analysis is performed off-line. A low level scheduler executes at run-time to enforce the processing of the transactions according to the respective priorities. Since part of the scheduling is performed on-line, this approach is more flexible, it may handle the aperiodic traffic with the help of a periodic server [25], and above all, does not require a master node. Notice however, that on-line changes to the set of communication requirements that may lead to increased needs are not allowed. In this case, the system must be halted and the schedulability analysis must be redone before putting the system on again. This approach can be used with most real-time communication systems where there are implicit or explicit static priorities associated to the communicating entities, either operating in Master-Slave, Token-Passing, Virtual Token-Passing, CSMA-DCR or CSMA-CA modes.

The alternative to static scheduling is *dynamic scheduling*. In this case, no prior knowledge concerning the communication entities is required. At run-time the system handles the communication requests dynamically, as they appear. This approach is highly flexible since any changes to the communication needs can be accommodated, or at least considered, on-line. In many situations, no pre-run-time schedulability analysis is performed and thus a *best-effort* approach is used: the communication system does its best to handle the current requirements but the scheduling is not guaranteed. This is the normal situation in most communication systems where the pre-run-time schedulability analysis is not carried out by the system itself but it is left for the user (except for the periodic traffic in WorldFIP which is always guaranteed due to the static table-based approach used).

To achieve guaranteed dynamic scheduling, on-line schedulability analysis must be used. This is known as a *planning approach*. In this case, whenever there is a request for extra resources (e.g. more bandwidth, new periodic (sporadic) streams, etc.) the schedulability analyzer is invoked to determine whether the new set of requirements can be met and guaranteed. If yes, the request is processed, otherwise, the requesting entity is notified that it is not possible, for the moment, to satisfy the request. This feature is also part of a larger concept known as Quality-of-Service (QoS) control. The only existing communication system that allows a planning approach is ATM which is based on a well defined QoS architecture [4]. It is possible to specify the required bit rate for a given stream, to specify the timing relation between source and destination (CBR - Constant Bit Rate, VBR - Variable Bit Rate) as well as the connection mode (either connection oriented or connectionless). ATM is particularly well suited to simultaneously handle streams of data of different nature such as voice, video, real-time control data and others. Either admission control, resource reservation and packet scheduling in ATM are still subject of research work, e.g. [27]. There is also recent research work concerning the use of the planning approach in existing communication systems such as WorldFIP [28] and CAN [26].

Support for Internet Protocol

The number of applications available for use on the Internet is enormous. It would be desirable to make use of some of them within manufacturing systems, particularly WWW-related applications. In fact, the World Wide Web (WWW) provides a powerful tool to view, monitor, control and analyze the contents of factory communication systems. The question here is whether it is possible to use the Internet Protocol (IP) on top of the above mentioned real-time networks. The answer is not immediate since there are several possibilities. For example, with *IP tunneling* each whole IP frame is sent in the data field of another networks' frame. This incurs in large communication overhead and thus can only be used on middle/high level networks such as FDDI, MAP or ATM which can handle large messages. However, this technique has the advantage of requiring no modifications to the original IP applications.

Another possibility is to use a *gateway* within each node that runs IP applications. Although this can be more efficient in terms of communication overhead, the processing overhead is considerably high, making it impossible to use it in most field devices. According to this approach, each IP service would be translated to a corresponding service in the real-time communication system and vice-versa. In some situations, a true bijective translation is not possible and compromises must be made.

In any case an address translation mechanism is also required, to convert the IP address into the right network address. Besides, notice that IP is based on a connection-oriented *client-server* communication model. The translation of this communication model to connectionless communication and/or to producer-consumer-based communication also requires several compromises and, in some cases, it is even not possible. The fact that IP does not support timing constraints information may also cause further problems concerning the timeliness of the factory communication system. In order to prevent that it is necessary to express the time constraints and/or priorities associated to each communicating entity in a way which can be understood by the underlying real-time communication system. This limitation has been considered by the IETF (Internet Engineering Task Force) in the new version of the protocol IPv6 which already allows for the definition of a QoS parameter associated to the desired bit rate.

4. CONCLUSION

In this paper the communication needs inside a manufacturing system have been identified and classified according to the respective data size and temporal constraints (either deadline and rate). In particular, the requirements for real-time communication have been identified and characterized. Then, the existing solutions for manufacturing communication systems have been analyzed and classified according to the degree in which they meet the real-time communication requirements.

Particular attention was devoted to some aspects of traffic scheduling which have a major impact on the real-time capabilities of a communication system, namely the medium access control policy and the static versus dynamic nature of traffic scheduling.

One aspect that results from the analysis is the non-existence of a single off-the-shelf solution that covers all the communication needs inside a manufacturing system. This leads to the need for several coexisting network systems with the inherent complexity in maintenance and management. However, some communication systems already offer different profiles suited for two of the three considered levels of the CIM architecture which tend to reduce this dispersion problem, e.g. MAP and Mini-MAP, Profibus/FMS and Profibus/DP, WorldFIP and DWF.

In such a scenario ATM networks are referred by some authors [25] as having a great potential to become the main choice for factory communication systems. This is mainly due to its embedded QoS architecture which allows an easy integration of various types of traffic particularly multimedia and real-time control. Besides, it is highly scalable, allowing for bit rates which go from low (tens of Kbit/s) to very high (few Gbit/s) and it may operate over optical fiber resulting in improved noise immunity. Since telecommunications operators are starting to offer ATM-based services over wide-areas, the use of ATM inside the factory may also simplify the setup of wide-area networks by the interconnection of remote facilities.

REFERENCES

[1] ISO TC 184/SC5/WG2, Architecture and Communications – User requirements for system supporting time critical communications, 1992.

[2] Z. Manna, A. Pnueli, The temporal logic of reactif and concurrent systems: Specification, Springer-Verlag, 1992.

[3] R. Alur, D.L. Dill, “A theory of timed automata”, Theoretical Computer Science, 126:183-235, Elsevier, 1994.

[4] J.R. Pimentel, “Implications of Emerging Technologies on Factory Communication Systems”, In: 2nd IEEE International Workshop on Factory Communication Systems - WFCS’97, pp. 3-7. Barcelona, Spain, 1997.

[5] G. Diaz, Z. Mammeri, J.P. Thomesse, “Distributed system architectures for multimedia applications: An analysis of the approaches”. In: World Multiconference on Systemics, Cybernetics and Informatics – SCI' 97, pp. 13-20. Caracas, Venezuela, 1997.

[6] D. Ferrari, “Multimedia network protocols: where we are?”, Multimedia Systems, Vol. 4, 1996, pp. 229-304.

[7] J.P. Thomesse, “A Review of the FieldBuses”, Annual Reviews in Control, 22:35-45, 1998.

[8] J.R. Pimentel, Communication Networks for Manufacturing, Prentice-Hall, Englewood-Cliffs, N.J., 1990.

[9] B. Abeysundara, A. Kamal, “High-Speed Local Area Networks and their Performance: A Survey”, ACM Computing Surveys, 23:221-264, 1991.

[10] DIN, German Standards 19245-1 to 19245-3, PROFIBUS, Process Fieldbus, 1990.

[11] DIN, German Draft Standard 19245-4, PROFIBUS-PA, Profibus for Process Automation, 1996.

[12] D. Galara, J.P. Thomesse, FIP: Proposition d' un système de transmission multiplexé pour l' échange entre capteurs, actionneurs et automates reflexes, Ministère de l' industrie et de la recherche, France, March 1984.

[13] AFNOR, French Standard NF C46601 to C46607, FIP bus for exchange of information between transmitters, actuators and programmable controllers, 1989.

[14] DS, Danish Standard, DS 21906, P-Net, Multi-master, multi-net fieldbus for sensor, actuator and controller communication, 1990.

[15] DIN, German Draft Standard 19258-1, Interbus-S, sensor and actuator network for industrial control systems, 1996.

[16] IEC, IEC Standard 1158, Digital Data Communications for measurement and control - Fieldbus for use in industrial control systems. Part 2 Physical Layer Specification and Service Definition. Part 3 Data Link Service Specification. Part 4 Data Link Protocol Specification. Part 5 Application Layer Service Specification. Part 6 Application Layer Protocol Specification., 1998.

[17] R. Bosch, GmbH, CAN Protocol Specification V2.0 (A, B), 1992.

[18] ISO DIS 11898, Road Vehicules Interchanges of Digital Information - Controller Area Network (CAN) for High Speed Communication, 1992.

[19] A. S. Tanenbaum, Communication Networks for Manufacturing, Prentice-Hall, Englewood-Cliffs, N.J., 1988.

[20] J.F. Kurose, M. Schwartz, Y. Yemini, “Multiple-Access Protocols and Time Constrained Communication”, ACM Computing Surveys, 16(1):43-70, 1984.

[21] N. Malcolm, W. Zhao, “Hard Real-Time Communication in Multiple-Access Networks”, Real-Time Systems, 8:35-77, 1995.

[22] Z. Mammeri, J.P. Thomesse, “Réseaux Locaux, Couche Physique et Couche Liaison de Données” TEKNEA, 1991.

[23] H. Kopetz, G. Grünsteidl, “TTP - A Protocol for Fault-Tolerant Real-Time Systems” IEEE Computer, 24(1):22-26, 1994.

[24] G. Le Lann, N. Rivierre, “Real-Time Communications over Broadcast Networks: the CSMA-DCR and the DOD/SCMA-CD Protocols”. In: Actes des Conférences de RTS' 94, pp. 67-84. Paris, France, 1994.

[25] J.K. Strosnider, T. Marchok, J. Lehoczky, “Advanced real-time scheduling using the IEEE 802.5 token ring”.In: Proceedings of the IEEE Real-Time Systems Symposium, pp. 42-52, 1988.

[26] L. Almeida, J. Fonseca, P. Fonseca, “Flexible Time-Triggered Communication on a Controller Area Network”, In: Proceedings of Work-In-Progress Session of IEEE Real-Time Systems Symposium, RTSS’98, Madrid, Spain, 1998.

[27] H. Hanson, M. Sjodin, “Response Time Guarantees for ATM-Networked Control Systems”, In: 2nd IEEE International Workshop on Factory Communication Systems - WFCS’97, pp. 213-222, Barcelona, Spain, 1997.

[28] L. Almeida, R. Pasadas, J. A. Fonseca, “Using The Planning Scheduler to Improve Flexibility in Real-Time Fieldbus Networks” Control Engineering Practice, Vol. 7, Nº 1, 1999, pp. 101-108, , Elsevier Science.