

Supporting real-time distributed computer-controlled systems with multi-hop P-NET networks

Eduardo Tovar

Department of Computer Engineering, Polytechnic Institute of Porto,
Rua de São Tomé, 4200 Porto, Portugal; Tel.: +351.2.8340500; Fax: +351.2.8321159
emt@dei.isep.ipp.pt

Francisco Vasques

Department of Mechanical Engineering, University of Porto, Rua dos Bragas,
4099 Porto Codex, Portugal
vasques@fe.up.pt

Alan Burns

Department of Computer Science, University of York,
Heslington, York, YO1 5DD, United Kingdom
burns@cs.york.ac.uk

Abstract

Fieldbus communication networks aim to interconnect sensors, actuators and controllers within process control applications. Therefore, they constitute the foundation upon which real-time distributed computer-controlled systems can be implemented. P-NET is a fieldbus communication standard, which uses a virtual token-passing medium-access-control mechanism. In this paper pre-run-time schedulability conditions for supporting real-time traffic with P-NET networks are established. Essentially, formulae to evaluate the upper bound of the end-to-end communication delay in P-NET messages are provided. Using this upper bound, a feasibility test is then provided to check the timing requirements for accessing remote process variables. This paper also shows how P-NET network segmentation can significantly reduce the end-to-end communication delays for messages with stringent timing requirements.

Keywords: Distributed computer-controlled systems; P-NET fieldbus networks; real-time communications; message response time analysis.

1. Introduction

Local area networks (LANs) are becoming increasingly popular in industrial computer-controlled systems. LANs allow field devices like sensors, actuators and controllers to be interconnected at low cost, using less wiring and requiring less maintenance than point-to-point connections (Lenhart, 1993). Besides the economic aspects, the use of LANs is also reinforced by the increasing decentralisation of control and measurement tasks, as well as by the increased use of intelligent microprocessor-controlled devices.

Broadcast LANs aimed at the interconnection of sensors, actuators and controllers are commonly known as fieldbus networks. In the past, the scope of fieldbuses was dominated by vendor-specific solutions, which were mostly restricted to specific application areas. Moreover, the concepts behind each proposed network were highly dependent on the manufacturer of the automation system. Each one had different technical implementations and claimed to fulfil different application requirements, or to fulfil the same requirements but with different technical solutions (Cardoso and Tovar, 1996). More recently, vendor-independent standardised fieldbuses, supporting the open system concept, have started to be commonly used. Particular relevance must be given to the European Standard EN 50170 (Cenelec, 1996), which encompasses three widely used fieldbuses: P-NET (Pnet, 1994), PROFIBUS (Profibus, 1992) and FIP (Afnor, 1990).

This paper addresses the ability of P-NET to cope with the real-time requirements of distributed computer-controlled systems (DCCS). In essence, timing requirements mean that messages must be sent and received within a bounded interval, otherwise a timing fault is said to occur. That means, for instance,

that a control device must be able to read data from a remote sensor within a specified interval, whatever the network load.

The P-NET medium-access-control (MAC) protocol is based on a virtual token-passing procedure, used by master stations to grant the bus access to each one of them, and a master-slave procedure used by master stations to communicate with slave stations. This master-slave interaction is called a message cycle: the master sends a request frame and the addressed slave immediately sends a response frame. At each of the visits of the token, a master is able to perform, at most, one message cycle.

Typically, the process-relevant devices (sensors and actuators) are accessed through a slave network-interface, whereas the distributed control algorithms reside at master stations. Therefore, in P-NET, the end-to-end communication delay (Tindell *et al.*, 1995) for master-slave transactions (those that typically deal with real-time traffic) is composed of the following four major components:

1. generation delay: the time elapsed between the release of the sender task and the queuing of the related message request;
2. queuing delay: the time taken by a message request to access the communication medium after being queued;
3. transmission delay: the time taken by a message request to be transmitted on the communication medium and processed at the slave side, added to the time taken by the message response to be transmitted back to the master;
4. delivery delay: the time taken by the master's application task to process a message response.

The queuing delay is a consequence of the contention not only between message requests from the same master but also with message requests from other masters. The impact of the first factor in the overall queuing delay depends on the policy used to queue the message requests, while the second factor depends on the behaviour of the token-passing procedure. Therefore, an evaluation of the worst-case queuing delay of message requests is of paramount importance to guaranteeing the messages' timing requirements. The end-to-end communication delay is referred to in this paper as the message response time.

The remainder of this paper is organised as follows. Section 2 describes the most important concepts behind P-NET networks. Particular relevance is given to the behaviour and timing characteristics of its MAC protocol. Based on these characteristics, an upper bound for the message response time is derived in Section 3. This upper bound gives a sufficient pre-run-time schedulability condition; that is, if the messages' relative deadlines are greater than their worst-case response time, then the real-time requirements of the supported application are guaranteed. As it will be shown, the upper bound for the message response time is a function of the number of masters in the network segment. This motivates the analysis performed in Section 4, which addresses the response time analysis for messages relayed through P-NET hopping devices. The idea behind this analysis is the following one. In P-NET networks several bus segments can be interconnected into a larger network using hopping devices, in such a way that any master on the network can transparently access any node within the network. Such a segmentation has effective advantages in terms of the real-time characteristics of the overall network. As each segment has independent virtual token-passing procedures, the token rotation time decreases. If nodes with more stringent message transactions are placed in the same network segment, the response time for those messages will be smaller than in the single-segment approach. Nonetheless, transactions that are relayed through one or more hopping devices may be required. Such multi-hop transactions will suffer an increase in their response times. In Section 4 an upper bound for the multi-hop message response time is derived. Both the basic and the multi-hop response time analysis assume worst-case peak-load situations. This allows the determination of sufficient but not necessary pre-run-time schedulability conditions. In the Appendix, the level of pessimism in these pre-run-time schedulability conditions is investigated. In Section 5 a numerical example is presented to show how to apply the pre-run-time schedulability conditions. With this example, the advantages and criteria for segmenting a P-NET network are presented, and some conclusions are drawn.

2. A brief description of P-NET

P-NET (Process network) was designed as a communications link between distributed process-control sensors, actuators and small programmable controllers. It has recently gained an increased role, as it became (along with PROFIBUS and FIP) a European Standard, the EN 50170 - General-Purpose Field Bus Communication System.

P-NET is a multi-master standard based on a virtual token-passing (VTP) scheme. In P-NET all communication is based on a message transaction principle, where a master sends a request and the addressed slave immediately returns a response. Fig. 1 illustrates the hybrid-operating mode of the P-NET's MAC.

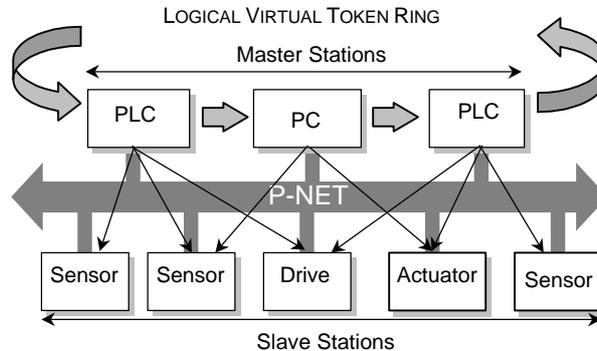


Fig. 1. Token passing and master-slave procedures in P-NET networks

The P-NET standard uses a data rate of 76800bps. This data rate resulted from weighing up the conflicting requirement for data to be transported as fast as possible, but not at such speed as to negate the use of standard microprocessor UARTS, or restrict the usable distance or cable type (Jenkins, 1997).

The VTP scheme is implemented using two protocol counters. The first one, the access counter (AC), holds the node address of the master currently transmitting. When a request has been completed and the bus has been idle for 40 bit periods (520 μ s at 76,8Kbps), each one of the access counters is incremented by one. The master whose access counter value equals its own unique node address is said to be holding the token, and is allowed to access the bus. When, as the access counter is incremented, it exceeds the "maximum number of masters", the access counter in each master is reset to one. This allows the first master in the cycling chain to gain access again.

The second counter, the idle bus bit period counter (IBBPC), increments for each inactive bus bit period. Should any transactions occur, the counter is reset to zero. As explained above, when the bus has been idle for 40 bit periods following a transfer, all the access counters are incremented by one, and the next master is thus allowed to access the bus.

If a master has nothing to transmit (or indeed is not even present), the bus will continue to be inactive. Following a further period of 130 μ s (10 bit periods), the idle bus bit period counter will have reached 50, (60, 70,...) so all the access counters will be incremented again, allowing the next master access. The virtual token-passing will continue every 130 μ s, until a master does require access.

The P-NET standard permits each master to perform only one message transaction (later defined as message cycle) per token visit. This is an important idea for the remainder of this paper.

After receiving the token, the master must transmit a request before a certain time has elapsed. This is denoted as the master's reaction time, and the standard imposes a worst-case value of up to 7 bit periods. A slave is allowed to access the bus, between 11 and 30 bit periods after receiving a request, measured from the beginning of the stop bit in the last byte of the frame. The maximum allowed delay is then 390 μ s (corresponding to 30 bit periods). Later on, this delay will also be denoted as the slave's turnaround time. For a better understanding of the basic MAC procedures and the notation used, refer to Fig. 2.

Before proceeding with the worst-case response time analysis, it is important to understand the idea of a P-NET message cycle length. A P-NET frame contains five fields: node address field (2 bytes); control/status field (1 byte); information length field (1 byte); information field (0-63 bytes); error detection field (1-2 bytes). The node address field may have up to 24 frame bytes. P-NET uses these complex addresses if multiple segments are used and special devices (P-NET hopping devices) are used to relay frames between the different segments.

As each frame byte in P-NET actually corresponds to 11 bits, a frame may have up to 759 bits (69 \times 11 bits). In P-NET all the frame bytes are sent asynchronously, with one start bit (logical zero), 8 data bits (with LSB first), one address/data bit and one stop bit. Within a frame, a start bit must immediately follow a stop bit.

Thus, considering the case that both the request and response frames have 759 bits (realistically it is more likely that either the request will be longer, in cases of data being written to a slave, or the response

will be longer, in cases of data being received from a slave), the overall sum for the longest message cycle is 1548 bit periods, corresponding to 20.15ms at 76800bps. This includes the worst-case slave's turnaround time (30 bit periods).

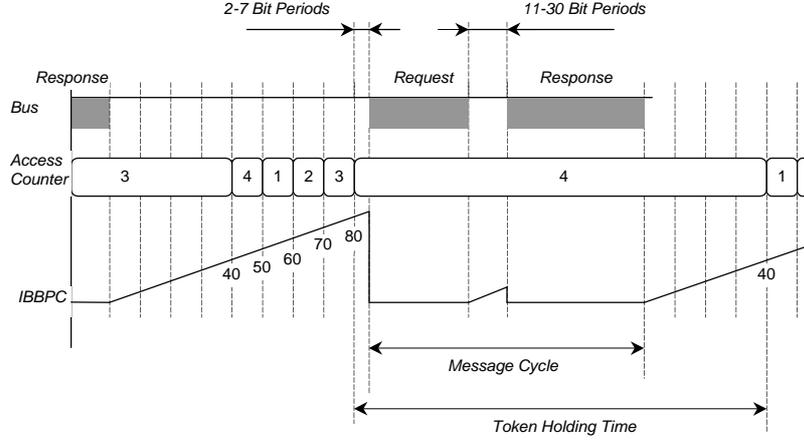


Fig. 2. An example of virtual token-passing in P-NET networks

3. A basic pre-run-time schedulability analysis

The following pre-run-time schedulability analysis provides sufficient conditions to guarantee the timing requirements of P-NET messages. Some of the results presented in this section were introduced in (Tovar and Vasques, 1998a) and (Tovar and Vasques, 1998b).

3.1. Network and message models

Consider a network with n masters, with addresses ranging from 1 to n . Each master accesses the network according to the VTP scheme; hence, first master 1, then masters 2, 3, ... until master n , and then again masters 1, 2, Slaves will have network addresses higher than n .

The following message stream model is assumed:

$$S_i^k = (C_i^k, D_i^k) \quad (1)$$

S_i^k defines a message stream i in master k ($k = 1, \dots, n$). A message stream is a temporal sequence of message cycles concerning, for instance, the remote reading of a specific process variable. C_i^k is the longest message cycle duration of stream S_i^k . D_i^k is the relative deadline of a message of the stream S_i^k , that is, the maximum admissible end-to-end communication delay for that message cycle.

3.2. Upper bound for the virtual token cycle

In P-NET each master is allowed to perform, at most, one message cycle per token visit. Therefore, the evaluation of the P-NET token cycle time (maximum time between two consecutive token arrivals at a master) is of paramount importance for the estimation of the end-to-end communication delay. It is obvious that an upper bound for the P-NET token cycle time (denoted as V) equals the sum of the maximum token-holding time in every master, and its value is:

$$V = \sum_{i=1}^n \left(7 \times bp + \max_{j=1, \dots, ns^i} (C_j^i) + 40 \times bp \right) \quad (2)$$

where $7 \times bp$ represents the longest master's reaction time and $40 \times bp$ represents the token-passing time after performing a message cycle. The longest message cycle in a master i , $\max_{j=1, \dots, ns^i} (C_j^i)$, includes the time needed to transmit both the request and response frames, the slave's turnaround time (which in P-NET is bounded to $30 \times bp$) and propagation delays.

3.3. A basic pre-run-time schedulability condition

In P-NET, the message requests are queued in a first-come-first-served (FCFS) queue. The maximum number of pending requests is ns^k , since if there were two pending message requests from the same S_i^k , this would mean that a deadline had been missed. The peak-load condition occurs when ns^k requests are simultaneously made just after the token passes to the next master. In this case, the transmission of the last of those ns^k requests is deferred by ns^k visits of the token. If the last message is the one with the most stringent deadline, a “priority inversion” with the length $(ns^k-1) \times V$ occurs.

If R_i^k denotes the upper bound for the message response time of stream S_i^k , this upper bound is given by:

$$R_i^k = ns^k \times V + 7 \times bp + C_i^k + I_i^k \quad (3)$$

where I_i^k aggregates both the upper bounds for the generation and the delivery delays for S_i^k .

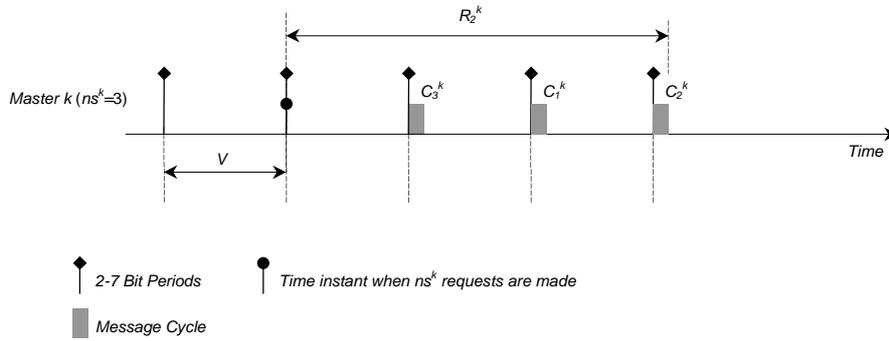


Fig. 3. An example of a message response time evaluation

It is now possible to introduce the pre-run-time schedulability condition, which is a sufficient condition to guarantee that real-time messages are transmitted before their deadlines. In P-NET, a set of message streams is schedulable if, at each master k :

$$D_i^k \geq R_i^k, \quad \forall_{i \in \{1, \dots, ns^k\}} \quad (4)$$

4. Schedulability analysis for multi-hop P-NET networks

Hopping P-NET devices (labelled as gateways in the standard, but termed as hopping devices in this paper) allow the interconnection of different network segments, each one with independent logical virtual token-passing schemes. In P-NET, the function of a gateway is to isolate two or more bus segments, and to automatically route a frame between the connected buses. In the authors' opinion, and according to ISO-OSI definitions, the P-NET gateways combine techniques used in bridges and routers, and thus the term “hopping devices” is preferred.

The P-NET multi-segment feature allows for routing through up to ten hopping devices. These multi-hopping capabilities are based on simple rules for address conversion inside the hopping devices.

P-NET supports four types of addresses: simple, complex, extended and response address types. The simple and response address types use only 2 bytes (destination and source addresses). The extended address uses 4 bytes (2 destination and 2 source address bytes) and the complex address may use up to 24 bytes. P-NET uses the complex addressing scheme to route frames through hopping devices. This complex address explicitly addresses each intermediate device.

In P-NET, hopping devices isolate traffic between P-NET segments. If the different segments group inter-related masters and slaves, the overall real-time capabilities are improved, as the virtual token cycle time in each single segment becomes smaller. However, if a particular stream relates a master and a slave in two distinct segments, that stream will have a higher response time. In this section an analysis for deriving the upper bound of the response time for messages that are to be relayed through a number of P-NET hopping devices is provided.

4.1. Motivation

Suppose a P-NET network composed of four masters (M1, M2, M3, M4) and four slaves (e1, e2, e3, e4), all connected to the same network segment. Each one of the masters deals with two message streams, as shown in Fig. 4.

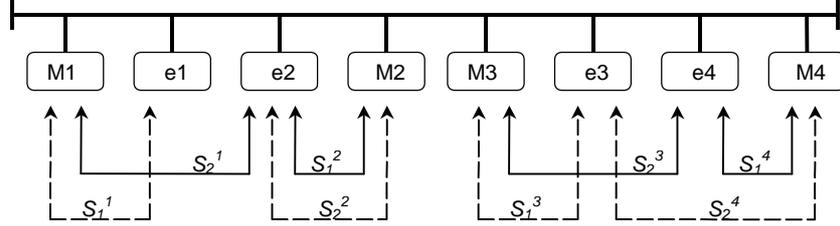
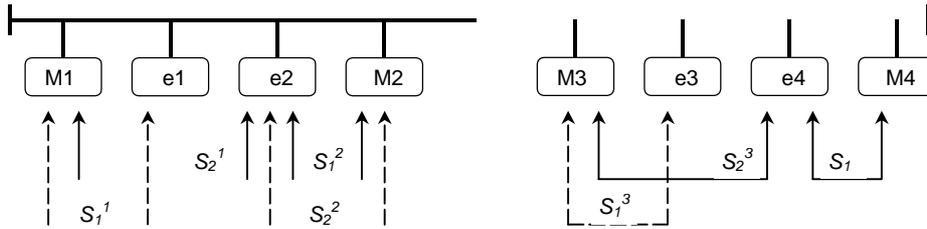


Fig. 4. A P-NET network example

Considering that the maximum token holding time in each station is $250 \times bp$ (this means that the longest message cycles of all the masters are equal: $C_i^k = 203 \times bp, \forall i,k$), then the pre-run-time schedulability condition (4) is (ignoring I): $D \geq [2 \times (4 \times (250) + (7 + 203)) \times bp] = 2210 \times 1/76800 = 28.8\text{ms}$.

The example in Fig. 4 illustrates, on a reduced scale, the advantages of segmentation. In fact, the whole network could be composed of two segments, grouping M1, M2, e1 and e2 in one segment, and M3, M4, e3 and e4 in another segment (Fig. 5). For simplification, any of the existing masters (M1, M2, M3 or M4) is used to implement the multi-hopping features.



4. When M4a gains access to segment 2, and the message containing the required information is the first one in the outgoing queue, M4a sends a request without a response to M5b.
5. When M5a gains access to segment 1, and the message is the first one in the outgoing queue, M5a sends a request (containing the required information) without a response to M1.

So, in general, if h represents the number of intermediate hopping devices through which a message stream is to be relayed, there will be $2 \times h + 1$ queuing and access delays to be considered.

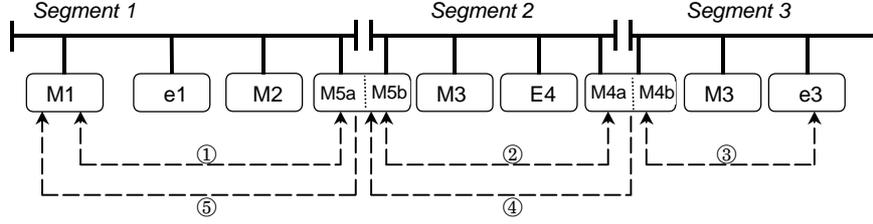


Fig. 6. An example of a multi-hop transaction

4.3. Pre-run-time schedulability condition for multi-hop message streams

Each P-NET segment has its own virtual token-rotation procedure. Thus, the maximum virtual token cycle time in a segment x can be defined as:

$$V_x = \sum_{i \in x} \left(7 \times bp + \max_{j=1..ns^i} (C_j^i) + 40 \times bp \right) \quad (5)$$

The upper bound for the response time of a message from stream S_i^k can be derived as follows. If the message stream is to be relayed through 0 hopping devices, then R_i^k is:

$$R_i^k = ns^k \times V(k) + 7 \times bp + C_i^k + I_i^k \quad (6)$$

where $V(k)$ corresponds to the upper bound for the virtual token rotation time of the network segment to which station k belongs.

If the message stream is to be relayed through 1 hopping device, then R_i^k is:

$$R_i^k = (ns^k + ns^{r_1}) \times V(k) + ns^{r_2} \times V(r_2) + \sum_{j=1}^3 (7 \times bp + C_j + I_j) + 2 \times f \quad (7)$$

where r_1 is the hopping master in the same segment as master k , and r_2 is the hopping master in the other segment. The summation represents the length of each of the $2 \times h + 1$ message transactions that will occur (using a simplified notation C_j) and the respective protocol overheads (using a simplified notation I_j). The symbol f stands for the time needed by the hopping device to transfer frames between communication stacks.

If the message stream is to be relayed through h hopping devices (with $h \geq 2$), then R_i^k is:

$$R_i^k = (ns^k + ns^{r_1}) \times V(k) + \sum_{j=1}^{h-1} \left[(ns^{r_{2 \times j}} + ns^{r_{2 \times j+1}}) \times V(r_{2 \times j}) \right] + ns^{r_{2 \times h}} \times V(r_{2 \times h}) + \sum_{j=1}^{2 \times h+1} (7 \times bp + C_j + I_j) + 2 \times h \times f \quad (8)$$

where r_1 is the hopping device master in the same segment as master k and $r_2, r_3, \dots, r_{2 \times h}$ are the hopping device masters which relay the message from master k to its destination. For example, for the scenario shown in Fig. 6, $r_1 = M5a$, $r_2 = M5b$, $r_3 = M4a$ and $r_4 = M4b$. As a consequence, in Eq. (8) r_i ($i=1, \dots, 2 \times h$) identifies the masters in hopping devices according to the physical sequence from master k to the addressed slave devices and not according to the sequence of transactions (see Section 4.3).

As for the non-segmented case (Eq. (3) or (6)), the same sufficient P-NET pre-run-time schedulability condition (inequality (4) - $D_i^k \geq R_i^k$) can be used to guarantee that real-time multi-hop message streams

are processed before their deadlines. Depending on the number of hopping devices a message is relayed through, Eq. (6), (7) or (8) is used to evaluate the upper bound of the message response time.

5. Numerical results

In the previous section, a sufficient pre-run-time schedulability condition for supporting real-time communications with P-NET has been provided. It was also demonstrated that the message response time can become much smaller if the network nodes are distributed by a number of P-NET network segments, each one grouping as many inter-related network nodes as possible. In this way tighter message deadlines can then be supported.

In this section, a numerical example is provided, which exemplifies what a user of P-NET can obtain from the proposed pre-run-time schedulability conditions. Although a limited number of message streams per node is assumed, some useful information can be obtained from this example:

1. how the maximum upper bound of each message-stream's response time, in both a non-segmented and segmented P-NET network can be evaluated;
2. how a P-NET network can be segmented in order to reduce the maximum upper bound for the message response time.

In this specific example, a comparison is also made between the response time's upper bound in a non-segmented and a segmented P-NET network, clearly showing the impact of network segmentation.

Assume that a distributed computer-controlled system must be implemented using eight master networks. Also assume that all message cycle lengths are bounded to $200 \times bp$ (2.6ms at 76800bps). The number of streams related to each master is shown in Table 1 (a total of twenty-eight message streams, distributed by eight masters).

Table 1. Number of message streams related to each master

Master	1	2	3	4	5	6	7	8
ns^k	3	4	3	2	1	4	5	6
$\max\{C_j^k\}$	$200bp$							

In this case, the upper bound for the virtual token cycle is $V=8 \times 247bp=25.7ms$. Therefore, using Eq. (3), the upper bound for the message response times is as shown in Table 2 (all streams in the same master will have the same upper bound for their response times). Note that the generation and delivery delays at the application process level are ignored, and must be evaluated at the level of the application process software. However, ignoring such delays is not of major importance, and as P-NET operates at 76800bps they will be usually much smaller than the transmission and queuing delays.

Table 2. Upper bound for the message response times

Master	1	2	3	4	5	6	7	8
R_i^k (ms)	79.8	105.5	79.8	54.1	28.4	105.2	131.2	156.9

If, for example, the application imposes deadlines smaller than 105.5ms for the message streams of master 2, or less than 28.4ms for the message streams of master 5, then the message stream set would not be schedulable.

Suppose that by re-organising the network into three network segments, as shown in Fig. 7, only two message streams are multi-hop streams: S_1^j and S_2^8 . Then, tighter deadlines can be supported for all but those two message streams.

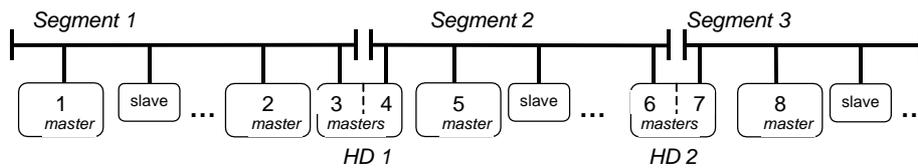


Fig. 7. Proposed segmentation of the network

Assume that streams S_1^1 and S_2^8 correspond to remote accesses, to slaves in segment 2 and segment 8, respectively. Table 3 illustrates the routing sequence for those streams.

Table 3. Routing sequence (master IDs) for the multi-hop streams

	h	r_1	r_2	r_3	r_4
S_1^1	1	3	4	-	-
S_2^8	2	7	6	4	3

These two streams will impose two additional message streams (resulting from messages being relayed through this hopping device) in masters 3 and 4, and one additional message stream in masters 6 and 7. Table 4 reflects the aggregate number of message streams per master station.

Table 4. Aggregate number of message streams related to each master

Master	1	2	3	4	5	6	7	8
ns^k	3	4	5	4	1	5	6	6

If, for simplification, f and l components from Eqs (7) and (8) are ignored, as well as the additional byte addresses in the multi-hop streams (thus maintaining $200 \times bp$ as the value for the longest message cycle in each master), then the results, shown in Table 5, are obtained.

Table 5. Upper bound for the virtual token Cycle in each segment (Eq. (5))

Segment	1	2	3
V (ms)	9.65	9.65	6.43

By implementing the proposed network segmentation, an important reduction of worst-case response times can be achieved, as illustrated in Tables 6 and 7.

Table 6. Upper bound for the response times (single-segment streams Eq. (6))

Master	1	2	3	4	5	6	7	8
R_i^k (ms)	31.65	41.3	50.95	41.3	12.35	50.95	41.28	41.28

Table 7. Percentage of the Upper Bound Obtained as Compared to Table 2

Master	1	2	3	4	5	6	7	8
R_i^k/R_i^k	39.6%	39.1%	63.8%	76.3%	43.5%	48.4%	31.4%	36.3%

Obviously, for the multi-hop message streams S_1^1 and S_2^8 the upper bounds for their response times increase as compared to the figures given in Table 2 (79.8 ms and 156.9 ms respectively).

The values are as follows. For R_1^1 Eq. (7) is used and Table 4 gives the number of streams for the masters. Its value is: $R_1^1 = (ns^1 + ns^3) \times V(1) + ns^4 \times V(4) + 3 \times 207bp = (3+5) \times 9.65 + 4 \times 9.65 + 8.08 = 123.88ms$.

For R_2^8 Eq. (8) is used and Table 4 gives the number of streams for the masters. Its value is: $R_2^8 = (ns^8 + ns^7) \times V(8) + (ns^6 + ns^4) \times V(6) + ns^3 \times V(3) + 5 \times 207bp = 12 \times 6.43 + 9 \times 9.65 + 5 \times 9.65 + 13.47 = 225.73ms$.

The upper bound for the response time of message stream S_1^1 (for which $h=1$) becomes 155% higher (related to the value given by Table 2), whereas message stream S_2^8 (for which $h=2$) becomes 144% higher.

The impact of network segmentation on the response time of both single-segment and multi-hop message streams is therefore clear.

6. Conclusions

In this paper, schedulability conditions for supporting real-time distributed computer-controlled systems with P-NET networks are provided. Both non-segmented and segmented P-NET networks are analysed.

This paper shows the advantages of using P-NET hopping devices for supporting network segmentation, since a significant reduction on response times can be achieved in most message transactions. However, the system designer must clearly understand that such reductions are not possible

for inter-segment message transactions. Therefore, care should be taken to group masters and slaves involved in message transactions with stringent deadlines in the same network segment.

The proposed schedulability analysis provides sufficient conditions to guarantee message timing requirements. As these schedulability conditions are derived under peak-load assumptions, they present a certain level of pessimism. Therefore, an investigation of this pessimism is provided in the Appendix. This may allow system engineers to design P-NET network-based DCCS, within which some message transactions have softer real-time requirements.

Acknowledgements

The authors acknowledge the useful comments made the anonymous reviewers. This work was partially supported by ISEP (project REMETER), by FLAD (project SISTER 471/97) and by DEMEGI/FEUP.

References

- Afnor (1990). Normes FIP NF C46-601 to NF C46-607. Union Technique de l'Electricité.
- Cardoso, A., & Tovar, E. (1996). Industrial communication networks: issues on heterogeneity and internetworking. *Proceedings of the 6th International Conference on Flexible Automation and Intelligent Manufacturing (FAIM'96)*, Atlanta, USA, 139-148.
- Cenelec (1996). General Purpose Field Communication System. EN 50170, Vol. 1/3 (P-NET), Vol. 2/3 (PROFIBUS), Vol. 3/3 (FIP), Cenelec.
- Jenkins, C. (1997). P-NET as a European Fieldbus Standard EN 50170 Vol. 1. *Institute of Measurement and Control Journal*, Vol. 30, April, pp. 75-79.
- Lenhart, G. (1993). A fieldbus approach to local control networks. *Advances in Instrumentation and Control*, Vol. 48, No. 1, pp. 357-365.
- Pnet (1994). The P-NET Standard. International P-NET User Organisation ApS.
- Profibus (1992). PROFIBUS Standard DIN 19245 part I and II. Translated from German, Profibus Nutzerorganisation e.V.
- Tindell, K., Burns, A., & Wellings, A. (1995). Analysis of hard real-time communications. *Journal of Real-Time Systems*, 9, pp. 147-171.
- Tovar, E., & Vasques, F. (1998a). Scheduling real-time communications with P-NET. *Digest of the IEE Real-Time Systems Colloquium*, York, England, Digest No. 98/306, pp. 9/1-9/5.
- Tovar, E., & Vasques, F. (1998b). Pre-run-time schedulability analysis of P-NET fieldbus networks. *Proceedings of the 24th Annual Conference of IEEE Industrial Electronics Society (IECON'98)*, Aachen, Germany, Vol. 1/4, pp. 236-241.

Appendix

The pre-run-time schedulability conditions derived in this paper are sufficient but not necessary. This is a result of trying to develop an approach to guaranteed timeliness based on the worst-case peak-load conditions. It is therefore pertinent to present an evaluation of the pessimism that exists in the pre-run-time schedulability conditions. Revisiting Eq. (3), used in the case of non-segmented networks, the following sources of pessimism are present:

- (1) In the priority inversion factor ns^k :
 - (a) it is a worst-case scenario that all messages would be requested to be transferred in the period between two token arrivals;
 - (b) it is a worst-case scenario that, simultaneously with condition 1(a), the specific message will be the last in a queue with a length of ns^k ;
 - (c) it is a worst-case scenario that, simultaneously with conditions 1(a) and 1(b), multiple request for message streams will occur just after a token visit, thus imposing an additional token rotation in the overall access time;
- (2) In the virtual token cycle factor (V):
 - (a) it is a worst-case scenario all master stations will use the token for sending a message;
 - (b) it is a worst-case scenario that, simultaneously with condition 2(a), all stations will transfer their longest message cycle;
 - (c) it is a worst-case scenario that the time $7 \times bp$ is the master's reaction time;
 - (d) and finally, it is a worst-case scenario that the time $30 \times bp$ is the slave's turnaround time.

Assume a scenario of a P-NET network with ten masters, where each master supports ten message streams. If all masters have a maximum message cycle length of $100 \times bp$ (request and response frame lengths, but, for this analysis without the slave's turnaround time), then the worst-case response time for a message cycle in that station is (again ignoring I): $V = \sum_{i=1, \dots, 10} (7 \times bp + 130 \times bp + 40 \times bp) = 17700 \times bp = 23.1 \text{ms}$, with a slave turnaround time equal to $30 \times bp$. Therefore, the upper bound for the response time of any message request in any master in the network will be: $R = 10 \times V + 137 \times bp = 232.9 \text{ms}$.

Considering Eq. (3), $R_i^k = ns^k \times \sum_{j=1, \dots, n} (7 \times bp + C_{max} + 40 \times bp) + 7 \times bp + C_i^k$, the following equation

$$R_i^k = A \times ns^k \times [(n - D) \times (B \times C_{max}^* + 40 \times bp + C \times 37 \times bp) + D \times 10 \times bp] + C_{max}^* + C \times 37 \quad (9)$$

can be used to access the impact of each of the factors identified as contributors to the overall pessimism level.

A represents the aggregate probability of the simultaneous occurrence of situations 1(a), 1(b) and 1(c). B represents the probability of a reduction in the average length of messages cycles in a complete token rotation. C represents the probability of a reduction in the average time for both the master's reaction and the slave's turnaround times. Finally, D represents the average number of masters not using the token in a complete token rotation. The term $D \times 10 \times bp$ is included because it represents the token-passing time for masters not using the token (refer to Fig. 2 for clarification). C_{max}^* denotes the message length excluding the slave's turnaround time ($C_{max} = C_{max}^* + 30 \times bp$).

Assuming the above scenario (ten masters, ten message streams in each, etc.), the Eq. (9) can be re-written as:

$$R = \{A \times 10 \times [(10 - D) \times (B \times 100 + 40 + C \times 37) + D \times 10] + 100 + C \times 37\} / 76.8 \text{ (ms)} \quad (10)$$

Fig. 8 illustrates the impact of each parameter on the evaluation of the upper bound of the response time. When all parameter values are set to 0.6, the upper bound for the response time is 58.6 ms, which compared to 232.9 ms, gives about a reduction to 25%.

In the case of the response times of multi-hop message streams (Eqs (7) and (8)), the pessimism level is obviously higher, and increases as the number of intermediate hopping devices increases.

A simplified version of Eq. (8) can be used, as $ns^l = ns^2 = \dots = ns^{10}$ (again, parameters f and I are ignored):

$$R = (2 \times h + 1) \times ns^k \times (V + 7 \times bp + C') \quad (11)$$

and the following equation can be used to access the level of pessimism:

$$R = (2 \times h + 1) \times A \times ns^k \times (B \times V + 7 \times bp + C') \quad (12)$$

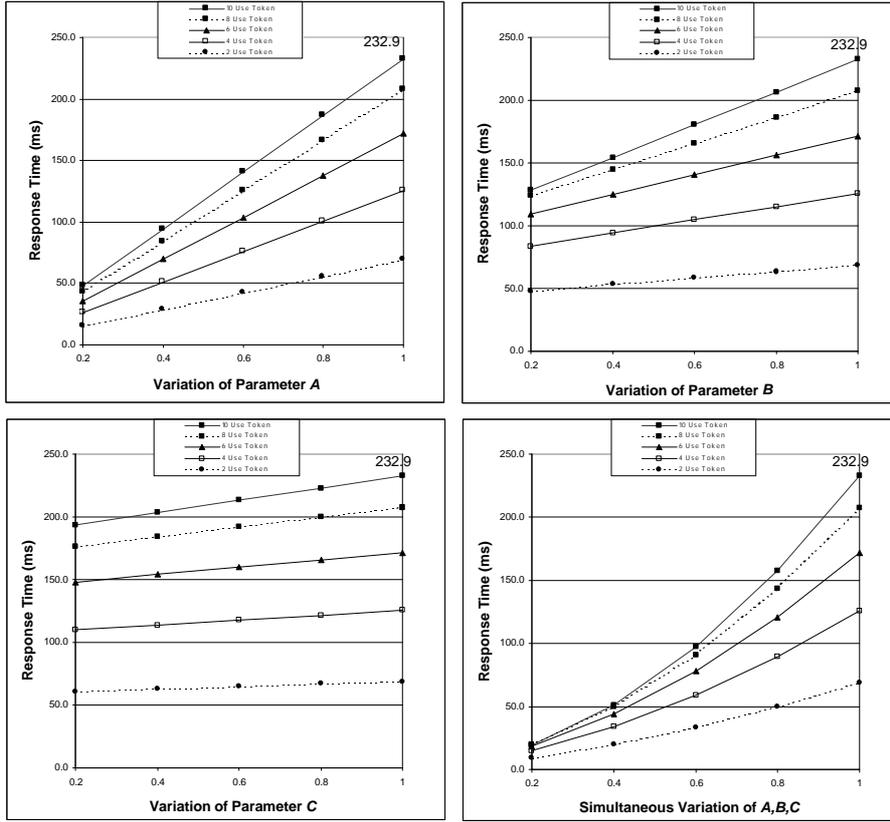


Fig. 8. Level of pessimism on the evaluation of the upper bound of the response time for single-segment message streams

In Eq. (12), h represents the number of hopping devices a message stream is relayed through. Parameter A is a reducing factor related to the number of streams in each outgoing queue. Parameter B is a reducing factor for the average token-rotation time.

Assuming that all segments have ten masters, each with ten message streams, and with $100 \times bp$ as the maximum length of the message cycles, then in each segment $V = 10 \times (7 \times bp + 100 \times bp + 40 \times bp) = 19.1$ ms. If C' is assumed to be, on average, $80 \times bp$, then $R = (2 \times h + 1) \times A \times 10 \times (B \times 19.1 + 1.1)$ ms.

Fig. 9 illustrates the level of pessimism as a function of the number of hopping devices a message stream is relayed through. It is easy to see, from the equation $R = (2 \times h + 1) \times A \times 10 \times (B \times 19.1 + 1.1)$, that the individual contributions of A and B are similar. Therefore, only trends for parameter A and aggregation of A and B are shown in the graphs.

