IPP-HURRAY! Research Group

Polytechnic Institute of Porto
School of Engineering (ISEP-IPP)

Timing Analysis of a Multiple Logical Ring Wired/Wireless PROFIBUS Network

Luís FERREIRA
Eduardo TOVAR

HURRAY-TR-0414 (**version 01**)
April-2004

relatório técnico

technical report

Timing Analysis of a Multiple Logical Ring Wired/Wireless PROFIBUS Network

Luís FERREIRA, Eduardo TOVAR

IPP-HURRAY! Research Group
Polytechnic Institute of Porto (ISEP-IPP)
Rua Dr. António Bernardino de Almeida, 431
4200-072 Porto
Portugal
Tel.: +351.22.8340502, Fax: +351.22.8340509
E-mail: {llf, emt}@dei.isep.ipp.pt
http://www.hurray.isep.ipp.pt

Abstract:
Recently, there have been a few research efforts towards extending the capabilities of fieldbus networks to encompass wireless support. In previous works we have proposed a hybrid wired/wireless PROFIBUS network solution where the interconnection between the heterogeneous communication media was accomplished through bridge-like interconnecting devices. The resulting networking architecture embraced a Multiple Logical Ring (MLR) approach, thus with multiple independent tokens, to which a specific bridging protocol extension, the Inter-Domain Protocol (IDP), was proposed. The IDP offers compatibility with standard PROFIBUS, and includes mechanisms to support inter-cell mobility of wireless nodes. In this paper we advance that work by proposing a worst-case response timing analysis of the IDP.

Timing Analysis of a Multiple Logical Ring Wired/Wireless PROFIBUS Network

Luís Ferreira, Eduardo Tovar
Polytechnic Institute of Porto (ISEP-IPP)
Rua Dr. António Bernardino de Almeida, 431
4200-072 Porto, Portugal
E-mail: {llf, emt}@dei.isep.ipp.pt

Abstract

Recently, there have been a few research efforts towards extending the capabilities of fieldbus networks to encompass wireless support. In previous works we have proposed a hybrid wired/wireless PROFIBUS network solution where the interconnection between the heterogeneous communication media was accomplished through bridge-like interconnecting devices. The resulting networking architecture embraced a Multiple Logical Ring (MLR) approach, thus with multiple independent tokens, to which a specific bridging protocol extension, the Inter-Domain Protocol (IDP), was proposed. The IDP offers compatibility with standard PROFIBUS, and includes mechanisms to support inter-cell mobility of wireless nodes. In this paper we advance that work by proposing a worst-case response timing analysis of the IDP.

1. Introduction

There has been an enormous eagerness towards extending the capabilities of fieldbus networks to encompass wireless support. The RFieldbus European project [2] was just one example of that effort, where PROFIBUS was extended to support hybrid wired/wireless communication systems. In RFieldbus, repeaters were proposed used to provide interoperability between wired and wireless network components (network domains). In any case, in the RFieldbus approach there is only one Single Logical Ring (SLR), with only one token rotating between all the masters (wired or wireless) in the network. The main advantage of such SLR approach is that the effort for is not significant protocol extensions to PROFIBUS, since the adaptation is essentially at the physical layer level.

However, there are a number of advantages in using a Multiple Logical Ring (MLR) approach to support such type of hybrid systems. This concept was introduced and discussed in [3], and further detailed in [4-5], where a bridge-based approach (thus, layer 2 interoperability) was outlined. Each logical ring is comprised of stations that communicate via a unique medium – a domain. Therefore, a wired domain corresponds to the set of (wired) stations that intercommunicates via a wired segment. Correspondingly, a wireless domain is a set of (wireless) stations intercommunicating via the air. Those works describe the functionalities provided by the Inter-Domain Protocol (IDP), which supports the communication between stations in different domains. They also describe the mechanisms and functionalities that allow the mobility of station between different wireless cells. These protocol extensions can be supported with mostly no impact to the original PROFIBUS protocol, and thus those extensions provide compatibility with legacy PROFIBUS technologies.

In this paper, we will specifically tackle the problem of the timing analysis of the proposed inter-domain transactions supported the bridge-associated IDP. This work builds on previous relevant works on PROFIBUS real-time assessment.

The research works on the timing behaviour of PROFIBUS networks [8-12] have proved the capabilities of this protocol to support distributed computer-controlled systems with stringent real-time requirements. Obviously, the wireless extensions and the bridge-based architecture of the network require the development of a new timing analysis, which accounts for the IDP and the effects of the mobility procedure. In this particular paper we will propose solutions to this problem, albeit, and because of space limitations, without considering the influence of the previously proposed inter-cell mobility protocols on the Worst-Case Response Time (WCRT) of the various types of message transactions.

The reminder of this paper is organised as follows. In Section 2, the main concepts related to bridge-based hybrid wired/wireless PROFIBUS architecture, and particularly IDP, are briefly presented. Then, in Section 3, we present a worst-case timing analysis of the IDP-related transactions, to which, in Section 4, a numerical example of this timing analysis is given. Based on the results presented in Section 4, in Section 5 we discuss the results and provide some hints towards refining the worst-case analytical formulations aiming at reducing the level of pessimism that typically exists in such type of formulations in the context of distributed systems. In Section 6, we draw some conclusions.

2. System architecture

The hybrid wired/wireless fieldbus network is composed of stations, with a wireless interface (radio), that are able to interoperate with wired (legacy) stations. The communications between the different stations is supported by the PROFIBUS protocol with specific extensions to support wireless communications and mobility.

In this Section we describe, to the extent required for reasoning the rest of the paper, the main characteristics of PROFIBUS and the MLR hybrid wired/wireless architecture.

2.1. Basics of the PROFIBUS protocol

The PROFIBUS Medium Access Control (MAC) protocol uses a token passing procedure to grant bus access to masters, and a master-slave procedure used by masters to communicate with slaves. A PROFIBUS master is capable of dispatching transactions during its token holding time (T_{TH}), which, for each token visit, is the value corresponding to the difference, if positive, between the target token rotation time (T_{TR}) parameter and the real token rotation time (T_{RR}). For further details, the reader is referred to [9].

A master station that sends an *Action Frame* (the first frame transmitted in a transaction) is said to be the initiator of the transaction, whereas the addressed one is the responder (a master or a slave). A transaction (or message cycle) consists on the request send/request frame from the initiator and of the associated acknowledgement or response frame of the responder. The acknowledgement (or the response) must arrive before the expiration of the *Slot Time* (T_{SL}), otherwise the initiator repeats the request a number of times defined by the *max_retry_limit*, a DLL parameter.

In order to maintain the logical ring, PROFIBUS provides a decentralised ring maintenance mechanism. Each PROFIBUS master maintains two tables – the *Gap List* (GAPL) and the *List of Active Stations* (LAS), and may optionally maintain a *Live List* (LL).

The *Gap List* consists of the address range from *TS* ('This Station' address) until *NS* ('Next Station' address, i.e., the next master in the logical ring). Each master station in the logical ring starts checking the addresses in its GAPL every time its *Gap Update Timer* (T_{GUD}) expires. This mechanism allows masters to track changes in the logical ring, such as addition (joining) or removal (leaving) of stations. This is accomplished by examining (at most) one *Gap address* per token visit, using the *FDL_Request_Status* (FDL stands for Fieldbus Data Link Layer) frame.

The LAS is a list of all the masters in the logical ring, and the LL contains all active stations (both masters and slaves).

2.2. Basics on system components

The wireless part of the fieldbus network is supposed to include at least one *radio cell*. Basically, a radio cell can be described as a 3D-space where all associated wireless stations are able to communicate with each other. In the example of Figure 1 the following set of wired PROFIBUS master (M) and slave (S) stations are considered: M10, M7, S22, S24, S25 and S27. Additionally, the following set of wireless stations is considered: M6, M1, S21 and S23. Within this set, only M6 and S23 are mobile. Three bridge devices are considered: B1, B2 and B3. Each bridge is constituted by two modified PROFIBUS masters, which implement the required protocol extensions. Each one of these masters is called a Bridge Master (BM).

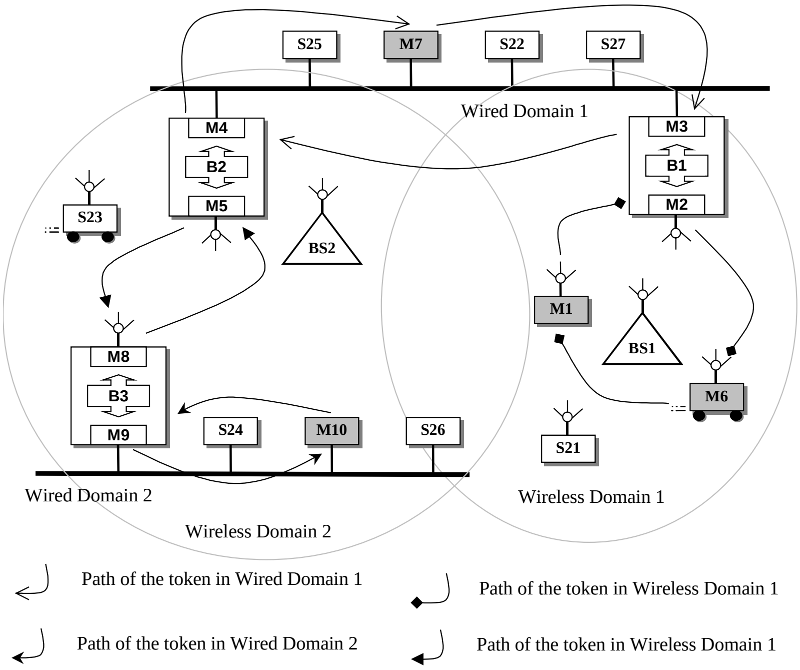

Figure 1. Wireless PROFIBUS example network

We are also assuming that the network has a tree-like topology, and that bridges perform routing based on MAC addresses.

In such a system, all communications are relayed through base stations: BS1 and BS2. We will assume, in the remainder of the paper, that M5 and M2 include the base station functionalities in their wireless front-end [1], thus, structuring wireless domains 1 and 2, respectively.

Network operation is based on the Domain-Driven Multiple Logical Ring (MLR) approach, described in [3]. Therefore, each wired/wireless domain has its own logical ring. In this example, four different logical rings exist: {(M1 → M2 → M6), (M3 → M4 → M7), (M5 → M8), (M9 → M10)}.

2.3. Basics on the Inter-Domain Protocol (IDP)

In PROFIBUS, a message cycle is composed by a request and related response. The interval between these two actions must be smaller than the value set for the slot time parameter. The use of intermediate bridges demands refining further this message cycle concept (for inter-domain message transaction), since unavoidably the response will not arrive immediately after the request.

An Inter-Domain Transaction (IDT) is a transaction that involves an initiator and a responder belonging to different domains, i.e. with one or more bridges in the communication path. An IDT is composed of a request (Inter-Domain Request), and its respective response (Inter-Domain Response).

We denote the first bridge master in the path from the initiator to the responder as BM_i (*i* standing for initiator). Similarly, the last bridge master in the path is denoted as BM_r (*r* standing for responder).

This solution specifies that when an initiator makes an Inter-Domain Request, only one of the BMs belonging to the domain - bridge BM_i, codes the frame using the IDP, and relays it. The remaining stations discard the frame.

Then this frame – the Inter-Domain Frame (IDF) is relayed by the bridges until reaching bridge master BM_r. This bridge decodes the original request frame and transmits it to the responder, which can be a standard PROFIBUS station (for example a PROFIBUS-DP slave). The response is again coded using the IDP and routed back until reaching bridge master BM_i, where it will be decoded and stored.

As the actual response to the original request takes more time than if the responder belongs to the same domain as the initiator, the initiator must periodically repeat the same request until receiving the related response. This means, in practice, that the request is not immediately responded. After BM_i having received (and stored) the correspondent response frame, then it is ready to respond to a new (repeated) request from the initiator. One of the objectives of this mechanism is to provide complete transparency from the point of view of both the initiator and the responder (to cope with PROFIBUS compatibility requirements). This is achieved by BM_i emulating the responder in a way that the initiator station considers the responder station as belonging to its domain, and by BM_r emulating the initiator in a way that the responder considers the initiator as belonging to its domain.

Considering the network scenario illustrated in Figure 1, Figure 2 represents a simplified timeline regarding a transaction between master M3 and slave S23.

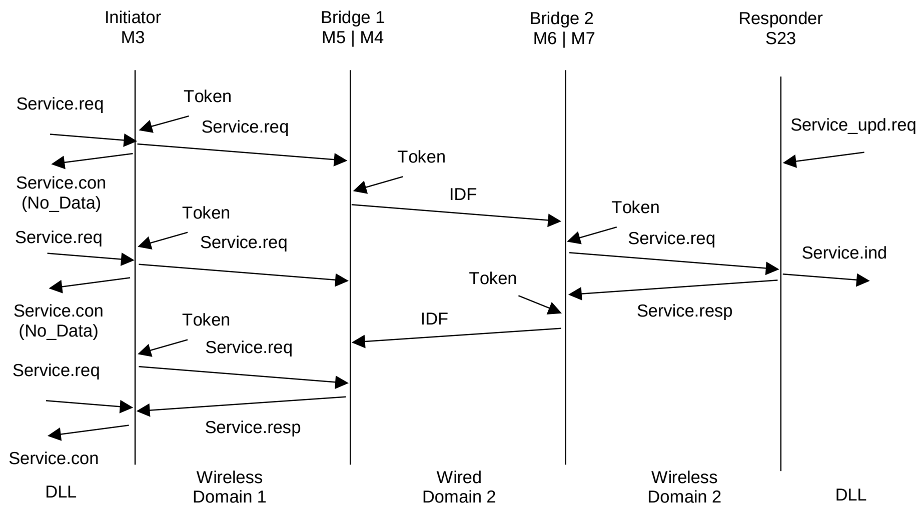

Figure 2. Example for an Inter-Domain Transaction (IDT)

Regarding Figure 2 and the operation of the IDP, we assume that slaves read their inputs periodically, placing their image in the Data Link Layer (DLL), using the PROFIBUS *Service_upd.req* primitive. The image of the input values is placed in a buffer, which is used by the DLL to build a response to a specific request. An indication is returned to the higher layers every time a slave receives a request. This behaviour is implemented by the PROFIBUS-DP protocol. On the initiator side, it is also necessary that the user of the DLL periodically repeats the same request. For every request, the DLL returns a confirmation, which can include "no data" if the response data is not available yet.

2.4. Mobility Mechanisms

The inter-cell mobility procedure proposed in [4] is hierarchically managed. One master in the system implements the global mobility management functionality, which is responsible for periodically starting the handoff procedure and controlling some of its phases – the *Global Mobility Manager* (GMM). In each domain, one master controls the mobility of stations belonging to that domain – the *Domain Mobility Manager* (DMM). Finally, the bridge stations implement specific mobility services.

The mobility procedure evolves through several phases. First the BMs must finish all ongoing IDT (for which they are responsible). After, the wireless DMMs can transmit beacon messages, which are used by the mobile wireless stations to assess the quality of the radio channels. Finally, the mobile wireless stations must enter into a new domain and the routing tables of the bridge must be updated. For further details the reader is referred to [4], where the mobility procedure is described in detail.

As previously said, the mobility procedure can have a significant impact on the worst-case response time for IDTs, and even for transactions between nodes in the same domain. However, and because of space limitations, in this particular paper we will be addressing the timing analysis not considering mobile nodes, and thus not considering the case of systems implementing the inter-cell mobility mechanisms.

3. WCRT analysis for IDT transactions

In this Section we start by describing the related work on PROFIBUS timing analysis. Then we present the proposed analysis for the IDP. Two types of Inter-Domain Transactions (IDT) are addressed: when a transaction involves a request and a response; and when a transaction only involves a request.

3.1. Related work on PROFIBUS timing analysis

The first work which addressed the schedulability of a PROFIBUS network [11], provided a just a sufficient schedulability test without presenting any estimation of the worst-case response time for PROFIBUS message streams. In [9] the authors improve that work by suggesting two different approaches to guarantee the real-time behaviour of the synchronous traffic in a PROFIBUS network, and propose a methodology for the calculation of the worst-case response time for a message transaction. Nevertheless, that methodology can lead to pessimistic results, and thus, in [12], the authors present a even more refined approach. Both approaches presented in [9] and in [12] will be used as a basis for the timing analysis of our MLR PROFIBUS networks.

The *Unconstrained Low-Priority Traffic Profile*, one of the analysable traffic profiles proposed in [9], will be described next, since it is a well know and accepted methodology for the calculation of the worst-case response time (WCRT) of a PROFIBUS message stream.

With this approach, the real-time requirements for the synchronous (high priority) PROFIBUS traffic are satisfied, even when only one synchronous message is transmitted per token visit, and independently of the asynchronous (low priority) PROFIBUS traffic load. In this way, it is possible to have a guaranteed real-time approach for the message streams, provided that the relative deadline for the synchronous message streams is larger than the related worst-case message response time, which can be computed as follows:

$$Rslr_i^k = Q^k + Ch_i^k = nh^k \times T_{cycle}^k + Ch_i^k \tag{1}$$

where Q^k is the queuing delay, nh^k is the number of synchronous high-priority message streams generated in master k, T^k_{cycle} is the worst-case token rotation time and Ch_i^k is the worst-case duration of a synchronous message cycle i issued by master k.

The exact characterisation of the token cycle time properties of PROFIBUS is described in [9], which permits the evaluation the T^k_{cycle} parameter. An upper bound on the token cycle time can be given by:

$$T_{cycle}^k = T_{TR} + T_{del}^k = T_{TR} + n \times C_\sigma \tag{2}$$

where T_{TR} is the PROFIBUS target token rotation time, n is the number of masters, and C_σ is the duration of the longest message cycle in the network. The evaluation of maximum token lateness T^k_{del} can be more precise and less pessimistic if the duration of the longest message cycles relative to a master is considered. In [9] the authors specifically address this aspect.

3.2. Proposal for WCRT analysis of IDT

One of the characteristics of the IDP is that the initiator periodically repeats a request until receiving a response. Consequently, the WCRT for a message stream in such conditions mainly depends on the stream period. Figure 3, depicts a scenario where that situation is obvious.

Consider that A_i^k is the maximum number of attempts performed by the initiator (master k) until receiving a valid response from bridge BM_i, related to stream i. T_i^k is the period of stream i from master k. Therefore, the last request, the one that obtains the response, requires a WCRT of $Rslr_i^k$, which can be calculated using the worst-case response time analysis presented in Section 3.1. That corresponds to the Single Logical Ring (SLR) case.

Then, the WCRT for a stream i from master k, on a MLR network ($Rmlr_i^k$), can be computed using the following formulation:

$$Rmlr_i^k = A_i^k \times T_i^k + Rslr_i^k \qquad \textbf{(3)}$$

As an example, in the scenario of Figure 3, the response time for the represented stream, is equal to: 2x T_i^k+ $Rslr_i^k$.

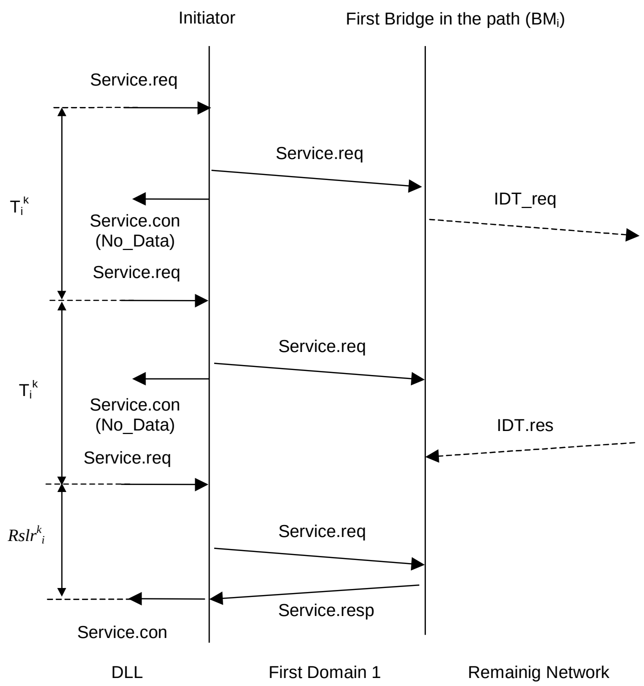

Figure 3. IDT timings

The maximum number of attempts (A_i^k) depends on the delay experienced by the IDT, from the reception of the request at BM_i, until the arrival of the respective response ($Rbmi_i^k$). To obtain A_i^k we must consider the worst-case situation on the side of BM_i, i.e. when the minimum amount of time, between the first request and the last request, is available at BM_i to obtain the response. Figure 4, depicts an example.

This situation occurs when the first request, the one that initiates the IDT, arrives at BM_i delayed by its worst-case ($Rslr_i^k$), and the last request, the one that obtains the response arrives at BM_i delayed by the best case (C_i^k). In this situation $Rslr_i^k + Rbmi_i^k < A_i^k \times T_i^k + C_i^k$ T. So, since A_i^k must be an integer its value can be obtained as follows:

$$A_i^k = \left\lceil \frac{Rslr_i^k + Rbmi_i^k - C_i^k}{T_i^k} \right\rceil \qquad \textbf{(4)}$$

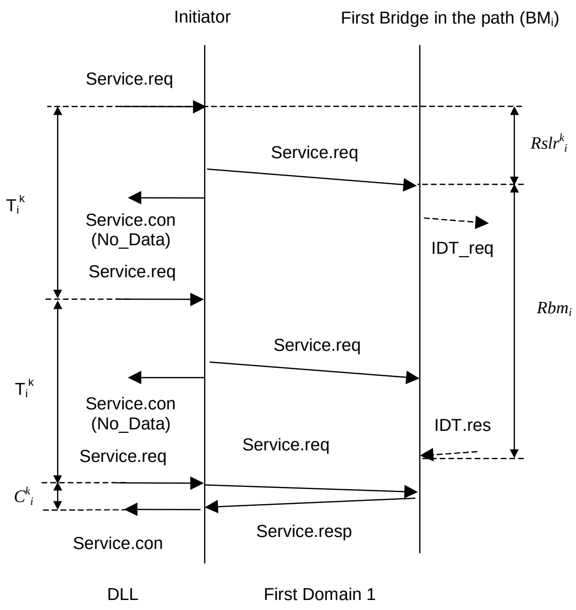

Figure 4. IDT timings

For the calculation of $Rbmi_i^k$, analysis can be adapted from the P-NET networks case [14] and from [3] (which describes a similar MLR architecture).

Consider, for example, again the network scenario depicted in Figure 1, and a message stream S^7_1 between master M7 and slave S24. In this case, BM_i is the bridge master M4, and BM_r is bridge master M9. To obtain the WCRT for an IDT transaction it is necessary to account for all the delays experienced by the IDT on the BMs, which depends on the number of streams processed by them and on the traffic conditions on the respective domains.

Thus, using the WCRT analysis for a SLR PROFIBUS network, the following equation allows the calculation of *Rbmi* for the example outlined:

$$Rbmi_i^1 = nh^5 \times T_{cycle}^{wl2} + (Creq_1^7)^{wl2} + \phi + nh^9 \times T_{cycle}^{wr2} + (Ch_1^7)^{wr2} + \phi + nh^8 \times T_{cycle}^{wl2} + (Cresp_1^7)^{wl2}$$

where nh^x is the number of high priority message streams processed by a BM x. T^d_{cycle} is the token cycle time in network domain d. $(C_{req}{}^1{}_7)^d$ is the duration of the request frame in a domain d. The symbol ϕ represents the time needed by the Bridge process a frame and forward it to the other BM. $nh^5 \times T^{wl2}_{cycle}$, $nh^9 \times T^{wr2}_{cycle}$ and $nh^8 \times T^{wl2}_{cycle}$ are, respectively, the queuing delays on BMs M5, M9 for the inter-domain request frame, and on M8 for the inter-domain response frame. $(Ch_i{}^7)^{wr2}$ is the transaction duration time between M9 and S24. $(C_{req}{}^1{}_7)^d$ and $(C_{resp}{}^1{}_7)^d$ are, respectively, the latency of the request frame in a domain d, and the latency of the response frame in a domain d.

To obtain a general formulation, we will refer to the BMs in the path, from the initiator to the responder as r_1, r_2 until and r_{2b}. Where r_1, r_2 refer to the BMs of the first bridge in the path, r_{2b-1} and r_{2b} refer the BMs of the last bridge in the path. b is the number of bridges between the initiator and the responder. Note that BM r_{2b} is attached to the same domain where the responder is located, thus it will execute a complete transaction (including a request and a response). The network domains are numbered on the same order, being the first domain of the path, domain number 1 and the last domain numbered as $b+1$. Then $Rbmi_i^k$ can be obtained as follows:

$$\begin{aligned} Rbmi_i^k &= \sum_{f=1}^{b-1} \left(nh^{r_{2f}} \times T_{cycle}^{r_{2f}} + (Creq_i^k)^{f+1}\right) + b \times \phi \\ &+ nh^{r_{2b}} \times T_{cycle}^{r_{2b}} + (Ch_i^k)^{b+1} + \\ &+ b \times \phi + \sum_{f=2}^{b} \left(nh^{r_{2f-1}} \times T_{cycle}^{r_{2f-1}} + (Cresp_i^k)^f\right) \end{aligned} \quad \textbf{(5)}$$

In this equation, $(Creq_i^k)^s$, $(Cresp_i^k)^s$, $(Ch_i^k)^s$, are, respectively, the time needed to transmit the request, the response and for completing a message cycle on network domain s. ϕ is the latency of the bridges.

It is also possible to rewrite equation (5), in a more compact format as follows:

$$Rbmi_i^k = \sum_{f=2}^{2b} Rslr_i^{r_f} + 2 \times b \times \phi \quad \textbf{(6)}$$

where $Rslr^{bm}_i$ is the response time on a single logical ring domain for a bridge master bm, which can be calculated by any one of the SLR WCRT analysis proposed for PROFIBUS.

It is important to note that IDP defines different frame formats for the frames exchanged between bridges [5], and this aspect must be taken into account when calculating transaction latencies on each domain.

3.3. Unicast transaction

PROFIBUS defines that a transaction can be made in unicast or broadcast mode, which is the case of the SDN service provided by the DLL. The IDP is capable of supporting such kind of communications, thus it is also necessary to provide a worst-case bound for such type of transactions.

The worst-case time required by a request, from a message stream i, to go from a master k to another station w ($Ru^{k \to w}_{i}$), can be obtained by adapting equation (5) as follows:

$$Ru_i^{k \to w} = Rslr_i^k + \sum_{f=1}^{2b} Rslr_i^{r_{2f}} + b \times \phi \qquad \textbf{(7)}$$

where the first part of the equation represents the latency on master k, and the other factors represent the latency associated with the forwarding by the bridges.

If station k is a BM, then the last equation might not be valid. As an example, consider the network depicted in Figure 5, and the following two transactions: a transaction 1 that involves stations M2 and S21 and a transaction 2 that involves stations M3 and S23. For transaction 1, the first leg of the path is between M2 and M3, with a delay equal to ϕ, thus the first station in the path, which transmits the message through the network, is M3. For transaction 2, the message is transmitted directly into the network segment by M3. This situation is particularly important for the timing analysis of the mobility procedure (not addressed in this paper), since most of the messages related to the mobility procedure are transmitted in unicast or broadcast modes.

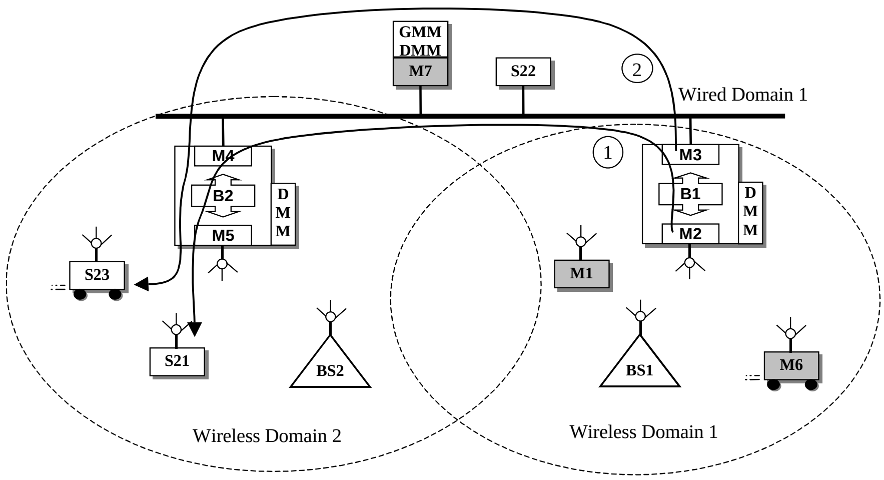

Figure 5. Bridge master cases

It is also possible that the destination station is a BM, e.g. M5. In that situation the message would be received by M4 and passed to M5.

Thus, in both cases equation (7) requires some adaptations. If the transaction is similar to transaction 2, then the first station in the path is the initiator itself. If the transaction is similar to transaction 1, then the first station in the path is the BM on other side of the bridge, and the message is delayed by ϕ. If the responder is a BM not directly connected to the last domain where the message is transmitted then, the message is also delayed by ϕ before being received by the destination BM. Thus, equation (7) can be rewritten as follows, to better reflect this actual behaviour:

$$Ru_i^{k \to w} = \sum_{f=2}^{2b} Rslr^{r_{2f}} + (b + di + df) \times \phi \quad \textbf{(8)}$$

where di is equal to zero if the initiator is a master station or if the initiator is a BM directly connected to the first domain in the path for message stream i. di is equal to one if the initiator is a BM not directly connected to the first domain in the path. df is equal to zero if the destination station is a master, a slave or a BM directly connected to the last domain in the path. df is equal to one if the destination station is a BM not directly connected to the last domain where the message is transmitted.

4. Numerical results

In this Section we will illustrate how the timing analysis presented in the previous section can be applied. The numerical results obtained will then be compared with the results obtained by the simulation models.

4.1. Applying the WCRT analysis

For the remainder of this section consider the network depicted in Figure 1, and the related network parameters as listed on Table 1. This table is based on typical PROFIBUS parameters, and assumes a wireless infrastructure similar to the RFieldbus case, where the physical layer of the wireless part of the network is based on the 802.11b protocol. Therefore, on the wireless part of the network the data rate is 2.0Mbit/s, and each frame must include a head and a tail for synchronisation purposes [13].

Table 1. Assumed network parameters for the numerical example

Parameter	Value	Notes
bit rate (Wired)	1.5 Mbit/s	-
bit rate (Wireless)	2.0 Mbit/s	-
bits per char (wired)	11	due to the start, stop and parity bits
bits per char (wireless)	8	-
T_{SDR}	60 bits	40μs (Wr) 30μs (Wl)
T_{ID}	65 bits	43.3μs (Wr) 33.5μs (Wl)
T_{TR}	300μs	-
Frame head length	32 bits	(only for wireless domains)
Frame tail length	16 bits	(only for wireless domains)
ϕ	30μs	bridge delay

Consider additionally that system masters have a set of high priority message streams as presented in Table 2. To calculate the latency associated with transmitting the requests and responses, the formulations outlined in [13] have been used. We are also assuming that all streams have a period of 8ms, and that all frames (request or response) have a size of 20 bytes.

Table 2. IDT streams for the numerical example

Stream	Source Address (SA)	Destination Address (DA)
S^1_1	1	22
S^1_2	1	24
S^1_3	1	27
S^1_4	1	25
S^6_1	6	22
S^6_2	6	23
S^6_3	6	25
S^6_4	6	27
S^7_1	7	23
S^7_2	7	21
S^7_3	7	24
S^7_4	7	22
S^7_5	7	22
S^{10}_1	10	22
S^{10}_2	10	24
S^{10}_3	10	23
S^{10}_4	10	21

Taking into the consideration the SA $\rightarrow$ DA path of the transactions listed in Table 2, the number of message streams (IDTs) handled by a BM *bm* in the system (nh_{IDT}^{bm}) , will be as listed in Table 3.

Table 3. High priority streams handled by the BMs

BM	nh_{IDT}^{bm}
M2	2
M3	9
M4	4
M5	6
M8	5
M9	2

For this scenario, and using the formulations outlined in Section 3, the WCRT for the system message streams will result as presented in Table 4, together with the value of $Rbmi^k_i$.

Table 4. WCRT for the system message streams

Stream	SA	DA	$Rbmi^k_i$ (ms)	$Rmlr^k_i$ (ms)
S^1_1	1	22	13.25	28.7
S^1_2	1	24	31.2	44.7
S^1_3	1	27	13.3	28.7

S^1_4	1	25	13.3	28.7
S^6_1	6	22	13.3	28.7
S^6_2	6	23	24.4	36.7
S^6_3	6	25	13.3	28.7
S^6_4	6	27	13.3	28.7
S^7_1	7	23	5.49	23.5
S^7_2	7	21	2.66	23.5
S^7_3	7	24	12.1	31.5
S^7_4	7	22	-	7.5
S^7_5	7	22	-	7.5
S^{10}_1	10	22	15.8	28.6
S^{10}_2	10	24	-	4.59
S^{10}_3	10	23	4.55	20.6
S^{10}_4	10	21	31.3	44.6

5. Working on reducing the pessimism

The results presented above have a level of pessimism that is inherent to considering the simultaneous occurrence of a number of worst-case situations. Although this may be a fate inherent to all guaranteed approaches based on worst-case scenarios, particularly in the case of distributed event-driven systems, it is important to investigate whether there is room for some improvements (in terms of reducing the pessimism level, in the analytical models).

In this Section we elaborate a bit further on this direction.

Eventually, one of the main sources of pessimism resides in the assumption made on considering that in the worst case only one high priority message can be processed at each token visit. Also, it may not be a negligible source of pessimism the assumption that all message streams relayed by a bridge master will be ready for transmission at the same time.

While fighting against the first would eventually collide with a basilar (from the real-time perspective) approach for handling such type of real-time guarantees in PROFIBUS networks, the latter probably deserves a second thought. In fact, being able to better model the maximum number of message streams that can be simultaneously queued by the BMs, may strongly impact the values for $Rbmi^k_i$.

For the calculation of $Rbmi^k_i$, we assumed a worst-case situation at each bridge, in which all message streams relayed by a BM could be queued for transmission just prior to the instant when a frame related to a message stream *i* from master *k* arrives at the BM. In fact, that assumption can be somehow relaxed. On a dual port bridge, the messages arriving at the bridge are received in sequence by one of the bridge ports, hereafter called the input BM. At the same time, these messages can be transmitted by the other bridge port – the output BM of the bridge. Consequently, in some cases, when a frame from message stream *i* from master *k* is queued on the output BM, the output queue will not, simultaneously, have frames from all message streams.

Figure 6 will support further intuition on this. The illustrated example assumes the network structure presented in Figure 1, and describes the sequence of events for transactions between masters M1, M6 and slaves S22, S27 and S25, related to message streams S_1^1, S_3^1, S_1^6 and S_3^6, assuming additionally that no other traffic exists in the network. For convenience messages transmitted by S22, S27 and S25 are depicted in the same line of BM M3.

In this example, we are considering that all request messages are queued just prior to the reception of the token by M1. As it can be observed, when a frame from message stream S_2^6 is ready for transmission, the response frame related to message streams S_1^1 and S_3^1 had already been obtained by bridge B1. In the figure, the queuing delay for S_2^6 in master M1 ($QM1_2^6$) is shown, as also the queuing delay in BM M7 ($QM7_2^6$). $QM7_2^6$ depends on the number of frames queued on M7 (2 considering one from S_2^6), at the arrival of a frame from message stream S_2^6 and the ongoing transactions on wired domain 1.

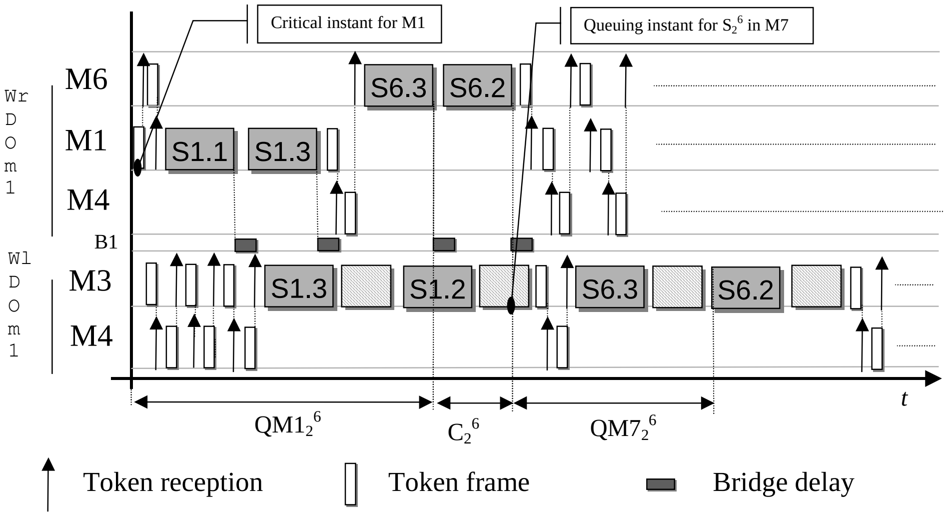

Figure 6. IDT arriving at a BM

What we add here for the computing the worst-case response time, is that it should not be needed to consider that, in all cases, all message streams relayed by a BMs are on their output queues simultaneously.

Therefore, we could introduce the notation nh'^{bm}_{IDT} as the maximum number of IDT transactions simultaneously queued by a bridge master *bm* (note the impact of *nh* in equation (5)).

The analysis presented in Section 3 guarantees that at least one high priority message is dispatched per token visit, thus it is guaranteed that the output queue of a BM is reduced by one element at least once every token visit.

We are now trying to devise a general formulation to this problem. Consider a bridge constituted by a bridge master *k* and a bridge master *l*. Bridge master *k* receives the incoming traffic from its domain - the input domain, and bridge master *l* forwards the traffic to another domain – the output domain.

The incoming traffic can be characterised as follows:

- all message streams related to IDT that use BM *k* arrive in sequence;
- we assume that all messages have the same size (*Cin*), equal to the minimum size of the input message streams;
- all messages arrive at BM *k* with minimal separation.

The traffic forwarded by BM *l* can be characterised as follows:

just prior to receiving the first message concerning input traffic, there is a transmission opportunity;

- the remaining transmitting opportunities are separated by the cycle time of the output domain (t^{out}_{cycle}).

Figure 8, depicts, on a simplified time-line, the arrivals at BM M4 and the transmissions by BM M7, which illustrates the assumptions.

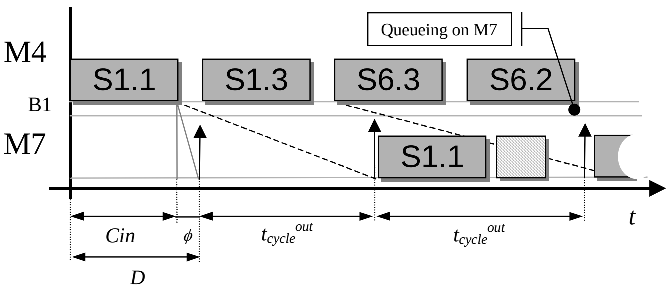

Figure 8. worst-case scenario

The initial delay, *D* is equal to $Cin + \phi$. To obtain the number of messages which can be transmitted by the output BM, we propose the following algorithm.

```
Cal_nh'_idt(Cin, nh_idt, tcycle, ϕ)
{
 // F: vector that contains if an IDT
 // had been forwarded by the output BM

 // Analysis interval
 ttotal = Cin × nh_idt;
 D = Cin + ϕ;
 // inicial time
 t = D + tcycle;
 nm = 0

 while t + Cin < ttotal
```

```
{
   // messeges available
   msg_av = floor(t / Cin);
   if t > nm × Cin and F(nm) == 0
      and nm ≤ masg_av then
   {
       F(nm) = 1;
       nm = nm + 1;
   }
   t = t + tcycle;
 }
 nh'_idt = nh_idt - nm
 return(nh'_idt)
}
```

Figure 9. Algorithm for obtaining nh'_{IDT}

Consider again the scenario depicted in Figure 1 and the conditions described in Section 4. Figure 10, plots the reduction in the number of message streams (i.e. $nh_{IDT} - nh'_{IDT}$) for computing $Rbmi^k_i$ for a BM as a function of the total number of message streams relayed by a BM, for M3, M5 and M9.

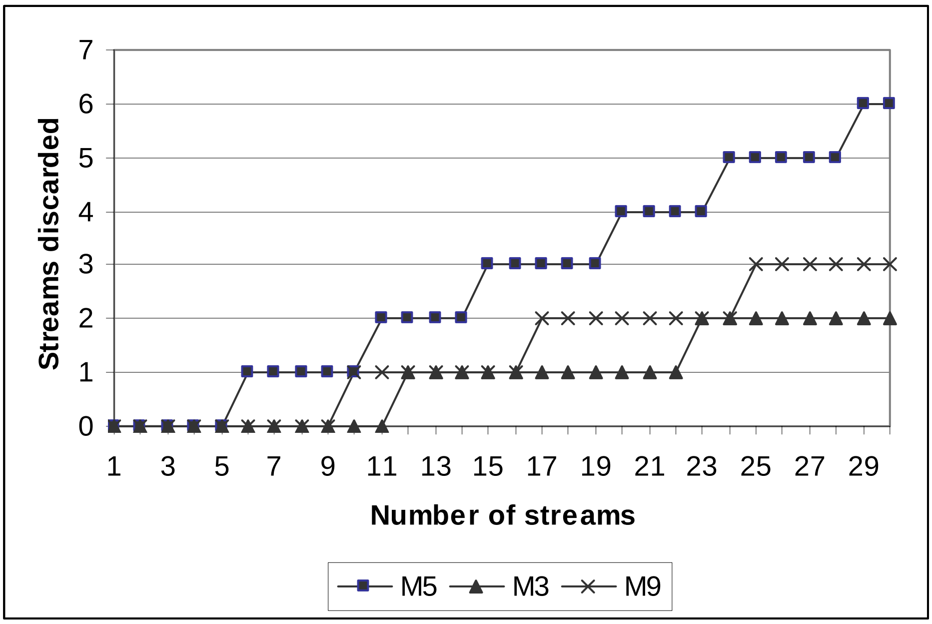

Figure 10. Discarded message streams on a BM

6. Conclusions

Recently, there have been a few research efforts towards extending the capabilities of fieldbus networks to encompass wireless support. In previous works [3-5] we have proposed a hybrid wired/wireless PROFIBUS network solution where the interconnection between the heterogeneous communication media was accomplished through bridge-like interconnecting devices. The resulting networking architecture embraced a Multiple Logical Ring (MLR) approach, thus with multiple independent tokens, to which a specific bridging protocol extension, the Inter-Domain Protocol (IDP), was proposed. The IDP offers compatibility with standard PROFIBUS and supports inter-cell mobility of wireless nodes. In this paper we advance that work by proposing a worst-case response timing analysis of the IDP.

This type of analysis has typically a certain level of pessimism, which is inherent to the consideration a simultaneous occurrence of a number of worst-case situations. Although this is the fate of guaranteed approaches based on worst-case scenarios, particularly for distributed event-driven systems, in the last section of this paper we also introduced some ways to improve further the proposed analytical formulations.

Ongoing work is now being carried out concerning incorporating the timing behaviour of the inter-cell mobility protocol already defined for the architecture [4-5].

7. References

[1] Alves, M., Tovar, E., Vasques, F., Roether, K. and Hammer, G., "Real-Time Communications over Hybrid Wired/Wireless PROFIBUS-based Networks", Proceedings of the 14th Euromicro Conference on Real-Time Systems, pp. 142-151, June 2002.

[2] Rauchhaupt, L., "System and Device Architecture of a Radio Based Fieldbus – The RFieldbus System", Proceedings of the 2002 IEEE International Workshop on Factory Communication Systems, pp. 193-202, August 2002.

[3] Ferreira, L., Alves, M., Tovar, E., "Hybrid Wired/Wireless PROFIBUS Networks Supported by Bridges/Routers", Proceedings of the 2002 IEEE International Workshop on Factory Communication Systems, Vasteras, Sweden, pp. 193-202, August 2002.

[4] Ferreira, L., Tovar, E., Alves, M., "PROFIBUS Protocol Extensions for Enabling Inter-Cell Mobility in Bridge-based Hybrid Wired/Wireless Networks", Proceedings of the 5th IFAC International Conference on Fieldbus Systems and their Applications, Aveiro, Portugal, pp.283-290 July 2003.

[5] Ferreira, L., Tovar, E., Alves, M., "Enabling Inter-Domain Transactions in Bridge-Based Hybrid Wired/Wireless PROFIBUS Networks", Proceedings of the 9th IEEE International Conference on Emerging Technologies and Factory Automation - ETFA2003, Lisbon, Portugal, pp. 15-22, July 2002.

[6] Tovar, E., "Supporting Real-Time Communications with Standard Factory-Floor Networks", PhD Dissertation, Faculdade de Engenharia da Universidade do Porto, July 1999.

[7] Grow, R., "A Timed Token Protocol for Local Area Networks", Proceedings of Electro'82, Token Access Protocols, Paper 17/3, 1982.

[8] Tovar, E. and Vasques, F., "Real-Time Fieldbus Communications Using PROFIBUS Networks", IEEE Transactions on Industrial Electronics, vol. 46, no. 6, December 1999.

[9] Tovar, E. and Vasques, F., "Cycle Time Properties of the PROFIBUS Timed-Token Protocol", Computer Communications 22 (1999), pp. 1206-1216, Elsevier.

[10] Tovar, E., Vasques, F. and Burns, A., "Supporting Real-Time Distributed Computer-Controlled Systems with Multi-hop P-NET Networks", Control Engineering Practice, Vol. 7, No. 8, pp. 1015-1025, Pergamon, Elsevier Science, August 1999.

[11] Vasques, F., Juanole, G., "Pre-run-time Schedulability Analysis in Fieldbus Networks", Proceedings of the 20th Annual Conference of the IEEE Industrial Electronics Society (IECON'94), Bologna, Italy, pp. 1200-1204, September 1994.

[12] Cavalieri, S., Monforte, S., Tovar, E., Vasques, F., "Evaluating Worst-Case Response Time in Mono and Multi-Master Profibus-DP Networks", Proceedings of the 4th IEEE International Workshop on Factory Communication Systems (WFCS'2002), Vasteras, Sweden, pp.233-240, August 2002.

[13] Alves, M., “Real-Time Communications over Hybrid Wired/Wireless PROFIBUS-Based Networks”, PhD Dissertation, Faculdade de Engenharia da Universidade do Porto, February 2003.

[14] Tovar, E., Vasques, F., Burns, A., “Supporting Real-Time Distributed Computer-Controlled Systems with Multi-hop P-NET Networks”, Control Engineering Practice, Vol. 7, No. 8, pp. 1015-1025, Pergamon, Elsevier Science, August 1999.