Towards resilient real-time wireless communications

Jeferson L. R. Souza and José Rufino
University of Lisboa - Faculty of Sciences
LaSIGE - Navigators Research Team
Email(s): jsouza@lasige.di.fc.ul.pt, ruf@di.fc.ul.pt

***Abstract*—The use of wireless networks on environments with real-time constrains requires timeliness and dependability guarantees to allow the execution of time-restricted networked operations. The current wireless standards and state-of-the-art solutions pay no or little attention to dependability aspects of communications, which are fundamental to secure any timeliness guarantee. This paper presents an innovative standard-compliant solution, dubbed *Mediator Layer*, which extends the MAC sublayer with additional components that can be incorporated within networked simulators and implemented in currently available hardware platforms. Thus, the *Mediator Layer* enhances and complements the services traditionally offered by wireless network standards, providing a set of fundamental abstractions useful for both system design and application programming.**

***Index Terms*—wireless sensor and actuator networks, real-time communications, dependability, timeliness, resilience.**

I. Introduction

The use of wireless networks on environments where communications may have real-time requirements is a current trend [1]. This trend is guided by the needs to reduce system size, weight, and power consumption (SWaP) without lessen timeliness and dependability guarantees. Industrial, vehicular, and aerospace environments are waiting for a solution to address an effective and efficient real-time support on wireless communications. For example, [2] discusses a set of pressing challenges on the use of wireless sensor networks (WSNs), more specifically wireless sensor and actuator networks (WSANs), in industrial automation, where dependability and real-time guarantees are a must.

So far, a lot of work has been done, proposing new medium access control (MAC) protocols [3]–[10], modifications on the existent standards [11]–[13], and abstract models [14] trying to enhance the reliability on wireless communications. However, all these works focus their analyses in the temporal aspects of the frame transmission service, paying no or little attention to the dependability aspects of communications, which are fundamental to secure any timeliness guarantee. An interesting conclusion draw in [2] is the necessity of improvements on the existent standards face to requirements of safe and secure networked operations.

Thus, this work-in-progress presents an innovative solution dubbed *Mediator Layer* [15], which is capable to enhance the dependability and timeliness of MAC sublayer services. In addition, a set of constructs useful to the programming of distributed applications, such as reliable communications, node failure detection, membership and clock synchronization may be offered immediately above the MAC sublayer. In this sense, our solution enhances and complements the services traditionally offered by wireless network standards, providing a set of fundamental abstractions useful for both system design and application programming. A prototype of the *Mediator Layer* is being integrated in the NS2 simulator [16] and in a commercial off-the-shelf (COTS) platform [17], using the IEEE 802.15.4 standard as a case study.

The presentation of our advances is organized as follows: Section II presents the system model. Sections III and IV describes our work-in-progress solution, including the incorporation of the *Mediator Layer* in the NS2 simulator and in the hardware platform. Finally, Section V presents some conclusions and future directions of this work.

II. System model

In this section we provide a brief description of our system model, which establishes a base foundation for our design and simulations. Our system model is formed by a set of wireless nodes $X = \{x_1, x_2, \ldots, x_n\}$, being $1 < n \leq \#A$, where A is the set of all wireless nodes using the same communication channel. A wireless node is a networked device capable to communicate with other wireless nodes. The set of nodes X itself defines a node relationship entity dubbed wireless network segment (WnS), which is established by all wireless nodes within $X \subseteq A$ that use a given communication channel and share a single hop communication range space.

For any WnS we use the following assumptions:

1) the communication range of X, i.e. its broadcast domain, is given by: $B_X = \bigcap_{j=1}^{n} B_D(x_j),\ \forall x_j \in X$, where $B_D(x_j)$ represents the communication range of node x_j;
2) $\forall x \in X$ can sense the transmissions of one another;
3) $\forall x \in A,\ x \in X \iff B_D(x) \bigcap B_X = B_X$ or, as a consequence of node mobility, $x \notin X \iff B_D(x) \bigcap B_X \neq B_X$;
4) $\exists x \in X$ which is the coordinator, being unique and with responsibility to manage the set X;
5) a network component (e.g. a node $x \in X$) either behaves correctly or crashes upon exceeding a given number

This work was partially supported: by the EC, through project IST-FP7-STREP-288195 (KARYON); by FCT/DAAD, through the transnational cooperation project PROPHECY; and by FCT, through the project PTDC/EEI-SCR/3200/2012 (READAPT), the Multiannual Funding Program, and the Individual Doctoral Grant SFRH/BD/45270/2008.

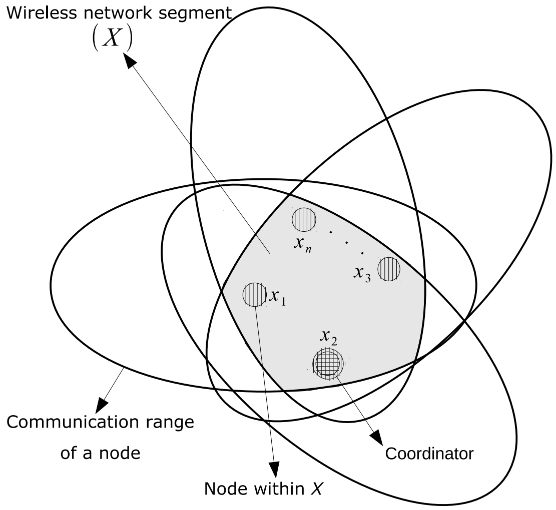

Fig. 1: Representation of a wireless network segment

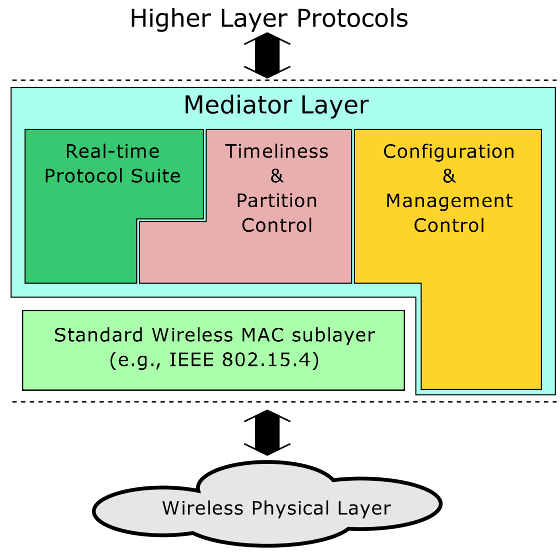

Fig. 2: The Mediator Layer and its components

of consecutive omissions (the component's *omission degree*, f_o) in a time interval of reference[1], $\mathcal{T}_{rd}$;

6) failure bursts never affect more than f_o transmissions in a time interval of reference, $\mathcal{T}_{rd}$;

7) omission failures may be inconsistent (i.e., not observed by all recipients).

Assumptions 1 to 3 define the physical relationship between nodes within the WnS. Our system model characterizes the relationship between nodes at MAC sublayer, where nodes must be in the communication range of each other to communicate, and are able to sense one another (assumption 2). Mobility may drive nodes away of the WnS (assumption 3).

In the context of network components, an omission is an error that destroys a data frame. Establishing a bound for the *omission degree* (assumptions 5 and 6) provides a general method for the detection of failed components. If each omission is detected and accounted for, the component fails once it exceeds the *omission degree bound*, k. The *omission degree* is thus a general measure of the reliability of network components with respect to accidental/intentional transient errors.

Figure 1 presents a graphical representation of a wireless network segment. In this figure we can see the communication range of each node within X, evidencing the intersection between all communication ranges of all nodes, which delimits the broadcast domain of X. One node within X assumes the role of coordinator (assumption 4). The management activities of the coordinator comprises the assignment of the current communication channel in use by the WnS, the allocation of guaranteed slots for frame transmissions, and so on.

[1]For instance, the duration of a given protocol execution. Note that this assumption is concerned with the total number of failures of possibly different wireless nodes.

III. Towards resilient real-time wireless communications

Our approach to enhance the dependability and timeliness of wireless communications consists of an extensible component layer, dubbed *Mediator Layer*, build around a standard MAC sublayer. No modification is required to existing standards, being the *Mediator Layer* standard-compliant solution easily implemented in currently available COTS platforms. In this sense, the *Mediator Layer* approach significantly differs from other solutions described in the literature [3]–[14].

The *Mediator Layer* provides enhanced frame transmission and management services through a minimal set of fundamental components (draw in Fig. 2), which handle the actions required to secure dependability and timeliness in communications.

The *Real-Time Protocol Suite* (Fig. 2) component is responsible for handling frame transmissions, which can be data or management frames. Different protocols serving requests with different types of requisites, ranging from unreliable unicast to reliable broadcast, can be incorporated within this component. This augments the applicability of wireless networks to broader class of applications, including those with mixed-criticality requirements. Our intuition is that replacing a classical timeout-based protocol design approach with a combination of positive and negative confirmations (acknowledgements) contributes to reduce protocol worst case termination times, and thus service timeliness. Timeout-based techniques may still be used but only to detect node crash failures.

The *Timeliness and Partition Control* (Fig. 2) component deals with the temporal aspects related to the frame transmission service. Controlling and monitoring the timing of the actions within the *Mediator Layer* relies on an internal *Time Service* that ensures the temporal awareness of all networked operations, including the occurrence of temporary network partitions (i.e., network inaccessibility [18]) or even timeliness violations of a specific transmission protocol. An example is the monitoring and verification of a deadline associated to a data request, which may be disturbed by inaccessibility incidents. The dynamic control of inaccessibility periods [15] allows protocol execution to be aware of the real duration of inaccessibility incidents, and to (self-)adapt its execution to actual network operating conditions. In particularly, optimal timeout values can be used in the dimensioning of protocol timers, which contributes to enhance the timeliness guarantees.

The *Configuration and Management Control* (Fig. 2) component manages and controls the configuration of all parameters of the MAC sublayer and the internal parameters of the *Mediator Layer*, respecting application requirements, resource limitations, and environment restrictions. Such requirements, limitations, and restrictions should be defined in an profile that is utilised to adjust the aforementioned parameters. Autonomous methods may be utilized by the *Configuration and Management Control* components, making configuration procedures self-adaptive, self-managed, and self-controlled.

IV. BUILDING *Mediator Layer* COMPONENTS

In order to illustrate how to improve the dependability and timeliness properties of a standard MAC sublayer, we select a set of key mechanisms that needs to be included in the *Mediator Layer* for that purpose.

The first mechanism we address introduces a minor extension in the standard frame check sequence (FCS) mechanism. The native FCS mechanism silently discards every frame received with errors, meaning omission errors will not be perceived, nor monitored nor accounted for. Conversely, the extension specified in Algorithm 1 provides a management indication of the status of the received frame to the *Mediator Layer*, even if the frame was received with errors. The management indication highlighted in line 15 signals: the instant the frame was received, represented by the variable $time\_stamp$, as obtained in line 6; the frame header, represented by the variable $frame\_header$, which is extracted in line 7; and the FCS error status represented by the variable fcs_error. This simple extension of the native standard mechanism enriches the monitoring capabilities of the *Mediator Layer*.

The second mechanism we address takes profit of the FCS extension mechanism to account for omission errors. The accounting of omission errors is expressed as a pseudo-code presented in Algorithm 2. The status of each frame received by a node is signalled through a MAC management indication in line 7. If a frame was received with errors, the number of consecutive omission errors, O_{degree}, is incremented by one (line 9). Otherwise, i.e., whenever a frame is received without errors, the value of O_{degree} is cleared (line 11). Should the omission degree bound, k, be exceeded, a management indication is provided (line 14). This is an indication that the communication channel in use has failed and that a channel switch action should be issued to MAC sublayer management entities.

Algorithm 1 Extending the FCS mechanism

```
 1: Initialization phase.
 2: fcs_error ←false;
 3: Begin.
 4: loop
 5:     when Channel.indication(frame) do
 6:         time_stamp ←MLA.get.time();
 7:         frame_header ←MAC.get.header(frame);
 8:         if MAC.FCS.check(frame) is OK then
 9:             fcs_error ← false;
10:             MAC.indication(frame);
11:         else
12:             fcs_error ← true;
13:             MAC.discard(frame);
14:         end if
15:         MAC.Mgmt.indication(time_stamp, frame_header, fcs_error);
16:     end when
17: end loop
18: End.
```

Algorithm 2 Accounting local omission errors

```
 1: Initialization phase.
 2: O_degree ← 0;
 3: k ← The value of the omission degree bound depends on environment
    conditions. The default value utilized within the IEEE 802.15.4 standard
    is k = 3.
 4: current_channel ← It represents which is the current channel in use.
 5: Begin.
 6: loop
 7:     when MAC.Mgmt.indication(time_stamp, frame_header, fcs_error)
        do
 8:         if fcs_error = true then
 9:             O_degree ← O_degree + 1;
10:         else
11:             O_degree ← 0;
12:         end if
13:         if O_degree > k then
14:             MLA.Mgmt.indication(time_stamp,current_channel,
                O_degree_exceeds_k)
15:         end if
16:     end when
17: end loop
18: End.
```

The real-time operation of nodes within the WnS is also monitored by a node failure detection service, which is part of the *Real-time Protocol Suite* component. The node failure detection is presented in Algorithm 3. This algorithm resides within every node of a WnS, establishing a distributed and decentralized node failure detection service.

Each node locally accounts omissions from other nodes of the WnS as follows: an indication representing the status of a received frame is detected (line 6). The source node index of the received frame is extracted from the $time\_stamp$ and $frame\_header$ variables (line 7). If a valid node index is found (e.g., the node source identifier in the frame header matches the foreseen time slot represented here by the $time\_stamp$ variable). If the frame was received with errors, the omission degree of this node is incremented by one (line 10), otherwise it is reseted to zero (line 12). When the omission degree of a given source node of a received frame exceeds k, a local indication is generated by the node failure detection service (line 15), which can be utilized by a membership

Algorithm 3 Node failure detection

* The detection of a node's crash is intentionally ommitted

Initialization phase.
$k \leftarrow 3$;
$node_{index} \leftarrow$ *It represents the index of a node.*
Begin.
loop
  when *MAC.Mgmt.indication(time_stamp,* $frame\_header, fcs\_error$*)* **do**
    $node_{index} \leftarrow$ *MLA.Mgmt.getNode(time_stamp,* $frame\_header$*)*;
    if $node_{index}$ *is valid* **then**
      if fcs_error = true **then**
        $O_{degree}[node_{index}] \leftarrow O_{degree}[node_{index}] + 1$;
      else
        $O_{degree}[node_{index}] \leftarrow 0$;
      end if
      if $O_{degree}[node_{index}] > k$ **then**
        MLA.FD.indication(time_stamp,current_channel, $node_{index}$, $O_{degree}[node_{index}]$*_exceeds_k)*
      end if
    end if
  end when
end loop
End.

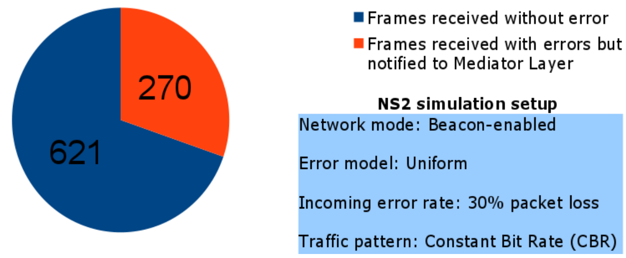

Fig. 3: Results obtained with the extension of the FCS mechanism implemented within NS2 simulator

service to help the update of the view that represents the active nodes within the WnS.

As a proof-of-concept the *Mediator Layer* has been implemented and incorporated within both NS2 simulator [16] and a COTS platform [17], using the IEEE 802.15.4 standard as a case study. The results achieved so far have shown that these mechanisms allow to achieve optimal latencies with respect to the detection of channel failures (Algorithm 2) and node failures (Algorithm 3). Any violation of the omission degree bound is also detected as soon as it occurs.

This happens because frame omissions are monitored and detected at the lowest level of communications (Algorithm 1). The effectiveness of this approach is illustrated in Fig. 3. In Fig. 3 the blue color represents the number of frames (621) received without any error (i.e., indicated with or without the presence of *Mediator Layer*), while the orange color represents the number of frames (270) received with errors, and detected by *Mediator Layer*.

V. Conclusion and Future Work

This paper presented an extensible component layer dubbed *Mediator Layer*, which enhances the timeliness and dependability properties of wireless communications. A dependable and timely service at lowest level of the network protocol stack helps the higher level protocol designers to keep their solutions as simple as possible, using the easy-to-design building block approach enabled by the *Mediator Layer*. Our standard-compliant approach has been implemented and incorporated within the NS2 network simulator and a COTS platform.

Future directions include, but are not limited to: adding protocols to the *Real-Time Protocol Suite* component to cover the requirements imposed by real-time environments; finishing the implementation and incorporation of the *Mediator Layer* within the NS2 network simulator and the COTS platform; performing analyses and evaluation of the *Mediator Layer* approach in such both simulator and COTS platform face to different error conditions and temporal requirements of environments and applications; and defining relevant real-time metrics to evaluate the wireless communications with regard to application requirements and environment restrictions.

References

[1] T. Stone, R. Alena, J. Baldwin, and P. Wilson, "A viable COTS based wireless architecture for spacecraft avionics," in *IEEE Aerospace Conference*, 2012, pp. 1–11.
[2] J. Åkerberg, M. Gidlund, and M. Björkman, "Future research challenges in wireless sensor and actuator networks targeting industrial automation," in *9th IEEE INDIN*, July 2011.
[3] A. Sahoo and P. Baronia, "An energy efficient MAC in WSN to provide delay guarantee," in *15th IEEE LANMAN*, June 2007.
[4] I. Aad, P. Hofmann, L. Loyola, F. Riaz, and J. Widmer, "E-MAC: Self-organizing 802.11-compatible MAC with elastic real-time scheduling," in *IEEE MASS*, October 2007.
[5] E. E-López, J. V-Alonso, A. M-Sala, J. G-Haro, P. P-Mariño, and M. Delgado, "A WSN MAC protocol for real-time applications," *Personal Ubiquitous Computing Journal*, January 2008.
[6] P. Bartolomeu, J. Ferreira, and J. Fonseca, "Enforcing flexibility in real-time wireless communications: A bandjacking enabled protocol," in *IEEE ETFA*, September 2009.
[7] X.-Y. Shuai and Z.-C. Zhang, "Research of real-time wireless networks control system MAC protocol," *Journal of Networks*, April 2010.
[8] T. Zhou, H. Sharif, M. Hempel, P. Mahasukhon, W. Wang, and T. Ma, "A novel adaptive distributed cooperative relaying MAC protocol for vehicular networks," *IEEE Journal on Sel. Areas in Comm.*, Jan 2011.
[9] M. Sha, G. Hackmann, and C. Lu, "Arch: Practical channel hopping for reliable home-area sensor networks," in *17th IEEE RTAS*, April 2011.
[10] X. Zhu, S. Han, P.-C. Huang, A. Mok, and D. Chen, "MBStar: A real-time communication protocol for wireless body area networks," in *23rd ECRTS*, July 2011.
[11] Yu-Kai, Ai-Chun, and Hui-Nien, "An adaptive GTS allocation scheme for IEEE 802.15.4," *IEEE Trans. on Parallel and Distributed Systems*, May 2008.
[12] M. Hameed, H. Trsek, O. Graeser, and J. Jasperneite, "Performance investigation and optimization of IEEE 802.15.4 for industrial wireless sensor networks," in *IEEE ETFA*, September 2008.
[13] A. Koubâa, A. Cunha, M. Alves, and E. Tovar, "i-GAME: An implicit GTS allocation mechanism in IEEE 802.15.4, theory and practice," *Springer Real-Time Systems Journal*, August 2008.
[14] F. Kuhn, N. Lynch, and C. Newport, "The abstract MAC layer," in *23rd DISC*, September 2009.
[15] J. L. R. Souza and J. Rufino, "An approach to enhance the timeliness of wireless communications," in *5th UBICOMM*, Lisbon, 2011.
[16] T. Issariyakul and E. Hossain, *Introduction to Network Simulator NS2*. Springer, 2009.
[17] ATMEL, *ATMEL AVR2025: IEEE 802.15.4 MAC Software Package - User guide*, ATMEL Coorporation, May 2012.
[18] J. L. R. Souza and J. Rufino, "Characterization of inaccessibility in wireless networks - a case study on IEEE 802.15.4 standard," in *3th IFIP IESS*, September 2009.