System Architectures, Protocols and Algorithms for Aperiodic Wireless Control Systems

José Araújo, Manuel Mazo Jr., *Member, IEEE*, Adolfo Anta, Paulo Tabuada *Member, IEEE,* and Karl H. Johansson *Fellow, IEEE,*

***Abstract*—Wide deployment of wireless sensor and actuator networks in cyber-physical systems requires systematic design tools to enable dynamic trade-off of network resources and control performance. In this paper, we consider three recently proposed aperiodic control algorithms which have the potential to address this problem. By showing how these controllers can be implemented over the IEEE 802.15.4 standard, a practical wireless control system architecture with guaranteed closed-loop performance is detailed. Event-based, predictive and hybrid sensor and actuator communication schemes are compared with respect to their capabilities and implementation complexity. A two double-tank laboratory experimental setup, mimicking some typical industrial process control loops, is used to demonstrate the applicability of the proposed approach. Experimental results show how the sensor communication adapts to the changing demands of the control loops and the network resources, allowing for lower energy consumption and efficient bandwidth utilization.**

***Index Terms*—Aperiodic wireless control, event-triggered, self-triggered, hybrid communication, wireless sensor and actuator networks, networked control systems.**

I. Introduction

We are witnessing an increasing interest in the implementation of control systems over wireless communication channels [1], [2]. The potential benefits of wireless industrial processes are numerous, for instance, the reduction of wiring costs, the ease of deployment, and the increased versatility for sensor placement. By introducing digital communication channels at the core of complex control systems, one obtains a cyber-physical system, in which a digital computing system interacts with a physical plant. Analyzing and designing such cyber-physical systems poses numerous challenging problems that the scientific community has only recently started to address [3]–[9].

A major technological concern in wireless networked systems is the power consumption of sensors equipped with small batteries. Transmitting measurements too often, as is normally the case in today's digital control systems, can drain the energy reserves of a battery-driven sensor in a few hours. New control theoretical advancements on this front have started to appear in the literature, inspired by the seminal work on event-based control by [10], [11]. Several techniques have been proposed providing suitable closed-loop performance guarantees [12], [13], [14], [15], [16], dealing with decentralized systems [17], [18], network effects [19], [20], [21] and being experimentally validated [22], [23], [24], [25]. In particular, the work of [12] and its follow-ups aims at reducing communication by proposing an aperiodic control scheme. While in a traditional sampled-data paradigm [26] new measurements and controller updates are performed periodically, regardless of the state of the plant, this new aperiodic paradigm is based on events triggered only when stability or a pre-specified control performance are about to be lost. Moreover, there is a strong effort to provide adequate communication protocols for wireless control. The standardization of low data rate and low power wireless networks is an ongoing process, yet there is no single widely accepted complete protocol stack for control [27].

In this paper we propose a new wireless control system architecture for the practical implementation of networked control that guarantees stability and performance while minimizing energy consumption and network bandwidth usage. The overall wireless networked control system is illustrated in Fig. 1, where several plants share a control node and a network manager node directs the access to the shared wireless medium. This architecture is well established from the current practice in process industry [28], [29], [1], where efficient integration of communication and control has been identified as a high-impact challenge for the next generation of industrial automation systems [30]. Relying on the techniques of [15], we propose three mechanisms for aperiodic implementation of the update times: *event-based*, *predictive* and *hybrid communication*. Each mechanism defines an aperiodic control scheme and a communication scheduling. The underlying idea in each of these cases is to introduce a feedback loop to decide the control update times, thus linking control and communication. Our design relies on the IEEE 802.15.4 protocol [31] as the communication medium between sensors, controller and actuators. The IEEE 802.15.4 protocol is the base of several emerging wireless communication standards for industrial control such as WirelessHART [32] and ISA100 [33]. We propose a few modifications to the IEEE 802.15.4 standard that enable our proposed implementations. We implemented all these proposals in a set of double-tank systems and performed experiments demonstrating the efficiency of our implementa-

Manuscript received February 27, 2012. Accepted for publication April 17, 2013.

J. Araújo and K. H. Johansson are with the ACCESS Linnaeus Center, Royal Institute of Technology, Sweden (email: {araujo,kallej}@kth.se).

M. Mazo Jr. is with the Delft Center for Systems and Control, Delft University of Technology, The Netherlands (email: m.mazo@tudelft.nl).

A. Anta is with the General Electric Global Research Europe, Germany (e-mail: anta@ge.com).

P. Tabuada is with the Department of Electrical Engineering, University of California, Los Angeles (e-mail: tabuada@ee.ucla.edu).

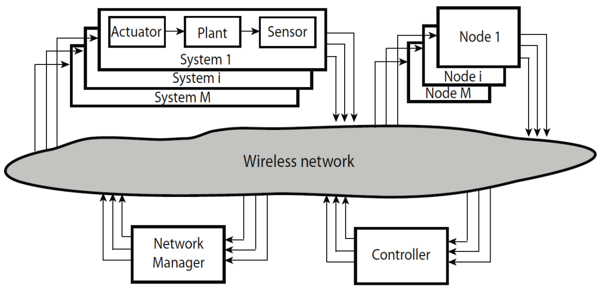


Fig. 1: Wireless networked control system.

tions in terms of energy and communications bandwidth usage.

The rest of the paper is organized as follows: Section II provides a brief review of the aperiodic control techniques in [15]; Section III presents the implementation mechanisms that we propose; Section IV describes the necessary modifications to IEEE 802.15.4 MAC and in Section IV-C the communications scheduling is introduced; finally the experimental setup and results are described in Sections V and VI; the paper ends with some brief conclusions in Section VII.

II. Aperiodic Control Techniques

In the current section we provide a brief review of recently proposed techniques for the implementation of control systems that does not require the periodic transmission of measurements and computation of control inputs. Instead, the techniques reviewed in this section provide times between controller updates that are not constant, and whose value depend on the state of the system. We also provide simple solutions to cope with delays in these techniques.

A. Event-Triggered Control

We consider a linear control system of the form:

$$\dot{x}(t) = Ax(t) + Bu(t), \quad x(t) \in \mathbb{R}^n, u(t) \in \mathbb{R}^m \tag{1}$$

where A and B are matrices of appropriate dimensions. A controller $u = Kx$ is designed to render the system asymptotically stable, hence, there exists a Lyapunov function of the form $V = x^T Px$ satisfying:

$$\dot{V} = \frac{\partial V}{\partial x}(A + BK)x = -x^T Qx \tag{2}$$

where Q is a positive definite matrix. The Lyapunov function V can be seen as a *certificate of stability*, since according to equation (2) V is always decreasing, but also of *performance* since (2) also ensures that the rate of decrease is at least $x^T Qx$.

However, in digital implementations the input u cannot be updated continuously but only at discrete time instants t_k, whenever an actuator message is received. Thus, we consider sample-and-hold implementations, in which the value of u remains constant between two consecutive controller updates, *i.e.* $u(t) = Kx(t_k), \forall\, t \in [t_k, t_{k+1}[$. In such implementations, the evolution of the Lyapunov function V is instead given by:

$$\dot{V} = \frac{\partial V}{\partial x}(Ax + BKx(t_k)). \tag{3}$$

One can specify the desired performance of the implementation by means of a function S required to upper bound the evolution of V:

$$V(t) \leq S(t). \tag{4}$$

Provided that (4) holds and S is decaying over time, the closed-loop system is stabilized, with a decay rate of its Lyapunov function no lower than the one specified through S. Following [34], one such S is obtained as the Lyapunov function:

$$S(t) = x_s(t)^T Px_s(t) \tag{5}$$

for the hybrid system:

$$\dot{x}_s(t) = A_s x_s(t) \qquad t \in [t_k, t_{k+1}[\tag{6}$$

$$x_s(t_k) := x(t_k), \tag{7}$$

where A_s is a Hurwitz matrix satisfying the following Lyapunov equation $A_s^T P + PA_s = -R$, with $Q - R$ positive definite. One such choice is $R = \sigma Q$, where $\sigma \in]0, 1[$ determines the rate of decay of S, as a proportion of that of V in an analog controller implementation.

To enforce (4), an event-triggered implementation defines the sequence of time instants t_k when the control input needs to be updated, as:

$$t_k = \min\{t > t_{k-1} \mid V(x(t)) - S(x_s(t)) = 0\}. \tag{8}$$

For convenience, we denote by $g_S : \mathbb{R}^n \to \mathbb{R}_0^+$ the function defined by $g_S(x(t_k)) = t_{k+1} - t_k$, when employing the performance function S. From here on, we also refer to $\tau_k = t_{k+1} - t_k$ as the inter-transmission time.

Following the proposed design it can be guaranteed that the minimum inter-transmission time is strictly greater than zero over all possible initial conditions in the operating region Ω:

$$\tau_{min} = \inf_{x_0} g_S(x_0) > 0, \qquad \forall\, x_0 \in \Omega. \tag{9}$$

This time always guarantees a certain level of performance (as defined by S) and stability of the closed loop system and can be computed using Lemma 4.1 in [34].

In our digital implementations, the triggering condition (8) is checked periodically at the speed at which measurements are acquired. We denote this period by T. This implies that the performance that can actually be guaranteed is slightly reduced, as analyzed in [34].

B. Self-Triggered Control

Self-triggered control implementations relax the requirement of continuously monitoring the triggering condition (8) by predicting when it will be violated. A self-triggered technique provides estimates of the inter-transmission times τ_k by relying on the plant model, the last measurement of the state of the system $x(t_k)$, and the performance specification. The prediction of the time between two consecutive updates is embodied in a self-triggering function $\tilde{g}_S : \mathbb{R}^n \to \mathbb{R}_0^+$ satisfying $\tilde{g}_S(x(t_k)) \leq g_S(x(t_k)), \forall\, x(t_k) \in \mathbb{R}^n$. In self-triggered control it is also customary to impose an upper bound t_{max} to the inter-transmission times in order to provide robustness guarantees of the self-triggered controller.

Several methods have appeared in the literature to perform self-triggered control [14], [34], [15]. We employ in our implementations the method proposed in [34], which is aimed at predicting the violation of the event-triggering conditions for linear systems reviewed in Section II-A. Nevertheless, our proposals also apply to other available techniques, including those for nonlinear systems [15].

In brief, the idea of [34] is that knowing the initial condition and the currently applied input, the solution of (1) at times $t = nT$, $n = 1, 2, \dots$ can be explicitly computed, allowing to check the triggering condition ahead of time.

C. Compensation of Delays

In the following we propose two simple solutions to compensate for delays between measurement acquisition and actuation updates, applicable to the event-triggered and self-triggered techniques reviewed in the previous sections. These are the first explicit consideration of delays in the techniques proposed in [34]. For convenience, in the explanations that follow we divide the kind of delays present in (wireless) networked systems in two types: delays in the access to the communication channel, with an upper bound Δ_a; and delays due to the actual transmissions and computation at the controller, with an upper bound denoted by Δ_t. Furthermore, we denote by $\Delta = \Delta_a + \Delta_t$. For the purpose of a proper comparison between event-triggered and self-triggered techniques we selected techniques sharing the same triggering condition. This selection, in the event-triggered case, requires collocated sensors. Thus, we can assume in what follows that the sensing node has access to the whole state vector. We remark at this point that completely decentralized event-triggered techniques also exist which can be adapted to tolerate upper-bounded delays [17], [35].

1) Event-Triggered Control: In order to compensate delays for event-triggered control implementations, we propose to check condition (8) ahead of time, by predicting the value of V and S some amount of time in advance so that we can guarantee that the control update will take place before the condition is violated. This approach has a slight predictive flavor, and requires the sensing node to compute the control input locally. Let $A_\Delta = e^{A\Delta}$ and $B_\Delta = \int_0^\Delta e^{A(\Delta - r)} B dr$. From a sample acquired at $t_s = nT \in [t_k, t_{k+1}[$ one can estimate the value of the state Δ units of time in the future as:

$$\hat{x}(t_s + \Delta) = A_\Delta x(t_s) + B_\Delta u(t_k). \tag{10}$$

As we wish to guarantee until the next controller update that $V(nT) \leq S(nT)$ holds, we check instead if $\hat{V}(t_s + \Delta) \leq S(t_s + \Delta)$, where $\hat{V}(t) = \hat{x}(t)^T P \hat{x}(t)$. If $\hat{V}(t_s + \Delta) > S(t_s + \Delta)$ then the next controller update time is $t_{k+1} = t_s + \Delta - T$, and the sensor sends the predicted value $\hat{x}(t_s + \Delta - T)$ computed from the measurement $x(t_s - T)$. The controller then applies the new control input $u(t_{k+1}) = K\hat{x}(t_s + \Delta)$ at the time $t_s + \Delta$. Note that we impose the upper bound of the delay to the system by waiting until $t_s + \Delta$. This allows us to reduce communications between sensors and actuators, while still assuring the minimum inter-transmission times guarantees.

2) Self-Triggered Control: Due to the predictive nature of self-triggered control implementations, channel access delays can be prevented, *i.e.* $\Delta_a = 0$. This can be done by dynamically scheduling their channel access making use of the prediction of the next update event. However, computation and transmission delays may still affect the control loop. In this case, the sensor nodes are scheduled to transmit their measurements at $t_k - \Delta_t$. The controller then receives the measurements early enough to compute $u(t_k) = k\hat{x}(t_k)$ so that the actuator can apply the input at t_k.

The controller estimates $\hat{x}(t_k)$ from the actual measurement $x(t_k - \Delta_t)$ as shown in the previous case. Then, the estimate is used to compute the control signal, and also to obtain the next update time $t_{k+1} = t_k + \tilde{g}_S(\hat{x}(t_k))$.

III. Communications for Aperiodic Control

We are now ready to introduce aperiodic communication mechanisms to perform control over WSANs. Each communication mechanism defines the usage of event-triggered and/or self-triggered controller implementations, specifies the scheduling policy and MAC characteristics. The proposed mechanisms follow the architecture presented in Fig. 2, in which a plant to be controlled is instrumented with wireless sensors and actuators. Each sensor provides, through a Zero-Order-Hold (ZOH), high rate measurements to the the embedded event-generator. A controller receives wirelessly measurements from the sensors and produces a control input that is sent (wirelessly) to the actuators, which hold the input constant until a new value is received. Finally, a Network Manager (NM) is in charge of scheduling the communications between the diverse elements of the system.

A. Event-Based Communication

Under this paradigm, the control update times for the system are based on the event-triggered control implementation presented in Section II-A and are not known a priori. It is the sensors that decide on-the-fly when it is time to update the controller with fresh measurements by evaluating (4). While event-based implementations are certainly robust to disturbances, as there is a continuous supervision of the state of the plant, and reduce the amount of measurements that nodes need to transmit, they also have a couple of clear shortcomings. First, the continuous supervision of the triggering condition imposes the availability of specific hardware dedicated to this task; and second, as the update times are not available a priori there is no possibility of implementing any dynamic scheduling policy. In order to guarantee the reliable communication between sensor, controller and actuator we propose the use of a TDMA MAC, with a fixed scheduling policy for the assignment of communication slots to the sensors.

B. Predictive Communication

Motivated by the fact that event-based communication imposes the restrictions of continuous supervision of the triggering condition as well as a fixed scheduling policy, we propose a predictive communication mechanism. Predictive implementations utilize the self-triggered control technique presented

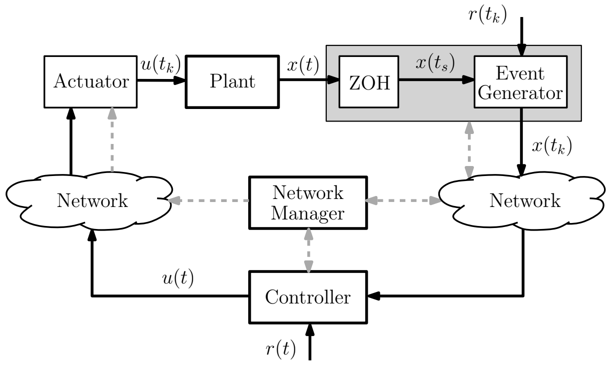


Fig. 2: System architecture for an aperiodic networked control loop.

in Section II-B where the controller will be responsible for computing the value $\tilde{g}_S(x(t_k))$ and calculating t_k in (8) for all plants in the network and transmitting these values to the NM. After all values of t_k are transmitted, the scheduling algorithms proposed in Section IV-C2 are executed at the NM. This node then informs all the sensor, controller and actuator nodes of the message transmission/reception slots. We remark that the sensor node may also compute t_k, and transmit this information to the centralized unit which performs the scheduling. A drawback of this mechanism when compared to event-based communications is that it is in general less robust to disturbances due to the fact that sensor nodes in between transmissions are set to a sleep mode in order to save battery life. If a disturbance affects the plant within this interval, no rejection takes place until a new sensor transmission. This is the motivation for the proposed hybrid communication mechanism proposed next.

C. Hybrid Communication

In order to extract as many of the benefits of the two previous mechanisms, we propose the use of a mechanism operating essentially as in the Predictive communication case, but with the addition of a number of slots in the communication schedule not assigned a priori to any sensor. These extra communication slots, are used to attempt transmissions on an event-triggered fashion when a disturbance (not accounted for in the self-triggered technique) takes place. The access to these slots will be done relying on a contention based mechanism, as supported by IEEE 802.15.4. Thus, no guarantees on channel access can be provided for event-based transmissions, which means that any strict performance guarantees are only provided by the predictive communication side, while the event-based part of this hybrid mechanism provides a best-effort approach to reduce latency when responding to disturbances.

IV. Communications Implementation

A. IEEE 802.15.4 MAC

For the implementation of the wireless communications we build upon the IEEE 802.15.4 MAC layer [31], which is used in some of the proposed protocols for control over wireless, *e.g.* WirelessHART [32]. What makes this protocol interesting for our purposes is the capability of this MAC layer to integrate both guaranteed slots and contention based slots in a single scheme. We focus on the beacon-enabled mode MAC specified in the standard. In such a setup, a centralized coordinator node, the NM, is responsible for synchronizing and configuring all the nodes in the network. The synchronization and configuration messages take place periodically at each *beacon message* which defines the time bounds of the *superframe* structure defined by the protocol. We denote by Γ_i the time instants at which the beacon is transmitted. The superframe length is named *Beacon Interval* (B.I.). The B.I. is further divided in *active* and *inactive* periods, as shown in Fig. 3. The active period has a time interval defined by *Superframe Duration* (S.D.) and is divided in 16 equally sized slots. The *active period* is further split into a Contention Access Period (CAP) and a Collision Free Period (CFP). During the CAP, nodes transmit *best effort* messages where the MAC scheme is Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA). On the other hand, the CFP is intended to provide real-time guaranteed service, by allocating *Guaranteed Time Slots* (GTS) to the nodes using a Time Division Multiple Access (TDMA) scheme. Since during the CFP there are no packet losses due to collisions or channel congestion, this mechanism is an attractive feature for time-sensitive wireless applications, as is the case of real-time control of several plants over a wireless network. The total number of GTS slots is limited in the standard to seven. The standard specifies a scheduling of the GTS by the NM following a *first-come-first-served* (FCFS) request-based scheme. At each CAP, the nodes requiring a GTS send a request to the NM which will allocate the slot to the node if there are available GTSs. An *inactive period* is defined in the end of the active period so that the network nodes and the NM enter a low-power mode and save energy.

TABLE I: Current consumption (mA) of a Telos wireless platform for different operation modes.

Mode	Description	Measured Current
1	μC active, Radio Tx	21.7 mA
2	μC active, Radio Rx/listen	22.8 mA
3	μC active, Radio OFF	2.4 mA
4	μC idle, Radio OFF	40 μA

B. MAC Limitations

Sensors are assumed to be battery powered, and so we aim at maximizing their life span by reducing their power consumption. In order to design an energy efficient communication, we look at how energy is spent in typical wireless nodes. The power consumption of the widely used wireless sensor platforms Telos [36] is given in Table I [37]. The table clearly shows that communications are very power expensive, and moreover, that the cost of listening and receiving messages is even more expensive than transmitting.

Naturally, to save large amounts of energy the nodes should keep their radios off and the microcontroller (μC) idle for as long as possible. This is achieved in two ways: reducing the communication slot size to reduce listening, reception and transmission times; and increasing the length of the inactive

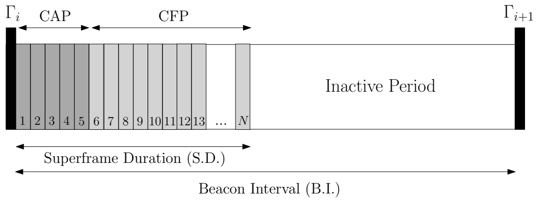


Fig. 3: Superframe structure of the modified IEEE 802.15.4 MAC.

period. In the standard two parameters are available to adjust the structure of the superframes are the S.D. and the B.I., and the rest of the parameters are inferred from these two.

While the standard covers most of our needs, it also poses some limitations which we address through the following modifications:

1) Allowing the number of slots assigned to CAP and CFP to be a free design parameter, as opposed to the maximum of 7 GTSs in the CFP, and a total maximum of 16 slots during CAP and CFP imposed by the IEEE 802.15.4 MAC standard. Moreover, we also allow the complete removal of the inactive period, which is not contemplated by the standard. These changes provide the desired increased versatility in adjusting latency in channel access and slot-size adjustments.
2) The use of a scheduling mechanism at the NM for the assignment of GTS slots instead of the *first-come-first-served* (FCFS) request-based scheme in the IEEE 802.15.4 MAC specification.

Note that these modifications do not require changing the overall structure of the IEEE 802.15.4 MAC protocol, but only adjustments on the maximum allowed number of superframe slots, and the GTSs scheduling methodology. The modified superframe structure is presented in Fig. 3. For more details on the GTS modification and implementation details see [38].

C. Proposed Implementation

In this section we present the scheduling policies for the three proposed communication mechanisms.

1) Event-based Communication: Under the event-based implementation, the time instants at which communication needs to occur are not known beforehand. We propose the use of a static TDMA scheme in which GTSs are reserved for each control loop in the network. Communication is then established at a particular GTS (assigned to sensor node i) only if the corresponding condition (4) is not satisfied for node i. This TDMA mechanism may introduce a delay between event generation and transmission of measurements, as a sensor may have to wait a certain time from the triggering of an event until a transmission in its assigned slot. Nevertheless, this delay is bounded and can be accommodated as presented in Section II-C.

In order to choose the B.I. length and the number of GTSs assigned to each control loop, the worst-case inter-event time for each triggering condition needs to be considered. The time between two consecutive slots assigned to node i must be lower than the minimum inter-event time τ_{min} (given by (9)) of the condition of node i. This guarantees that no more than one transmission is needed for node i between two consecutive GTSs allocated to node i. Notice that, since the schedule is static, the GTSs that are not used cannot be reassigned to other nodes. The next mechanism overcomes this drawback.

2) Predictive Communication: A dynamic scheduling scheme can be used in the case of predictive implementations since a future control update time can be known in advance. In order to allow for efficient usage of the available network resources, we propose to schedule the messages in the network according to an Earliest Deadline First (EDF) approach, which is known to be optimal for time-constrained schedules [39]. Under this policy, the NM reads the estimate of the next event time τ_{k+1} for node i (given by $\tilde{g}_S$) and assigns a GTS to this node so that its message is sent no later than τ_{k+1}. In other words, the event time represents the deadline for the control-related message. Notice that the event times have to be adjusted since transmissions can only occur during the CAP.

As in the case of event-based implementations, τ_{min} has to be considered to choose the length of B.I. in order to guarantee schedulability of all the messages under worst-case conditions. For space reasons, details on the schedulability conditions are omitted and we refer the interested reader to [40].

3) Hybrid Communication: The hybrid communication mechanism utilizes the same scheduling policy defined for the predictive implementation, where GTSs are scheduled in an EDF fashion. Additionally, no GTSs are provided for the event-triggered messages since these are only granted access during the CAP, for ”best-effort” data transmission.

V. Experimental Setup

In order to evaluate the performance of the proposed aperiodic communication mechanisms, we built a lab process with a wireless network shared by two control loops and several independent monitoring nodes transmitting auxiliary messages, with no hard deadlines. The control loops are regulating two double-tank processes [41], where the tanks are collocated with the sensors and actuators and communicate wirelessly with a controller node. Fig. 4 shows the setup of two double-tank systems and eight independent monitoring nodes. Each double-tank is composed of one sensor and one actuator node. The controller node is also the NM in our setup. A scheduler node is added and connected to the NM. This unit performs scheduling computations for each mechanism, reducing the computation load of the NM.

A. Double-Tank System

The double-tank system consists of a pump, a water basin and two tanks of uniform cross sections. Fig. 4 depicts the experimental apparatus and a diagram of the physical system. The liquid in the lower tank flows to the water basin. A pump is responsible for pumping water from the water basin to the upper tank, which flows to the lower tank. The sensing of the water levels is performed by pressure sensors placed under each tank. One wireless sensor node interfaces the sensors with an ADC, in order to sample the pressure sensor values

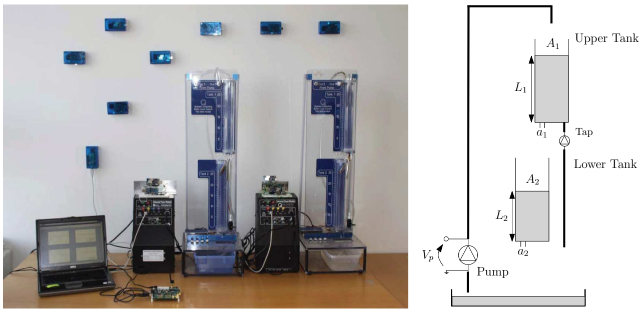


Fig. 4: WSAN setup with double-tank system and monitoring nodes. Block diagram of a double-tank system.

for both tanks. The plant actuation is made through the DAC of the wireless actuator node which actuates in the pump motor.

The linear model of the double-tank system obtained by linearizing its nonlinear model [41] around an operating point L_{10}, L_{20} is given by:

$$\begin{aligned}\Delta \dot{L}_1 &= -\frac{a_1}{A_1}\sqrt{\frac{g}{2L_{10}}}\Delta L_1 + \frac{K_p}{A_1}\Delta V_p \\ \Delta \dot{L}_2 &= \frac{a_1}{A_2}\sqrt{\frac{g}{2L_{10}}}\Delta L_1 - \frac{a_2}{A_2}\sqrt{\frac{g}{2L_{20}}}\Delta L_2,\end{aligned} \tag{11}$$

where, a_i is the outflow diameter of upper and lower tanks, A_i is the diameter of the upper and lower tanks, g is the gravitational acceleration in cm/s^2, V_p is the voltage applied to the pump motor, K_p is the pump motor constant, and L_1 and L_2 are the height of the water in the upper and lower tanks, respectively. Additionally, $\Delta L_1 = L_1 - L_{10}$, $\Delta L_2 = L_2 - L_{20}$ and $\Delta V_p = V_p - V_{p_0}$ represent the incremental values of the state and the input with respect to the operating point.

The goal of the experiment is to control the water level of the lower tank L_2 by adjusting the motor voltage V_p accordingly. Tracking of a reference signal $r(t)$ can be achieved by using feedforward tracking, with the control input defined as,

$$u(t) = Kx(t) + Mr(t), \tag{12}$$

where $x = [\Delta L_1, \Delta L_2]$ and the state-feedback matrix K is assumed to be chosen so that the closed-loop system matrix $\bar{A} = A - BK$ is Hurwitz. Matrix M is calculated to ensure setpoint tracking of the undisturbed closed-loop system for a constant command signal $r(t) = \bar{r}$.

In order to apply the aperiodic implementations proposed earlier, the state $x(t)$ must be converging to the origin and $u(t)$ must be a state-feedback controller. This is achieved by shifting the system's origin to the reference value we wish to track. If we assume that the reference is constant, we have the new continuous-time state-space system:

$$\begin{aligned}\Delta \dot{x} &= A\Delta x + B\Delta u, \\ \Delta u &= K\Delta x,\end{aligned} \tag{13}$$

for appropriate values of A and B, where $\Delta x(t) = x(t) - \bar{r} = [\Delta L_1 - \Delta L_{1\text{ref}}, \Delta L_2 - \Delta L_{2\text{ref}}]$, and $\Delta L_{i\text{ref}}$, for all $i = 1, 2$, is the steady-state value of the upper and lower tank and $\Delta u(t) = u(t) - M\bar{r}$, which achieves $\lim_{t\to+\infty} \|\Delta x(t)\| = 0$. By selecting the desired reference value of the lower tank $\Delta L_{2\text{ref}}$, the upper tank reference $\Delta L_{1\text{ref}}$ follows by solving the state-space equations in steady-state, i.e. $\Delta \dot{x} = 0$.

TABLE II: MAC Parameters

Scheme	CAP	CFP
Event-based	1	10
Predictive	1	10
Hybrid	7	4

In order to guarantee robustness of the predictive scheme with respect to disturbances [34], an upper bound on the inter-sampling times t_i needs to be imposed. We fix that bound in 10 s. The performance function $S(t)$ in (5) is defined by $R = 0.1Q$ and Q is selected as the identity matrix, for all mechanisms. The state-feedback matrix $K = \begin{pmatrix} -0.39 & -0.95 \end{pmatrix}$. We compute the minimum inter-transmission time τ_{min} for this system (9) using Lemma 4.1 in [34], which, for this physical system, gives a minimum time $\tau_{min} = 1$ s. Hence, the inter-transmission times τ_k for the control-related messages will be in the range $[1, 10]$ s. However, these times may be (conservatively) adjusted to be allocated at a GTS.

A periodic implementation of the control loops is implemented for comparison purposes. The sampling period of the periodic implementation is set to 0.64 s (the closest value from below to 1s that the protocol allows), given by (9), since stability has to be guaranteed under all possible conditions.

B. Communication Network

The wireless sensor platform chosen for this experiment is the Telos platform [36]. These nodes are equipped with a 250kbps 2.4 GHz Chipcon CC2420 IEEE 802.15.4 compliant radio and on-board sensors. The operating system used is TinyOS [42]. The implementation of the protocol used in our setup is based on [43] with the modification presented in Section IV-B, and detailed in [38].

MAC parameters: The number of CAP slots and GTSs are defined in Table II for each of the communication mechanisms. Moreover, S.D.= 323.1ms, B.I.= 646.3ms and each of the 11 slots has a duration of 29.4ms. When performing feedback control of the double-tank system, two slots for sensing and actuation communication must be defined for the mechanisms. Therefore, the minimum number of GTSs to be defined must be larger or equal to four.

Auxiliary messages from monitoring nodes are scheduled during the available GTSs for the event-based and predictive mechanisms, while in the hybrid mechanism monitoring nodes communication takes place during the CAP.

VI. Experimental Results

We now provide a description of the performed experiments and the obtained results with respect to control performance and energy efficiency as well as the network bandwidth utilization of the proposed implementations.

The initial water level of the lower tank is set to 5 cm and the reference $\bar{r} = 10$. An input disturbance of magnitude 1 V is applied continuously to the pump actuation starting at time $t = 130$s which will allow an analysis on how well each mechanism rejects disturbances. The experiment has a

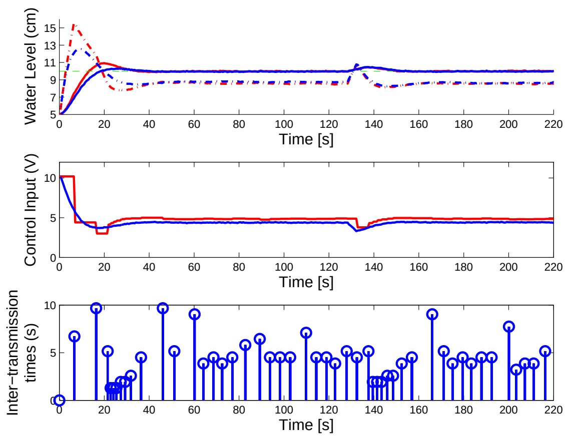


Fig. 5: Experimental results for event-based communication. The upper plot depicts the evolution of the water tank level and middle plot the control input values for event-based mechanism (red) and a periodic controller (blue). Upper (dash-dotted line) and lower (solid line) water levels are presented.

total duration of 220 s. The expected upper bound on the communication and computation delay is set to $\Delta_t = 35$ ms and will be compensated by each communication mechanism.

A. Control Performance and Energy Consumption

The control performance and energy efficiency is evaluated for one double-tank system. With respect to control performance, we analyze both the time response of the water levels and the Integral of the Absolute Error (IAE) of the lower tank water level, which is calculated as IAE $= \int_0^\infty |r - \Delta L_2(s)| ds$. The energy efficiency of each communication mechanism is given by the wireless sensor battery lifetime expectation. For this calculation we sum the total current consumption of the wireless node over the complete experiment and repeat it until a full consumption of 2900 mAh of battery capacity.

1) Event-Based Communication: The event-based communication mechanism is implemented with $T = 0.64$s as a new measurement is acquired 5ms before the start of the GTS allocated to the sensor. In this case, the time delay in the access of the communication channel will be $\Delta_a = 0.64$s as this is the time between consecutive GTSs allocated to the same control loop. Fig. 5 shows the time response and inter-transmission times of one double-tank system for event-based and periodic mechanisms. It is observed that both control implementations track the reference signal with similar behavior. Table III depicts the values for the IAE, number of transmissions and battery life span of the wireless sensor nodes. The IAE analysis show that the event-based scheme outperforms the periodic implementation. A faster rise time is achieved, followed by a fine adjustment of the water flow when closer to the reference. Additionally, the disturbance is efficiently rejected. Even though the number of transmissions of the event-based scheme is only 14.1% of the periodic, the battery lifetime increase is of 54.4% and not 700% as it could be expected if only the number of transmissions would consume energy. This difference originates from the fact that

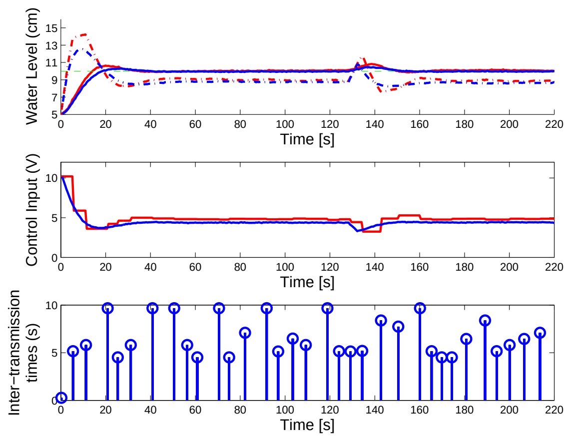


Fig. 6: Experimental results for predictive communication. Signal legends are shown as in Fig. 5.

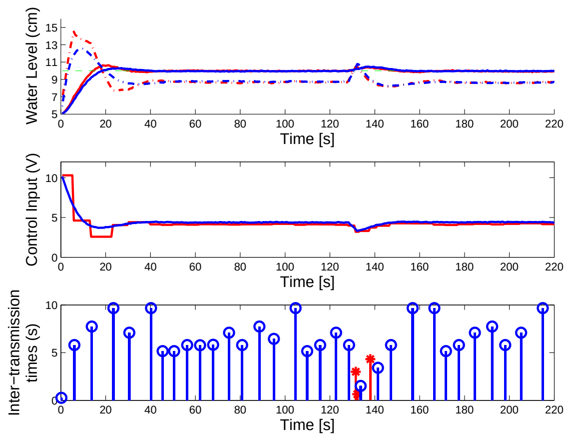


Fig. 7: Experimental results for hybrid communication. Signal legends are shown as in Fig. 5. The red star marker represents an event-based transmission and the blue round marker represents a predictive transmission.

the wireless nodes still need to turn on the radio and μC to receive beacon messages at each B.I., and spend energy during the inactive period. Thus, high reductions in the number of transmissions do not imply the same ratio of energy savings.

2) Predictive Communication: In order to provide the same performance guarantees as the event-based mechanism, the predictive mechanisms is implemented with $T = 0.64$s. Fig. 6 shows the time response and inter-transmission times of the double-tank system for the predictive and periodic mechanisms. As in the previous case, both control implementations track the reference signal with similar behavior. From Table III, IAE analysis show that the predictive scheme outperforms the periodic and event-based schemes during the transient but has a much worse performance when rejecting the disturbance. This occurs due to the fact that the sensor node is only active at scheduled transmission times, and not at every superframe as the periodic and event-based schemes. Since the event-based mechanism and predictive share the same performance criterion, both are expected in theory to have similar behavior, in the absence of disturbances. Different performances before the introduction of the disturbance can be explained by model inaccuracies and noise affecting the real plant. The number of transmissions using this scheme is 9.8% of the periodic, and lower than the event-based mechanism. This can be explained using the above arguments, due to the noise affecting the sensor readings. The battery lifetime is increased by 58.6% compared to the periodic scheme, while maintaining good control performances.

3) Hybrid Communication: The hybrid communication mechanism was implemented with two different measurement acquisition periods in its event-based component: a fast acquisition with $T = 10$ms (Hybrid$^{\text{F}}$) and slow acquisition every $T = 0.64$s (Hybrid$^{\text{S}}$). In the Hybrid$^{\text{F}}$ case, the event-based component will "continuously" check if (8) is violated throughout the whole superframe. On the other hand, in the slow implementation Hybrid$^{\text{S}}$, (8) is checked once during at the beginning of the CAP. Fig. 7 depicts the results for the implementation of Hybrid$^{\text{F}}$. The implementation of Hybrid$^{\text{S}}$ showed a similar behavior. Moreover, the hybrid mechanism tracks the reference signal and rejects the disturbance.

The inter-transmission times τ_{ki} are depicted for the case in which the transmission was generated by the event-based (blue circle) or the predictive mechanism (red star). As seen in the figure, only predictive transmissions take place during transient, and event-based transmissions occur during the disturbance rejection phase. From Table III, the IAE during the transient is kept close to the predictive scheme for both hybrid implementations as expected. The benefit of using the hybrid scheme become clear when the disturbance occurs. In this case, event-based transmissions occur when rejecting the disturbance. In addition, the Hybrid$^{\text{F}}$ has a very low battery life duration since the μC is kept computing for all times (mode 3 in Table I) and never sleeps. In the Hybrid$^{\text{S}}$ implementation, the battery life increases to the same levels as the other aperiodic schemes since the node is set to sleep if no transmission takes place. The number of transmissions of the Hybrid$^{\text{S}}$ is 10.1% of the periodic, lower than the event-based scheme and close to the predictive scheme. Note that a higher battery consumption is obtained by the hybrid mechanism when compared to a predictive scheme since the triggering condition is verified at every superframe.

B. Network Bandwidth Utilization

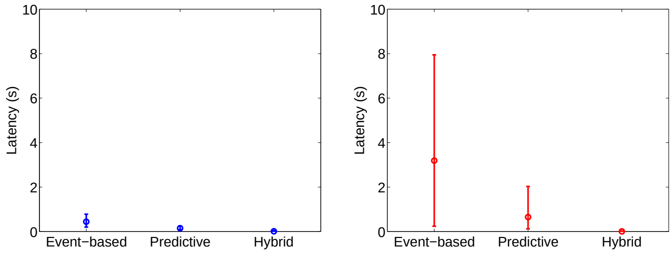
The network bandwidth utilization is characterized by how well the network is shared among the wireless nodes. To evaluate the network bandwidth utilization of each mechanism we define two message deadline types for the soft messages, which represent different traffic patterns that could be found in real WSANs. Each soft message has a size of 64 bytes.

- Slow traffic: Slow periodic transmissions, with period $T_m = 5$B.I. $= 3.3$ s, through the whole experiment.
- Bursty traffic: Fast periodic transmissions, with period $T_m = 1$B.I. $= 0.64$ s, during 25B.I. $= 16$ s, starting at $t = \{0, 120, 200\}$ s and slow periodic transmissions during the rest of experiment.

We analyze the latency experienced by these nodes in each of the mechanisms, where by latency we mean the time between

TABLE III: Performance evaluation of the proposed aperiodic mechanisms. The IAE performance indicator for different experiment phases, number of updates (N_{updates}) and battery life in days, are depicted for each of the mechanisms.

Scheme	IAE$^{[0,130]}$	IAE$^{[130,220]}$	IAE$^{[0,220]}$	N_{updates}	Battery life (days)
Event-based	75.78	12.48	88.26	49	879.84
Predictive	72.87	28.78	101.65	34	1010.18
Hybrid$^{\text{F}}$	67.05	16.51	83.55	36	63.66
Hybrid$^{\text{S}}$	68.19	16.87	85.06	35	910.14
Periodic	73.08	15.42	88.50	347	636.81



(a) Latency for Slow Traffic (b) Latency for Bursty Traffic

Fig. 8: Latency analysis of the monitoring wireless nodes, with respect to the traffic pattern. For each mechanism, the plot represents the minimum (lower bar), mean (round marker), and maximum delay (upper bar).

the each transmission is generated by the application and an acknowledgment was received for that particular message. Fig. 8 depicts the latency analysis for both traffic patterns. For each mechanism it is shown the minimum, maximum and mean value of the latency for all the eight nodes during the experiments. These values are the averages of three experimental runs. By using a bursty traffic pattern, the latency increases in all schemes, with a higher impact in the event-based mechanism as it is based on a static scheduling mechanism. In this case, the queue of soft messages is large, since only six slots of the superframe are available to be shared among eight monitoring nodes. The periodic scheme latency results are the same as the event-based scheme, since static scheduling is also performed. For the predictive scheme, the benefit of using a dynamic scheduling mechanism is clearly observed. By adjusting the GTS scheduling as a function of the control requirements, more space is available for the monitoring nodes to transmit. The hybrid mechanism was evaluated with eight, ten and twelve monitoring nodes, but no differences in the latency values were observed. This mechanism shows the most efficient network bandwidth utilization since its latency is low, when comparing to the other schemes and is in the interval $[2, 40]$ ms. Each node is able to transmit messages during the CAP, where several other nodes may attempt to transmit. However, no GTS scheduling queueing will occur in this case. Naturally, there is an advantage to allow monitoring nodes to transmit soft messages during the CAP, instead of GTSs.

VII. Conclusions

In this work we have proposed aperiodic implementations of control systems that are specially designed for WSANs. We provided a joint design of the aperiodic sampling technique, the wireless MAC protocol and a scheduling algorithm, that together guarantee a required control performance while efficiently using the network resources. In order to implement these mechanisms, we also identified limitations of the current IEEE 802.15.4 MAC protocol and propose slight modifications to increase its flexibility and enable our implementations.

Experimental results demonstrated the efficiency of the proposed communication mechanisms with respect to control and communication performance. All the mechanisms achieve set-point tracking and disturbance rejection, with closed-loop control performances close to the ones obtained with a traditional periodic paradigm. Finally, we illustrated how these improvements translate in terms of energy savings and network bandwidth utilization. While in the present paper we focus on the linear systems case, we remark that nonlinear systems can be addressed in a similar fashion [12], [15].

ACKNOWLEDGMENT

The authors would like to thank the late David Andreu, Aitor Hernandez and João Faria for their contributions to the development of the experimental testbed used in this work and the IEEE 802.15.4 MAC implementations in TinyOS.

REFERENCES

[1] V. Gungor and G. Hancke, “Industrial wireless sensor networks: Challenges, design principles, and technical approaches,” *Industrial Electronics, IEEE Transactions on*, vol. 56, no. 10, pp. 4258 –4265, oct. 2009.

[2] T. Sauter, S. Soucek, W. Kastner, and D. Dietrich, “The evolution of factory and building automation,” *Industrial Electronics Magazine, IEEE*, vol. 5, no. 3, pp. 35 –48, sept. 2011.

[3] P. Antsaklis and J. Baillieul, “Special issue on technology of networked control systems,” *Proceedings of the IEEE*, vol. 95, no. 1, 2007.

[4] J. Moyne and D. Tilbury, “The emergence of industrial control networks for manufacturing control, diagnostics, and safety data,” *Proceedings of the IEEE*, vol. 95, no. 1, pp. 29 –47, jan. 2007.

[5] M. Baldi, R. Giacomelli, and G. Marchetto, “Time-driven access and forwarding for industrial wireless multihop networks,” *Industrial Informatics, IEEE Transactions on*, vol. 5, no. 2, pp. 99 –112, may 2009.

[6] J. Chen, X. Cao, P. Cheng, Y. Xiao, and Y. Sun, “Distributed collaborative control for industrial automation with wireless sensor and actuator networks,” *Industrial Electronics, IEEE Transactions on*, vol. 57, no. 12, pp. 4219 –4230, dec. 2010.

[7] X. Cao, J. Chen, Y. Xiao, and Y. Sun, “Building-environment control with wireless sensor and actuator networks: Centralized versus distributed,” *Industrial Electronics, IEEE Transactions on*, vol. 57, no. 11, pp. 3596 – 3605, Nov 2010.

[8] A. Ulusoy, O. Gurbuz, and A. Onat, “Wireless model-based predictive networked control system over cooperative wireless network,” *Industrial Informatics, IEEE Transactions on*, vol. 7, no. 1, pp. 41 –51, feb. 2011.

[9] L. Palopoli, R. Passerone, and T. Rizano, “Scalable offline optimization of industrial wireless sensor networks,” *Industrial Informatics, IEEE Transactions on*, vol. 7, no. 2, pp. 328 – 339, May 2011.

[10] K. J. Åström and B. Bernhardsson, “Comparison of periodic and event based sampling for first-order stochastic systems,” in *Proceedings of the 14th IFAC world congress*, vol. 11, 1999, pp. 301–306.

[11] K.-E. Årzén, “A simple event-based pid controller,” in *Proc. 14th IFAC World Congress*, vol. 18, 1999, pp. 423–428.

[12] P. Tabuada, “Event-triggered real-time scheduling of stabilizing control tasks,” *Automatic Control, IEEE Transactions on*, vol. 52(9), pp. 1680–1685, 2007.

[13] W. Heemels, J. Sandee, and P. van den Bosch, “Analysis of event-driven controllers for linear systems,” *Int. J. of Control*, pp. 81(4), 571–590, 2008.

[14] X. Wang and M. Lemmon, “Self-triggered feedback control systems with finite-gain l2 stability,” *Automatic Control, IEEE Transactions on*, vol. 45, pp. 452–467, 2009.

[15] A. Anta and P. Tabuada, “To sample or not to sample: Self-triggered control for nonlinear systems,” *Automatic Control, IEEE Transactions on*, vol. 55, no. 9, pp. 2030–2042, 2010.

[16] J. Ploennigs, V. Vasyutynskyy, and K. Kabitzsch, “Comparative Study of Energy-Efficient Sampling Approaches for Wireless Control Networks,” *Industrial Informatics, IEEE Transactions on*, no. 99, p. 1, 2010.

[17] M. Mazo Jr. and P. Tabuada, “Decentralized event-triggered control over wireless sensor/actuator networks,” *Automatic Control, IEEE Transactions on*, vol. 56, no. 10, pp. 2456 –2461, oct. 2011.

[18] G. S. Seyboth, D. V. Dimarogonas, and K. H. Johansson, “Control of multi-agent systems via event-based communication,” in *World Congress*, vol. 18, no. 1, 2011, pp. 10 086–10 091.

[19] C. Ramesh, H. Sandberg, L. Bao, and K. H. Johansson, “On the dual effect in state-based scheduling of networked control systems,” in *American Control Conference (ACC), 2011*. IEEE, 2011, pp. 2216–2221.

[20] A. Cervin and T. Henningsson, “Scheduling of event-triggered controllers on a shared network,” in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, 2008, pp. 3601 –3606.

[21] E. Henriksson, D. E. Quevedo, H. Sandberg, and K. H. Johansson, “Self-triggered model predictive control for network scheduling and control,” in *Advanced Control of Chemical Processes*, vol. 8, no. 1, 2012, pp. 432–438.

[22] J. Yook, D. Tilbury, and N. Soparkar, “Trading computation for bandwidth: reducing communication in distributed control systems using state estimators,” *Control Systems Technology, IEEE Transactions on*, vol. 10, no. 4, pp. 503 –518, jul 2002.

[23] A. Camacho, P. Martí, M. Velasco, C. Lozoya, R. Villa, J. Fuertes, and E. Griful, “Self-triggered networked control systems: An experimental case study,” in *IEEE International Conference on Industrial Technology (ICIT)*, 2010, pp. 123–128.

[24] D. Lehmann and J. Lunze, “Extension and experimental evaluation of an event-based state-feedback approach,” *Control Engineering Practice*, vol. 19, no. 2, pp. 101 – 112, 2011.

[25] S. Trimpe and R. DAndrea, “An experimental demonstration of a distributed and event-based state estimation algorithm,” in *Proc. of the 18th IFAC World Congress*, 2011, pp. 8811–8818.

[26] K. Åström and B. Wittenmark, *Computer controlled systems*. Prentice Hall Englewood Cliffs, NJ, 1990.

[27] J. Akerberg, M. Gidlund, and M. Bjorkman, “Future research challenges in wireless sensor and actuator networks targeting industrial automation,” in *Industrial Informatics (INDIN), 2011 9th IEEE International Conference on*. IEEE, 2011, pp. 410–415.

[28] T. Samad, P. McLaughlin, and J. Lu, “System architecture for process automation: Review and trends,” *Journal of Process Control*, vol. 17, no. 3, pp. 191 – 201, 2007.

[29] A. Willig, “Recent and emerging topics in wireless industrial communications: A selection,” *Industrial Informatics, IEEE Transactions on*, vol. 4, no. 2, pp. 102 –124, May 2008.

[30] Socrades roadmap - the future of soa-based factory automation. 6th Framework Programme - European Commission. [Online]. Available: http://www.socrades.eu/Documents/objects/file1274836528.84

[31] IEEE 802.15.4, *IEEE 802.15.4 standard: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)*, IEEE, 2006. [Online]. Available: http://www.ieee802.org/15/pub/TG4.html

[32] HART Communication Foundation, *WirelessHART Data Sheet*, http://www.hartcomm2.org/hart_protocol/wireless_hart/wirelesshart_datasheet.pdf, 2007, datasheet.

[33] International Society of Automation, “ISA-SP100 wireless systems for automation website,” http://www.isa.org/isa100, 2010.

[34] M. Mazo Jr., A. Anta, and P. Tabuada, “An ISS self-triggered implementation of linear controllers,” *Automatica*, vol. 46, no. 8, pp. 1310–1314, 2010.

[35] X. Wang and M. Lemmon, “Event-triggering in distributed networked control systems,” *Automatic Control, IEEE Transactions on*, vol. 56, no. 3, pp. 586 –601, march 2011.

[36] J. Polastre, R. Szewczyk, and D. Culler, “Telos: enabling ultra-low power wireless research,” in *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*. IEEE, 2005, pp. 364–369.

[37] A. Prayati, C. Antonopoulos, T. Stoyanova, C. Koulamas, and G. Papadopoulos, “A modeling approach on the telosb wsn platform power consumption,” *J. Syst. Softw.*, vol. 83, no. 8, pp. 1355–1363, 2010.

[38] A. Hernandez, “Modification of the IEEE 802.15.4 implementation extended gts implementation,” KTH Royal Institute of Technology, Tech. Rep., July 2011. [Online]. Available: http://tinyos.cvs.sourceforge.net/viewvc/tinyos/tinyos-2.x-contrib/kth/index.html

[39] G. Buttazzo, *Hard real-time computing systems: predictable scheduling algorithms and applications*. Springer-Verlag New York Inc, 2005.

[40] J. Araújo, A. Anta, M. Mazo, J. Faria, A. Hernandez, P. Tabuada, and K. H. Johansson, "Self-triggered control over wireless sensor and actuator networks," in *Distributed Computing in Sensor Systems and Workshops (DCOSS), 2011 International Conference on*. IEEE, 2011, pp. 1–9.

[41] K. Åström and M. Lundh, "Lund control program combines theory with hands-on experience," *Control Systems, IEEE*, vol. 12, no. 3, pp. 22 –30, jun 1992.

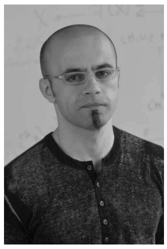
[42] P. Levis, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, A. Woo, D. Gay, J. Hill, M. Welsh, E. Brewer *et al.*, "TinyOS: An operating system for wireless sensor networks," *Ambient Intelligence*, 2004.

[43] A. Hernandez and P. Park, "Ieee 802.15. 4 implementation based on tkn15. 4 using tinyos," Technical report, KTH Electrical Engineering, Stockholm, Tech. Rep., 2011.

José Araújo received the M.Sc. degree in electrical and computer engineering with a specialization in control and robotics in 2008 from the Faculty of Engineering, University of Porto (FEUP), Portugal. He is currently working towards a Ph.D. degree in automatic control at KTH Royal Institute of Technology, Sweden. He has held visiting Ph.D. positions at the University of British Columbia (2008) and the University of California, Los Angeles (2012). His research interests include the design and implementation of network control systems and wireless systems.

Manuel Mazo Jr. is an assistant professor at the Delft Center for Systems and Control, Delft University of Technology (The Netherlands). He received the Ph.D. and M.Sc. degrees in Electrical Engineering from the University of California, Los Angeles, in 2010 and 2007 respectively. Previously he received a Telecommunications Engineering "Ingeniero" degree from the Polytechnic University of Madrid (Spain), and a "Civilingenjr" degree in Electrical Engineering from the Royal Institute of Technology (Sweden), both in 2003. Between 2010 and 2012 he held a joint post-doctoral position at the University of Groningen (The Netherlands) and the innovation centre INCAS3 (The Netherlands). His main research interest is the formal study of problems emerging in modern control system implementations, and in particular the study of networked control systems and the application of formal verification techniques to control.

Adolfo Anta (M11) received the Licenciatura degree from ICAI Engineering School, Madrid, Spain, in 2002 and the M.Sc. and Ph.D. degrees from the University of California at Los Angeles, in 2007 and 2010, respectively. From 2002 to 2005 he was a Design Engineer at EADS-Astrium, and from 2010 to 2012 he was a postdoctoral researcher at TU Berlin and the Max Planck Institute, Germany. He is a currently with General Electric Global Research Europe. His research interests include robust control, networked control systems, and power systems. Dr. Anta received the Fulbright Scholarship in 2005, the Alexander von Humboldt Fellowship in 2011, was a Finalist for the Student Best Paper Award at the IEEE Conference on Decision and Control in 2008, received the 2010 EMSOFT Best Paper Award and the IEEE CSS George S. Axelby Award in 2011.



Paulo Tabuada was born in Lisbon, Portugal, one year after the Carnation Revolution. He received his "Licenciatura" degree in Aerospace Engineering from Instituto Superior Tecnico, Lisbon, Portugal in 1998 and his Ph.D. degree in Electrical and Computer Engineering in 2002 from the Institute for Systems and Robotics, a private research institute associated with Instituto Superior Tecnico. Between January 2002 and July 2003 he was a postdoctoral researcher at the University of Pennsylvania. After spending three years at the University of Notre Dame, as an Assistant Professor, he joined the Electrical Engineering Department at the University of California, Los Angeles, where he established and directs the Cyber-Physical Systems Laboratory.

Paulo Tabuada's contributions to cyber-physical systems have been recognized by multiple awards including the NSF CAREER award in 2005, the Donald P. Eckman award in 2009 and the George S. Axelby award in 2011. In 2009 he co-chaired the International Conference Hybrid Systems: Computation and Control (HSCC'09) and in 2012 he was program co-chair for the 3rd IFAC Workshop on Distributed Estimation and Control in Networked Systems (NecSys'12). He also served on the editorial boar of the IEEE Embedded Systems Letters and the IEEE Transactions on Automatic Control. His latest book, on verification and control of hybrid systems, was published by Springer in 2009.

Karl H. Johansson is Director of the KTH ACCESS Linnaeus Centre and Professor at the School of Electrical Engineering, Royal Institute of Technology, Sweden. He is a Wallenberg Scholar and has held a six-year Senior Researcher Position with the Swedish Research Council. He is Director of the Stockholm Strategic Research Area ICT The Next Generation. He received MSc and PhD degrees in Electrical Engineering from Lund University. He has held visiting positions at UC Berkeley (1998-2000) and California Institute of Technology (2006-2007). His research interests are in networked control systems, hybrid and embedded system, and applications in smart transportation, energy, and automation systems. He has been a member of the IEEE Control Systems Society Board of Governors and the Chair of the IFAC Technical Committee on Networked Systems. He has been on the Editorial Boards of several journals, including Automatica, IEEE Transactions on Automatic Control, and IET Control Theory and Applications. He has been Guest Editor for special issues, including the one on "Wireless Sensor and Actuator Networks" of IEEE Transactions on Automatic Control 2011. He was the General Chair of the ACM/IEEE Cyber-Physical Systems Week 2010 in Stockholm and IPC Chair of many conferences. He has served on the Executive Committees of several European research projects in the area of networked embedded systems. In 2009, he received the Best Paper Award of the IEEE International Conference on Mobile Ad-hoc and Sensor Systems. In 2009, he was also awarded Wallenberg Scholar, as one of the first ten scholars from all sciences, by the Knut and Alice Wallenberg Foundation. He was awarded an Individual Grant for the Advancement of Research Leaders from the Swedish Foundation for Strategic Research in 2005. He received the triennial Young Author Prize from IFAC in 1996 and the Peccei Award from the International Institute of System Analysis, Austria, in 1993. He received Young Researcher Awards from Scania in 1996 and from Ericsson in 1998 and 1999. He is a Fellow of the IEEE.