CISTER
Research Center in
Real-Time & Embedded
Computing Systems

Technical Report

A Self-Adaptive Approximate Interpolation Scheme for Dense Sensing

Maryam Vahabi
Eduardo Tovar
Michele Albano

CISTER-TR-140512
Version:
Date:

A Self-Adaptive Approximate Interpolation Scheme for Dense Sensing

Maryam Vahabi, Eduardo Tovar, Michele Albano

CISTER Research Unit

Polytechnic Institute of Porto (ISEP-IPP)

Rua Dr. António Bernardino de Almeida, 431

4200-072 Porto

Portugal

Tel.: +351.22.8340509, Fax: +351.22.8340509

E-mail:

http://www.cister.isep.ipp.pt

Abstract

Very dense networks offer a better resolution of the physical world and therefore a better capability of detecting the occurrence of an event; this is of paramount importance for a number of industrial applications. However, the scale of such systems poses huge challenges in terms of interconnectivity andtimely data processing. In this paper we will look at efficient scalable data acquisition methods for such densely instrumented cyber-physical systems. Previous research works have proposed approaches for obtaining an interpolation of sensor readings from different sensor nodes. Those approaches are based ondominance protocols, presenting therefore excellent scalability properties for dense instrumented systems. In this paper we propose an important advance to the state-of-the-art. Our novel approach not only incorporates a physical model to enable more accurate approximate interpolations but it also detects and self-adapts to changes in the physical model.

A Self-Adaptive Approximate Interpolation Scheme for Dense Sensing

Maryam Vahabi, Eduardo Tovar, Michele Albano
CISTER/INESC-TEC, ISEP, Polytechnic Institute of Porto
Porto, Portugal
{mamvi, emt, mialb}@isep.ipp.pt

***Abstract*— Very dense networks offer a better resolution of the physical world and therefore a better capability of detecting the occurrence of an event; this is of paramount importance for a number of industrial applications. However, the scale of such systems poses huge challenges in terms of interconnectivity and timely data processing. In this paper we will look at efficient scalable data acquisition methods for such densely instrumented cyber-physical systems. Previous research works have proposed approaches for obtaining an interpolation of sensor readings from different sensor nodes. Those approaches are based on dominance protocols, presenting therefore excellent scalability properties for dense instrumented systems. In this paper we propose an important advance to the state-of-the-art. Our novel approach not only incorporates a physical model to enable more accurate approximate interpolations but it also detects and self-adapts to changes in the physical model.**

***Keywords*— Sensor Networks; Data Acquisition; Aggregate Quantities; Dominance-based MAC Protocols**

I. INTRODUCTION

Although the information technology transformation of the 20th century appeared revolutionary, a bigger change is in progress. The term Cyber-Physical Systems (CPS) has come to describe the research and technological efforts that will ultimately allow the interlinking of the real-world physical objects and the cyberspace efficiently [1]. The integration of physical processes and computing is not new. Embedded systems have been in place for a long time and these systems often combine physical processes with computing. The revolution is coming from massively deploying networked embedded computing devices allowing instrumenting the physical world with pervasive networks of sensor-rich embedded computation. As Moore's law continues, the cost of a single embedded computer equipped with sensing, processing and communication capabilities drops toward zero. This makes it economically feasible to densely deploy networks with very large quantities of such nodes.

Accordingly, it is possible to take a very large number of sensor readings from the physical world, compute quantities and take decisions out of them. Very dense networks offer a better resolution of the physical world and therefore a better capability of detecting the occurrence of an event; this is of paramount importance for a number of applications where high-spatial sensing (and actuation) resolution is needed.

Structural health monitoring (SHM) of physical infrastructures (bridges, aircrafts, etc.) is an example of CPS applications where high-spatial resolution sensing is required [2]. Other examples are the ongoing efforts within the aircraft industry to respond to environmental concerns by developing technologies that will allow sustained air travel growth while minimizing overall carbon footprint [3].

Let us elaborate a bit further on an Active Flow Control (AFC) application for drag reduction in aircraft.

II. INTERCONNECT CHALLENGES IN AFC

The drag breakdown of a commercial aircraft shows that the skin friction drag and the lift-induced drag constitute the two main sources of drag, approximately one half and one third of the total drag for a typical long range aircraft at cruise conditions [4].

Skin friction drag is therefore the main component of the aerodynamic drag. Skin friction arises from the friction of air against the skin of the aircraft moving through it. The primary source of skin friction drag during the flight is boundary layer separation. The boundary layer is the layer of air moving smoothly in the immediate vicinity of the aircraft (wing, fuselage, tail). The smooth flow (laminar flow) is disturbed by the boundary layer separating from the surface creating a low pressure region and, ultimately, increasing the skin friction drag (turbulent flow). Fig. 1 illustrates these.

There are various approaches to reduce the turbulent skin friction, involving different mechanisms, such as: reducing turbulent friction drag through riblets; deformable active skin using smart materials (compliant walls), or by (locally) postponing the boundary layer separation using vortex generators such as dimples or Synthetic Jet Actuators - SJAs. In this latter case, suction from the surface of the wing can be used to remove the low-energy air directly from the boundary layer. Along with this method, additional momentum can be

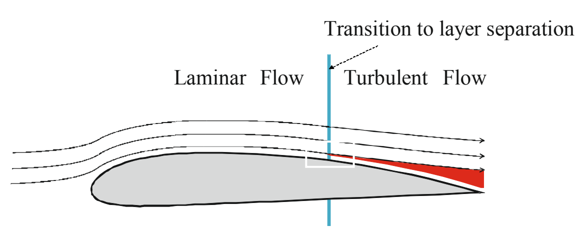


Fig. 1. Boundary layer separation exemplified with the wing.

achieved by generating streamwise vortices near the edge of the boundary layer that reenergize the boundary layer flow.

We aim at designing, validating and demonstrating a novel cyber-physical system able of performing an efficient suppression of the turbulent flow by using a dense sensing deployment to detect the low pressure region and a similarly dense deployment of actuators to manage the turbulent flow. With this concept, only the actuators in the vicinity of a separation layer are activated, minimizing power consumption and also the induced drag.

A recent research work [5] uses SJAs running at key positions on the wing to continuously energize the boundary layer and thus delay its separation. However, that approach does not use sensors to detect and trace the separation, and is therefore static and not proactive in nature. It results that the efficiency of active flow control (AFC) is compromised and energy resources are wasted when there is no boundary layer separation or when it lies outside the actuators' optimal control field.

We aim a smarter use of the actuators for AFC, and the SmartSkin project [6] ongoing at CISTER-ISEP is one such effort. Implementing AFC through this smart skin approach implies a reliable, highly fault tolerant network of active skin friction reduction components. Importantly, given the characteristics of the physical quantities to be tracked, sensors (e.g., pressure sensors, vibration sensors) might need to be only a few centimetres apart.

Therefore, even in the case of a medium-sized passenger aircraft, the sensor/actuator network will be composed of potentially hundreds of sensors, controllers and actuator systems (smart skin patches) that will be embedded across the aircraft wings and fuselage. Such densely instrumented active skin poses a huge challenge in terms of interconnectivity and timely data processing. We plan to develop novel sensor/actuator network paradigms and mechanisms able to deal efficiently with the large-scale processing requirements. Scalability will be a concern as well as timeliness.

We have already proposed algorithms for obtaining an interpolation of sensor readings from different sensor nodes, and those algorithms present excellent scalability properties for dense instrumented systems [7, 8].

III. BACKGROUND AND RELATED WORK

Applications of sensor and actuator (dense) networks behave typically as follows:

```
1: do forever
2:    each sensor node takes a new sensor reading;
3:    sensor nodes form a (potentially approximate)
          representation of all sensor readings;
4:    the representation of sensor readings is used, for
          example, to compute an actuation command.
```

The execution of line 3 in the above pseudo code requires an algorithm that acquires a (potentially approximate) representation of all sensor readings. Ideally, this algorithm should offer a small deviation of the representation of sensor reading as compared to the physical world.

A. Quantity Aggregation

Various interesting features of dominance-based protocols [9] (CAN [10] and WiDom [11, 12] are examples) can be exploited to obtain aggregate quantities in large scale dense networks, with a time-complexity that is very low and independent of the number of nodes. Such mechanisms are being used as a key building block in densely instrumented Cyber-Physical Systems as is discussed next.

By associating the priorities of messages to physical quantities (such as temperature, acceleration, luminosity, etc.), several high-performance algorithms for data processing can be devised in which the time-complexity is independent of the number of nodes.

For instance, if each node uses the value of its sensor reading instead of an arbitrary priority, the node winning the contention for the medium will be the one with the minimum (MIN) of the sensed values. In [8], it is demonstrated that CAN-enabled platforms can be used to compute various aggregate quantities, such as MIN (or MAX).

B. Approximate Interpolation

Previous work [8] proposed an algorithm for obtaining an interpolation of detected signals. The interpolation is a function $f(x, y)$ where x and y are space coordinates and the function $f(x, y)$ approximates sensor readings throughout the area of interest. The function $f(x, y)$ is represented by a set of control points, denoted as S, where each control point $q_k \in S$ has three attributes x_k, y_k and v_k, with the meaning that evaluating the interpolation at the location (x_k,y_k) should give the value v_k. On locations where no control point exists, the function $f(x, y)$ is defined as a weighted average of control points; this is called *weighted-average interpolation* (WAI). Formally, the function $f(x, y)$ is defined as:

$$f(x,y) = \begin{cases} 0 & if\ S = \emptyset \\ v_k & if\ \exists q_k \in S:\ x_k = x \wedge y_k = y \\ \frac{\sum_{k \in S} v_k \cdot w_k(x,y)}{\sum_{k \in S} w_k(x,y)} & otherwise \end{cases} \quad (1)$$

where weights, $w_k(x, y)$, are given by:

$$w_k(x,y) = \frac{1}{(x_k - x)^2 + (y_k - y)^2} \quad (2)$$

Let N_i denote a sensor node. Let (x_i,y_i) denote the location of this sensor node and let v_i denote the sensor reading of this sensor node. We let e_i denote the error of the interpolation at sensor node N_i and we let e denote the maximum error over all sensor nodes. Formally, we express this as:

$$e_i = |v_i - f(x_i, y_i)| \quad (3)$$

and

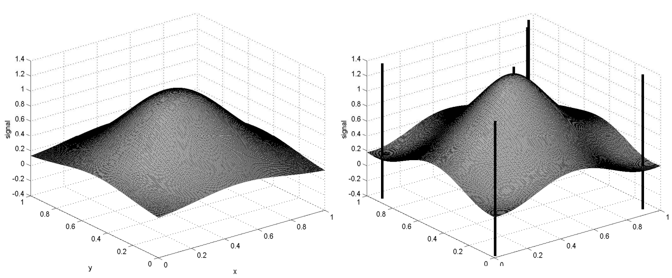
$$e = \max_{i=1..m} e_i \quad (4)$$

where m is the number of nodes.

An algorithm for efficiently constructing S is proposed in [8] (we refer to it as *Basic Interpolation Algorithm)*. The main idea is that initially the interpolation is zero on each location (this is represented by setting S to the empty set). Then, each sensor node evaluates the interpolation at its location and compares it with its sensor reading and the sensor node with the maximum error is granted the medium for transmitting its location and sensor reading, and this information is added to S. This is repeated k times (where the value of k is selected by the designer). Pseudo code for this algorithm is shown below (each sensor node executes the algorithm and a sensor node can read the variable i to obtain its identifier):

```
S ← ∅
for j ← 1 to k do
    calculate the interpolation function f(x_i,y_i) based on S
    calculate e_j
    select a sensor node N_k with the maximum e_k, that is
        e_k = e. This can  be achieved using the MAX (MIN)
        computation mentioned in Section IIIA
    the location and the sensor reading of N_k forms a control
        points; add this control point to S
end for
```

Fig. 2 illustrates the operation of the interpolation scheme. Fig. 2(a) shows how a physical quantity varies as a function of space coordinates x and y. Fig. 2(b) shows an interpolation which is an approximate representation of this physical quantity; the lines indicate the location of control points in S.



(a) Original Signal (b) Interpolated Signal

Fig. 2. Interpolation example (taken from [8]).

There are however some limitations in that algorithm. The basic interpolation algorithm does not consider the dynamics of the monitored signal. Therefore, when the speed of change in the monitored signal is much faster than the speed of execution of the interpolation algorithm, the first points taken into the set S become outdated and as result the approximate interpolation becomes inaccurate. Note that this problem is more acute when the monitored signal is more complex (the one shown in Fig. 2 is relatively simple) and therefore requiring more control points to be added to the set S.

To address this problem, in [7] the authors proposed to embed a model of dynamics of the physical world into the algorithm: after the selection of a new control point, all previously selected control points in the set S update their three attributes (the x, y coordinates and the value v) according to the defined change model.

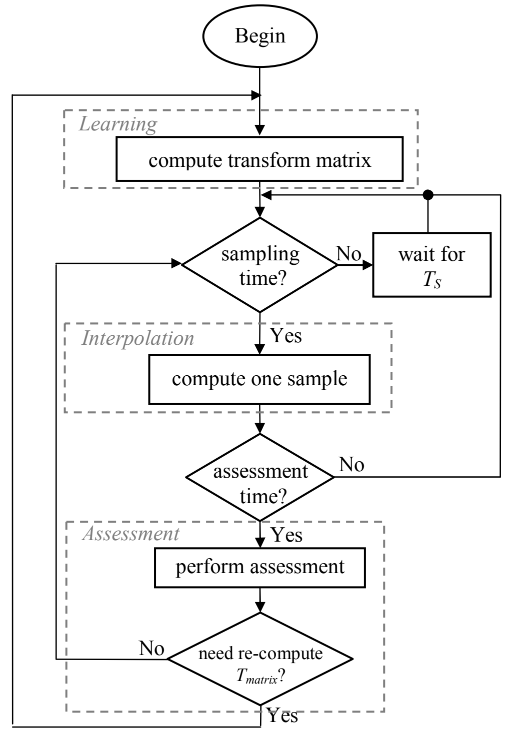


Fig. 3. Flow chart of the proposed algorithm.

That approach has still a number of significant limitations: (i) in that approach there is no provision on how to obtain the physical model parameters to be incorporated into the algorithm (potentially without having any prior knowledge about the signal behavior) and (ii) there is no provision on any feedback control of the quality of the approximate interpolation enabling the periodic re-computation of the physical model parameters that are embedded in the approximate interpolation algorithm.

IV. A Self-Adaptive Approx. Interpolation Scheme

In this paper we report ongoing research aiming at a more elaborated algorithm. Besides the approximate interpolation functionality (that performs compensation in the control points to accommodate the dynamics of the physical signal under observation) the novel algorithm also includes a functionality that is called *Learning*. Learning is performed at the beginning and every time need for re-computing the physical model parameters is determined by a third functionality that is called *Assessment*. The flow chart in Fig. 3 illustrates this approach. For simplification of terminology, we use the term “sample” to denote an approximate interpolation. The time between two consecutive samples is denoted by T_S.

A. Learning Mechanism

During Learning, a matrix that models the change pattern of the physical signal is computed; we call this matrix the transformation matrix (T_{matrix}). To define the parameters of the matrix, $T_{i,j}$, the trend of the signal change is observed and a system of equations based on this observation is solved. To do so, we need to define a time window during which the tracking is performed.

The term time-slot represents the time it takes to execute an iteration in the approximate interpolation (i.e. taking one control point and re-computing $f(x,y)$). As explained in the previous section, the node with highest error e_i can be found and added to the set S during one time-slot. By nullifying the set S (line 2 of the following pseudo code) at the beginning of each iteration, it is possible to determine the location of the node with the highest error (i.e. the location of the physical signal peak, assuming that the shape of the signal is preserved) in consecutive time-slots. Repeating five times this procedure provides sufficient information to solve a system of equations through which the T_{matrix} is computed. The following pseudo code summarizes the learning mechanism:

1: **for** $j \leftarrow 1$ to 5 **do**
2: $S \leftarrow \varnothing$
3: calculate the interpolation function $f(x_i,y_i)$ based on S
4: calculate e_j
5: select a sensor node N_k with the maximum e_k, that is $e_k = e$
6: save the location (x, y) and the sensor reading (v) of the control point and form a system of equations
7: **end for**
8: solve the system of equations to define the parameters of T_{matrix}, $T_{i,j}$ (further details on §4.1 of [13])

B. Interpolation

While performing the interpolation, our algorithm uses the T_{matrix} to update the data collected in the previous iteration. In particular, the algorithm selects a new control point at each time-slot, but before computing the approximate interpolation $f(x, y)$, all the previously selected points in the set S update their three attributes (x,y,v) by applying the affine transformation expressed by the T_{matrix} (see Equation (5), $xnew_i$, $ynew_i$ and $vnew_i$ are the new (updated) version of x coordinate, y coordinate and value of the control point q_i in the current time-slot).

$$\begin{cases} \mathrm{xnew_i} \leftarrow \mathrm{T_{1,1}} \times \mathrm{x_i} + \mathrm{T_{1,2}} \times \mathrm{y_i} + \mathrm{T_{1,3}} \times \mathrm{v_i} + \mathrm{T_{1,4}} \\ \mathrm{ynew_i} \leftarrow \mathrm{T_{2,1}} \times \mathrm{x_i} + \mathrm{T_{2,2}} \times \mathrm{y_i} + \mathrm{T_{2,3}} \times \mathrm{v_i} + \mathrm{T_{2,4}} \\ \mathrm{vnew_i} \leftarrow \mathrm{T_{3,1}} \times \mathrm{x_i} + \mathrm{T_{3,2}} \times \mathrm{y_i} + \mathrm{T_{3,3}} \times \mathrm{v_i} + \mathrm{T_{3,4}} \end{cases} \tag{5}$$

C. Assessment

The assessment functionality is performed with a periodicity, T_A, that is a multiple of T_S. During Assessment, network communications are TDMA like. A sub set of the control points, ζ, taken into the previous sample send their observed error value in their allocated TDMA-slot. The observed error of a control point is the difference between the measured value and the value of the interpolated signal in its locations. The following pseudo code illustrates Assessment:

Requirements:
(i) All nodes are aware of the start of the Assessment;
(ii) latest set S is preserved: $S=\{q_1, q_2, \ldots, q_k\}$;
for $j \leftarrow 1$ to ζ **do**
if $id == q_j.id$ **then**
compute the error and broadcast it
else
receive the packet and store the observed error value
endif
endfor
*re-compute_*T_{matrix} ← FALSE
if Average observed error is bigger than ε **then**
*re-compute_*T_{matrix} ← TRUE
endif

The first requirement listed in the above pseudo code is satisfied by utilizing the same synchronization technique applied in the prioritized MAC protocol [10-12], since all nodes agree on a common reference time to either start an iteration of interpolation or a TDMA-slot.

The observed error computed in Assessment is used to decide whether re-computing the transformation matrix or not. If the average observed error is bigger than a threshold value (ε), the current parameters of T_{matrix} are deemed as not valid anymore and the Learning mechanism will execute.

V. EVALUATION

Through simulations we studied the average error of the samples provided by the proposed self-adaptive approximate interpolation scheme. We consider a dense network with 2500 nodes deployed in square grid fashion over an area of 1m^2. For simplification we consider having an input physical signal that can be modeled by a *single peak middle* shape signal (similar to the one given in Fig. 2(a)) with the following Gaussian function:

$$p(x,y) = e^{-7((x-0.5)^2+(y-0.5)^2} + 0.1 \tag{6}$$

We consider in the experiments abrupt change patterns in the signal, expressed by the displacement of the whole signal up and down. We express the displacement in terms of relative speed compared to the duration of a time slot.

Fig. 4(A) shows the speed of the change during the simulation time (4000 time-slots). The positive values and negative values represent "movements" up and down, respectively. Note, as written previously, that the values change equally in all points (x_i, y_i) of the grid.

We run the simulation for 4000 time-slots and set the T_S to be 25 time-slots and T_A to be $3 \times T_S$, which means that a new sample is computed every 25 time-slots and the assessment runs every 75 time-slots. The number of control points is set to 6. In the assessment mechanism, we chose the first five control points that contributed in computing the latest interpolation sample (ζ=5).

Since we set T_A to be $3 \times T_S$, in the worst case, we may observe three samples with lower quality. In our described scenario, this happens for example after time-slot 100 when the change rate increases from 12% to 14%. Since the last assessment has occurred at time-slot 86, and the next assessment occurs at time-slot 161 (75 time-slot later), the samples taken between time-slot 100 to 161 are computed with an outdated T_{matrix}, hence we see three low quality samples in a row. After the assessment in time-slot 167 since the average observed error is not small enough (ε is chosen 0.01), all nodes

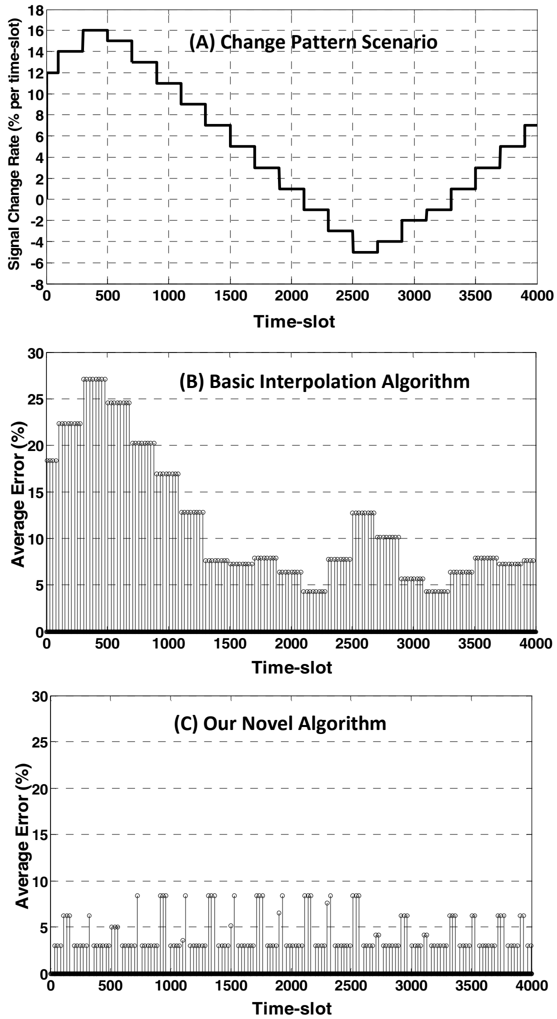


Fig. 4. (A) Change pattern scenario, (B) Average error with BIA and (C) Average error with our novel algorithm.

prepare to execute the learning mechanism in order to re-compute the T_{matrix}.

Our novel algorithm provides more accurate results (average error of about 3%), compared with the Basic Interpolation Algorithm (BIA). It is noteworthy to mention that for the simple physical signal with low change rate, the performance of the BIA algorithm is roughly the same as that of our proposed algorithm, but for more complex signals (where there is need to have more points in the set S for computing one sample) the new approach outperforms BIA even for slow change rates.

Besides, in worst case, and for this experiment, the novel algorithm provides samples with 9% as the largest average error, while BIA produces samples with an average error of 27%, especially when the signal change rate is high.

VI. Future Work

Ongoing work involves synthesizing such an algorithm from a model of the physical world by incorporating changes in the assessment rate as part of the feedback control as well as more sophisticated techniques to model more complex change patterns (not monotonous in the area of observation). Notably we will also address the implementation fit of those algorithms in real platforms. It is foreseeable that some tradeoffs (in accuracy) will need to be addressed to fit execution complexity.

Acknowledgment

This work was partially supported by National Funds through FCT (Portuguese Foundation for Science and Technology) and by ERDF (European Regional Development Fund) through COMPETE (Operational Programme 'Thematic Factors of Competitiveness'), within SMARTSKIN project (FCOMP-01-0124-FEDER-020312) and SENODS project (FCOMP-01-0124-FEDER-012988); also by FCT and ESF (European Social Fund) through POPH (Portuguese Human Potential Operational Program), under PhD grant SFRH/BD/67096/2009.

References

[1] J. A. Stankovic, I. Lee, A. Mok, and R. Rajkumar, “Opportunities and obligations for physical computing Systems” IEEE Computer, vol. 38, no. 11, pp. 23–31, November 2005.

[2] J. Caffrey, R. Govindan, E. Johnson, B. Krishnamachari, S. Masri, G. Sukhatme, K. Chintalapudi, K. Dantu, S. Rangwala, A. Sridharan, N. Xu, and M. Zuniga, “Networked Sensing for Structural Health Monitoring,” In: Proceedings of the 4th International Workshop on Structural Control, Columbia University, NY, June 2004.

[3] C. Dickey, “The Flying Prius”, Newsweek, July 2010.

[4] T. Washburn, “Airframe Drag/Weight Reduction Technologies”, Green Aviation Summit - Fuel Burn Reduction, NASA Ames Research Centre, September 2010.

[5] M. Watson, A. J. Jaworski, and N. J. Wood, “Application of Synthetic Jet Actuators for the Modification of the Characteristics of Separated Shear Layers on Slender Wings” Aeronautical J., vol. 111, no. 1122, pp. 519–529, Aug. 2007.

[6] E. Tovar, N. Pereira, I. Bate, L. Indrusiak, S. Penna, J. Negrão, J. Viana, F. Philipp, D. Mayer, J. Heras, F. Pacheco, and J. Loureiro, "Networked Embedded Systems for Active Flow Control in Aircraft", In the proceedings of the 11th RTN2012, Pisa, Italy, July 2012.

[7] A. Ehyaei, E. Tovar, N. Pereira, and B. Andersson, “Scalable Data Acquisition for Densely Instrumented Cyber-Physical Systems”, In proceedings of the ACM/IEEE Second International Conference on Cyber-Physical Systems (ICCPS 2011), pp. 174–183, 2011.

[8] B. Andersson, N. Pereira, W. Elmenreich, E. Tovar, F. Pacheco, and N. Cruz. “A scalable and efficient approach to obtain measurements in CAN-based control systems,” In IEEE Trans. Industrial Informatics vol. 4, no. 2, pp.80–91, 2008.

[9] A.K. Mok, and S. A. Ward, “Distributed Broadcast Channel Access,” Computer Networks, Vol. 3, November 1979.

[10] Bosch GmbH, Stuttgart, Germany. CAN Specification, ver. 2.0, 1991.

[11] B. Andersson, N. Pereira, and E. Tovar. “Widom: A dominance protocol for wireless medium access,” IEEE Transactions on Industrial Informatics, vol. 3, no. 2, May 2007.

[12] N. Pereira, R. Gomes, B. Andersson and E. Tovar, “Efficient aggregate computations in large-scale dense WSN,” in Real-Time and Embedded Technology and Applications Symposium (RTAS 2009), pp. 317–326, 2009.

[13] M. Vahabi, and E. Tovar, “Data Gathering Approach in Dense Sensor Networks", Technical Report, available at http://www.cister.isep.ipp.pt/docs/, Oct. 2012.