Self-Diagnosis and automatic configuration of smart components in advanced manufacturing systems

Rui Pinto, João Reis, Vitor Sousa, Ricardo Silva, Gil Gonçalves
Institute for Systems and Robotics
Faculty of Engineering of University of Porto
Porto, Portugal
e-mail: { rpinto , jpcreis , vdsousa , rps , gil }@fe.up.pt

***Abstract*—One of the key elements for the next generation of Intelligent Manufacturing is the capability of self-diagnosis, where the machinery used can itself report any breakdown or malfunction based on data, and self-reconfiguration as a way to improve responsiveness in case of sudden requirement changes, either by customer request or production line downtime. All these capabilities allow for quicker and improved systems reliability, leveraging the critical production phases as ramp-up, scheduled and unscheduled maintenance. Based on these premises, the main intent of the project Intelligent Reconfigurable Machines for Smart Plug&Produce Production (I-RAMP^3) is to develop innovative concepts such as NETwork-enabled DEVices (NETDEVs) acting as a technological shell to all the shop-floor equipment, converting it into an agent-like system and tackling the existing gaps between hardware and software for improving the European Industry.**

Keywords*—Wireless Sensor Networks; Intelligent Systems; Manufacturing Systems; Sensor Diagnostics and Validation; Over the air Programming.

I. Introduction

Considering the current European industrial panorama, there's still a discrepancy between the mass production hardware solutions and the easy access and monitoring of generated data using software implementations. The low abstraction level of controllers used nowadays in industry do not ease the integration with other existing solutions like Information Systems or other statistical analysis applications. Most of these systems rely on controllers and corresponding Human-Machine Interfaces (HMIs) to monitor the process, and no information is easily accessed for further analysis. Moreover, due to the low level of coding required for any change on these hardware solutions, it's hard to adapt to new production requirements and modify the process parameters. In terms of Assembly Line life-cycle, the impact of the previous constraints lead to high ramp-up times in the early stages of the production, as well as right after scheduled and unscheduled maintenance phases.

Based on these facts, the European Project I-RAMP^3 aims for developing innovative concepts for shop-floor devices' virtualization, enabling the easy access, process monitoring and control, as a way to foster the European Industry competitiveness. This virtualization is accomplished by using an agent-like concept called NETwork enabled-Device (NETDEV) with standardized communication, self-description of device's capabilities, negotiation techniques and plug'n'produce concept for easy device integration. The NETDEV entities explored in this document are divided into Device NETDEVs, which virtualize shop-floor machinery, and Sensor & Actuator (S&A) NETDEVs, which encapsulates shop-floor sensors or motes.

One of the key factors explored in I-RAMP^3 is the use of Wireless Sensor Networks (WSNs) and its capabilities for self-organization and self-diagnosis in the industrial domain. The flexibility in wireless communication, along with low energy consumption, reliable data acquisition and easy deployment *in situ* are just few of the benefits explored so far. Sensor data on a WSNs can be highly susceptible to errors due to external influences, communication conditions and network problems [18]-[21]. Together with the NETDEV encapsulation, sensors become self-aware and consequently diagnose themselves when a breakdown occurs.

Other intelligent feature about S&A NETDEVs is its self-reconfiguration capability. Since the Over-The-Air Programming (OTAP) concept was introduced to WSNs, updating a sensor node firmware on site turned out to be outdated and not efficient. This technology is used in the I-RAMP^3 project not to update or reconfigure a sensor node firmware, but to program from scratch a new one connected to the network with no measurement capabilities what so ever. The goal is to force the network to configure the new sensor node with sensing capabilities according to the task needed to be performed at the moment on the system.

The paper is organized in five different Sections. Section II talks about the latest advances of WSNs in the European industry, mostly implemented on the I-RAMP^3 project. Section III depicts the Sensor Data Validation techniques used in the present work, together with a quick overview about the latest applications of WSNs in various scenarios. Section IV presents the developments of OTAP and its detailed process in the I-RAMP^3. Then, an open discussion about the benefits of a NETDEV-like approach and all its embedded functionalities taking into consideration the end users of the system is presented in Section V. Ultimately, the present paper ends up with an acknowledgement and final remarks about the developed technology and its significant importance as a next step for intelligent manufacturing.

II. Wireless Wensor Network in Industry

Sensor usage on industrial applications has become extremely important, since monitoring the behavior of a machine is crucial to adapt its operation due to regular

 ISBN: 978-1-61208-437-4

changes on product demand. On a shop-floor environment, sensors should not be treated as an integrant part of a machine, but a separated component, which like complex machines, should be flexible enough to change its operations according to process demands. In I-RAMP^3 were explored new concepts on WSNs applied in industry, aiming for the addition of an intelligence layer on sensors, which empower them to be as complex as machines, both sharing plug'n'produce features and both capable of communicating with each other on an agent-like system environment.

Intelligent WSNs rely on some features such as easy integration of sensor nodes from different manufactures using, e.g., the PlugThings Framework [1] technology, along with automatic calculation of the nodes' physical location, self-diagnosis capabilities using sensor data validation methods, and self-reconfiguration capabilities using OTAP technologies to reprogram new sensor nodes on the network.

A. Sensor Integration

In the I-RAMP^3 project, the integration of multiple types of sensor nodes on the system is made using the PlugThings Framework, which contains a *Universal Gateway* (UG) to parse raw sensor data from the different sensor nodes. As can be seen in Figure 1, each sensor node of the network communicates directly to this gateway node, where the received measurements are processed and translated from raw data (stream of bytes) into readable form (measurement values). These data are compiled on Extensible Markup Language (XML) based format files that are part of the *Sensor & Actuator Abstraction Language* (SAAL), which is used to communicate with *Sensor & Actuator Abstraction Middleware* (SAAM), where all the intelligence related to the sensors is implemented. When the SAAM receives a new message from a sensor node, it will collect the sensor board identification number (ID) and the Media Access Control (MAC) Address that identifies the communication protocol. Both board ID and MAC Address are the unique identifier of a sensor node.

Joining a new sensor node to the network will imply the creation of a new S&A NETDEV corresponding to that sensor node, letting transparent to all the entities on the network what measuring tasks it can perform. Since a sensor node can have multiple sensors integrated, the corresponding S&A NETDEV will be able to perform different tasks related with the different sensor types of the sensor node. It will have one task per sensor integrated in the mote, being this way able to provide sensor information in a standardized way.

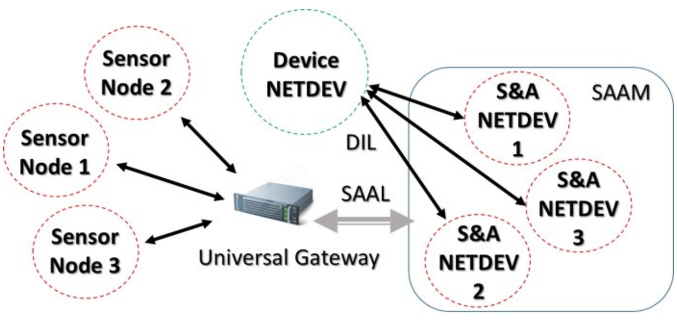

Figure 1. Sensor Integration on an I-RAMP^3 Environment

B. Device NETDEV and S&A NETDEV

NETDEV communication is accomplished by using a task-driven language - Device Integration Language (DIL) - which is composed by four main XML schemas: NETDEV Self-Description (NSD), which describes the capabilities that a specific NETDEV can perform, by defining conditions, goals and process parameters; Task Description Document (TDD), which is a request for task execution, specifying the conditions, goals, process parameters and the period of the task execution; Quality Result Document (QRD), which is the result of a task iteration, detailing the process quality; Task Fulfillment Document (TFD), which is an acknowledge document that represents the task finalization. A task request process is represented in Figure 2.

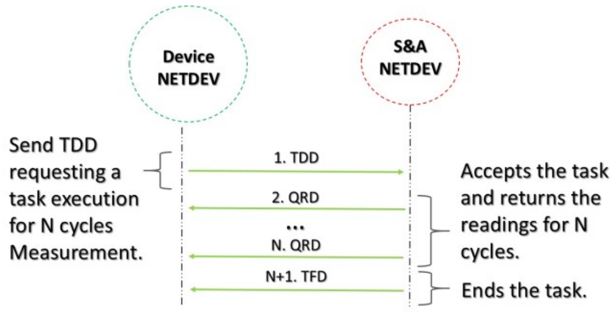

Figure 2. DIL Communication

This communication is initiated every time a NETDEV depends on one another to execute a task. Generally, S&A NETDEVs are requested by Device NETDEVs via TDD to execute tasks for sensing the environmental conditions during a given number of cycles. If the S&A NETEV is able to execute the requested task, it will successfully acknowledge the request by answering with QRDs containing the sensor data, during the number of cycles specified. A TFD will be sent back to the Device NETDEV denying the task if the S&A NETDEV is not capable of meeting the task goals and conditions or if it is already occupied executing a task for other NETDEV entity. A TFD is also used to acknowledge a successful task execution finalization.

III. Sensor Data Validation

Sensors are used at the shop-floor level to monitor the surrounding environmental and/or physical conditions of machines and all manufacturing components. The measured data will be used as an input for complex machines to control the manufacturing process and to adapt themselves according to these external conditions. This adaptation allows the machine to be flexible enough to change its variable inputs and internal processing, controlling the production process to maintain product quality despite fluctuations. Machine's process depends on data measured from sensors, so it's very important that these data stays the most reliable as possible when delivered to the machine. Data samples collected from sensors, especially from WSNs, are prone to be faulty due to internal and external influences, such as environmental effects,

 ISBN: 978-1-61208-437-4

limitations of resources, energy problems, hardware malfunctions, software problems, network issues, among others, as shown in [18]-[21]. Sensor data validation consists on a set of methods applied to the data provided by the sensors with the main goal of detecting anomalies and malfunctions on these sensors and take action accordingly on the corresponding S&A NETDEVs.

A. Methodology

Data validation methods are applied to data received from sensors. Finding deviations from normal sensor readings doesn't mean that they occur due to a malfunction of the sensor node, but rather due to an abnormal variation of conditions being measured. Despite being a sensor-based cause or a conditions-based cause, the WSN is self-aware and self-diagnosis of the task execution's process state.

Anomaly detection methods generally classify data into correct or faulty. There is no right method that works better than all the others and no method guarantees success, because they all depend on several factors such as type of monitored variable, the overall measurement conditions, the sensor used and the characteristics of the environment being perceived [2][15]. In [2][3] is proven that anomaly detection should not rely on just one method, but instead on a number of methods applied successively for detecting different types of data faults. Furthermore, there are methods [2] suitable to be used online, and other more complex and demanding on the processing level, suitable for offline validation, such as Bayesian Networks (BNs), Artificial Neural Networks (ANNs), Regression Techniques like Partial-Least Squares Regression, etc., used in many different contexts such as aerospace, energy, electric power systems, urban environment, among others [8]-[14]. Regarding S&A NETDEVs, techniques that provide a quick WSN diagnostics were used, such as Min/Max, Flat Line [3][5], Modified Z-Score [7] and No Value detection.

The Min/Max approach is based on a heuristic rule, which defines upper and lower bounds that refer to hardware specifications or/and conditions that are not likely to occur in the current context. Therefore, if sensed data is within bounds, data are likely good, otherwise, the sensor may be faulty. The Flat Line technique is based on temporal correlation of a big chunk of latest data collect. If the difference between successive data samples remains zero, this means that the sensor is probably faulty. Modified Z-Score is a statistical-based technique used as an outlier detection mechanism. It takes into account averaged values and deviations to assess if a certain value do not follows the same behavioral trend as the others. The No Value detection technique finds gaps in datasets. If the difference between the current time and the timestamp of the last measurement is unusually large, then probably the sensor has stopped the communication with the gateway.

B. Implementation

On I-RAMP[3], the sensor data validation is characterized by four main steps, as shown in Figure 3: 1) First, raw data is acquired from the sensor nodes; 2) Raw data is converted into a readable form by the UG and sent to the SAAM; 3) While a S&A NETDEV executes a task, the received sensor data is validated by a sequence of internal methods to detect anomalies; 4) If anomalies on data are detected, the corresponding S&A NETDEV is marked as probably faulty, which results, depending on the severity of the error detected, in the inability of accepting future task executions or termination of the current task's execution.

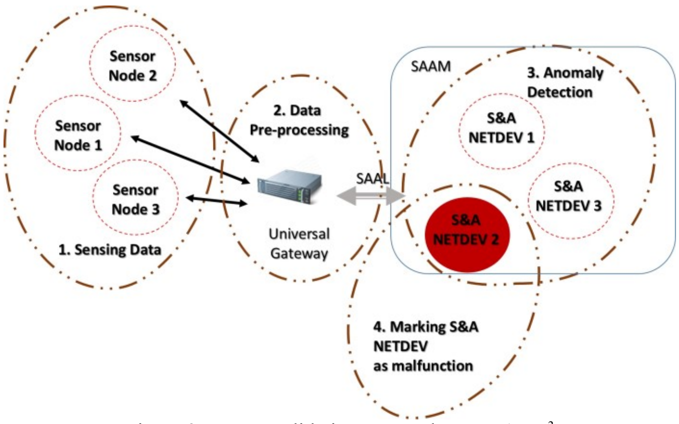

Figure 3. Data Validation approach on I-RAMP[3]

While the S&A NETDEV is executing a task, the dataset of the corresponding sensor node will go through two validation modules: Module A, which is intended for detecting sensor malfunctions and Module B, which is intended for detecting abnormal behavior from the sensor node.

Module A validates the received sensor data using Flat Line [3][5][17] and No Value detection methods, aiming to identify a malfunction sensor node. If Flat Line method returns positive for error detection, it means that, on the sensor node, the board is reading the same electrical quantity for an unusual amount of time, which means that the sensor doesn't detect any variation on the environment quantity being measured. Hence, it's most likely that the sensor is not correctly connected to the board. On the other hand, if the No Value method detects gaps in the dataset, most likely the battery as run out or the sensor node just broke down. Facing a malfunctioning sensor node, the corresponding S&A NETDEV is responsible to terminate prematurely the task execution, without any human interaction and making itself unavailable to take on other task requests. Module B is intended for methods that detect outliers, such as the Min/Max detection [3][5][16], which detects readings out of system limit thresholds, and the Modified Z-Score [4][5] that detects spikes and abnormal readings. This module returns a strong probability about the malfunctioning state of the sensor, despite lower than the one returned by Module A. This probability is based on the defective readings that, in this case, can be caused by sensor failing or abnormal behavior of the system itself. In such circumstances, the S&A NETDEV waits for the normal task termination to change its process state to unavailable (for future task executions), while a maintenance process doesn't occur on the corresponding sensor node.

 ISBN: 978-1-61208-437-4

IV. Over The Air Programming

OTAP is a technology developed originally to update firmware for mobile devices. Since the use of this type of equipment rely greatly on wireless internet access, OTAP has been used on the past years from manufactures and network operators to deliver firmware updates to equipment with internet access. However, because of the widely use of WSNs and the growing complexity of them, OTAP was taken to a new direction towards WSNs [22].

A WSN could have thousands of sensor nodes and the maintenance of these nodes could be very time-consuming. Therefore, since they must all be re-programmed one by one, this is not a very cost-effective solution. Moreover, the WSN may have nodes located in difficult access places, so updating firmware in sensor nodes on site can be challenging. Several sensor nodes from different manufactures are already embedded with the OTAP technology, which relies on updating firmware on sensor nodes from the gateway node, using the existing wireless communication between them, such as XBee, Wi-Fi or 3G.

A. *OTAP Methodology in I-RAMP³*

The WSN consists on different sensor nodes, gateway nodes connected to the UG and the communication topologies between them. The Sensor Nodes used in the I-RAMP³ that make the OTAP implementations possible are the Libelium Waspmote PRO (v1.2) [23] sensor boards with the XBee module for the 802.15.4 communication protocol [24]. Updating firmware on the Waspmote PRO (v1.2) requires using the Libelium OTA technology [6], which divides the OTAP process on two main steps: 1) Node discovery on the network and 2) Firmware upload. The OTA-Shell application [6] is used at the UG level to control the options available in OTA, sending commands to the sensor nodes to be reprogrammed. A firmware upload occurs when the shop floor operator replaces sensor node hardware due to a severe malfunction detection on a sensor node (using the methods discussed previously). The logical representation of OTAP methodology is depicted in Figure 4.

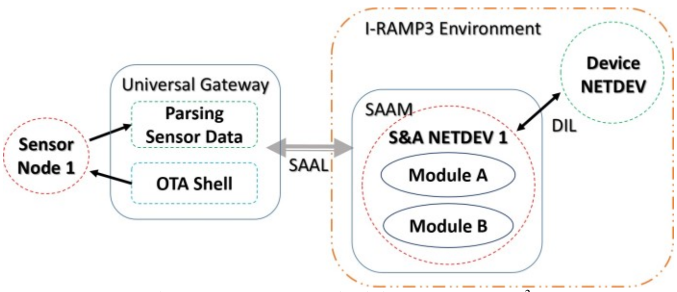

Figure 4. OTAP Methodology on I-RAMP³

When a S&A NETDEV is executing a task and a sensor node failure is detected, the malfunction could be caused by irreversible problems that require equipment replacement on the nodes, such as: 1) Replacement of the bad sensor/communication module; 2) Replacement of a bad sensor board; 3) Replacement of the entire sensor node. Since sensors and communication protocols are not directly related with the program that is running on the sensor node, 1) doesn't require firmware update of any kind. On the other hand, when 2) or 3) occurs, a firmware update is required, which can be done traditionally or using the OTAP approach.

Traditionally, before a new sensor board is connected, it needs previously to be manually programed with the right program. This approach may be counterproductive on a smart factory context, since the ramp-up time of replacing a sensor board could be very high. With the OTAP approach, when a new sensor board is connected, the sensor node is informed from the network of what to do, by being programmed automatically over the air. The basic idea is to previous store on the UG the replaced sensor node's program in form of an automatic generated binary image after compiling the code and program the new sensor node over the air with the stored binary image, replacing a malfunction one.

1) *Replacement of Faulty Sensor/Communication Module*

Malfunctions on the sensor node may have its root cause on specific components of the node, leading to the replacement of only the bad component. A malfunctioning S&A NETDEV detected by, e.g., a Flat Line could be possibly caused by a broken sensor that was used on the task execution requested and, therefore, the replacement process requires only the exchange of one sensor. On the other hand, if the malfunction is detected by, e.g., a No Value method, probably it is caused by problems on the communication protocol. The S&A NETDEV shuts down temporarily, until the component exchange is finished.

The moment a sensor or communication protocol replacement occurs and the sensor node is turned on, the S&A NETDEV will detect incoming readings from the same sensor board, as it used to, and associates this sensor node to the same S&A NETDEV making it available for task execution once again. If the communication protocol was replaced, the MAC Address associated with the S&A NETDEV is updated by the new one.

2) *Replacement of Sensor Board*

In the I-RAMP³ context, OTA is applied not for firmware update but for programming a new sensor board for the first time it joins the network, after replacing a failing sensor node. The process begins the moment a malfunction sensor node is detected during task execution, which imply replacing a failing sensor board, without exchanging the components connected to it, such as sensors and communication protocol. With the OTA approach, the shop floor operator avoids programming manually the new sensor board before it is connected to the system. The sensor board runs a program that sends to the gateway specific messages, meaning it is "*Alive*" and lacks contextual information, and waits to receive instructions for an OTA process. This "*Alive*" message is a defined string, containing information about the new sensor node, such as sensor board ID and MAC Address. Because only the sensor board is replaced, the MAC Address received on the "*Alive*" messages was already associated with an

 ISBN: 978-1-61208-437-4

existing S&A NETDEV, so the corresponding sensor ID will be updated by the new one. This means that the sensor board changes, but the virtual representation (S&A NETDEV) of the sensor node remains the same.

Since "*Alive*" messages are received instead of sensor readings and the sensor node is associated with an existing S&A NETDEV (due to the MAC Address), an OTAP process begins. First, SAAM identifies which program is the right one to be used for sensor programming via OTA, based on the previously created S&A NETDEV capabilities. Hence, SAAM orders the UG to start a new instance of the OTA Shell, using the identified binary image to program that specific sensor node. The UG runs the OTA Shell, which first scans the network to locate the new node to be programmed and sends the binary file to the identified node, which stores the file on the Secure Digital (SD) card. The sensor node reboots in order to start the execution of the new program, after receiving the program successfully. The program is copied from the SD card to the Flash Memory and the sensor node starts running the new binary file.

After restoring its configuration, the sensor node is ready to operate again, starting to measure and sending data to the corresponding S&A NETDEV, which changes its internal state becoming available for task execution.

3) Replace the Entire Sensor Node

The malfunctions detected may be severe to the point where none of the component on the sensor node can be saved, forcing the replacement of the entire node. When this happens, a new sensor node is connected to the network, which has a sensor node ID and MAC Address that are new in the system, resulting on the creation of a new S&A NETDEV, available to take requests for task execution.

The only way SAAM knows which tasks the new S&A NETDEV is able to perform, is by parsing the messages received from the sensor node and detect which are the sensor types connected to it. This occurs if the new sensor node is already programmed with the right program for the task pretended. On the other hand, if "*Alive*" messages are received, SAAM can't possibly know which tasks the new sensor node is able to perform, because it doesn't have any sensor readings and no background to associate the sensor node to an existing S&A NETDEV with capabilities already identified.

V. Discussion

The use of sensors in the industrial domain for condition-based monitoring and machine's parameterization always was a key element in the industrial domain. In most recent technological trends in manufacturing, machines need to act and adapt according to the environmental conditions to perform its tasks as reliable, effective and efficient as possible. Sensors generally assess not only the machine's condition by means of, e.g., temperature – essential to monitor the temperature of motors used in Linear Axis; Humidity – in sealing applications, the skin formation is driven by several parameters, and one of them is humidity; Luminosity – when using an optical sensor, most of the times is peremptory to calibrate the exposure time of the device in order to maximize the quality of image acquisition. Therefore, one of the cornerstones of I-RAMP^3 is to explore the applicability of WSNs and all the benefits it can bring to manufacturing environments, shielded with innovative concepts as NETDEVs enabling peer-to-peer device communication, and also plug'n'produce that shortens the time of device readiness to use.

Based on fact the WSN can take part on the intelligent manufacturing systems, two main functionalities were explored in this work. As previously explained, the use of reliable WSN compels the use of Sensor Data Validation techniques to assess the sensor functioning conditions and diagnose when there's a sensor breakdown for rapid responsiveness of the maintenance personnel, and consequently shorten the production system's down-time. As a way of shorten the ramp-up time after a sensor breakdown, the use of OTAP is explored to automatically program a mote becoming ready to use in a matter of seconds.

Assuming a perspective of a shop-floor Operator or Maintenance Engineer, the use of NETDEV entities like S&A NETDEV for sensor virtual encapsulation reveals to bring many benefits to the manufacturing environment. Since there are many dependencies from machine's execution and sensor readings, if there's not available an online functionality to permanently assess the reliability of sensors in case of faulty data, it can lead to machine damage and even put at risk the safety of shop-floor personnel. The use of NETDEV entities as a shell on the shop-floor devices can avoid this situations and also improve the knowledge about the process life-cycle. Moreover, in terms of System Design, since all these sensor validation techniques are embedded on the S&A NETDEV, it avoids the technical personnel to know in detail and implement the used techniques for sensor validation. In terms of integration, as previously explained, NETDEVs have a standardized way of communication, so easily any tool or software solution can interact with this system.

On the other hand, when a sensor is faulty, there's always the need to change the sensor for a new one, or schedule a downtime for sensor maintenance purposes. In the case of sensor exchange, being the most used practice due to cheap cost of motes when compared to machine, OTAP is a flexible and quick way of reverberate it. In terms of flushing a mote with the correspondent code for execution, the OTAP approach used in I-RAMP^3 is a totally automatic process that needs no physical direct interaction with the computer, which is most of the times made using a Universal Serial Bus (USB) cable, and not knowing which program is needed for flushing. On a shop-floor Operator's perspective, the only thing needed to perform sensor node programming is knowing which one is faulty, remove the mote, put a new one in the same position and switch it on. As described in Section IV, the process of mote identification in the network and program to be used to flush the mote is totally automatic. This way, only the physical mote removal and addition is necessary.

Additionally, and out of I-RAMP[3] bounds but still worth to highlight, is OTAP use to easily exchange the functionality of a sensor node with limited physical access. An example of that is changing the acquisition frequency of environmental conditions due to changes on customer requirement, having to a direct impact on the process. Another example is to automatically reprogram a mote to interpret different sensors physically connected with the sensor board. Once again, no manual flushing is needed and only the physical exchange of sensors is necessary.

VI. CONCLUSIONS AND FUTURE WORK

Automation and responsiveness always made part of the industrial mindset, where more flexibility and reliability means better production, better production means more customer confidence and more customer confidence means more competitive advantage toward more income.

I-RAMP[3] is a key enabler for this kind of philosophy since it aims to convert a production environment into an agent-like system, and therefore, to allow for machine-to-machine communication, self-aware capabilities due to NETDEV encapsulation becoming self-diagnosable and self-reconfigurable with all the functionalities developed so far in the project. The NETDEV concept, together with online and automatic Sensor Data Validation, and along with OTAP, are the main foundations that can turn the use of WSN in the industrial domain into a reality, paving the way for next generation of Factories of the Future. Based on all the afore mentioned topics, we can state that the use of sensors still have an important evolution to take place in industry, and all the technological baseline is being yield.

ACKNOWLEDGMENT

This research was supported by project I-RAMP[3] (FoF.NMP.2012-3) – Intelligent Network Devices for fast Ramp-up – funded by the European Commission under the Seventh Framework Programme for Research and Technological Development.

REFERENCES

[1] G. Gonçalves, J. Reis, R. Pinto, M. Alves, and J. Correia "A step forward on Intelligent Factories: A Smart Sensor-oriented approach." Emerging Technology and Factory Automation (ETFA), 2014 IEEE. IEEE, 2014. pp. 1-8.

[2] N. Branisavljevic, Z. Kapelan, and D. Prodanovic "Improved real-time data anomaly detection using context classification." Journal of Hydroinformatics, 13(3), 2011, pp. 307-323.

[3] A. B. Sharma, L. Golubchik, and R. Govindan "Sensor faults: Detection methods and prevalence in real-world datasets." ACM Transactions on Sensor Networks (TOSN), 6(3), 2010, p. 23.

[4] L. S. Jayashree, S. Arumugam, and A. R. Meenakshi "A communication-efficient framework for outlier-free data reporting in data-gathering sensor networks." International Journal of Network Management, 18(5), 2008, pp. 437-445.

[5] J. Ravichandran and A. I. Arulappan "Data Validation Algorithm for Wireless Sensor Networks." International Journal of Distributed Sensor Networks 2013 (2013).

[6] Libelium. [Online]. Available: http://www.libelium.com/uploads/2013/02/over_the_air_programming.pdf. [retrieved: September, 2015]

[7] B. Iglewicz and D. Hoaglin “How to detect and handle outliers”, The ASQC Basic References in Quality Control: Statistical Technique 16. Milwaukee, WI: American Society for Quality Control, 1993.

[8] O. J. Mengshoel, et al. "Probabilistic Model-Based Diagnosis: An Electrical Power System Case Study", Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transaction on, 40(5), 2010, pp. 874-885.

[9] O. J. Mengshoel, A. Darwiche, and S. Uckun "Sensor Validation using Bayesian Networks." In Proc. of the 9th International Symposium on Artificial Intelligence, Robotics, and Automation in Space (iSAIRAS-08), Los Angeles, CA, 2008.

[10] C. Jianhong, L. Hongkun, S. Deren, and L. Wei “A hybrid data-driven modeling method on sensor condition monitoring and fault diagnosis for power plants”, International Journal of Electrical Power & Energy Systems, 71, 2015, pp. 274-284.

[11] A. Messai, A. Mellit, I. Abdellani, and A. Massi Pavan “On-line fault detection of a fuel rod temperature measurement sensor in a nuclear reactor core using ANNs”, Progress in Nuclear Energy, 79, 2015, pp. 8-21.

[12] S. Dauwe, et al. “Multi-criteria anomaly detection in urban noise sensor networks”, Environ. Sci.: Processes Impacts, 16(10), 2014, pp. 2249-2258.

[13] K. Zhang, S. Shi, H. Gao, J. Li “Unsupervised Outlier Detection in Sensor Networks Using Aggregation Tree”, Advanced Data Mining and Applications, 4632, 2007, pp. 158-169.

[14] C. C. Castello, J. Sanyal, J. S. Rossiter, Z. P. Hensley, J. R. New "Sensor Data Management, Validation, Correction, and Provenance for Building Technologies." Technical paper TRNS-00223-2013.R1. In Proceedings of the ASHRAE Annual Conference and ASHRAE Transactions 2014, Seattle, WA, June 28-July 2, 2014.

[15] J. Bertrand-Krajewski, J. Bardin, M. Mourad, and Y. Branger "Accounting for sensor calibration, concentration heterogeneity, measurement and sampling uncertainties in monitoring urban drainage systems." Water Science & Technology, 47(2), 2003, pp. 95-102.

[16] M. Mourad and J. L. Bertrand-Krajewski "A method for automatic validation of long time series of data in urban hydrology." Water Science & Technology, 45(4-5), 2002, pp. 263-270.

[17] G. Olsson, M. Nielsen, Z. Yuan, A. Lynggaard-Jensen, and J. P. Steyer "Instrumentation, control and automation in wastewater systems." (2005).

[18] G. Tolle, et al. "A macroscope in the redwoods." Proceedings of the 3rd international conference on Embedded networked sensor systems. ACM, 2005, pp. 51-63.

[19] G. Barrenetxea, et al. "Sensorscope: Out-of-the-box environmental monitoring." Information Processing in Sensor Networks, 2008. IPSN'08. International Conference on. IEEE, 2008, pp. 332-342.

[20] N. Ramanathan, et al. "Rapid deployment with confidence: Calibration and fault detection in environmental sensor networks." Center for Embedded Network Sensing (2006).

[21] R. Szewczyk, A. Mainwaring, J. Polastre, J. Anderson, and D. Culler "An analysis of a large scale habitat monitoring application." Proceedings of the 2nd international conference on Embedded networked sensor systems. ACM, 2004, pp. 214-226.

[22] I. Adly, H.F. Ragai, A. El-Hennawy, and K. A. Shehata "Over-The-Air Programming of PSoC sensor interface in wireless sensor networks." MELECON 2010 - 15th IEEE Mediterranean Electrotechnical Conference. IEEE, 2010, pp. 997-1002.

[23] Libelium. [Online]. Available: http://www.libelium.com/uploads/2013/02/waspmote-technical_guide_eng.pdf. [retrieved: September, 2015]

[24] P. Baronti, P. Pillai, V. W. Chook, S. Chessa, A. Gotta, and Y. F. Hu “Wireless sensor networks: A survey on the state of the art and the 802.15. 4 and ZigBee standards.” Computer communications, 2007, 30(7), pp. 1655-169.