WSN Evaluation in Industrial Environments
First results and lessons learned

W-B. Pöttner[*], L. Wolf[*], J. Cecilio[†], P. Furtado[†], R. Silva[†], J. Sa Silva[†], A. Santos[†], P. Gil[†], A. Cardoso[†],
Z. Zinonos[‡], J. do Ó[§], B. McCarthy[¶], J. Brown[¶], U. Roedig[¶], T. O'Donovan[||], C. J. Sreenan[||],
Z. He[**], T. Voigt[**], A. Jugel[††]

[*]Technische Universität Braunschweig
Brunswick, Germany

[†]University of Coimbra
Coimbra, Portugal

[‡]University of Cyprus
Nicosia, Cyprus

[§]Petrogal
Lisbon, Portugal

[¶]Lancaster University
Lancaster, UK

[||]University College Cork
Cork, Ireland

[**]Swedish Institute of Computer Science
Kista, Sweden

[††]SAP AG
Walldorf, Germany

Abstract—The GINSENG project develops performance-controlled wireless sensor networks that can be used for time-critical applications in hostile environments such as industrial plant automation and control. GINSENG aims at integrating wireless sensor networks with existing enterprise resource management solutions using a middleware. A cornerstone is the evaluation in a challenging industrial environment — an oil refinery in Portugal. In this paper we first present our testbed. Then we introduce our solution to access, debug and flash the sensor nodes remotely from an operations room in the plant or from any location with internet access. We further present our experimental methodology and show some exemplary results from the refinery testbed.

I. INTRODUCTION

Large industrial sites need good means for monitoring and control of operations. Usually they have already complex process-control systems in operation. For instance, oil refineries do monitoring of the status of various in-field devices such as motors, valves, pumps to control process variables such as temperature, pressure etc.

Such existing control systems are typically based on fixed wired networks. They usually work relatively well today, but lack flexibility making it very complicated and expensive to extend the system or to adapt to changed requirements. Wireless systems using wireless sensor nodes and network components could provide this flexibility and allow for reduced costs and installation time. To be applicable to such industrial environments, specific wireless sensor networks (WSNs) have to be used which are compliant to the application scenario demands and being capable of providing the required reliability and timeliness. In particular, to be usable in such context, WSNs have to provide application-specific performance guarantees regarding delivery delay, reliability and energy consumption and must also integrate with existing enterprise resource management solutions. Since wireless sensor networks have not been widely deployed in time-critical applications, the EU-funded GINSENG [2] project aims to develop a performance-controlled WSN solution that employs

novel algorithms and software components to provide such performance guarantees to applications. A carefully planned and provisioned approach to deployment is a particular focus of GINSENG.

To have the most realistic conditions when evaluating the GINSENG solutions, a WSN testbed has been created at the Sines oil refinery in Portugal. In this paper we give a brief overview of the GINSENG solution, present the testbed and discuss problems that arise when having a testbed in a live, functioning industrial plant. In addition, we present our solution to remotely program and debug the nodes in the field. Furthermore, we present our methodology for evaluation in this industrial testbed. Finally, we demonstrate exemplary results from the refinery.

The remainder of this paper is structured as follows. In Section II existing testbeds are briefly discussed, while Section III gives an overview of GINSENG. Sections IV and V show the industrial testbed and our remote access solution respectively. Section VI presents our evaluation methodology and exemplary results are given in Section VII. Finally, Section VIII concludes this paper.

II. RELATED WORK

WINTeR [8] is a testbed specifically targeted at radio-harsh environments that can be found in the oil and gas industry. The testbed mimics the industrial surroundings with complex multipath propagation, provides the means to generate interference and has software that allows remote access. However, unlike the GINSENG testbed, WINTeR is not in a real industrial plant but replicates real industrial surroundings.

Krishnamurthy et al. [7] have trial deployments of a wireless sensor network in two industrial settings: a semiconductor plant and an oil tanker. Based on the application of vibration analysis to determine if maintenance is necessary, various sensor nodes are deployed and hardware platforms are compared. The authors have deployed the network for a continuous

four month period to prove its practicality, but do not have a permanent testbed deployment.

The WISEBED [3] project aims to create a heterogeneous virtual WSN testbed comprised of several smaller existing testbeds that can be interconnected. This not only allows researchers access to more nodes than available at each of the individual locations, but also provides heterogeneous hardware platforms that eventually work together. Actual testbeds of various sizes exist at different locations but are limited to lab conditions.

III. GINSENG Architecture

The GINSENG system is a cross layer designed system that is comprised of a number of different interrelated components. These components and their relationships are illustrated here in Figure 1. The system has been designed in a modular fashion in order to support the evaluation of individual components as well as the full system as a whole whenever required.
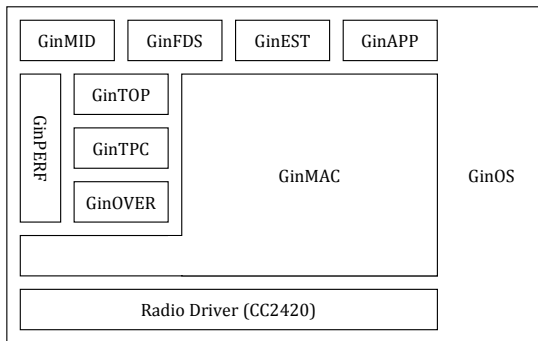

Figure 1. GINSENG Stack on the sensor nodes

Figure 1 shows that the core component of the GINSENG system is the GinMAC protocol. All the other components rely on it to produce the underlying reliable network topology that carries the sensor data. By default, the GinMAC component produces a tree based routing topology that utilises TDMA MAC layer access. The other components then augment this functionality to overlay new capabilities and services and improve the performance of the network in general, resulting in a comprehensive system capable of supporting real world wireless sensor networking deployments that require hard data delivery guarantees in challenging environments. In this section we provide an overview of the functionality of each of the components in the GINSENG system.

Operating System (GinOS): The Operating System used by the GINSENG system is based on the Contiki Operating System [4] and utilises a number of additional modules for improved performance in reliable wireless sensor network deployments. Principal amongst these are Real Time Scheduling and the Ring Buffer logging system. The Real Time scheduling functionality in the GinOS component utilises Contiki's rtimer module to meet hard real time deadlines. In particular, the Contiki rtimer is used to support the operation of GINSENG's MAC protocol which requires accurate time slot-based scheduling. The Ring Buffer logging system used in GinOS was created to ensure sensor nodes in the GINSENG system have access to a file system that allows reading of data from the beginning of a file and appending data to the end. Also, the ring buffer logging system is optimized for time-critical systems such as GINSENG.

Medium Access Control (GinMAC): The Medium Access Control used within the GINSENG system provides addressing and channel access control mechanisms to allow GINSENG nodes that are within radio range to communicate. It is a multihop system that uses an exclusive TDMA scheme for channel access with a pre-dimensioned virtual tree topology and hierarchical addresses. This TDMA-based model provides the fundamental building blocks that allow the GINSENG system to provide reliable, deterministic data delivery in wireless sensor networks by carefully separating the transmission of data to prevent data loss occurring from collisions. GinMAC uses hierarchical addresses to identify individual positions in the tree that have the form X-Y-Z. Each of the variables identifies the branch at a certain level while 1-1-1 denotes the first branch in all three tiers of a tree. 1-0-0 gives the child to the left of the sink node, while 2-1-0 is the left child of the second child of the sink node.

Topology Control (GinTOP): The Topology Control component is responsible for constructing the topology that the sensor nodes will form in order to transmit data. It is designed to connect all nodes in the network and organize them in a tree structure topology to serve the needs of GinMAC. GinTOP does not rely on any central entity to have full knowledge of the network and instead is responsible for topology related decision making locally on each individual sensor node. GinTOP is responsible for processes such as Neighbour Discovery, Virtual Tree Topology Construction and Virtual Tree Topology Maintenance and Optimization. The basic aim of the GinTOP component is to automatically generate a resulting tree topology that utilises the best available links (based on signal strength) to interconnect all of the sensor nodes in a GINSENG deployment. In addition GinTOP also allows for both static and dynamic configuration as well as mobility with both soft and hard handoffs.

Transmission Power Control (GinTPC): The Transmission Power Control component in the GINSENG system is responsible for minimising the energy that is utilised when transmitting unicast packets to other sensor nodes in a deployment by individually adapting the transmission power towards each neighbour. This technique also reduces the geographical coverage of a GINSENG network and allows for more efficient channel reuse. Since reliability in a GINSENG deployment is of major importance, the GinTPC mechanism has been designed to safely determine if a lower transmission power can be utilised without jeopardizing performance.

Overload Control (GinOVER): The Overload Control component in the GINSENG system is responsible for queue management. In particular, it is designed to maintain and optimise the transmission queue to ensure smooth and efficient communication throughout a GINSENG deployment's

operation. Whilst other components in the GINSENG system are designed to try and prevent conditions such as bad links or excessive traffic from arising, it is the role of the GinOVER module to ensure the ordered transmission of packets and minimum delays in processing if they do actually occur. In addition to its base functionality, the GinOVER component also offers prioritised packet transmissions based on packet type and deadline, thus enabling the preferential handling of certain packets during operation (i.e. control packets).

Performance Debugging (GinPERF): The Performance Debugging component of GINSENG is responsible for recording and monitoring all performance aspects of the GINSENG System. It provides a Management Information Base (MIB) that is used by all of the components of the system to store performance information throughout their operation. The Performance Debugging component utilises the information stored in the MIB to monitor ongoing performance and reports any anomalies to the sink. The information provided by the Performance Debugging component can also be employed by many other tasks where a comprehensive understanding of the operational state of a GINSENG network deployment is required, in particular it is used extensively during debugging and evaluation exercises.

System Application (GinAPP): The System Application component is the application layer component and therefore the conceptual driver of the GINSENG system, since it generates the digitised data that must be transmitted over the GINSENG deployment. The typical role of the GINSENG System Application is to monitor or control some physical aspect of an industrial environment by gathering data from its associated sensors or altering the state of an actuator. It then utilises the GINSENG system to transmit packets to and from the network sink.

Middleware (GinMID): GinMID comprises the System Integration and Performance Monitoring components of the GINSENG middleware. The System Integration enables the upwards and downwards communication between data sources (including the GINSENG WSN), backend applications and further middleware components and is controlled by the middleware-internal Performance Monitoring that ensures the flexible management of performance information and the optimal behaviour in overload situations (performance-driven load shedding).

Fault Diagnosis and Supervision (GinFDS): The Fault Diagnosis and Supervisory component monitors the GINSENG system and application to detect faults. The GinFDS component is comprised of a fault diagnosis component at node level, which aims to provide preliminary fault detection and isolation by utilising various analytical approaches to data analysis. In addition there is a supervisory component in the middleware that provides methodologies to analyse the information coming from all the fault diagnosis components of the WSN and to generate inputs for the other modules of the middleware as alarms or indicators about fault diagnosis status in the distributed system.

Energy Estimator (GinEST): The Energy Estimator component of the GINSENG system monitors the use of energy within the system and predicts system lifetime. This component enables the user to calculate system lifetime which is used in formulating a battery replacement strategy. The Energy Estimator retrieves cumulative on-time of each node's hardware components using Contiki's software-based energy estimation module [5], and periodically reports the values to the sink.

IV. TESTBED

Evaluation is very important for the GINSENG project to prove that the envisioned solutions work in real industry settings and that the system meets the application-specific performance targets. Consequently, we have installed a wireless sensor network testbed within the Sines oil refinery in Portugal which is operated by Petrogal S.A. We have used T-Mote Sky sensor nodes for our testbed installation.
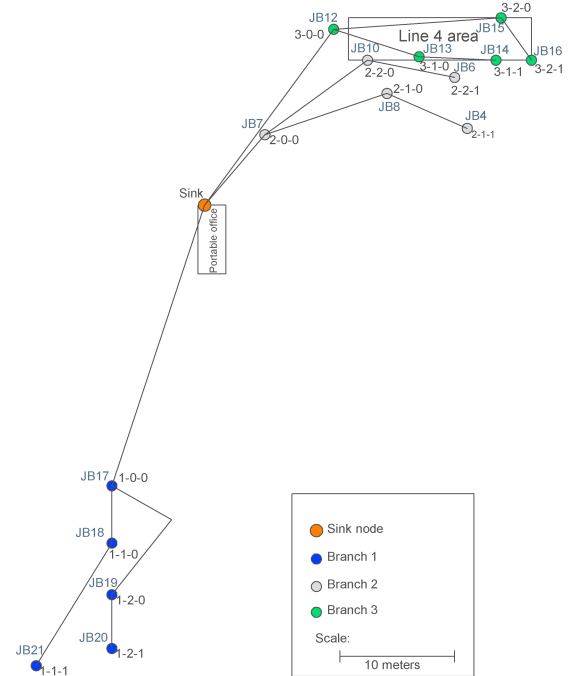

Figure 2. Sines testbed overview

The operational basis for the testbed is a portable office container that houses the testbed backend equipment such as the sink PC and acts as DCS (Distributed Control System) for the GINSENG project. Around the container, we have 16 nodes permanently installed in the refinery as shown in Figure 2. Also, we have picked locations for 14 additional nodes that we expect to install within the next month. Each node is attached to a real sensor in the refinery using the analogue to digital converter of the T-Mote Sky. Since the industry sensors encode their signal into current between 4 and 20 mA, we have installed signal converters that "translate" the signal into 0-2.5 V. The current testbed installation covers more than 2500 m^2 while the expanded testbed will cover more then 3500 m^2.

The 16 installed nodes can be divided into three groups:

- 1 node is the sink and attached to the portable office
- 10 nodes are attached to flow sensors
- 5 nodes are attached to pressure sensors

One of the sensor node attached to a flow sensor is also connected to an actuator. However, for safety reasons this actuator is a demo and does not really actuate something in the refinery.

Figure 3. Sines testbed photo showing nodes 17 to 20 (front to back)

Figure 3 tries to give an idea of the installation situation of nodes that are shown in the lower left corner of Figure 2. Although the distances between the nodes are well below 10 m, the communication environment is still challenging because of multipath propagation and interference.

As already introduced in the previous section, the TDMA MAC layer of the GINSENG project [9] organises the network as a tree rooted at the sink node. Figure 2 contains the tree structure that has been used for our evaluation relying on static topology. However, since GINSENG deploys mechanisms to dynamically form a network, the final tree may look different.

A. Safety Regulations

The Sines refinery has several safety regulations to avoid interruptions of the plant operation on the one hand and to ensure safety of workers and the environment on the other hand. In our experience, these regulations limit the implementation of equipment and also greatly increase the overhead of any evaluation that may take place. Working in the refinery is only possible, after a full day of safety training. Operating electrical equipment in the open is not permitted, expect for special certified devices. People entering the refinery have to wear special fire-proof overalls, boots, safety glasses and a helmet.

Implementing a wireless sensor network in potentially explosive atmospheres requires the components to be ATEX certified [6]. Obtaining ATEX certification for sensor nodes may be possible, but embedding the nodes into certified ATEX junction boxes is the simpler solution. These boxes are intended to be used for electrical interconnections and are sold in various sizes. Due to practical constraints, we have used boxes with a footprint of 150x150 mm and a height of 94 mm. On the inside, these boxes have a hexagonal shape

and contain screw terminals to connect incoming and outgoing wires. Such cables are routed through special ATEX compliant cables glands to keep the ATEX compliance of the box.

Although early experiments have shown no impact on signal attenuation by specific ATEX boxes [1], the ATEX boxes employed in the refinery have a high impact on the radio propagation due to metal structures on the in- and outside. The attenuation of the boxes prevented communication with neighbouring nodes. Consequently, we have attached antennas which are located outside of the ATEX boxes to the nodes. These antennas have a gain of 9 dBi and finally allow wireless communication between the nodes in the refinery.

V. REMOTE REPROGRAMMING SYSTEM

The refinery testbed allows testing and evaluating software components in a real industry setting. However, the manual process of reprogramming the nodes is not only time consuming and error prone; it also limits the efficiency with which the testbed can be used. In place programming of the nodes using a laptop is not allowed due to safety regulations in the refinery. Consequently, manual reprogramming involves someone walking to the node, opening the ATEX case, taking the node to a safe place, programming it via USB and bringing it back to the ATEX box. Of course, only during acceptable weather conditions and only whenever someone is available to do the programming job, new software components can be evaluated. Therefore, we have implemented a remote reprogramming system to be able to program all nodes belonging to the testbed from a central point and also over the Internet.

In order to be a replacement for a person manually programming the nodes and to support a thorough evaluation of the GINSENG system, a number of requirements have to be met. The system should allow nodes to be flashed in-situ from a central point while also allowing to reset the nodes. The system should provide constant power to all nodes to allow repeatable experiments and exclude the influence of depleting batteries. The system should collect the serial outputs of the nodes and time stamp these as precise as possible to provide an out-of-band debugging facility that does not interfere with the network operation. Finally, the central point should be accessible via the Internet to allow efficient usage of the testbed and the system should rely on off-the-shelf hardware as far as possible to avoid excessive cost and hardware development time.

Despite the requirements, the remote reprogramming system has to face a number of constraints. Nodes as well as the remote reprogramming system have to be situated within ATEX boxes which greatly limits the available room for hardware. The box also limits the amount of heat that can be dissipated without damaging the hardware. Further, the ATEX boxes are placed up to 120 m away from the central point and power is not available in any of the boxes. Finally, the portable office container does not have any form of internet access.

Based on the requirements and constraints, the following section will present the design of the remote reprogramming

system employed in the industry testbed of the GINSENG project.

A. System Design

Based on the requirements our first approach was to use long USB cables between the central point and the nodes. However, even using USB extenders that are available on the market, we were unable to support distances of up to 120 m and still flash code onto the nodes. Also, some of the available USB extenders require too much room and do not fit into the ATEX box. Consequently, we have decided that the most viable solution is an embedded Linux computer inside the ATEX boxes that can power the node and can connect to the portable office container using Ethernet. Although Ethernet over copper cables is not specified for distances above 100 m, this limitation is mainly caused by the minimum packet length for collision detection and should not cause problems when gradually exceeding the maximum distance.
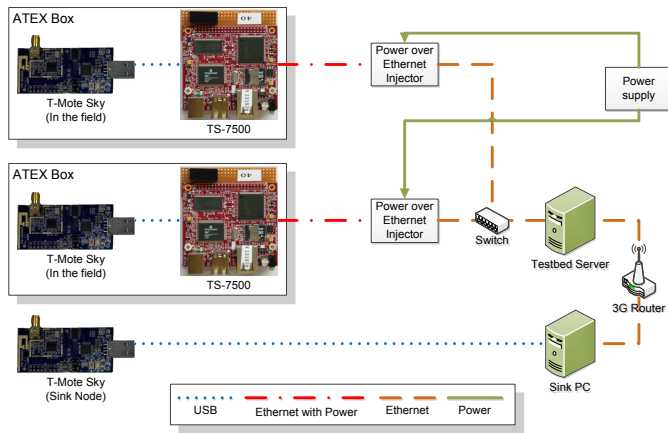

Figure 4.   Remote Reprogramming Component Overview

1) Hardware Design: The hardware architecture of our system is outlined in Figure 4. Each T-Mote Sky node is connected to a Technologic Systems TS-7500 embedded computer that runs the Linux operating system. The footprint of the TS-7500 board is 74.3x66 mm which is small enough to fit into the ATEX box. The board has an ARM9 CPU, a micro SD card slot, two USB host ports, an Ethernet connector and a 44-pin connector that exposes various other pins of the board. Power is supplied to the nodes via the USB connector and the TS-7500 boards are connected to a central switch using Ethernet cabling. To supply power to the boards, we have decided to use Power over Ethernet (PoE) to avoid running additional cables for power supply means. Since the TS-7500 does not support PoE out of the box and to reduce the necessary room, we have decided to use a proprietary PoE solution that does not adhere to IEEE 802.3af.

The proprietary PoE solution uses the two unused pairs of the Ethernet cable; one pair to supply ground and the other pair to supply 24V to the TS-7500 boards. The power can be introduced into the Ethernet cables using passive PoE injectors and the TS-7500 board allows to access the two PoE

pairs on the multi-purpose 44-pin connector. To transform the incoming voltage into 5V that are required by the board, we have implemented a step-down converter on a circuit board that is attached to the TS-7500. The step-down converter has an efficiency of 90% which minimises the dissipated heat.

2) Software Design: On the software side, we have a standard ARM Debian Linux distribution running on the TS-7500 boards that uses the Network Time Protocol (NTP) to synchronise its clock with the testbed server. The boards run an SSH server and also a daemon process that listens to the serial port of the attached node. The process has three core duties:

1) Time stamp messages arriving on the serial port and send them to the testbed server using reliable TCP communication
2) Wait for incoming flash images via secure file copy and flash images onto the node
3) Receive commands from the testbed server to reboot the board, reset the node, etc. and process these commands

On the testbed server, we use a standard Ubuntu Linux equipped with an SSH and NTP server. A daemon process takes care of receiving log messages from the TS-7500 boards and stores them into a separate file for each node. Also, the daemon distributes flash images and commands to boards that should be programmed, reset or rebooted. The daemon has a tool for interaction with the testbed operator. Nodes are addressable by individual address or by group ids to enable programming multiple or all nodes at once.

3) Internet Access: Since no internet connectivity is available in the portable office container, we have used a cellular 3G internet connection to enable access to the testbed from the outside. We have used a 3G router that is connected to the testbed server and the sink PC via Ethernet as shown in Figure 4 and provides internet access to these two computers. 3G internet access in Portugal is unrestricted and provides public IPs that can be directly reached from the outside. As a backup solution, we have set up a VPN connection to a server outside of the refinery that can be used in case the direct access to the testbed router is blocked for some reason.

B. Current status and early experiences

The system as outlined above is currently installed in the refinery testbed in Portugal. 15 nodes are connected to TS-7500 boards and can be programmed remotely while the hardware for another 14 boards is prepared and awaits installation. The 3G internet uplink is working as expected and people using the system find it very helpful, since it offers hands-on abilities to people being situated remotely. Also, it greatly increases the efficiency with which the testbed can be used.

Initial issues with a subset of the boards have revealed problems with the power supply boards. The step-down converters did not behave as documented in the datasheet and some had to be replaced. Also, additional fuses have been implemented to prevent further damage from the step-down converter and the TS-7500 boards.

Another problem that we experienced are T-Mote Sky nodes that become unresponsive and do not accept new programs to be flashed onto them. We have mitigated this issue by replacing the respective nodes.

Finally, the Ethernet cables above 100 m produce packet loss ratios of up to 50 %. These are caused by the fact that the TS-7500 board thinks to have lost the Ethernet link just to regain link after a couple of seconds. This problem is related to implementation issues in the Linux device driver provided by Technologic Systems and could not be solved to date. According to the company, the problem is known and a fix is worked upon.

VI. EVALUATION METHODOLOGY

Each software component within the GINSENG system can be run with a number of different configurations. For example, the Performance Debugging component may use frequent periodic reporting or in-network processing with less frequent reporting. The Medium Access Control component may use individual transmissions or may use data aggregation to conserve bandwidth. As the GINSENG system is assembled using the individual components, a very large number of different GINSENG system configurations is possible. Due to the large number of possible system configurations it is impossible to test all combinations in the testbed deployment at Sines (separate lab experiments were employed to test individual components). For the evaluation at Sines, there were five main experiments carried out. Firstly, a GINSENG system including all of the components in a standard configuration was used to define an evaluation baseline (GINSENG Baseline). Thereafter we modified the configuration of specific individual components in order to evaluate the impact of these configuration changes on the overall system performance. We avoided the modification of multiple components at a time as this would prevent us from being able to easily quantify the impact of individual component configurations on the overall GINSENG system. Furthermore, we carefully selected a component configuration modification that purposely has significant impact on the overall GINSENG system performance. Figure 5 summarizes our experiments.
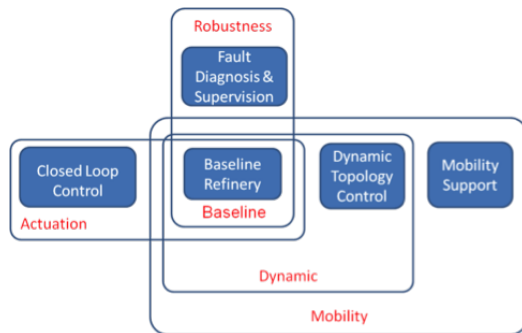

Figure 5. Planned Experiments

The experiments in Figure 5 were organized as follows: after the initial baseline experiment, our second set of experiments

utilised the Topology Control component with its configuration set to dynamic topology control (GINSENG Dynamic). The third set of experiments used for comparison with the GINSENG Baseline profile utilised the Topology Control component with its configuration set to mobile topology control (GINSENG Mobile). The following two experiments after these utilised exactly the same network configuration as the GINSENG Baseline, only with additional application level functionality enabled. As such, in the fourth experiment we also introduced the use of an actuator. This tested the ability of the GINSENG system to not only periodically report sensor data, but to also act upon the data collected and use the GINSENG network to trigger a physical alteration to the refinery equipment (GINSENG Actuation). Finally, the last set of experiments utilised the GinFDS component to activate Fault Diagnosis and Supervision, therefore testing the application functionality that introduces robustness to the network's operation (GINSENG Robustness). For all experiments (carried out for individual component configurations at partner labs and overall GINSENG system evaluation at Sines) the same system metrics are evaluated. This allowed us to compare component evaluations at partner labs with the system evaluation at Sines. For example, energy consumption over time at each node is a performance metric that should be collected for all experiments. In addition to these generic metrics, it is necessary to collect component specific metrics to evaluate functionality of a component.

A. Evaluation Metrics

We deploy generic GINSENG Metrics and Component Specific Metrics. The generic metrics are needed to understand the general characteristics of the GINSENG deployment at the Sines Refinery. These metrics consist of:

Energy Consumption: efficient operation with regards to energy consumption helps to ensure that the wireless sensor network deployment can run for longer without any intervention required, and help maintainers ensure that batteries are replenished before they run out of power.

End-to-End Data Delivery Delay: Understanding what delay a specific node may encounter in its data delivery can be essential to meet the deadlines imposed by the stringent requirements in industrial scenarios.

End-to-End Data Delivery Reliability: Packet loss should be reduced to an absolute minimum in GINSENG, so it is important to analyse its occurrence and causes.

Component Specific Metrics include (summary list):

System Application (GinAPP): measures for this component include the alarm delay (time from an event triggering an alarm to its delivery in the sink node or control station) and the actuation time (closed-loop control time);

Medium Access Control (GinMAC): measures for this component include the number of retransmissions

Topology Control (GinTOP): measures for this component include the tree size, the tree forming time and the number of control messages that were necessary. The tree size indicates whether all nodes have been attached, and the tree forming

time indicates how long it takes for all nodes to attach to the tree;

Mobile Nodes: for mobility evaluation metrics include downtime and packet losses. The mobile node downtime is the time between disconnection and re-attachment of a node while switching attachment points and packet losses accounts for the packets that are lost during the hand over process;

Overload Control (GinOVER): measures for this component include the average queue length and the maximum queue packets. The average queue length gives a good indication if the network capacity is dimensioned high enough, and the maximum number of packets in the queue indicates if the network and queue capacity can cope with current traffic conditions.

Performance Debugging (GinPERF): performance debugging metrics include overheads and anomaly detection accuracy and delay.

Transmission Power Control (GinTPC): we measure layer 2 packet losses and power consumption, besides the generic metrics already described before.

Energy Estimator (GinEST): measures for this component include cumulative, average and current power consumption;

Operating System (GinOS): measures for this component include rtimer scheduling success rate and scheduling overhead.

VII. RESULTS

To develop an understanding of how the basic functionality of the GINSENG network was performing in the Sines refinery deployment we began by carrying out tests in the GINSENG Baseline configuration whilst analysing the Generic GIN-SENG metrics. Namely, this meant focusing on the energy consumption of the network, its overall reliability and the end-to-end delay experienced by nodes when transmitting sensor data. Data was collected during a period of 30 minutes while influences of the radio-harsh environment onto the radio propagation conditions cannot be quantified.

Reliability metric is significant with regards to the GIN-SENG project because one of the primary goals is to develop a solution with very little data loss during operation. The end-to-end packet loss value in our deployment is calculated by using the GinMAC sequence number to calculate the number of messages that have been sent by a node, and from that it calculates the number of lost messages (number of messages received vs. number of messages sent). The final value is then calculated as (messages lost / messages sent) * 100. Figure 6 shows the packet loss values for each node during the network operation. In operation it is observed that the value reported in this field is initially high when the first instance of loss occurs because there has typically been a relatively small amount of messages sent. With time, as more messages are delivered this value then begins to stabilize to a more accurate representation of the actual level of packet loss experienced. To acquire the data used for the plots in these graphs we combined all of the packet loss rate values reported for each node and then divided them by the number of reports each node made throughout the duration of the testing run. Based on Figure 6 the majority of

nodes in the deployment experienced zero end-to-end packet loss throughout the entire duration of our testing run and this is obviously the ultimate aim of the project. Node 1-2-1 was affected by the highest level of packet loss, but still this total value was under 0.25%.
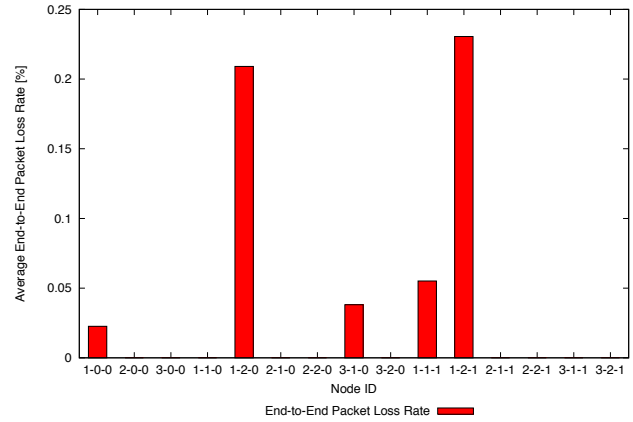

Figure 6. Baseline Average Packet Loss Rate

In addition to the overall reliability the energy consumption of a wireless sensor network is one of the key considerations in its effectiveness. Figure 7 depicts the overall energy consumption recorded for each node at the end of a testing run. Energy consumption in wireless sensor networks is important, so we want to determine exactly how much energy is used by each node in our testbed deployment during our experiments. In this graph we present a separate individual bar for each node in the deployment.
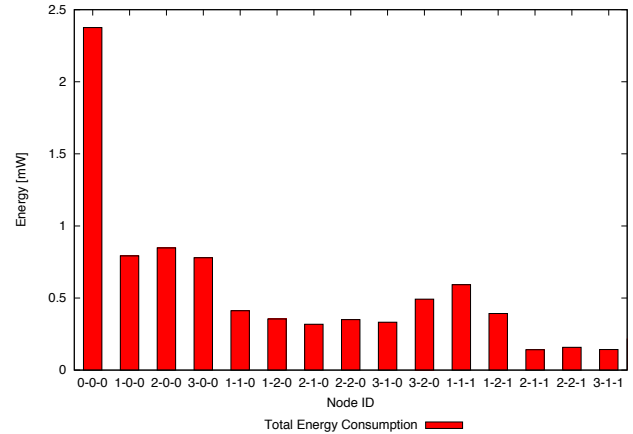

Figure 7. Baseline Total Energy Consumption

The other key metric we measure in the Generic GINSENG experiments is the end-to-end delay imposed on traffic delivery. Delay is another key consideration because in industrial processes the time it takes for information to reach a control unit can be critical. Not only are the average delay levels important but also the fluctuation in the level of delay experienced is significant. In addition to reliability, determinism is also a characteristic that the GINSENG project has strived to

attain, ensuring high reliability may in some cases be futile if the delay experienced fluctuates wildly and means that controllers cannot be accurately guaranteed when they will receive data (even though it may be possible to guarantee that they will definitely receive data at some point). The delay value is calculated by comparing the time the message was generated with the time it was received by the sink node.

Figure 8 shows that the average delay is never above 900ms for any of the nodes in the deployment and that for the majority of nodes it is under 500ms. In addition to the node identification itself it is also possible to identify each node's hopcount from the sink through its Node ID notation. This highlights an interesting aspect about the delay imposed on end-to-end delivery in multihop TDMA networks like GINSENG.

In the graph, a number of instances occur where nodes with higher hopcounts (i.e. that are further away from the sink) have significantly lower delay than those which are only one hop from the sink. In a non-TDMA based solution this outcome would be unexpected and may indicate possible problems, however in a TDMA based solution like GINSENG this result highlights the significance of the schedule used to transmit traffic. In the cases where 3 hop nodes deliver their traffic with less delay than a 1 hop node in the same branch, the 3 hop node has generated data and very soon after is given a slot to transmit that data in. In the meantime the nodes further up the tree must queue their data and wait even though it has already been generated which increases the delay. If this continues up the tree the data from the node that is 3 hops deep will pass through each hop quickly, whilst the intermediate nodes data must continue to wait in a queue.
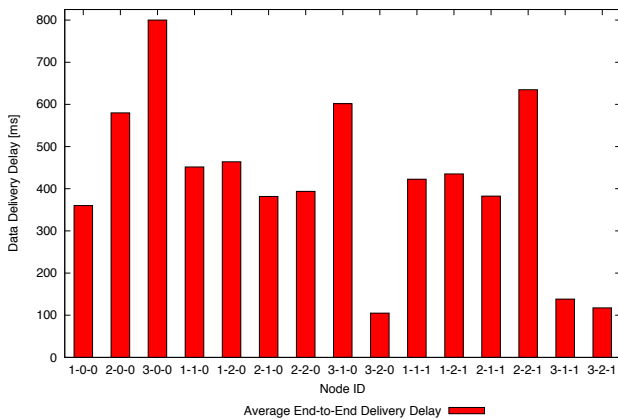

Figure 8.   Baseline Average End-to-End Delivery Delay

VIII. CONCLUSIONS

In this paper we have presented an industrial testbed situated in an oil refinery in Portugal that is actively used for the evaluation of the GINSENG project. We have discussed, that existing in-lab testbeds cannot reproduce the complex characteristics of industrial plants and that a testbed within the oil refinery produces realistic results. Further, we have given a brief introduction into the GINSENG system and outlined the parameters of our testbed. We have also given a slight insight into how complex deploying nodes in an actual industrial plant can be. Also, we have augmented our testbed with additional hardware to remotely debug and program nodes because manual programming of the nodes is tiresome and greatly limits the efficiency with which the testbed can be used. We have discussed the methodology to evaluate a complex system as GINSENG in a challenging environment such as the refinery. Finally, we have presented exemplary results.

Overall, the results show that the GINSENG system can meet application-specific performance targets while being energy efficient at the same time. With up to 0.25 % of end-to-end packet loss in a harsh radio environment that cannot be quantified and with delivery delays well below the target of 1 s, the energy consumption especially of the leaf nodes is low as expected. The sink node consumes significant energy but is connected to a power adaptor by design.

REFERENCES

[1] C.A. Boano, N. Tsiftes, T. Voigt, J. Brown, and U. Roedig. The Impact of Temperature on Outdoor Industrial Sensornet Applications. *Industrial Informatics, IEEE Transactions on*, 6(3):451 –459, aug. 2010.

[2] J. Brown, U. Roedig, T. O'Donovan, C. J. Sreenan, Z. He, T. Voigt, B. Pöttner, L. Wolf, A. Klein, J. Sa Silva, V. Vassiliou, and J. do Ó. Invited poster: GINSENG: Performance Control in Wireless Sensor Networks. In *8th European Conference on Wireless Sensor Networks (EWSN'11)*, Feb 2011.

[3] Ioannis Chatzigiannakis, Stefan Fischer, Christos Koninis, Georgios Mylonas, and Dennis Pfisterer. WISEBED: An Open Large-Scale Wireless Sensor Network Testbed. In Ozgur Akan, Paolo Bellavista, Jiannong Cao, Falko Dressler, Domenico Ferrari, Mario Gerla, Hisashi Kobayashi, Sergio Palazzo, Sartaj Sahni, Xuemin (Sherman) Shen, Mircea Stan, Jia Xiaohua, Albert Zomaya, Geoffrey Coulson, and Nikos Komninos, editors, *Sensor Applications, Experimentation, and Logistics*, volume 29 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pages 68–87. Springer Berlin Heidelberg, 2010. 10.1007/978-3-642-11870-8_6.

[4] A. Dunkels, B. Grönvall, and T. Voigt. Contiki - a lightweight and flexible operating system for tiny networked sensors. In *Proceedings of EmNetS-I*, Tampa, Florida, USA, November 2004.

[5] A. Dunkels, F. Österlind, N. Tsiftes, and Z. He. Software-based on-line energy estimation for sensor nodes. In *Proceedings of the Fourth Workshop on Embedded Networked Sensors (Emnets IV)*, Cork, Ireland, June 2007.

[6] European Commission. Equipment and protective systems for potentially explosive atmosphere - ATEX. http://ec.europa.eu/enterprise/sectors/mechanical/atex/index_en.htm.

[7] Lakshman Krishnamurthy, Robert Adler, Phil Buonadonna, Jasmeet Chhabra, Mick Flanigan, Nandakishore Kushalnagar, Lama Nachman, and Mark Yarvis. Design and deployment of industrial sensor networks: experiences from a semiconductor plant and the north sea. In *Proceedings of the 3rd international conference on Embedded networked sensor systems*, SenSys '05, pages 64–75, New York, NY, USA, 2005. ACM.

[8] J. Slipp, Changning Ma, N. Polu, J. Nicholson, M. Murillo, and S. Hussain. WINTeR: Architecture and Applications of a Wireless Industrial Sensor Network Testbed for Radio-Harsh Environments. In *Communication Networks and Services Research Conference, 2008. CNSR 2008. 6th Annual*, pages 422 –431, May 2008.

[9] Petcharat Suriyachai, James Brown, and Utz Roedig. Time-Critical Data Delivery in Wireless Sensor Networks. In *6th IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS '10)*. IEEE, June 2010.