Self-organizing Maps versus Growing Neural Gas in detecting Anomalies in Data Centers

M. Zapater , D. Fraga , P. Malagón , Z. Banković , J.M. Moya

Abstract. Reliability is one of the key performance factors in Data Centers. The out-of-scale energy costs of these facilities lead Data Center operators to increase the ambient temperature of the data room to decrease cooling costs. However, increasing ambient temperature reduces the safety margins and can result in a higher number of anomalous events. Anomalies in the Data Center need to be detected as soon as possible to optimize cooling efficiency and mitigate the harmful effects over servers. This paper proposes the usage of clustering-based outlier detection techniques coupled with a Trust and Reputation System (TRS) engine to detect anomalies in Data Centers. We show how SOM and GNG can be applied to detect cooling and workload anomalies respectively in a real Data Center scenario with very good detection and isolation rates, in a way that is robust to the malfunction of the sensors that gather server and environmental information.

Keywords: anomaly detection, data centers, self-organizing maps, growing neural gas

1 Introduction

During the last few years, there has been a rapid increase in the number of data center facilities over the world. Data Centers provide the required infrastructure for a wide range of traditional applications (social and business networking, Webmail, Web search, etc.) as well as new-generation applications such as e-Health or Smart Cities. Advances in the underlying manufacturing process and hardware design technologies have continuously made possible the constant increase in computing capacities. However, the increase in computational capabilities has not come for free. These facilities consume huge amounts of electrical power, accounting for 2% of the total USA energy budget [1]. They also generate a tremendous amount of heat that has to be extracted to ensure the reliable operation of server and other computational (IT) equipment. The energy consumption needed to cool down servers accounts for around 30% of the total energy cost of the infrastructure [2]. Even though increasing the Data Center room temperature has proven to be a way to save cooling energy, there are some important concerns regarding reliability, which is one of the key performance

factors in Data Centers. The American Association of Heating and Cooling (ASHRAE) describes that the inlet temperature of servers should be kept below 30ºC to avoid CPU redlining [3]. Failures in either the room or the server cooling systems could lead to reliability issues that would reduce the Mean Time To Failure (MTTF) of IT equipment [4]. Temperature anomalies in the Data Center, as well as any other type of anomaly that might affect the reliable behavior of IT equipment, need to be detected as soon as possible to mitigate the harmful effects.

To this end, this paper proposes the usage of clustering-based outlier detection techniques coupled with a Trust and Reputation System (TRS) engine to detect anomalies in Data Centers. In our previous work [5] we have demonstrated that clustering-based outlier detection approaches offer numerous advantages for detecting insider attacks, such as high adaptability, flexibility, possibility to detect unknown attacks, no restrictions on training data, etc. Data center anomalies exhibit a similar behavior, making clustering techniques a good candidate for their detection.

Within the scope of clustering-based approaches, we encounter different deployment possibilities: i) k-means or k-Nearest Neighbour (k-NN) techniques, or ii) topology-preserving competitive methods, such as Self-organizing maps (SOM) or Growing Neural Gas (GNG). Topology preserving techniques are very convenient for our application scenario, since one of the main parameters that reveal the presence of outliers is the average distance of a cluster to its closest neighbors.

The main contributions of this paper can be summarized as follows:

- We show an exhaustive analysis on the taxonomy of anomalies in Data Centers and the information sources used to detect and isolate them.
- We present a Trust and Reputation System (TRS) coupled with topology-preserving clustering algorithms to detect and isolate anomalies related to data room cooling failures, server workload anomalies and anomalies related to the data room monitoring infrastructure.
- We validate our results with data gathered in a real Data Center room with heterogeneous servers. Our experimental setup allows runtime monitoring of the facility, as well as the controllable generation of anomalies.

The remainder of the paper is organized as follows: Section 2 describes the related work on the area. Section 3 describes the taxonomy of anomalies and information sources, whereas Section 4 describes the Trust and Reputation environment and the clustering algorithms used in this scenario. Experimental results are shown in Section 5. Finally, the most important conclusions are drawn in Section 6.

2 Previous Work

Next-generation applications, such as the ones found in Smart Cities, e-Health, Ambient Intelligence or Weather analysis, require constantly increasing high computational demands that can only be provided in Data Centers [6,7]. Several techniques to reduce energy consumption in Data Centers are based on increasing the supply temperature of air conditioning units to reduce cooling costs. However, increasing the inlet temperature of servers has some drawbacks. A report by the Uptime Institute [8] showed that for every 10ºC degrees of temperature in excess of

21°C in the inlet temperature of servers, long-term reliability could be reduced by 50%. Even though recent research [9] shows that the effect of high temperatures on reliability is smaller than what had been assumed, as the ambient temperature increases the safety margin for the server thermal shutdown is decreased.

Moreover, the temperature distribution in a Data Center is not uniform and tends to have hot spots, which are areas significantly hotter than the average. To prevent server thermal shutdown, the highest CPU temperature limits the maximum Computer Room Air Conditioning (CRAC) air-supply temperature. Thus, it is important to be able to detect and localize any anomaly taking place at the Data Center. Anomalies can be due to failures in the cooling system, in the servers, or misbehaviors in the workload assignment, that affect the thermal conditions of the server and room.

There is much research in the area of anomaly detection in Data Centers. Some approaches try to model and estimate the temperature conditions with Computational Fluid Dynamics (CFD) simulations [10]. CFD is time and cost expensive, and results are not robust to changes in the Data Center. Other works use regression models with historic data [11] or threshold-based anomaly detection [12]. All the previous techniques rely on considering static Data Center layouts. However, data center environments are subjected to constant changes in the placement of servers and racks.

Learning and training techniques based on fuzzy control have been previously used by Sedano et.al.[13] for temperature control in buildings to maximize energy efficiency. For the particular case of Data Centers, machine learning approaches based on Neural Networks (NN) aim to find relationships between the thermal features. Other works use Self-Organizing Maps (SOM) [14] but only to discover network attacks in the Data Center, not as a methodology for anomaly detection.

In this paper we leverage the usage of topology-preserving clustering algorithms such as SOM or Growing Neural Gas (GNG) to detect and isolate anomalies. The most similar work to ours is the research by Yuan et.al. [15]. The authors propose the usage of a hierarchical neural network to detect temperature anomalies both at the server level and at the data center level. As opposed to ours, they do not show a complete taxonomy of anomalies, and do not provide metrics such as detection time or isolation capacity.

3 Decomposing Anomalies in Data Centers

3.1. Taxonomy of anomalies

Thermal anomalies are not the only ones having a relevant impact in the behavior of the overall data room. In general, we can classify anomalies according to their cause in the following taxonomies:

- Data room cooling: caused by failures in the cooling equipment of the data room. Their impact depends on the number of CRAC units failing and the nature of the failure.
- Server level: refers to failures in the electronic components of the servers. The effect is local to the server (i.e. thermal redlining in the CPUs). However, local effects can also have an impact on the room dynamics.

- Workload execution: workload is allocated to the computing nodes via a resource manager. Failures can be understood as tasks assigned to a certain computing node that aborted or did not complete properly. Their effect is local to a server but can be extended to the nodes absorbing the unattended demand, which might become potential hot spots.
- Information sources: caused by failures in the environmental or in-server sensors used to gather information to detect anomalies. Malfunction can come because of battery-powered sensors running out of power, environmental sensors being moved by data center operators, server sensors providing random incorrect values, etc.

A last taxonomy would be attacks on the information or networks of the data center. The scope of these attacks can be very broad, but they are generally related to gaining access to the computing nodes to retrieve sensitive information. The aim of this work is not to detect anomalies due to foreigner attacks on the data center, which falls under the area of security, but to discover anomalies inherent to the data center.

3.2 Taxonomy of information sources

Current Data Centers are constantly monitored by a large number of sensors to enable overall IT and cooling management. Generally speaking, the information gathered in the data center can be classified as follows:

- Environmental sensors retrieve relevant thermal characteristics of the data room. In a real-life scenario, these sensors are: i) temperature sensors to measure the inlet and outlet of servers, ii) data room relative humidity sensors, iii) differential pressure sensors for raised-floor air-cooled data centers and iv) CRAC air supply temperature sensors.
- Integrated server sensors: these sensors are embedded in the electronics of the servers during their manufacture, and can be polled without performance overhead. The most relevant sensors are: i) CPU, memory and ambient temperature, ii) fan speed sensors and iii) server power consumption sensors.
- Server workload information: this information is obtained directly through the OS of the server (e.g. CPU and memory utilization, disk accesses, etc.).
- Workload allocation: the resource manager provides information about the particular workload allocation to each node, i.e. number of tasks assigned, wallclock execution time, start and end time, etc.

4. Clustering algorithms coupled with Trust and Reputation Systems

Most of the anomalies that take place at the Data center have a direct impact on the thermal behavior of the data room. To apply SOM or GNG clustering techniques we assume the anomalies demonstrate themselves as spatial and temporal inconsistencies, no matter what their source is. The explanation on the next subsections applies both

for SOM and GNG, as both algorithms follow the same standard steps. They only differ in the fact that the size of SOM is fixed from the start, whereas the size of GNG grows during the training. Fixed size can be a limitation, as it might not possible to know the optimal number of clusters from the start, leading GNG to perform better in some scenarios where SOM does not obtain adequate detection and isolation rates. Due to space reasons, the reader is referred to [17] and [18] for a deeper explanation on the SOM and GNG techniques used in this paper.

4.1 Feature Extraction and Model Formation

Following the idea of temporal inconsistency in the presence of anomalies, we provide the data model that captures these properties and allows us to deploy machine learning. For the case of sensed values, we follow the idea presented in our previous work [19] based on extracting n-grams and their frequencies within different time windows. We give a short example for a boolean sensor. Let the sensor give the following output during the time window of size 20: 1 1 1 1 0 0 0 0 0 0 1 1 1 1 1 1 0 0 0 0. If we fix the n-gram size on 3, we extract all the sequences of size 3 each time moving one position forward. In this way we can observe the following sequences and the number of their occurrences within the time window: 111 - occurs 6 times, 110 - 2, 100 - 2, 000 - 6, 001 - 1, 011 - 1. Thus, we can assign them the following sequences: 111 - 0.33, 110 - 0.11, 100 - 0.11, 000 - 0.33, 001 - 0.05, 011 - 0.05. In our model, the sequences are the features and their frequencies are the corresponding feature values. This characterization is performed in predefined time instants and takes an established amount of previous data, e.g. we can perform the characterization every 20 time periods based on previous 40 values.

As the extracted feature vectors are not of the same size, we calculate the distance function using the approach presented in [20], which calculates distance between sequences. The same solution is applied to a continuous magnitude by normalizing the values to a fixed range (e.g. from 0 to 5) and quantifying the sensor values to reduce the amount of n-grams without losing relevant information.

4.2. Anomaly Detection

Our goal is to detect unknown behaviors which have not been seen during the training phase, thus, we aim to detect outlying data that belongs to non-outlying clusters. For this reason, we calculate the quantization error (QE) of each input as the distance from its group center. The deployed distance function [20] is equivalent to Manhattan distance after making the following assumption: a feature that does not exist in the first vector while exists in the second (and vice versa) actually exists but occurs with 0 frequency. In this way, we get two vectors of the same size and the distance between the centre and an input is between 0 (when they are formed of the same features with the same feature values) and 2 (when the features with the values greater than 0 are completely different). Similarly, if the set of the features of one is the subset of the feature set of the other, the distance is between 0 and 1.

During the testing, n-grams not seen in the training appear when a sensor starts providing data significantly different than before. When this happens, the distance (i.e., the QE value), between the n-gram and its corresponding centre is greater than 1, showing evidence of abnormal behavior in the sensor or the data room.

Sensors are arranged in areas according to the events they report information about. All sensors providing information about the same observation (e.g. a thermal anomaly in a certain rack or room area), are assigned to the same area. The sensors in each area are examined by one or more independent agents. Agents are trained separately and execute the clustering algorithms. The system of agents is coupled with a reputation system where each sensor has its reputation.

For our purpose, the reputation value of the sensors is used in two different ways: i) individual sensor reputation reflects the level of confidence that other sensors have in this sensor, and is used to detect sensor malfunctioning. On the other hand, ii) area-wide reputation is calculated as the average reputation value for a specific area, and reflects the real anomalies occurring in the Data Center (e.g. CRAC malfunctioning).

The individual reputation of each node (rep) is calculated as follows:

if (QE < 1) rep = 1; else rep = 1 - QE/2;

Depending on whether the current reputation is below or above the established threshold reputation is updated in a different way. If the current reputation is above the threshold and the node starts behaving suspiciously, its reputation falls quickly. However, recovering from lower reputation takes more time, as the node has to redeem itself. The reputation update can be described in the following way:

```
if (last_reputation[node] > threshold)
        new_reputation[node] = last_reputation[node] + 0.8*(rep + log(1.5*rep));
else
        new_reputation[node] = last_reputation[node] + 0.05*(rep+log(1.5*rep));
```

5. Experimental Results

5.1. Experimental setup

In this section we show the experimental methodology used for the experiments performed in this paper. All data has been collected from a data room belonging to the research group. For the purpose of this paper, we restrict our experiments to the enterprise servers in one rack. The rack contains two types of servers, different in terms of architecture and power consumption: i) SunFire V20z with 2 Dual-Core AMD Opteron CPU and 4GB of RAM and ii) Fujitsu RX300-S6 servers with 1 Quad-Core Intel Xeon processor and 16GB of RAM. The servers are arranged in three different partitions: i) one containing all intel servers, ii) one containing one half of the AMD servers and iii) a last one containing the other half of AMD servers.

All servers execute a controllable workload consisting on different tasks of the SPEC CPU 2006 benchmark [21], each requiring a different amount of CPU cores,

arriving with a Poisson statistical distribution. The workload is assigned via the SLURM resource manager [22] that distributes workload across partitions. Thus, each partition exhibits its own workload profile. A Wireless Sensor Network (WSN) developed by the research group is deployed in the Data Center to measure the inlet and outlet temperature of all servers as well as per-server power consumption. Internal server sensors are collected via the Intelligent Platform Management Interface (IPMI) tool that enables us to obtain, for each server: CPU, memory and server ambient temperature, and average fan speed.

Our experimental setup allows full controllability on the data room environmental conditions, as well as on the workload execution, enabling the generation of normal and abnormal training and test sets, in a fully controlled way. In particular, we generate different conditions in the Data centers that lead to two different anomalies:

- Anomalies in the data room cooling due to a CRAC fan failures
- Anomalies in the workload execution.

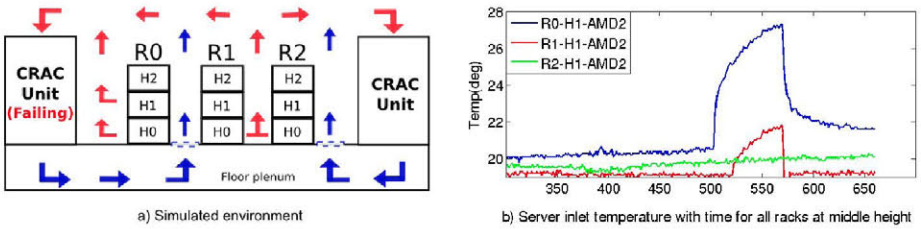
Moreover, these anomalies take place together with anomalies in the sensing infrastructure of the Data Center, i.e. malfunctioning sensors. Anomalies are detected with a Trust and Reputation System Engine, called Trustware-Engine, developed by our research group and implemented using the C++ programming language.

The next subsections describe how each type of anomaly is generated, which are the information sources needed to detect and isolate them, and how random sensor failures can be detected within this scope. To systematize this analysis, we provide results on detection ratios, detection time, and isolation time.

5.2 Anomalies in the data room cooling

In our experimental setup, during the normal operation of the air conditioner, the inlet temperature of the servers varies between 16°C to 23°C. CRAC anomalies can be generated by suddenly turning off the air conditioning unit for a certain time.

For these experiments, we simulate a CRAC failure in a real raised-floor air-cooled real Data Center environment composed of three racks (R0, R1, R2) with servers at three heights (H0, H1, H2) that are cooled via 2 CRAC units. Figure 1a shows the simulated rack and CRAC distribution in the data room, and the failing CRAC unit, whereas Figure 1b shows the inlet and ambient temperature sensor for a server in the middle height (H1) in all three racks.



a) Simulated environment

b) Server inlet temperature with time for all racks at middle height

Fig: 1. CRAC failure simulated environment and server inlet temperature data

The information provided by inlet and ambient temperature sensors of servers at the same rack and height is highly correlated, comes from two different information

sources (WSN and internal server sensors) and is sufficient to detect and isolate CRAC failures. We arrange data in areas according to their physical position in the data center and run the Trustware-Engine to test the anomaly detection with SOM and GNG algorithms, both when all sensors are working properly and when some sensor malfunction exists during the testing phase.

The best results for both cases are obtained with SOM, using a training set of 300 ticks (each tick representing 1 minute) and an n-gram size of 3. Usually, n-gram size varies from 2 to 5. Higher n-gram sizes give more sensibility to anomaly detection but, at the same time, increase the false positive rate [23]. An n-gram size of 3, provides the best tradeoff between detection and false positive rate in our setup.

Figure 2a shows the results provided by the Trustware-Engine for SOM with a CRAC failure starting around tick 500 that highly affects rack 0 (R0), moderately affects rack 1 (R1) and does not affect rack 2 (R2) at all. Red and purple colors represent low reputation values and yellow color represents reputation values near 100 percent. In the horizontal axe, information source IDs are represented for the different racks are presented. CRAC-failures are calculated by averaging the reputation of sensors in the same area. If reputation is below 40, we consider that an anomaly takes place. Figure 2b shows the malfunction of two sensors in Rack 0 (one in H2 and another in H0) around time instant 550. Regarding individual sensors, we consider that a sensor is malfunctioning when its reputation drops below 60 whereas the reputation of its neighbors if stable. Around tick 800 all sensors have a drop in their reputation. Because all sensors provide the same values, our system detects a CRAC anomaly around tick 800, instead of a sensor malfunction.

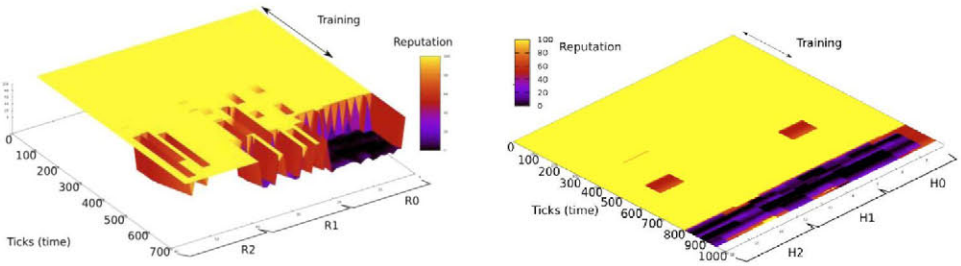


Fig: 2. CRAC fan failure detection and isolation with individual anomalies in sensors

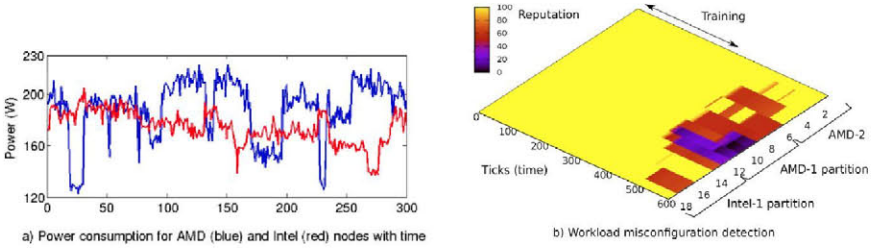
For our experiments, we obtain a CRAC failure detection rate of 100%, with a false positive rate of 0%, a very low detection and isolation time of 2 and 5 ticks respectively and a recovery in reputation values of 40 ticks.

5.3 Anomalies in the workload execution

Detecting anomalies in the workload execution in a heterogeneous Data Center is not an easy task mainly because of the temporal variation usually exhibited by the workload. Power consumption gathered via the WSN shows different profiles depending on the workload under execution and the server architecture (AMD vs

Intel, see Figure 3a). CPU temperature is correlated with power consumption and gathered via the internal server sensors, making these two metrics good candidates to detect anomalies. Because the SLURM resource manager assigns the incoming workload to three different partitions, to detect and isolate anomalies, we arrange the sensors depending on the partition they refer to. In this case, GNG techniques with a training set of 300 ticks and an n-gram size of 2, outperform SOM in terms of false positive rate. Figure 3b shows the detection and isolation of workload anomalies in a rack composed of 9 servers belonging to the three previously described partitions. Around tick 400 servers in AMD2 partition start having an abnormal behavior that extends to more servers around tick 500. When the behavior of the server workload changes partially its reputation drops. To avoid false positives, however, we only consider that an anomaly exists when the area-wide reputation drops below 40.

For our experiments, we obtain a workload misconfiguration detection rate of 100% and again immediate detection and isolation times, as in the previous case.



a) Power consumption for AMD (blue) and Intel (red) nodes with time

b) Workload misconfiguration detection

Fig: 3. Power profile in two different architectures and workload misconfiguration detection with individual anomalies in sensors

6. Conclusions

In this work we have presented a clustering-based detection methodology based on SOM and GNG coupled with reputation systems to detect and isolate cooling and workload anomalies. By making use of sensor topological information and arranging data in different areas we differentiate between individual sensor reputation and area-wide reputation, splitting CRAC and workload data center anomalies from anomalies due to the malfunction of information gathering sensors.

We show how SOM provides better results for CRAC anomaly detections, yielding detection rates of 100%, in training data with malfunctioning sensors. We also show that GNG yields better detection and isolation rates for workload anomaly detection, reducing the false positive rate when compared to SOM. It is important to note the very low detection and isolation rate, that allows rapid actuation upon a Data Center anomaly and that is a very important feature. In the future, we plan to extend our results to wider scenarios and detect anomalies related to the failure of specific server components such as power supplies or fans.

Acknowledgments. Research by Marina Zapater has been partly supported by a PICATA predoctoral fellowship of the Moncloa Campus of International Excellence (UCM-UPM). This work has been partially supported by the Spanish Ministry of Economy and Competitiveness, under contracts TEC2012-33892 and IPT-2012-1041-430000, and INCOTEC.

References

1. Koomey, J. "Growth in data center electricity use 2005 to 2010." *Oakland, CA: Analytics Press. August 1* (2011): 2010
2. Rasmussen, N. "Calculating total cooling requirements for Data Centers", *American Power Conversion*, White paper #25, 2007.
3. ASHRAE, TC. "Thermal guidelines for data processing environments-expanded data center classes and usage guidance." *Whitepaper by ASHRAE TC* 9 (2011).
4. Atienza, D. et al. "Reliability-aware design for nanometer-scale devices." *ASPDAC 2008. Asia and South Pacific* 21 Mar. 2008: 549-554.
5. Banković, Z., et.al. "Self-Organizing maps versus Growing Neural Gas in detecting data outliers for security applications". *HAIS* 2012: 89-96
6. Lima, L., et.al. "Group decision making and Quality-of-Information in e-Health Systems". *Logic Journal of IGPL* 19.2 (2011): 315-332
7. Corchado, E., Arroyo, A., Tricio, V. "Soft computing models to identify typical meteorological days". *Logic Journal of IGPL* 19.2 (2011): 373-383
8. Sullivan, R. F., "Alternating cold and hot aisles provides more reliable cooling for server farms", *Uptime Institute*, 2000
9. El-Sayed, N. et al. "Temperature management in data centers: Why some (might) like it hot." *ACM SIGMETRICS Performance Evaluation Review* 40.1 (2012): 163-174.
10. Romadhon, R. et al. "Optimization of cooling systems in data centre by computational fluid dynamics model and simulation." *Innovative Technologies in Intelligent Systems and Industrial Applications, (CITISIA)* 2009 : 322-327.
11. Haaland, Ben et al. "A statistical approach to thermal management of data centers under steady state and system perturbations." *Journal of the American Statistical Association* 105.491 (2010): 1030-1041.
12. Lee, E., Kyung, H. V., and Dario Pompili. *"Model-based Thermal Anomaly Detection in Cloud Datacenter"*
13. Sedano, J. et al. "Learning and training techniques in fuzzy control for energy efficiency in buildings." *Logic Journal of IGPL* 20.4 (2012): 757-769.
14. Depren, O. et al. "An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks." *Expert systems with Applications* 29.4 (2005): 713-722.
15. Ma, J., Guanzhong D., and Zhong X.. "Network anomaly detection using dissimilarity-based one-class SVM classifier." *Parallel Processing Workshops, 2009. ICPPW'09. International Conference on* 22 Sep. 2009: 409-414.
16. Yuan, Y., Lee, E. K., Pompili, D., Liao, J. "Thermal anomaly detection in datacenters". Journal of Mechanical Engineering Science, 226(8) 2104-2117. 2011.
17. Haykin,S. "Neural networks. A comprehensive foundation", 2nd ed. Prentice-Hall (1999)
18. Fritzke, B. "Growing Neural Gas Network Learns Topologies". In *Advances in Neural Information Processing Systems*, vol. 7, pp. 225–632. MIT Press, Cambridge (1995)
19. Moya, J. et al. "Improving Security for SCADA Sensor Networks with Reputation Systems and Self-Organizing Maps". *Sensors* 9.11 (2009): 9380-9397.
20. Lopez, Javier et al. "Trust management systems for wireless sensor networks: Best practices." Computer Communications 33.9 (2010): 1086-1093.
21. "SPEC CPU2006 Benchmark Descriptions". 2006. 23 Oct. 2013 http://www.spec.org/cpu2006/publications/CPU2006benchmarks.pdf
22. Yoo, A.B., Morris A.J., Grondona, M. "SLURM: Simple linux utility for resource management." *Job Scheduling Strategies for Parallel Processing* (2003): 44-60.
23. Bankovic, Z. "Detecting Unknown Attacks in Wireless Sensor Networks That Contain Mobile Nodes", Sensors 12 (2012): 10834-10850