Contemporary Engineering Sciences, Vol. 11, 2018, no. 94, 4665 - 4679
HIKARI Ltd, www.m-hikari.com
https://doi.org/10.12988/ces.2018.89496

Data Analysis Smart Systems in a Nanodevices - Based Middleware

Alberto López Pacheco[1] and Jose Aguilar[1]

[1] CEMISID, Facultad de Ingeniería
Universidad de Los Andes, Mérida, Venezuela

Ángel Pinto[2] and Juan Torres[2]

[2] Departamento de Ingeniería de Sistemas
Universidad del Sinú, Montería, Colombia

Abstract

In this paper, we propose a Data Analysis Smart System (DASS) for the ARMNANO architecture. ARMNANO is a reflective middleware for the management of nanodevices in a smart environment that has been developed in previous works. The ARMNANO architecture comprises a base level that specifies the nanodevices to sense and act, meanwhile the upper layers contain the set of components for the management of these devices. DASS is one of the upper layers, composed of a set of autonomic cycles (ACs) of nanodata analysis tasks. These ACs are offered as services in the hierarchized architecture of the middleware, and each one has specific goals in the nano-environment platform, in order to generate useful knowledge of the current conditions in the context. The ACs proposed in this paper for DASS are three: one to configure the nanodevices of the context in order to reach certain goals, another to optimize the behavior of the nanodevices, and a last one to fix/determine the environmental variables to reach in the nano-environment due to a sudden change in the supervised conditions in the smart environment (AmI). Each AC is instanced according to the context where ARMNANO is deployed. In this paper, we present a case study to describe the ACs.

Keywords: Data Analysis tasks, autonomic architecture, nano-smart environment

1. Introduction

Nanotechnology has become an important economic factor, and has coped the interest of research nowadays. On the other hand, nanodata is compromising the process of data treatment nowadays. Its massive production claims for new mechanisms to face the big data volume from different sources of nanomaterials and nano-particles, since the body, infrastructures, vehicles, and much more. The nanodata must be quickly and efficiently convert into useful knowledge for applications. Discerning within this information is crucial for reasons attaining health, security, among other. In this sense, are required systems, platforms, etc., which can continually, autonomously and instantaneously collect and analyze nanodata, in order to generate useful information for the context since it is produced.

ARMNANO is a middleware that allows the incorporation of nanodevices in AmIs [1]. This layer is composed of 2 horizontal levels and a transversal level. The 2 horizontal layers are the classical Base and Meta level of a reflexive middleware, in order to reflex and adapt the nanodevices to the requirements in the smart environment; meanwhile, the transversal layer is responsible to analysis the nanodata to help in the reflection and adaptation processes of ARMNANO. Specifically, the base level is composed of the nanodevices and the microgateways in the platform, which are modeled in ARMNANO by the NaS and NaA agents. The meta level holds 7 layers with different agents for the diverse tasks required by ARMNANO, such as data authentication, nano-communication management. These levels have been presented in detail in previous works [1]. The transversal level includes DASS. DASS is an architecture based on ACs of data analysis tasks. It offers the ACs as services, which can be locally or at the cloud, and are smart services of decision making of an AmI. DASS works with the nanodata of the context where ARMNANO is deployed. In this sense, DASS reuse the data generated/used by the different nanodevices of ARMNANO, such as the Nanosensors nodes (NS), Nanoactuator nodes (NA), and Nanorouter (Ro).

In this paper, we present DASS, which is based on the MAPEK loop concept [2, 4], and particularly, in the concept of ACs of data analysis tasks. This concept has been proposed in [2, 3], which is a type of autonomous intelligent supervision that allows reaching strategic objectives around a given problem. The autonomic cycles integrate a set of data analysis tasks, which autonomously and collectively work to achieve the strategic objectives pursued by these cycles [3, 4]. According to this concept, an AC is composed of four elements, for the monitoring, analyzing, planning and **e**xecution of the decisions, and a knowledge base [4]. The ACs continuity and autonomously works, in order to generate the knowledge required by ARMNANO to reach the goals of the AmI.

This paper proposes three generics ACs for DASS, which process the nanodata in the AmI. The first AC defines a set of activities to initially adequate the nanodevices in the AmI to the setpoints of the nano-data for a given context. The second AC is when there is a performance degradation on the AmI or when a new constraint is detected in the AmI by the middleware, and the nanodevices of the AmI must selfoptimize. The last AC is when the setpoints of the nanodata must be recalculated, due to a sudden change in the AmI, which imply self-configurate the nanodevices in the AmI. This paper is organized as follows: first section presents the ARMNANO architecture and the Communication Management Nanoagent (CmNA). The next sections present the design of DASS and, in section 5 the utilization of DAAS in an AmI with nanotechnology. Section 6 compares DAAS with other works. Finally, conclusions are made in section 5.

2. ARMNANO

ARMNANO is a multilayer architecture that provides services for nanodevices in an AmI, offering the ubiquity, real-time monitoring, self-responsiveness, self-learning, as key properties [1]. The layers that belong to ARMNANO are distributed in two levels (see Fig. 1). The base level and the meta level of ARMNANO have a transversal level called the Data Analysis Smart System (DASS), which is in charge of performing the data analysis tasks using nanodata, delivering appropriate knowledge services for the different agents in the AmI. Furthermore, the platform is decentralized and autonomic, in order to allow the self-configuration, the self-repairing, the interconnection, ubiquity in data treatment and commands delivery, and self-analysis and solutions delivery.

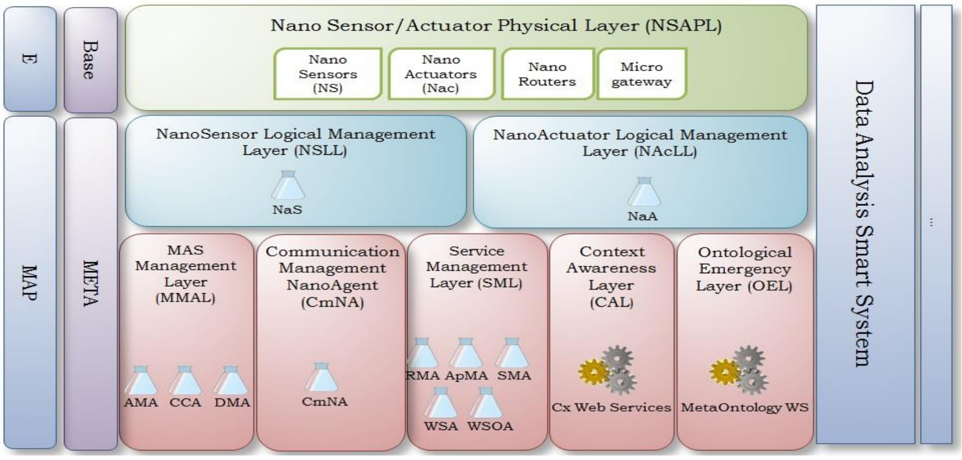

Fig. 1. ARMNANO Architecture

The base level, named NSAPL, contains the physical devices in the AmI, such as the nanodevices and the microgateway. The abstract views of the nanodevices in the NSAPL layer are deployed as agents, called NaS and NaA, in the middle (NSLL and NAcLL). Finally, ARMNANO comprise 5 layers (based on the works [6, 7, 8, ,9]), which are MMAL, CmNA, SML, CAL, and OEL, to provide assistance respect to context characterization and understanding, service provision at the cloud, emergence responsiveness in front of unexpected behavior, etc.Now are described briefly each layer in ARMNANO (for more details see [2]): i) *Nanosensor and Nanoa-*

ctuator Physical Layer (NSAPL): it describes the nanosensors, nanoactuators, nanorouters and microgateway in the AmI. ii) *Nanosensor Logical Management Layer (NSLL)*: It is composed of the logical abstractions of the nanosensors in the AmI, defined as NaS agents, with software for the authentication of the sensed data. iii) *Nanoactuator Logical Management Layer (NacLL)*: It holds the NaA agent, which represent the nanoactuators in the AmI. Their logical functions include to execute the commands and send feedback up to DASS. iv) *MAS Management Layer (MMAS)*: defined by a set of agents standardized by IEEE, called AMA, CCA and DMA, which has been defined previously [7]. v) *Communication Management Nanoagent (CmNA)*: it addresses the communication and authentication protocols for the nano-data. It has been defined in detail in [1]. vi) *Service Management Layer (SML)*: it connects the multiagents (MAS) and SOA paradigms. It is a crucial layer to deploy web services. It has been defined in detail in [6, 7, 8]. vii) *Context Awareness Layer (CAL)*: it deploys context-based services such as context discovery, modeling and reasoning. See [9, 10] for more details. viii) *Ontological Emergence Layer (OEL)*: it deploys services for the emergence of ontologies (see [10, 11, 12] for more details.

3. Data Analysis Smart Systems in ARMNANO

The ARMNANO platform is an autonomous distributed system that provides data analysis services supported in nanodevices. The major part of this data analysis is carried out by the DASS (see **Fig. 1.**). In particular, DASS manages a set of ACs of nanodata analysis tasks in order to make decisions. DASS will execute the set of ACs as services that depend on the context. These ACs are described by three types of data analysis tasks according to [2, 3]: i) *Monitoring Tasks*: are tasks related to observation. This action is performed by the NS nodes. ii) *Interpretation Tasks*: they carry out theprocesses of interpretation of the data. iii) *Decision Making Tasks*: they define the decisions based on the analyzed data, which are sent to the NA nodes. Particularly, in DASS are defined 3 generic ACs, which are **Conditioning**, **Optimizing** and **Fixing**. The multidimensional data model for these ACs is defined in the next section.

A. AC for the Conditioning of the Nano-environment

This AC has the goal to define the set of activities to set the values of the variables in their setpoints, based on the context defined by the nanodata. This AC is activated once the architecture is installed in a given context, and in the case of abnormal conditions. The abnormal situations can be detected when: a) The measured values by the nanosensors are biased respect to the reference. b) There are oscillations at the measurements, even considering some of the measurements are within the range. c) There are too many values out of the trending describing the target behavior. According to the previous issues, when ARMNANO is installed in a new context or there is an abnormal condition, then this AC is activated to provide a solution. The analytical tasks of this AC are (see Figure 2):

Monitoring Tasks: 1. *Capture variables of the nanosensor node*. 2. *Authentication data*: in this task is detected and verified the current situation. It interacts with the NaS agents and the authentication mechanism of CmNA.
Interpretation Tasks: 1. *Determination of current situation*: in this task is determined the new context or the abnormal situation, and its possible causes.
Decision Making Tasks: 1. *Self-Conditioning*: it determines a set of tasks that must execute the NA nodes, in order to reestablish the values of the variables in the AmI that are not in their setpoints.

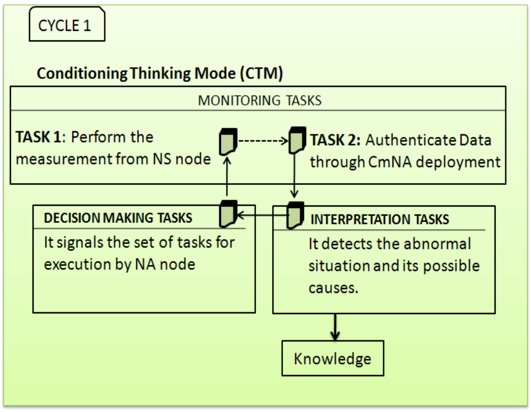

Fig. 2. Autonomic Cycle for Conditioning of the nano-environment

B. AC for Optimizing the Nanodevices

This AC is used when there is performance degradation on the AmI or when a new constraint is detected in the AmI by the middleware. The typical causes to execute this AC are: a) The response time is very large, b) There is a lot of resource consummation, c) The general goals in the AmI are not reached, d) There are new constraints about the nanodevices, the context, among others. In this way, this AC allows the optimization of the ARMNANO architecture deploys in the environment. In this case, the data analysis tasks are (see figure 3):
Monitoring Tasks: 1. *Capture variables of the nanosensor node*. 2. *Authentication data*: in this task is detected and verified the current situation, to be sure of a performance degradation in the AmI. It interacts with the NaS agents and the authentication mechanism of CmNA. This is a monitoring task.
Interpretation Tasks: 1. *Determination of the performance degradation of the AmI*. This task determines the different types of degradation, and the possible causes. 2. *Determination of new constraints in the AmI*. This task determines the constraints in the environment, nanodevices, etc.
Decision Making Tasks: 1. *Self-Optimization*: it determines a set of tasks that must execute the NA nodes, in order to improve the performance in the AmI or follow the new constraints in the environment. In some cases, it requires the definition of an optimization model, to determine the specific activities to be executed in the NA nodes.

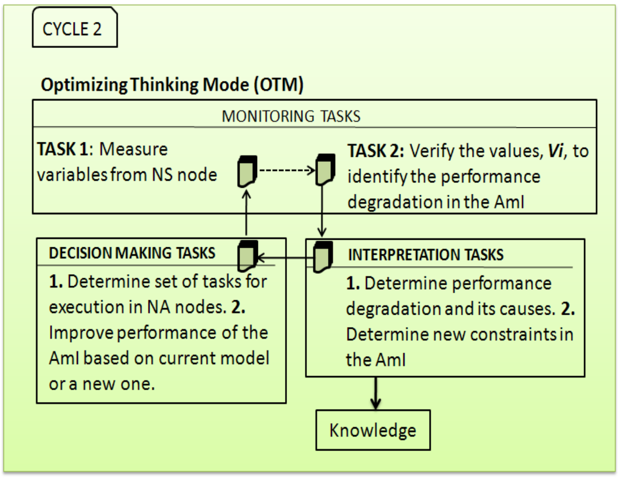

Fig. 3. Autonomic Cycle (AC) for the Optimization of the Nanodevices

C. AC to Fix Setpoints due to a sudden change in the AmI

This AC acts when there is a sudden change in the AmI and it is required quickly to act to reestablish the conditions of performance of the nanodevices in the system. In that way, it is first necessary to determine the target objectives and the variables linked to them to follow, and then, to condition the different nanodevices to follow them. That is, in this case must be determined the references points that the variables must follow, to reestablish the ideal conditions of the supervised condition.
This AC is complementary to the previous Acs, but it supervises the environment in order to react quickly when there is a sudden change at the level of the conditions to be kept in the AmI (for example, the health conditions in a patient in a health smart environment). It has important real-time demands, in order to process quickly the information and act according. This AC acts when: a) There is an abrupt change in the conditions to be kept in the environment, b) A catastrophic failure, c) Impossibility to achieve steady conditions in the target. In this case, the data analysis tasks of this AC are (see Fig. 4):

Monitoring Tasks: 1. *Capture variables of the nanosensor node*. 2. *Authentication data*: in this task is verified the current situation. It interacts with the NaS agents and the authentication mechanism of CmNA.
Interpretation Tasks: 1. *Determination of the degradation of the supervised conditions to be reached by the AmI*. It detects when there are problems in the supervised conditions. 2. *Determination of new setpoints*. This task defines the variables that require new setpoints, and these values.
Decision Making Tasks: 1. Self-Configuration: this task specifies the set of activities to fix the values in the AmI, normally using the NA nodes to this task.

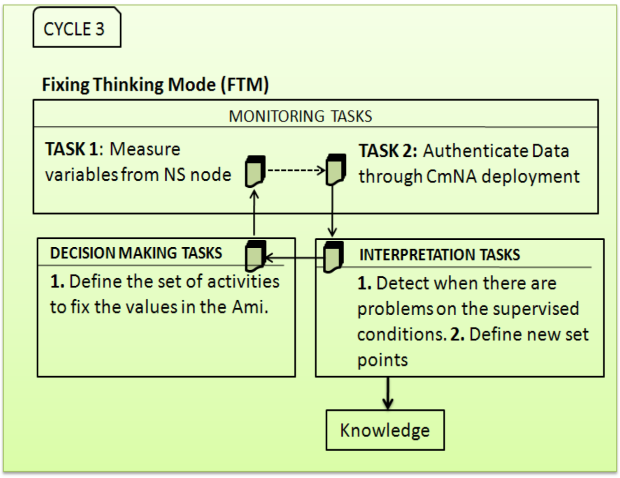

Fig. 4. Autonomic Cycle (AC) for Fixing Thinking Mode (FTM)

4. The Multidimensional Data Model for the Dass

Herein is described the data model required by the DASS, to collect the information about the system where the services are provided. To this aim, it is used a multidimensional model: i) ***User dimension***: it refers to the person or group of persons that directly receive the services from ARMNANO, where the nanosensors and nanoactuators can be implanted. ii) ***Thing/Object dimension*****:** it describes the objects where the nanosensors and nanoactuators can be implanted. An object can receive services from ARMNANO. The object can be, in the context of smart homes, a refrigerator or a kitchen. iii) ***Activity dimension***: it describes the different processes in the environment supervised by ARMNANO, and based on them, the possible actions and functionalities of the nanosensors (in general, the required capabilities of the nanodevices). iv) ***Time dimension***: it describes the parameters as the frequency of measurement, the granularity of the time unit used by nanosensors or nanoactuators, among other things. v) ***Device dimension*****:** here are included the different nanodevices, as the nanosensors and nanoactuators. It describes their functionalities, define their technical features,among other things. vi) ***Service dimension*****:** it describes the services provided by the nanodevices of ARMNANO. vii) ***Variable dimension*****:** it describes the variables (nanodata) that must be observed, analyzed, and modified, in the ARMNANO middleware. Normally, ARMNANO provides services based on the variables observed in the objects/people. This is a generic model that can be used in different contexts (see **Table No. 1**).

Table Nº1. ARMNANO multidimensional model

Context	User	Object	Activity	Time	Device	Service	Variables
Military	A military	weapon	Measure weapon temperature	After each weapon activation	**NS**:Temperature **NA**: deliver polymer-based heat absorber	- Keep weapon performance	Temperature
Health	A sick person	Brain	Monitor malign tumor brain growth, at back cortex	Every minute	**NS**:Voltage and malign cell counter. **NA**: cut gradually the tumor using out-to-in strategy	- Retain brain-back-cortex health . Remove tumor gradually	- Electrical signals - Tumor size

5. Case Study

A. Description of the Case Study

The case study consists of a cancer diseased woman in a hospital that has a smart room to provide health care and uses ARMNANO. ARMNANO allows monitoring constantly health variables, aided by a nanosensors array distributed internally and externally of the patient. This person has the diagnosis: "Breast's cancer patient at intermediate stage".

The breast cancer is defined as the proliferation of cells attaining the breast zone in women (it can appear in man, but are negligible the cases), which initially generate small nodules, but in a regular fashion produce remarkable tumors. As any kind of cancer, this affection is a neoplastic condition that conveys an accelerated cell replication, and propend to flood the whole body (metastasis) if it is not stopped in time. The detected symptoms are the following: a) Infrequent pain in the breast, b) Appearance of nodules closed to both nipples, c) Inflammation in the zone within nipples and arms, c) Beasts' redness and infrequent heating in this zone and around. ARMNANO acts in a decentralized fashion to monitor constantly the patient. As the affection of this individual is referred to as Breast Cancer, then the sensing property is addressed to monitor the specific biomarker to this cancer type, given by the **Cathepsin–D**. In this sense, the adequation of ARMNANO to this patient is: *Nanodevices for this patient:* a) Nanosensors (NS), it is an array of 12 NS injected locally in the breasts to monitor the Cathepsin-D protein, at the level of nmol/g (normal Cathepsin-D level in women is 10 nmol/g or less), b) Nanoactuator (NA), it is a12 NA-composed mesh able to perform mainly 2 tasks: b.1) 6 NA devoted to destroy/cut fatty tissue or malignant tissue (nodules, tumors, etc.), b.2) 6 NA devoted to inject locally the chemotherapy, c) Nanorouter (Ro),there are 2 Ro deployed to transmit the measurement values until the Mc at the frequency of observation of 1 / 6 min (see **Table 3**), d) Microgateway (Mc), deploys the CmNA agent to recognize the validity of Cathepsin-D concentration measured in blood.

DASS for this patient: the three ACs defined in section 5 are invoked in different moments, according to the events in the context in a given moment. Now, we describe its instantiation.

B. Multidimensional Data Model

Herein is observed uniquely the Cathepsin-D protein concentration ([Cathepsin-D]), thus the instantiation of the dimensions is around this variable (see Table 2).
User dimension: the patient, the doctors and the nurses.
Object dimension: the breasts and rest of woman body, the bed and the rest of objects in the room.

Device dimension: it is conformed by the nanosensors (12 nodes) and nanoactuators (12 nodes). Each of these nodes acts independently and reports to their respective nanorouters (2 Ro). The Mc unit is common (see CmNA layer of ARMNANO). Other devices are depending on the necessities: communication antennas, medical devices, etc. They must respond to the context requirements.

Service dimension: the main service delivered by ARMNANO is to keep the Cathepsin-D concentration under control. To this objective, it brings two services: to inject the chemotherapy locally, or to destroy the nodules in the breast zone as long as they appear. The continued monitoring of ARMNANO allows determining when these actions must be done.

Activity dimension: There is a continuous observation of the cancerous condition of the woman patient, in order to decrease the tumor size, and avoid the nodules replication. The woman perceives a constant attention by the ARMNANO actions that inject the chemotherapy locally, or extract the malignant tissue, among other things.

Variable dimension: the main observed variable is the Cathepsin-D concentration. Other variables observed in the patient are the temperature and arterial pressure, among others.

Table Nº2. Instantiation of the Data Model for the Breast Cancer case study

User	Object	Activity	Time	Device	Service	Variables
Woman patient Doctors, etc.	Both Breasts	Measure Cathepsin-D concentration	Every 6 min	Cathepsin-D sensible sensor	Keep woman with regular Cathepsin-D concentration level	Cathepsin-D concentration, temperature and arterial pressure.

At the Table 3 are signaled the set points of the variables in this case study. The first AC uses these set points to set the initial activities.

Table Nº3. Data model set points for the multidimensional view in case scenario

Variable	Nanodevice	Set Points
Monitored [cathepsin-D]	Nanosensor	25 nmol/g in blood
Frequency of Observation	Microgateway	1 / 6min
Period of Analysis	Microgateway	24/7
Number of NS	Microgateway	12
Size of the target	Nanoactuator	0,1 m^2
Context features of target	Nanoactuator	Blood density

We suppose that the sick woman is about to suffer metastasis, due to that, the level of the biomarker is approximately high, ranging 25 nmol/g as signaled in the set points (see Table 3). The nanosensors are spread out the breast zone, and report 12 values every 6 min (see the frequency of observation in Table 3). ARMNANO allows a 24/7 monitoring, leaving no space to lose drastic change detection in the sick woman.

C. AC for Conditioning the Nanodevices

The woman patient is injected in the breasts with 12 NS, and 12 NA, distributed 6 at each breast. The actions are foreseen to be symmetrical, in order to keep an equilibrium at chemotherapy dose applications. Also, it is included 1 Ro for the NSs and 1 Ro for the NAs. They are synchronized with the external Mc unit according to the set points signaled in Table 3. The AC tasks for this case study are explained as follows:

Monitoring Tasks: 1. The 12 NSs perform the regular measurement, according to the frequency determine in Table 3. 2. The cathepsin-D concentration values are validated.
Interpretation Tasks: 1. They determine if there is an abnormal condition given by an augmentation of the cathepsin-D concentration.
Decision Making: 1. One decision is a soft chemotherapy treatment that starts with the application of chemical inhibitors locally and non-invasively.

According to the tasks, the affected person is treated autonomously by the AC capabilities, so the ARMNANO architecture assures to run appropriately in the woman patient with the initial set points standardized according to Table 3.

D. AC for Optimizing the Nanodevices

In this scenario, there is an abnormal distribution of the measured values in the Cathepsin-D concentration. Thus, it can misguide a correct interpretation in the patient. Also, the data authentication task is not converging fast. This scenario is an ARMNANO degradation. The continued failures in ARMNANO generate the invocation to the "Optimizing Thinking" AC. The AC tasks for this case study are explained as follows.

Monitoring Tasks: 1. The observation is made partially. 2. Some values have not been appropriately authenticated.
Interpretation Tasks: 1. There is a one or more NSs that fail.

Decision Making: 1. The failure is corrected and tested. the NS nodes are updated, by substituting the failing ones with new.

E. AC to Fix Setpoints due to a sudden change in the AmI

The woman patient reaches a critical nodule formation, such that the malign cellular formation rate seeds new tumors. That is, in this scenario, there is a Breast cancer patient with abrupt tumor nodule formation. The “AC to Fix Setpoints due to a sudden change in the AmI” is called, in order to determine the gravity level of the individual, and to perform a decision making enabled to diminish the adverse effects the quickest possible (main advantage of continuing observation with ARMNANO). The tasks to this new case scenario are.

Monitoring Tasks: 1. The 12 NSs measure the Cathepsin-D concentration. With these values is determined a critical damaged tissue formation.
Interpretation tasks: 1. Both breasts show loss of consistency and shape.
Decision Making Tasks: 1. 6 NAs are commanded to remove the nodules that generate bad cellular production. These NAs acts locally to remove this tumor. 6 NAs are commanded to inject a chemical inhibitor.

This AC acts until the nodules appearance was stopped, since the damaged tissue was deleted physically. The NAs possess the advantage to act non-invasively. Damaged tissue is cut from inside and extracted with a cannula pipe at both breasts.

6. Comparison with Previous Works

This section carries out a comparison of several middleware from the perspective of the next criteria: i) They support the ACs paradigm. ii) They propose the utilization of data analysis tasks (DA). iii) What is the context of application (CA): transportation, health, communication, military, etc.

Table Nº4. Comparison among different middleware

Work	AC	DA	CA
Our approach	MAPEK loop	3 ACs of DA tasks (3 modes): Conditioning, Optimizing and Fixing.	Health, Smart cities, Smart homes.
[16]	NA	1 mode: Predictive analysis for the police	Smart Cities
[13]	NA	1 mode: Rules-based data analysis	Smart Buildings
[14]	NA	1 mode: Predictive analysis	Wearable technology, automotive, medical field
[15]	Autonomic paradigm	1 mode: Self-management of Cloud platform	Smart cities

According to the Table 4, previous works do not develop ACs. To our perspective, this is disadvantageous, since an AC allows a permanent discovery of knowledge, which can improve the response times. Particularly, DASS in ARMNANO select the appropriate AC according to the event in the context. It proposes different DAs according to the needs in a given moment. Additionally, new ACs can be included in ARMANO, if the specific context requires it. It is a scalability property very important in our approach.

In the previous works are proposed data analysis tasks, but not like ACs. They do not use the MAPEK concept for an autonomic answer of the middleware, according to the events in the environment. Additionally, the reach of these architectures is limited when facing a variety of events in a context. DASS capabilities are provided with 3 modes: to set the initial context, to optimize in case of a system degradation or failure, or to provide immediate answering when there is an abrupt change in the context. The data analysis ACs are fitted to each situation; thus, the response is more specific.

In general, the previous works lack the autonomic property, and are able to provide data analysis in specific contexts, such as wearable technology or smart cities, but they are not generic, and require constant external supervision. ARMNANO acts autonomously in a decentralized fashion, to provide solutions in the form of services to a wider spectrum of contexts.

7. Conclusions

The nanosensors field has received few attentions in the domain of adaptive computing architectures. One of the reasons for this is the difficulty to work with this size of the matter, and the fear to progress into non-invasive platforms of massive data recording and reporting, without clear rules. These issues were visualized at our ARMNANO middleware, which looks to provide effective and efficient solutions for supporting nanodata based applications. The ARMNANO architecture is composed of the hardware layer with nanosensors (NS), nanoactuators (NA) and nanorouters (Ro). In the microgateway unit are developed important logical functions as the data validation in the CmNA layer, and DASS allows the nanodata analysis tasks, and decision-making processes.

This last is the core of this paper, which proposes three AC for ARMNANO. The first AC allows the self-configuration of the nanodevices in the AmI, with the setpoints of the nanodata for a given context. The second AC allows the self-optimization when there is a performance problem or new condition in the AmI, and the last AC redefines the setpoints of the nanovariables, due to a sudden change in the AmI, and according to them self-configurate the nanodevices.

In the case study to provide services in a Breast Cancer Patient has been demonstrated the versatility, robustness and efficiency of our middleware. The different ACs were applied in different situations: The first was applied initially for a constant monitoring of the patient, after its diagnosis. In this sense, the patient was observed 24/7. The second AC was run to test the architecture degradation. It occurred when some values did not follow their normal distribution or when several NS fail, which was autonomously corrected with new NS. The last AC was run when an abrupt change happened in the woman patient, due to the detection of nodules appearance. In this case, the NA nodes were configured to submit continuous chemical inhibitor doses, or to remove the nodule formation.
Next works will involve the development of an appropriate ontology for the data analysis tasks in ARMNANO. Also, the implementation of the ACs in real contexts, and the generalization of their tasks in different contexts, are specific studies to carry out.

References

[1] A. Lopez-Pacheco, J. Aguilar, Autonomic Reflective Middleware for the Management of NANOdevices in a Smart Environment (ARMNANO), Submitted to Publication, 2018.

[2] J. Aguilar, J. Cordero, O. Buendia, Specification of the Autonomic Cycles of Learning Analytic Tasks for a Smart Classroom, *Journal of Educational Computing Research*, **56** (2017), 866-891. https://doi.org/10.1177/0735633117727698

[3] J. Aguilar, C. J., L. Barba, M. Sanchez, P. Valdiviezo, L. Chamba, Learning Analytics Tasks as Services in Smart Classroom, *Universal Access in the Information Society Journal*, **17** (2018), no. 4, 693–709. https://doi.org/10.1007/s10209-017-0525-0

[4] M. Huebscher, J. McCann, A survey of autonomic computing: degrees, models and applications, *ACM Computing Services*, **40** (2008), 1-28.

https://doi.org/10.1145/1380584.1380585

[5] J. Vizcarrondo, J. Aguilar, E. Exposito, A. Subias, ARMISCOM: Autonomic reflective middleware for management service composition, *2012 Global Information Infrastructure and Networking Symposium*, (2012). https://doi.org/10.1109/giis.2012.6466760

[6] J. Aguilar, M. Jerez, M. Mendoca, M. Sanchez, MiSCi: Autonomic Reflective Middleware for Smart Cities, *Technologies and Innovation*, Vol 658, Springer, 2016, 241-253. https://doi.org/10.1007/978-3-319-48024-4_19

[7] J. Aguilar, A. Rios, F. Hidrobo, M. Cerrada, Sistemas Multiagentes y sus Aplicacions en Automatizaion Industrial, Talleres Graficos. Universidad de los Andes, 2013.

[8] M. Sánchez, A. Aguilar, J. Cordero, P. Valdiviezo, Basic features of a Reflective Middleware for Intelligent Learning Environment in the Cloud (IECL), *Asia-Pacific Conference on Computer Aided System Engineering*, (2015). https://doi.org/10.1109/apcase.2015.8

[9] M. Sánchez, A. Aguilar, J. Cordero, P. Valdiviezo, A Smart Learning Environment based on Cloud Learning, I*ntertl Journal of Advanced Information Science and Technology*, **39** (2015), 39-52.

[10] M. Jerez, J. Aguilar, E. Exposito and T. Viellemur, CARMiCLOC: Context Awarenes Middleware in CLOud Computing, L*atin American Computing Conference (CLEI)*, (2015), 532-541. https://doi.org/10.1109/clei.2015.7360013

[11] M. Mendoca, J. Aguilar, N. Perozo, MiR-EO: Reflective Middleware for Ontological Emergency in Intelligent Environments, *Latinamerican Journal of Comp.*, **3** (2016), 25-39.

[12] J. Aguilar, M. Jerez, M. Sanchez, M. Mendoza, Emergencia Ontológica Basada en Análisis de Contexto, como servicio para Ambientes Inteligentes, *DYNA*, **84** (2017), 28-37. https://doi.org/10.15446/dyna.v84n200.59062

[13] P. Arjunan, M. Srivastava, A. Singh, P. Singh, OpenBAN. An Open Building ANalytics Middleware for Smart Buildings, *Proceedings of the 12th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, (2015), 70-79. https://doi.org/10.4108/eai.22-7-2015.2260256

[14] R. Kumar, Smart Micro/Nanosensors and their Application in Intelligent Sensory Nertwork System, *Intl. Journal of Sensor Networks and Data Communications,* **7** (2018). https://doi.org/10.4172/2090-4886.1000e113

[15] F. Pop, C. Dobre, A. Costan, AutoCompBD: Autonomic Computing and Big Data Platforms, *Soft Comput.*, **21** (2017), 4497-4499. https://doi.org/10.1007/s00500-017-2739-8

[16] G. Lu, Z. Li, W. Zhang, L. Yang, Autonomic Deployment decision Making for Big data Analytics Applications in the Cloud, *Soft Computing*, **21** (2017), 4501–4512. https://doi.org/10.1007/s00500-015-1945-5

Received: September 28, 2018; Published: November 7, 2018