



# **Towards Self Healing Publish/Subscribe System on MANET**

Imene Lahyani, Ismael Bouassida Rodriguez, Mohamed Jmaiel, Christophe  
Chassot

## **► To cite this version:**

Imene Lahyani, Ismael Bouassida Rodriguez, Mohamed Jmaiel, Christophe Chassot. Towards Self  
Healing Publish/Subscribe System on MANET. 2012. hal-00694385v1

**HAL Id: hal-00694385**

**<https://hal.science/hal-00694385v1>**

Submitted on 4 May 2012 (v1), last revised 26 Apr 2012 (v2)

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Towards Self Healing Publish/Subscribe System on MANET

Imene Lahyani<sup>1,2,3</sup>, Ismael Bouassida Rodriguez<sup>1,2,3</sup>, Mohamed Jmaiel<sup>3</sup>, and Christophe Chassot<sup>1,4</sup>

<sup>1</sup> CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France

<sup>2</sup> Univ de Toulouse, LAAS, F-31400 Toulouse, France

<sup>3</sup> University of Sfax, ReDCAD, B.P.W, 3038 Sfax, Tunisia

<sup>4</sup> Univ de Toulouse, INSA, LAAS, F-31400 Toulouse, France

{iabdenna,ibouassi,chassot}@laas.fr; mohamed.jmaiel @enis.rnu.tn

**Abstract**—This paper presents a novel approach for Quality of Service aware publish/subscribe systems on Mobile Ad-hoc Network. Crisis management application seems to be a relevant case study to our approach. On the one hand, our proposed solution monitor the system and analyzes his state towards predicting Quality of Service degradations. Real time prediction is based on the Auto Regressive Integrated Moving Average Formula. On the other hand, our solution defines new reconfiguration actions to be applied once Quality of Service degradation are predicted. In order to evaluate the proposed reparation actions, we define a protocol describing different transformation rules for adaptation using graphs and graph grammar. The Graph Matching and Transformation Engine (GMTE) is used to apply rules to the publish/subscribe system.

**Keywords**—MANET, publish/subscribe, crisis management applications, Quality of service, reconfiguration actions, Graph grammar.

## I. INTRODUCTION

Mobile Ad Hoc Network (MANET) is a network consisting of a set of mobile hosts communicating with each other without the assistance of base stations [12]. MANET includes mobile nodes that can dynamically self organize into arbitrary ad-hoc network topologies, allowing people and devices to seamlessly work in areas without a preexisting communication infrastructure such as, disaster recovery environments. An efficient way to communicate nodes in such environments is publish/subscribe (pub/sub) paradigm [3] due to the decoupling feature characterizing communicated entities.

A typical case study to our study is crisis management systems (CMS). CMS is composed of a set of mobile devices that communicate via a pub/sub architecture. CMS involve assessing potential threats and finding the best ways to avoid these threats. In such urgent situations, crisis management requires dealing with serious situations before their occurrence. Thus, CMS require high Quality of Service (QoS) and need to consider several criteria's that could affect the quality of nodes and links in the network.

Facing this problem, several studies [5], [11] react at the middleware level by introducing some reconfigurations actions in order to repair the system. Although the mentioned systems ensure a better system state by substituting faulty

links, they do not give any guarantee to have a fault free system. The main point of failure of these systems remains failure occurrence since its undesirable effects on the system especially on CMS.

This paper provides quality of service (QoS) guarantees in an efficient and scalable manner for CMS where (pub/sub) systems are deployed on MANET. The paper elaborates on an architecture that can support QoS in case of dynamic topology. It presents a decisional model aiming to monitor the system and analyzes his state at runtime by means of a predictive approach using the Auto Regressive Integrated Moving Average model (ARIMA) to forecast future latency values. The same method is used to estimate battery level at runtime towards avoiding nodes crash. Once a QoS degradation is predicted, diagnostic module reasons about the cause of the degradation. This creates opportunities to repair the system. Reconfiguration actions may be introduced to ensure system survivability when QoS degradation is predicted. To achieve the mentioned goal, adaption can be accomplished by migrating a faulty broker to another node in the network ensuring a better QoS. Adaption may also be achieved by splitting an overloaded broker in order to minimize his load. An other alternative is to substitute faulty link by introducing a broker in the link.

In order to evaluate the proposed reconfiguration actions at design time, we propose to build a dynamic protocol of the architecture defining transformation rules. Grammar graph is used to describe the architecture. The Graph Matching and Transformation Engine GMTE is used to apply transformation rules to the graph describing the architecture. This enables to give a general overview describing the network after applying reconfiguration actions.

In the following of this paper, we first survey existing systems dealing with QoS aware pub/sub system. In section II, we present the decisional model predicting failure occurrence. A detailed description of the proposed reconfiguration actions is described in Section III. Thereafter, we present, in Section IV, tools used to evaluate reconfiguration actions on the monitored system. Finally, we conclude and discuss future work in Section V.

## II. RELATED WORK

Work dealing with QoS aware pub/sub system when deployed on MANET do not adopt predictive schemes to fulfill their aims. We here present these systems classified in two categories.

In the first one, systems [5], [11] react after failure occurrence by proposing a reconfiguration action as a solution to link break or removal. These systems aim to ensure system survivability by maintaining connected peers. The mentioned systems ensure a better system state by substituting faulty links. However, they do not give any guarantee to have a fault free system.

The second category reacts before failure by periodically optimizing system performance taking into account some QoS parameters. For instance, Q[10] which is an infrastructure for pub/sub systems adapts the overlay network to the changing topology by means of cross-layer interaction. Indeed, to increase the efficiency of communication, the overlay network interacts with the routing layer to obtain information concerning the topology of the network. Armed by these information, it reconfigures itself in a way to keep the shortest paths between brokers. However, the shortest path is not always the optimal path ensuring a minimum of latency. In response to changes in physical network, opportunistic overlays introduced in [2] dynamically adapt event dissemination structures (i.e., broker overlays) with the goal of optimizing end-to-end delays in event delivery. As shortcoming, [2] does not provide a monitoring module to supervise the system, nor a mechanism to specify latency requirement of applications. However, in our work, we continuously monitor latency and we predict failures by computing estimated latency with adaptive threshold. Finally, the work in [1] proposes a QoS aware messaging system called Harmony. Separately, for each message topic, Harmony specifies performance requirements. Therefore, this constraint renders this approach applicable only for topic based pub/sub systems. Our approach, however, is not limited to a specific subscription language and is considered as a generic approach.

Besides, contrarily to Harmony, we use adaptive threshold according to the requirements of applications and the variation of the network. Furthermore, Harmony uses a technique consisting in sending topics across multiple paths continuously. This assumes the existence of multiple paths in the network and restricts the approach to only cyclic brokers' topology. However, our approach supports general broker overlay topologies and is not restricted to a specific topology. Although these approaches react before failure, none of them use a predictive strategy. However, all of them optimize the system periodically which can lead to wasting resources by performing unnecessary optimizations operations. Besides, the failure may occur in the system and there is no guarantee to have a free failure system. Thus,

up to now, the proposed solutions do not succeed to find an efficient solution to the problem. In this paper, we propose a failure prediction approach based on statistical methods to overcome shortcomings raised by the mentioned work. We ensure system survivability and safety by avoiding failures occurrence and prohibiting QoS degradations.

## III. THE PROPOSED APPROACH

The general objective of a crisis management systems (CMS) is to ensure that an abnormal or a catastrophic situation is detected and solved at time and does not go out of hand. Thus, it may require a minimum delay to manage the crisis or even minimize it by handling the situation.

The CMS is composed of a set of mobile devices that communicate via a pub/sub architecture. Thus, the CMS seem to be a relevant case study for our evaluation. CMS require QoS guarantee prohibiting communications interruption or even degradation.

In this paper, we describe our autonomic approach adding valuable self-adaptation to existing pub/sub systems without lowering their performances and characteristics. On the one hand, the proposed approach introduces a decisional model in nodes towards monitoring and analyzing nodes and link states. On the other hand, we describe new reconfiguration actions that can be applied to the system when QoS degradation are predicted. However, how to choose the appropriate reconfiguration action ? How to have a general overview on the network after applying reconfiguration actions?

To reach this goal, we propose to build a dynamic protocol that evaluate repairing actions at design time. A first level of the solution consists in giving network modeling. A second level can be reached by generating rules describing adaptation actions. Finally, the transformation engine GMTE is used to apply rules to the network and to generate the new architecture of the network after repair. This gives the opportunity to evaluate rules regarding performance parameters such network survivability.

### A. The decisional module

The decisional module involves monitoring and analysis component. It introduces in each node a component responsible for monitoring and analyzing system state towards providing quality delivery to crisis management applications. The proposed analytical module monitors latency and battery level using the Auto regressive Integrated moving average formula (ARIMA). Finally, decisional module decides about node and link state and reasons about the source of degradation once predicted.

Figure 1 describes the proposed analytical module.

1) *Monitoring approach:* Monitoring the system aims to observe and store relevant QoS parameter values to assess the current system health. MANETs suffer from limited nodes resources and frequent nodes mobility. Thus, in order to supervise the system, we monitor node state as well as link

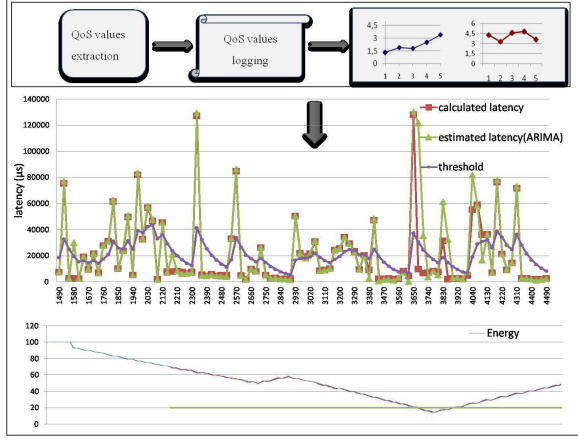


Figure 1. Decisional module based prediction approach

Algorithm 1. Latency monitoring

```

1: On_publish event()
2: sending_time = System.currentTimeMillis()
3: send_neighbor(event, sending_time)
4: On_receive event()
5: Receiving_time = System.currentTimeMillis()
6: latency = Calcul_latency(Receiving_time,
7: sending_time, Offset);
8: save_log(latency)

```

state. Monitored QoS parameter for node is battery level. Monitoring module tries to further monitor for total energy consumption in each node and thus conserves network lifetime. Besides, monitoring link state needs to supervise latency measurements at runtime.

♦**latency**: Latency is defined as the time elapsed between the time,  $T_1$ , when the first broker  $B_1$  sends the first bit of a packet to his neighbor  $B_2$  and the time,  $T_2$ , when the second receives the last bit of that packet at  $T_1 + \Delta T$ . Thus, the latency that takes the event between  $B_1$  and  $B_2$  is defined with the following formula:

$$latency(B_2, B_1) = T_2 - T_1 \pm (offset) \quad (1)$$

where the clock offset refers to difference in the time reported by the two clocks. Monitoring module integrated in each broker intercepts event when translating between neighboring brokers in order to compute latency value. However, if there is no event delivery during a long time, specific messages are used in order to analysis QoS link state. Latency is calculated locally by the broker receiving the event as depicted in Algorithm 1.

♦**Battery level**:

One important characteristic of MANETs is that nodes are energy constrained. Nodes are battery-operated and frequent recharging or replacement of batteries may be undesirable

or even impossible. This makes energy an important QoS parameter, that we should supervise. The above mentioned QoS parameter is located at the OS level and low level of the protocol stack. Thus, we explore a multi layer approach in order to provide information to the middleware layer.

The final step related to the monitoring module is saving QoS parameter values in log files.

2) *Analysis approach*: To deal with QoS issues, we propose a multi objective QoS model for pub/sub system on MANET.

The proposed approach enables energy efficient by pro-actively analyzing energy level per node towards reacting before node crush. Moreover, the analysis module analysis latency by estimating latency values using ARIMA model and comparing these values with adjustable thresholds. Adaptation is done using the Auto Regressive Integrated Moving Average Formula.

♦**latency analysis**: It's based on comparing estimated latency values with adaptive thresholds.

Analyzing monitored data allow us to detect symptoms that indicate an upcoming failure. Thus, by applying basic mathematical model related to ARIMA model on the monitored latency, we can rewrite :

$$Latency_t = \mu + \sum_{i=1}^p (\varphi_i) * (Latency_{t-i}) + \varepsilon_t - \sum_{i=1}^q (\Theta_i * \varepsilon_{t-i}) \quad (2)$$

Failure prediction is performed by comparing estimated latency with appropriate threshold. In our approach, we take use of adjustable decision threshold. Adaptive thresholds are computed dynamically tacking into account the latest available information using the Exponential Moving Average Formula [4].

$$S_{t+1} = \alpha Latency_t + (1 - \alpha) S_t \quad (3)$$

where  $\alpha$  is a constant and  $Latency_t$  is the measured latency value and  $S_t$  is the threshold value at present time.

♦**Energy analysis**: Node crush may cause system functionality interruption. Thus, energy degradation should be predicted at runtime. In order to forecast energy measures, an algorithm is needed though, that will estimate energy values in the future, then compare these values with thresholds. Therefore, we propose a three step algorithm that first computes estimated energy values using the Auto Regressive Integrated Weighed Moving Average Formula(ARIMA) [?] by applying basic mathematical model related to the ARIMA model on the monitored energy:

$$energy_t = \mu + \sum_{i=1}^p (\varphi_i) * (Energy_{t-i}) + \varepsilon_t - \sum_{i=1}^q (\Theta_i * \varepsilon_{t-i}) \quad (4)$$

Second, the proposed algorithm compares estimated latency with threshold and then trigger alarms when QoS violation

is noticed. Generally, battery level is considered to be critical when it becomes under 20%.

3) *Diagnostic approach*: After predicting a QoS degradation, we execute an alternative algorithm that effectively looks for the source behind latency degradation.

The key idea of source degradation identification is to study the dependency to any statistical relationship between two random variables or two sets of data. Doing this, we use the correlation method[6] towards studying the relation between latency value variation with parameters affecting latency such as hop count variation or overload variation.

Correlation method requires two steps. The first step consists in computing the Pearson correlation coefficient  $PCC$  value studying the dependency between two distributions  $X1$  and  $X2$  as described in the following equation.

$$PCC = cov(X1, X2) / \sqrt{(var(X1) - var(X2))}$$

The second step consists in executing a significance test of the correlation coefficient by comparing the computed coefficient with a theoretical value issued from the table of the critical value related to  $PCC$ . Thus, if  $PCC(X1, X2)$  is positive and greater than theoretical  $PCC$ , this means that  $X1$  and  $X2$  are positively and significantly correlated.

According to the significance of the computed correlation coefficient, our algorithm proceeds in different scenarios resumed in Figure 2.

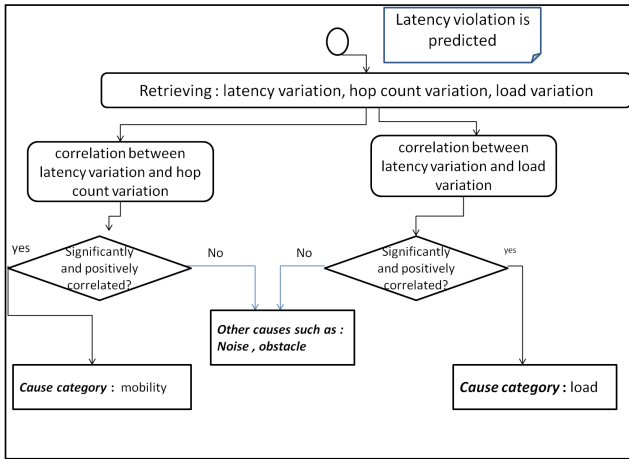


Figure 2. Diagnostic approach

### B. The proposed reconfiguration actions

Once a QoS degradation is predicted, we introduce reconfiguration actions in order to repair the system and ensure its safety and survivability required by CMS.

#### 1) Migration operation

Migration operation means code mobility of brokers as well as data encapsulated within it and which are necessary to its execution. Migration operation

consists in three different steps: Initialization, Data Transfer and finally Informing neighbors about the new locality.

#### 1. Initialization

The first step in broker migration consists in preparing the destination node to host the mobile broker. To this end, a new broker is launched in the selected node. This broker has initially an empty subscription table.

#### 2. Data Transfer

It is the most important step in the migration operation. It consists in sending the subscription table to the new broker deployed in the last step. The latter updates its routing table. Consequently, when the new broker receives an event corresponding to one of the subscription stocked in its routing table, it notifies the corresponding subscriber. The defaulting broker sends each subscription received during the migration process to the new one. This prevents messages from being lost. During this step, the initial broker continues to work.

#### 3. Informing Neighbors about the New Locality

After the reception of the subscription table, the new broker becomes able to notify the subscribers figuring in its subscription table up on receiving an event. This is not the case for forwards which cannot deliver events to the substituting broker since they do not know its location. In order to solve this problem, we intend to inform forwards about the new location of the defaulting broker. So the new broker looks at the list of forwards and sends to them its address, which corresponds to the new location of the broker. During the migration time, the defaulting broker continues accepting messages from forwards until the end of the migration. This ensures that the system continues to work during migration operation. Once informed about the new location of the defaulting broker, the forwards start sending messages to the substituting broker. Consequently, we ensure that messages are never lost during the migration operation. Finally, the old broker shut down when the new broker starts its operation.

Figure 3 summarizes migration operation. Our previous work [9] gives more detail about this adaptation action.

- 2) Splitting nodes: An overloaded broker becomes enable to bypass messages to neighbors. An advantageous solution is to split the broker and especially his subscription table to two others nodes in the network.
- 3) link substitution: When the link becomes overloaded, substitution is a beneficial solution. This may be done by intercalating a node in the defaulting link.



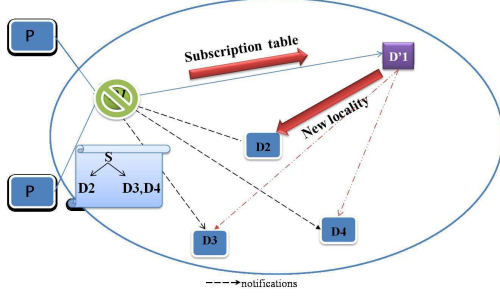


Figure 3. Migration operation

### C. Exploration of deployment solutions for publish/subscribe systems on MANET

After enumerating the proposed reconfigurations actions, we aim to explore deployment solutions for pub/sub systems on MANET. Doing this, we plan to build a simulator that give first a modeling of the pub/sub system on MANET. Second, the simulator describes rules to each reconfiguration action and finally generates the output graph by applying rules using GMTE<sup>1</sup> as a transformation engine.

1) *System modeling*: Pub/sub system on MANET is modeled by a graph AG containing logical as well as physical nodes and links. Thus, in the proposed graph, there is two types of vertices as follow:

- vertices modeling logical entities, ie, brokers, characterized by : ID broker, IP node, broker state.
- vertices modeling physical entities, characterized by :IP node, energy, node state.

Also, there is two types of arcs. The first one indicates a physical or a logical link characterized with: type(either physical or logical),hop count between brokers, link state. For example:

- (P, -, Hop, state): indicates a physical link (1 hop),
- (-, L,Hop, state): indicates a logical link (n hops),

2) *Rules description*: The rule represents the reconfiguration action to apply. Rules are modeled by a rule graph RG containing 4 zones: Inv, Abs, Del, Add, where :

- Inv is the fragment that remains invariant before and after the application of the rule.
- Del is the fragment that will be deleted while the rule is applied.
- Abs is the fragment that should be absent from the graph in order to apply the rule.
- ADD is the fragment that will be added after the application of the rule.

3) *The output Graph*: After applying rules on the input graph modeling network configuration, the transformation engine GMTE generates the output graph modeling the network with a new configuration.

<sup>1</sup>The Graph Matching and Transformation Engine (GMTE), available at <http://homepages.laas.fr/khalil/GMTE>

## IV. CASE STUDY

CMS involves structured groups of robots or military personnel who cooperate for the realization of a common mission. Such application is composed of participants having different roles: a global controller of the mission, coordinators, and several groups of investigators. Each group of investigators is supervised by a coordinator. This hierarchy could be schematized as follows:

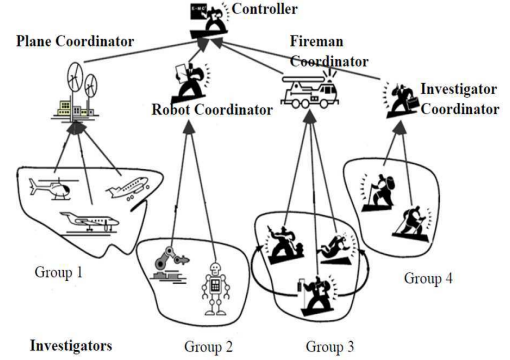


Figure 4. CMS architecture

We propose to elaborate communication between mobile investigators through a content based pub/sub system. In fact, the weak coupling between information producers' and consumers' identities provides P/S system the advantages of asynchronous, multi-point communication, hence freeing the communicating entities from temporal and special requirements for connection.

The communication begins when an investigator either advertises or publishes damage assessment information to find out some rescue in a particular region. Other members can selectively subscribe to the information germane to their task. Typically, the pub/sub system matches the published messages with the dynamically changing interests and routes them to the appropriate recipients.

Every investigator in the network is potentially mobile, it can join, leave the system and have access to the pub/sub service, while also acting as a broker for message forwarding.

Let consider a section of investigator. The graph AG corresponding to the section of investigators (group 3) is depicted in Figure 5.

Our decisional module predicts a QoS degradation in node ( $N_2$  in Figure 5) , Thus, applying the migration rule to the graph in Figure generates the new graph (Figure 6)

Future work will evaluate the output graph in order to decide about the appropriate reconfiguration action to be applied in case of QoS degradation. Parameters to evaluate the better output graph may involve liveness of the network. Thus, the better reconfiguration action corresponds to the rule that ensure a longer lifetime to the network.

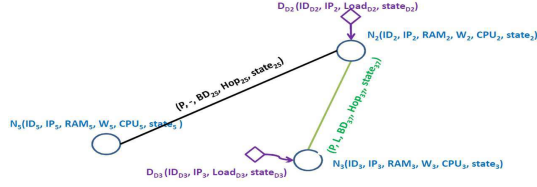


Figure 5. Network configuration

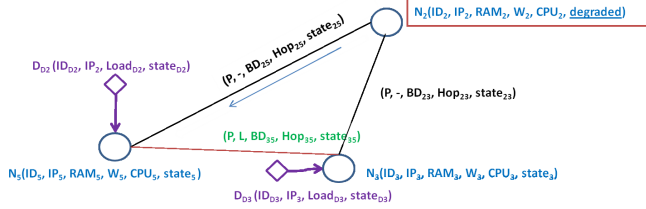


Figure 6. Network configuration after migration

## V. CONCLUSION

In this paper, we focus on QoS guarantees for crisis management applications composed by a pub/sub system deployed on MANET. Our proposed solution monitor the system and predict failure using the ARIMA model.

In the second part of this paper, new reconfiguration actions are proposed to repair the system. In order to evaluate such reconfiguration actions, we propose a dynamic protocol at design time describing different transformation rules for reconfiguration actions on formal technics of graph grammar. GMTE is used to transform these rules towards enumerating all possible adaptation solutions. Experimentation results will evaluate the proposed adaptation rules.

## REFERENCES

- [1] K. Minkyong and K. Kyriakos and Y. Fan and R. Johnathan and L. Hui and S. Konstantin, *Efficacy of techniques for responsiveness in a wide-area publish/subscribe system*, Proceedings of the 11th International Middleware Conference Industrial track, 2010.
- [2] C. Yuan and S. Karsten, *Opportunistic overlays: efficient content delivery in mobile ad hoc networks*, Proceedings of the ACM/IFIP/USENIX 2005 International Conference on Middleware, 2005.
- [3] A. Carzaniga, and D. Rosenblum and S. David and A. L. Wolf, *Design and evaluation of a wide-area event notification service*, ACM Trans. Comput. Syst, 2001.
- [4] M. Lucas and S. Saccucci and Jr. Baxley and V. Robert and H. Woodall and D. Maragh and W. Faltin and W. Fedrick and J. Hahn and T. Tucker and William T. and Hunter and J. Stuart and F. MacGregor and J. Thomas J., *Exponentially weighted moving average control schemes: properties and enhancements*, Technometrics, 1990.
- [5] G. Cugola, and G. .Picco, *REDS: a reconfigurable dispatching system*, Proceedings of the 6th international workshop on Software engineering and middleware, 2006.
- [6] <http://www.nvcc.edu/home/elanthier/methods/correlation.htm>.
- [7] [homepages.laas.fr/khalil/GMTE](http://homepages.laas.fr/khalil/GMTE)
- [8] Mr. Ruey and S. Tsay, *ARMA Models*, 2008
- [9] I. Lahyani and S. Marzouk and M. Jmael, *QoS Aware Publish/Subscribe System Deployed on a Mobile Ad-hoc Network*, Communications of SIWN, 2009.
- [10] M. Avvenuti and A. Vecchio and G. Turi, *A Cross-Layer Approach for Publish/Subscribe in Mobile Ad Hoc Networks*, Mobile Agents for Telecommunication Applications, 2005.
- [11] M. K. Denko, and E. Shakshuki, and H. Malik, *A Mobility-Aware and Cross-layer Based Middleware for Mobile Ad Hoc Networks*, Advanced Information Networking and Applications, 2007.
- [12] S. Corson, and J. Macker, *Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations*, 1999.