# Fault Detection in Autonomic Networks using the Concept of Promised Cooperation

Remi Badonnel[1,2] and Mark Burgess[1]

[1] Faculty of Engineering, Oslo University College
Pb 4 St. Olavs Plass, 0130 Oslo, Norway
[2] LORIA - INRIA, Nancy University
BP 239, 54506 Vandœuvre, France

**Abstract.** Fault detection is a crucial issue in autonomic networks for identifying unreliable nodes and reducing their impact on the network availability and performance. We propose in this paper to improve this situation based on the concept of promised cooperation. We exploit the promise theory framework to model voluntary cooperation among network nodes and make them capable of expressing the trust in their measurements during the detection process. We integrate this scheme into several distributed detection methods in the context of ad-hoc networks implementing the OLSR routing protocol. We quantify how the fault detection performances can be increased using this approach based on an extensive set of experimentations performed under the ns-2 network simulator.

**Key words:** Fault Detection, Autonomic Networks, Promise Theory

## 1 Introduction

Autonomic networks are self-organized networks capable of managing themselves and adapting to changes in accordance with high-level policies and objectives [17]. These networks are typically formed from a distributed set of nodes that are themselves autonomic, just as an organism is a collection of autonomous cells. They can also interact with other autonomic networks in order to provide extended services in a cooperative manner. Fault detection is a crucial issue in autonomic networks for detecting and isolating unreliable nodes that may impair the network performance.

A typical example of autonomic networks can be given with ad-hoc networking. Ad-hoc networks are spontaneously deployed from a set of mobile devices (laptops, personal digital assistants, mobile phones, sensors) without requiring any preexisting infrastructure. The deployment of an ad-hoc network is performed dynamically by the mobile devices themselves: each device is both a terminal that communicate with the others and a router that can forward packets on behalf of the other devices through multi-hop communications. The network topology can therefore be highly dynamic, since nodes might come and go based on user mobility, out-of-reach conditions and energy exhaustion. The network resources are scarce because the network is self-maintained by devices that usually operate under bandwidth and energy constraints.

Detection mechanisms are required in such autonomic networks for identifying a large range of faults including faults at the physical layer (due to physical errors), at the routing layer (due to misconfiguration) and from the energy viewpoint (due to battery failures). We designed and evaluated in [3, 4] a distributed fault detection scheme for ad-hoc networks, where we infer network faults by analysing the intermittence of nodes. The nodes observe themselves in a cooperative manner: they monitor the behaviour of the neighbour nodes based on routing layer information, then share and synthesize the measurements among them. A local node implementing the detection service can identify faulty nodes in the direct neighbourhood, but the service may generate biased local views in this configuration. We addressed this issue by defining several distributed detection methods in order to improve the detection performances. These methods correlate the measurements performed by different observer nodes so that such biased local views are discarded. However, we considered in this distributed scheme that the measurements are of equal importance during the detection process, whatever the observer node generating the measurements is, and whatever the trust the observer node has in its own measurements.

We propose in this paper to extend this fault detection scheme using the concept of promised cooperation. Promise theory [10] defines a graph theoretical description of autonomous agents that are independent and have private knowledge. These agents can learn about each other only if each agent promises to make information about itself (and its local view) available, and they promise to use such information. Each interaction by this voluntary cooperation therefore requires a set of promises to give and receive information [6]. Our objective is to model and instantiate the detection service as a set of promises among the network nodes, so that each node can express the trust it has in its local detection service. In that manner, a node is capable of voluntarily reducing the relative importance of its detection measurements in the distributed scheme when it considers the measurements may deteriorate the detection performances. The main issue that we address can be stated in three simple questions: how can the network nodes define the detection service using promise theory and express the trust in their local measurements? How can we integrate these parameters into distributed detection methods? How does this promise-based scheme impact on the whole performances of the detection?

The paper is structured as follows: we overview in Section 2 a distributed fault detection scheme for autonomic networks based on the analysis of routing plane information to detect faulty nodes. Section 3 describes how this fault detection scheme can be extended using the promise theory so that the network nodes can voluntarily tune their importance during the detection process in a beneficial manner. We detail how these promise properties can be expressed by the nodes and can be coupled with trust and reputation mechanisms. We evaluate the performance of our extended detection scheme through an extensive set of experimentations in Section 4. A survey of related work is given in Section 5. Finally, Section 6 concludes the paper and presents future research efforts.

## 2 Fault Detection in Autonomic Networks

Detection of faults in autonomic networks is performed by the nodes themselves based on a self-management scheme. Fig. 1 illustrates an example scenario with a network composed of 4 nodes $V = \{v_1, v_2, v_3, v_4\}$. Each node $v_i \in V$ locally monitors the behaviour of the direct neighbor nodes in order to detect misbehaving nodes. The monitoring data can be shared among the network nodes using a distributed detection scheme. In that case, the detection process takes into account and synthesizes a set of measurements performed by different network nodes to detect faulty elements.

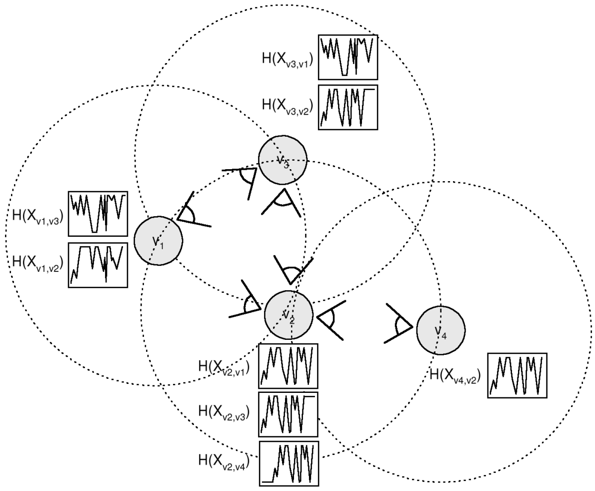


**Fig. 1.** Fault detection in an autonomic network where nodes observe themselves

### 2.1 Local detection

The first step for designing a fault detection scheme is to define what a faulty behavior corresponds to and how the nodes can observe it locally. The example scenario illustrated in Fig. 1 corresponds to the fault detection scheme that we designed in [3] based on the analysis of node intermittence in an ad-hoc network. Intermittence is a relatively normal condition due to causes that are inherent to such an autonomic network, e.g. node mobility. However, intermittence might have also different causes related to faulty behavior such battery failures, errors at the physical layer, routing misconfiguration. In our context, the faulty behavior is defined in terms of node intermittence and is detected by analysing the information already available at the routing plane. Each network node $v_i$ observes the intermittence of the direct neighbor nodes by monitoring the number $X_{v_i,v_j}$ of periodic hello messages received from another node $v_j$ during a time interval. An entropy-based metric $H(X_{v_i,v_j})$ is then applied to the random variable $X_{v_i,v_j}$ (see Eq. 1).

$$H(X_{v_i,v_j}) = \sum_{k=0}^{b_{max}} P(X_{v_i,v_j} = k).log(\frac{1}{P(X_{v_i,v_j} = k)}) \tag{1}$$

Based on a Markov chain-based analytical model, we approximated this entropy-based metric asymptotically via analytic depoissonization (see Eq. 2 where $b_{max}$ is the maximal number of packets that can be received during the measurement interval and $p_{up}$ is the state probability that the given node is up) and showed that this metric can be used to locally characterize a faulty intermittent node.

$$H(X_{v_i,v_j}) \asymp \frac{1}{2}ln(b_{max}) + ln\sqrt{2\pi p_{up}(1-p_{up})} + \sum_{k\geq 1} a_k b_{max}^{-k} \tag{2}$$

### 2.2 Distributed detection

The second step is typically extending the detection scheme in a distributed manner, so that the measurements performed by different nodes are synthesized at the network scale, in order to drop out any local biased views. In particular, we have defined three distributed detection methods based on thresholding in [3].

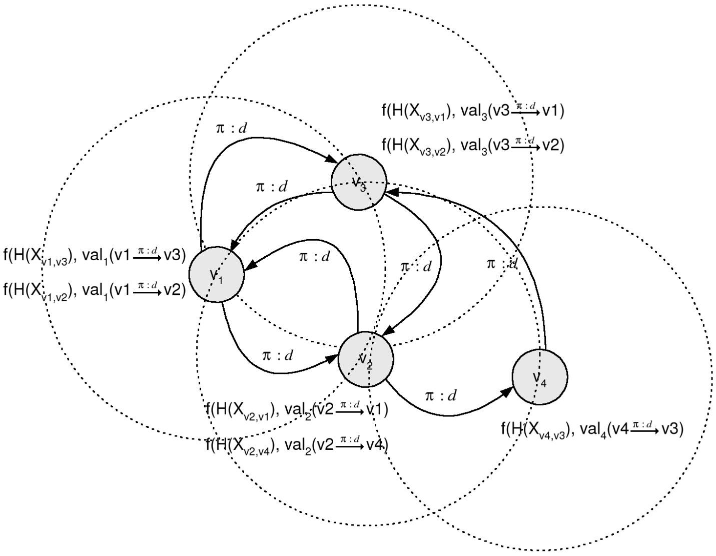
A threshold-based detection consists in (1) ranking the network nodes according to a criterion $c$ and then (2) identifying faulty nodes according to a threshold value $\lambda$ (faulty nodes are those presenting a criterion value $c(v_j) > \lambda$). We briefly overview below these three detection methods:

- The first detection method $m_1$ (also called majority voting) ranks the network nodes based on the number of observer nodes that locally observed and detected the node as a faulty one.
- The second method $m_2$ (also called sum of values) takes into account the number of observer nodes, but also the measurement values generated by these nodes by ranking the network nodes in function of the sum of measurement values at the network scale.
- The last method $m_3$ (also called average of values) consists in ranking the network nodes based on the average of measured values. $m_3$ does not focus on the number of observing nodes, but favors the measurement values at the network scale.

These methods can increase the detection performances by correlating the measurements performed by different observer nodes.

## 3 Fault Detection with Promises

We propose in this paper an extended fault detection scheme based on the concept of promised cooperation. Biased measurements impact on the performances of fault detection schemes, it would be valuable if network nodes could voluntarily decrease the relative importance of their measurements when they consider these measurements have a high probability to be biased. Of course, a simple solution would be that network nodes only report measurements when they consider them to be trusted. However, this solution may lose substantial information.



f(H(X_{v3,v1}), val_3(v3 $\xrightarrow{\pi:d}$ v1)
f(H(X_{v3,v2}), val_3(v3 $\xrightarrow{\pi:d}$ v2)
π : d
π : d
v_3
f(H(X_{v1,v3}), val_1(v1 $\xrightarrow{\pi:d}$ v3)
f(H(X_{v1,v2}), val_1(v1 $\xrightarrow{\pi:d}$ v2)
v_1
π : d
π : d
π : d
π : d
v_2
π : d
v_4
f(H(X_{v2,v1}), val_2(v2 $\xrightarrow{\pi:d}$ v1)
f(H(X_{v2,v4}), val_2(v2 $\xrightarrow{\pi:d}$ v4)
f(H(X_{v4,v3}), val_4(v4 $\xrightarrow{\pi:d}$ v3)

**Fig. 2.** Fault detection in an autonomic network with promises

The objective of our approach is to report all measurements, but to enhance them with a trust factor: the importance of a measurement given by a detection service should be proportional to the trust the observer node has in this measurement. We consider the logical framework of promise theory which provides support for specifying the interactions and constraints among autonomous agents (see Fig. 2). The concept of voluntary cooperation is central to this modeling and is clearly appropriate in the context of autonomic networks. The framework defines a service view of interactions. We consider the local detection service as the service promised by an autonomic agent that is not externally controllable, to another agent. In the promise logical graph, a promise (notation $\pi$) corresponds to a labeled edge of the form $v_i \xrightarrow{\pi:s} v_j$ where $v_i$ and $v_j$ are agents and $s$ is the type of promise. We introduce the type of promise $d$ corresponding to the local detection service. Each promise of giving the service (plotted on Fig. 2) is implicitly completed by a promise of receiving it.

Trust (notation $\tau$) is closely coupled to the concept of promises and can be defined as a judgment, or more specifically as a valuation of a service promise [6]. The general notation to define that agent $S$ trusts agent $R$ to ensure that agent $T$ keeps a promise of type s to agent $U$ is given by Eq. 3.

$$S[T] \xrightarrow{\tau:s} R[U] \tag{3}$$

In our context, we propose that a network node $v_i$ can express its own trust in the detection service it provides to another node $v_j$ i.e. the trust in keeping the promise $v_i \xrightarrow{\pi:d} v_j$ true. This trust (see Eq. 4) can be quantified by a valuation $\mathrm{val}_i$ made by node $v_i$ to determine the trust in the measurements generated by its detection service.

$$v_i[v_i] \xrightarrow{\tau:d} v_i[v_j] \leftrightarrow \mathrm{val}_i(v_i \xrightarrow{\pi:d} v_j) \tag{4}$$

The valuation $\mathrm{val}_i(v_i \xrightarrow{\pi:d} v_j)$ corresponds to a normalized value and takes therefore values between 0 (trustless detection service) and 1 (trustful detection service). We consider that the default value for a promise of type $d$ is the maximum value i.e. 1, so that a network node can only reduce the relative importance of its measurements in the distributed detection process. In our scenario, the network nodes share the intermittence measurements $H(X_{v_i,v_j})$, but also specify the trust $\mathrm{val}_i(v_i \xrightarrow{\pi:d} v_j)$ valuation in the local detection service. The promise logical graph is exploited during the distributed fault detection in order to improve its performances.

### 3.1 Integration into distributed detection methods

These valuations are then integrated into the distributed detection methods. Let's consider the three threshold-based detection methods previously presented, the measurements $H(X_{v_i,v_j})$ can be weighted in function of the trust valuation provided by the promise graph: $f(H(X_{v_i,v_j}, \mathrm{val}_i(v_i \xrightarrow{\pi:d} v_j))$. In that manner, the threshold-based detection methods can be refined as follows:

- The first detection method $m_1$ (majority voting) is then defined by the sum of observer nodes that locally detected the node as a faulty node, weighted by the trust valuations $\mathrm{val}_i(v_i \xrightarrow{\pi:d} v_j)$. The importance of an observer node in this detection method is thus equal to the trust value. This means that the vote of an observer node is not taken into account when the trust is equal to 0.
- The second method $m_2$ (sum of values) is now defined by a new criterion $c_2(v_i)$ corresponding to the sum of $H(X_{v_i,v_j})$ weighted by the trust valuations $\mathrm{val}_i(v_i \xrightarrow{\pi:d} v_j)$ by varying the $j$ value.

- The last method $m_3$ (average of values) consists in ranking the network nodes based on the average of measured values. As previously done with the two first detection methods, the measure values are weighted by the trust factor as follow: $H(X_{v_i,v_j}) \times \mathrm{val}_i(v_i \xrightarrow{\pi:d} v_j)$.

The trust valuations are defined by the nodes themselves in a voluntary manner based on the promise graph. As a consequence, malicious or faulty nodes may parameterized false values. However, as previously mentioned, the trust normalized value is by default set to the maximum value, so that a network node can only decrease its participation in the fault detection process i.e. decrease the relative importance of its measurements.

### 3.2 Trust and reputation

In our application scenario, promises permit network nodes to express the trust in the measurements they have performed: $v_i[v_i] \xrightarrow{\tau:d} v_i[v_j]$. A local node can typically detect bias by analysing and comparing the set of local measurements. For instance, if a node detects all the other nodes as intermittent, there is a high probability that these measurements are biased and that the node itself is intermittent. This endogenous factor (how the node perceives itself) is complementary to the exogenous factor (how the node perceives the other nodes). The trust valuations of other nodes $v_i[v_j] \xrightarrow{\tau:d} v_i[v_i]$ can be performed in an indirect manner by exploiting the detection results. In particular, the network nodes detected as faulty nodes by the distributed detection methods should be dynamically discarded from the detection process, as they may introduce biased information data and may significantly deteriorate the detection performances. A two-phase scheme can typically be instantiated in order to drop out the measurements performed by nodes that are suspected to be faulty and to determine if the detection results are unchanged when we discard these nodes from the detection scheme. During a first phase, a node can change from the normal state to the potential faulty state. In that case, the node is not considered as faulty, but the measurements generated by it are not taken into account during the following of the detection process. During a second phase, a node can change from the potential faulty state to the faulty state if the node is still selected by the detection method.

## 4 Experimental results

We evaluated the performance of our fault detection scheme through an extensive set of experimental results, and we determined to what extent promising nodes can improve the detection in an autonomic network. The simulations were performed with the discrete event network simulator ns-2 [1]. We simulated an ad-hoc network composed of a set of 50 nodes moving in a 1500 m x 300 m rectangular area during a time period of 900 simulated seconds. We considered the IEEE 802.11 MAC protocol at the data link layer. The routing protocol is the OLSR ad-hoc network protocol (implementation of the NRL [2]).

The node mobility corresponds to the wide-spread (RWP) random waypoint model (individual mobility of nodes) [7]. Each network node moves at a constant speed less than *speed* m/s to a destination point selected uniformly in the rectangle area and then waits during a pause time *pause* before moving to a new destination. The RWP model may generate initialization discrepancy issues [18] , we therefore used the steady-state mobility model generator *mobgen-ss* to guarantee an immediate convergence and obtain reliable simulation results.

| Parameter | Value |
|---|---|
| Simulator | ns-2 |
| Simulation time | 900 s |
| Simulation area | 1500 m x 300 m |
| Network nodes | 50 nodes |
| Faulty nodes | 0 - 5 node(s) |
| Promising nodes | 0 - 50 node(s) |
| Cooperative method | average of values |
| Mobility model | random waypoint<br>*mobgen-ss* |
| Speed | 0 - 10 m/s |
| Pause time | 0 - 120 s |
| Physical Layer | FSP / 2-RGR |
| MAC layer | IEEE 802.11 |
| Routing layer | NRL OLSR |

**Table 1.** Simulation parameters

For each experiment, we have randomly chosen a set $F$ of faulty nodes (from 0 to 5 nodes) and a set $P$ of promising nodes (from 0 to 50 nodes). The faulty nodes follow a two-state Markov chain faulty behavior model that we proposed in [3]. The promising nodes express the trust they have in the measurements they provide to the other nodes. We arbitrarily considered that the trust valuations are set to 50% for the promising nodes that are faulty, and are set to 100% for the promising nodes that are regular, as defined by Eq. 5 and 6.

$$\forall v_i \in P \cap F, \forall v_j \in V, \mathrm{val}_i(v_i \xrightarrow{\pi:d} v_j) = 0.5 \tag{5}$$

$$\forall v_i \in P \cap \bar{F}, \forall v_j \in V, \mathrm{val}_i(v_i \xrightarrow{\pi:d} v_j) = 1 \tag{6}$$

We quantified the performances (sensitivity and specificity) of the detection scheme with promises by comparing the initialized set of faulty nodes to the the set of nodes identified by the detection scheme. A detection scheme can be seen as a diagnostic test, where we test if a network node is a faulty node (positive test) or a regular node (negative test). By comparing the initialized set of faulty nodes to the set of positive-tested nodes, it is possible to determine if the test provides true or false results.

We compared the performances of our detection (method $m_3$) with different percentages of promising nodes in the network (0%, 25%, 50%, 75% and 100%). We plotted the Receiver Operating Characteristic (ROC) [21], a graphical plot of sensitivity (Sn) versus 1-specificity (1 - Sp) for each scenario. The sensitivity

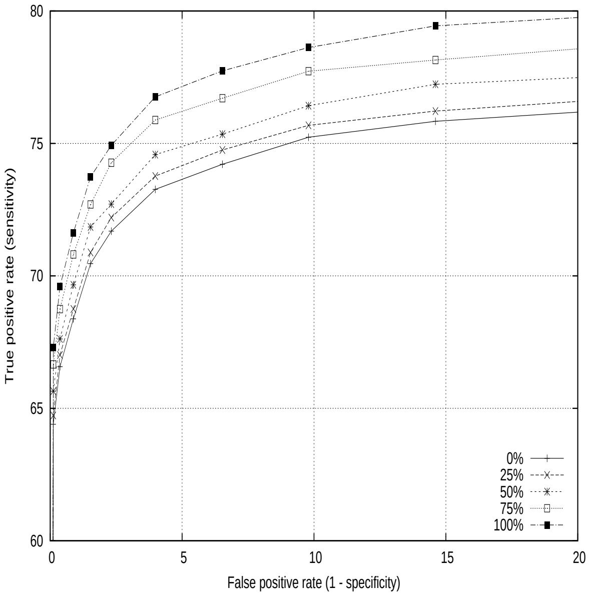


**Fig. 3.** Detection performances based on the percentage of promising nodes

quantifies how well the method picks up true cases, by defining the proportion of cases having a positive test result of all positive samples tested, while the specificity quantifies how well it detects false cases, by comparing the proportion of true negatives of all the negative cases tested. A diagnostic test is always a tradeoff between specificity and sensibility. The ideal diagnostic test shows a plot that is a point in the upper left corner of the ROC space.

We were interested in analysing the performances of method $m_3$ (that showed the best performances in [3]) but in that case by considering the concept of promises. As we are looking for a low positive false positive rate, we limited the plotting of ROC curves to a false positive rate no more than 20%. The comparison of ROC curves clearly depicts that the percentage of promising nodes impact on the detection performances. We did not know a priori to what extent the promises will impact on the cooperative methods. The experimental results shows that the detection performances can be improved of up to 4% in our configuration. The performances could be further improved by considering lower trust valuations: the lowest trust valuations (nodes part of $P \cap F$) were of 50% in our simulation scenarios. Moreover, reputation mechanisms could be integrated in the detection process in order to evaluate the ability of promising nodes to keep their promises. The detection is done in a lightweight manner based on information available at the routing plane, and management information are propagated using the extensible OLSR messages. Scalability and overhead are therefore directly dependent on the OLSR routing protocol performances.

## 5 Related work

Among the pioneering approaches of fault management, Jakobson introduced in [12] an approach for correlating events and faults with temporal constraints. Conceptual anomaly detection in distributed environments is proposed in [5] using an analytical method based on principal component analysis. The method does not address a specific deployment, but relates to the distributed detection methods that we defined. In both cases, distributed data from different hosts are correlated to provide an importance ranking of anomalies.

Detection schemes permit to identify misbehaving nodes based on different criteria in the context of network management. In particular, such a management approach was proposed in [15] by analysing the packet forwarding behavior of nodes in an ad-hoc network. We have proposed in [3] a detection scheme based on the analysis of the distribution of routing packets and complete it in this paper using the concept of promises. Failures detection algorithms based on keep-alive messages were also experimented in [20] and their performance are evaluated in overlay networks. Moreover, an active mechanism is detailed in [8] to characterize dynamic dependencies and to determine the root-causes of an anomaly at the application layer. Research efforts in ontology-based knowledge representation [11] contribute also to improve automated reasoning in this context.

The problem of distributed network monitoring and of its optimization has been addressed by [14] where heuristics are defined in order to optimize instrumentation location in bandwidth constrained environments. To a certain extent, these heuristics can be adapted to autonomic networks and systems. The DAMON architecture [16] defines a distributed monitoring system for ad-hoc networks, supporting multiple data repositories and including an auto-discovery mechanism of data repositories by the agents. This generic architecture is not dedicated to specific network parameters and could therefore be appropriate for the storage of fault detection data. A multi-tier architecture for efficient monitoring and management is proposed in [19] where nodes are grouped to ensure that they can be reached via a member of its group and is applicable to autonomic networks where network ressources are also constrainted. Finally, an excellent overview of trust and reputation issues in autonomic networks is given in [13].

## 6 Conclusions and Future Work

Fault detection is a management activity of crucial importance in autonomic networks for increasing network availability and reliability. This activity is performed by the nodes themselves in a self-organized manner in order to identify and isolate unreliable nodes. We propose in this paper to apply the concept of promised cooperation to fault detection methods in autonomic networks. The objective is to improve the detection performances by modeling voluntary cooperation among nodes and making them capable of expressing the trust they have in the measurements they provide to the other nodes.

We have shown in this paper how the network nodes can express this trust based on the framework of promise theory. We have specified a promise $d$ for the local detection service: $v_i \xrightarrow{\pi:d} v_j$ and defined the trust in the measurements given by a node as a normalized valuation $\text{val}_i(v_i \xrightarrow{\pi:d} v_j)$. We then have integrated these parameters into several threshold-based cooperative detection methods, so that a network node can voluntarily reduce the relative importance of its measurements in the distributed scheme. Promise theory therefore served as a support to bring an endogenous factor (how the node perceives itself) complementary to an exogenous factor (how the node perceives the others). We have evaluated these detection methods with promises in the context of ad-hoc networks by using our entropy-based intermittence measure. We have quantified the performances in terms of sensitivity and specificity by plotting the corresponding ROC curves. The intermittence measure also provides a practical and inexpensive approach to classifying the observed behaviour of nodes in keeping their promises, and does not require any kind of centralized management.

Our future work will consist in experimenting our extended fault detection methods with more complex fault models with higher order Markov chains and more developed internal chain architectures. We will also use the promise logical graph to evaluate the performances of the detection methods coupled with reputation mechanisms in an analytical manner. While our scheme mainly focuses on fault management, it would be interesting to extend it to security management and, in particular, to determine how collaborative attacks can affect the solution. Finally, the entropy-based intermittence measure is a complementary metric to that already used in cfengine [9] (*last seen time*). We have now implemented the present method into cfengine and are testing it in a variety of networks.

## Acknowledgments

We would like to gratefully acknowledge Radu State and Olivier Festor for their helpful discussions and comments. This work is supported by IST-EMANICS Network of Excellence #26854.

## References

1. Ns-2 network simulator. http://www.isi.edu/nsnam/ns/.
2. OLSR Extension for Ns-2. Navy Research Laboratory OLSR Project, http://pf.itd.nrl.navy.mil/projects/olsr/.
3. R. Badonnel, R. State, and O. Festor. Fault Monitoring in Ad-Hoc Networks Based on Information Theory. In *Proc. of the 5th International IFIP-TC6 Networking Conference (NETWORKING'06)*, Coimbra, Portugal, May 2006. Lecture Notes in Computer Science 3976, Springer Verlag.

4. R. Badonnel, R. State, and O. Festor. Self-configurable fault monitoring in mobile ad-hoc networks. *Elsevier Journal of Ad-Hoc Networks*, 2007. To be published.
5. K. Begnum and M. Burgess. Principle Components and Importance Ranking of Distributed Anomalies. *Machine Learning Journal*, 58(2):217–230, 2005.
6. J. Bergstra and M. Burgess. Local and Global Trust Based on the Concept of Promises. Technical report, Oslo University College, Norway, 2006.
7. J.-Y. Le Boudec and M. Vojnovic. Perfect Simulation and Stationarity of a Class of Mobility Models. In *Proc. of IEEE International Conference on Computer Communications (INFOCOM'05)*, Miami, FL, USA, March 2005.
8. A. Brown, G. Kar, and A. Keller. An Active Approach to Characterizing Dynamic Dependencies for Problem Determination in a Distributed Application Environment. In *Proc. of the 7th IFIP/IEEE International Symposium on Integrated Network Management (IM'01)*, Seattle, WA, USA, May 2001.
9. M. Burgess. A site configuration engine. *USENIX Computing systems*, 8(3), 1995.
10. M. Burgess and S. Fagernes. Promise theory - A Model of Autonomous Objects for Pervasive Computing and Swarms. In *Proc. of the International Conference on Networking and Services (ICNS'06)*, Silicon Valley, USA, June 2006.
11. E. Lehtihet and J. Strassner and N. Agoulmine and M. O. Foghlu. Ontology-Based Knowledge Representation for Self-governing Systems. In *Proc. of 7th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management (DSOM'06)*, Dublin, Ireland, October 2006.
12. G. Jakobson and M. D. Weissman. Real-time Network Management: Extending Event Correlation with Temporal Constraints. In *Proc. of the 4th IFIP/IEEE International Symposium on Integrated Network Management (IM'95)*, Santa Barbara, CA, USA, 1995.
13. L. Buttyan and J.-P. Hubaux. *Security and Cooperation in Wireless Networks.* Cambridge Press, 2007.
14. L. Li, M. Thottan, B. Yao, and S. Paul. Distributed Network Monitoring with Bounded Link Utilization in IP Networks. In *Proc. of IEEE International Conference on Computer Communications (INFOCOM'03)*, San Francisco, USA, 2003.
15. O. Gonzalez Duque and G. Pavlou and M. Howarth. Detection of Packet Forwarding Misbehavior in Mobile Ad-Hoc Networks. In *Proc. of the 5th International Conference on Wired/Wireless Internet Communications (WWIC'07)*, Coimbra, Portugal, May 2007.
16. K. Ramachandran, E. Belding-Royer, and K. Almeroth. DAMON: A Distributed Architecture for Monitoring Multi-hop Mobile Networks. In *Proc. of IEEE International Conference on Sensor and Ad Hoc Communications and Networks (SECON'04)*, Santa Clara, CA, USA, October 2004.
17. Richard Murch. *Autonomic Computing.* IBM Press, 2004.
18. J. Yoon, M. Liu, and B. Noble. Random Waypoint Considered Harmful. In *Proc. of IEEE International Conference on Computer Communications (INFOCOM'03)*, pages 1312–1321, San Francisco, CA, USA, April 2003.
19. M. Younis, P. Munshi, and E. Al-Shaer. Architecture for Efficient Monitoring and Management of Sensor Networks, E2EMON Workshop. In *Proc. of the 6th IFIP/IEEE International Conference on Management of Multimedia Networks and Services (MMNS'03)*, Belfast, Northern Ireland, UK, September 2003.
20. S. Q. Zhuang, D. Geels, I. Stoica, and R. H. Katz. On Failure Detection Algorithms in Overlay Networks. In *Proc. of IEEE International Conference on Computer Communications (INFOCOM'05)*, Miami, FL, USA, March 2005.
21. M.H. Zweig and G. Campbell. Receiver-Operating Characteristic (ROC) Plots. *Clinical Chemistry*, 29(4):561–577, 1993.