Performance Tuning of Failure Detectors in Wireless Ad-hoc Networks: Modelling and Experiments

Corine MARCHAND†, Jean-Marc VINCENT† ⋆

†Laboratoire ID - IMAG, MESCAL project (CNRS - INRIA - INPG - UJF)
ZIRST 51, Avenue Jean Kuntzmann, 38330 Montbonnot Saint Martin, France
{Corine.Marchand, Jean-Marc.Vincent}@imag.fr

Abstract. We consider wireless ad-hoc networks and implement failure detections mechanisms. These failure detectors provide elementary information for high level distributed algorithms such as consensus, election or agreement. The aim is to guarantee a quality of service for these mechanisms. Stochastic models for tuning failure detectors are proposed based on frequency analysis and contention modelling. Tuning methods are suggested for setting time-out delays. The theoretical results were validated experimentally on a wireless platform, based on a statistical analysis of the measurements.

1 Introduction

Technological advances in wireless devices such as laptop computers, personal digital assistants (PDAs), or mobile phones, bring significance to new wireless technologies. Progress in wireless communication protocols, e.g. Bluetooth, WIFI, allow the use of new ad-hoc networking schemes. It follows that new challenges arise from the communication variability in wireless networks and the unpredictable disconnections of those heterogeneous devices, creating very dynamic topologies called ad-hoc wireless networks.

In this context, the distributed environment we consider is composed of heterogeneous devices which form a dynamic group. This environment is completely distributed (no predefined memory or stable server in our case). In addition, this environment is also unstable: due to unpredictable disconnections of devices and the variability of communication latencies, failures can occur.

In this unreliable environment, the main goal is that each device should offer its local resources and services to one another, and could benefit from services provided by other devices. So, to manage services and resource sharing and to maintain the consistency of the group regarding newcomers and devices that voluntarily disconnect themselves, we have developed middleware modules in order to be able to make some decisions. Accordingly, our previous works [**?**,**?**] focus on distributed agreement problems in unreliable environments, and more specifically on consensus protocols.

To solve the agreement problem, several algorithms have been proposed. In particular Chandra & Toueg [**?**] establish that the consensus problem could be solved in an asynchronous context with unreliable failure detectors. These detectors provide local

⋆ This work was partially supported by FT R&D CRE MIRRA and DECORE-IMAG project.

estimation of the state of entities on the network. Thus a detector either suspects an other site, or not. The estimation of the detector is clearly unreliable, but if the information is asymptotically correct, the agreement is eventually obtained.

From an implementation point of view, failure detectors on each site communicate with each others. The estimates of the failure detector about the status of all other devices are delivered to an upper layer in form of a list containing the suspected devices. These failure detectors implement a function that, according to some information, make the decision to suspect or not. A typical function is a time-out delay : if the failure detector has not heard from a site since some time-out period, then it suspects the remote site.

The objective of this paper is to analyze the quality of service of such failure detector and apply the modelling approach to a wireless ad-hoc architecture. The infrastructure have been implemented and tested in an industrial context (CRE MIRRA with France-Télécom R&D) and in the RNRT SIDRAH project. Configurations with heterogeneous devices (PC, Laptop, PDA) have been used. Experiments shows that parameter tuning should be set according to the type of device and the global load of the network. Stochastic analysis of the system is then confirmed.

The paper is organized as follows. Section 2 introduces the failure detectors. Then, Section 3, stochastic models are derived and quality of service factors computed. The last part is devoted to experimental results and analysis.

2 Failure Detectors

2.1 Theoretical Concept

The working principle of failure detectors is to provide, at a given time and for a given process, a list of suspected devices. As failure detectors are considered unreliable, this list can contain wrong information about remote devices (suspicion of a device correctly present or no suspicion of a failed device).

Each device e_i included in the system has its own failure detector module. So, with this module, each device can obtain information, periodically or on demand, concerning the global state of the system.

However, information provided by a local failure detector does not necessarily indicate the real state of the system. The failure detector only suspects that some devices have crashed or are disconnected. Note that failure detectors are inherently unreliable because the information they provide may be incorrect.

Chandra & Toueg [**?**] characterize failure detectors with two properties: the *accuracy* property, which restricts false suspicions that failure detectors can make, and the *completeness* property, which requires that failure detectors eventually suspect every failed devices. In this paper, we focus on the $\Diamond\mathcal{S}$ class of failure detectors, called *Eventually Strong* [**?**].

2.2 Failure Detector Implementation

From among the several strategies that have been proposed to implement failure detectors, e.g., heartbeat or query (pinging), we choose to use the classical heartbeat detection model.

The heartbeat technique is based on the periodic emission of messages from each failure detector to everyone. In our implementation we divide the failure detector into two modules. We distinguish between the spreading information, which is included in the failure detector export module, and the gathering information, which is treated by the failure detector import module.

As a consequence, every export module periodically broadcasts a message (see figure **??**) to inform other devices of its reachable state.

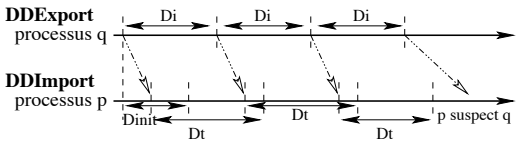

Fig. 1. Heartbeat principle

When an failure detector import module of an entity $e1$ receives a message from another device $e2$, it invokes its suspicion estimation function. This function in the simplest case works by arming a timeout. This mechanism is repeated until every one received a message from $e2$. Otherwise, if the import module of $e1$ does not receive a new message from $e2$ after the expiration of the timeout, it adds $e2$ to its list of suspected devices. The device $e2$ will be remove from this list when $e1$ receives a new message from $e2$.

This implementation technique introduces two parameters: the heartbeat period and the timeout delay. The heartbeat period is the time between two successive emissions in the failure detector export module of each device. The timeout delay is used in the import module. This parameter is the waiting period after which the failure detector of a device $e1$ starts to suspect a device $e2$ of having failed.

2.3 Quality of Service of Failure Detectors

Intuitively, the failure detectors' quality of service can be defined by: (1) the failure detector reactivity, which should be the fastest possible and (2) the failure detector should avoid false suspicions. Thus, the quality of a failure detector depends on its reactivity against external events and on its capacity to provide correct information. This quality of service notion was introduced and developed in [**?**] [**?**].

At run-time the failure detector is influenced by the two parameters [**?**] : D_i, the time period between two emissions of device i, and $\theta_j(i)$, the timeout delay for device i in the failure detector import module of device j. Therefore, the failure detector quality of service closely depends on the tuning of these parameters.

To define the quality of service of failure detectors, we have to address several tradeoffs. First, there is a tradeoff between the failure detector's reactivity and the number of sent messages over the network. Indeed, a decrease in heartbeat emission time period D_i allows for a better reactivity, thus limiting the duration of time devices are under false suspicion. However, this is at the cost of increased network utilization, which in turn may degrade overall system performance.

The desired properties of a failure detector are to 1) avoid suspecting devices that are available and 2) suspect devices that are not available as fast as possible. As the reactivity is related to the value of the time period, the failure detector's reliability depends on the timeout tuning. Thus, one has to balance the existing tradeoff between failure detector reliability and reactivity.

3 Stochastic Models

In this section we present stochastic modelling of failure detector mechanisms based on heartbeat. The goal is to provide a model that allows for tuning of the failure suspicion function. In fact, according to a set of parameter values, the model establishes the quality of service offered by the failure detector. This quality of service can be tuned by the user to fit the needs of the application.

The two quality of service criteria studied in this section are the reactivity of the failure detector and the quality of information given by the detector. The difficulty is to establish the tradeoffs between these two properties. The reactivity is the delay needed by a failure detector to detect the crash of the process. It is directly related to time-out and heartbeat period. The quality of information given by the detector is estimated by a false suspicion rate and the probability that the failure detector is in a state of suspicion. The reactivity is a decreasing function of the time-out value, as is the suspicion rate, that is also decreasing.

3.1 False detection probability

The difficulty for modelling such systems is the complexity of latency estimations. Figure **??** shows that the reception delay between two heartbeats send by the same failure detector depends on (1) the time taken by the beat in the communication stack of the sender, (2) the latency on the network taking losses into account, (3) the time spent in the communication stack of the receiver (4) and finally the time needed by the receiver failure detector process to access the information.

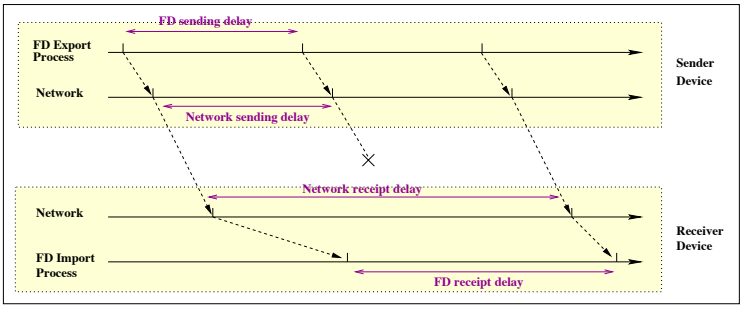

Fig. 2. Running principle between two failure detectors

In the failure detector , suspicion occurs when the reception module has not received a beat during some fixed time-out period. In this paper, the time-out θ is supposed to be constant in all experiments.

The false suspicion rate $\phi_I(\theta)$ is defined by the asymptotic ratio of the suspecting period (grey blocks on figure **??**) to the observation period.

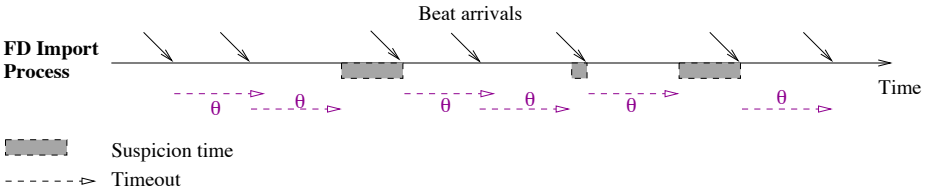

Fig. 3. Receptions and Suspicions

Denote by λ_0 the emission beat rate and λ the reception rate. The mean inter-arrival time of beats is $\frac{1}{\lambda}$ and $\lambda = \lambda_0.(1-p)$ with p the loss rate of messages on the network. Let $\{X_n\}_{n\in\mathbb{N}}$ be the sequence of inter-arrivals of beats on the receiver. So

$$\lim_{n\to+\infty} \frac{1}{n}\sum_{i=1}^{n} X_i = \frac{1}{\lambda}.$$

With this notation, it is possible to give an asymptotic expression for $\phi_I(\theta)$.

$$\phi_I(\theta) = \lim_{n\to+\infty} \frac{\sum_{i=1}^{n}(X_i-\theta)^+}{\sum_{i=1}^{n} X_i} = \lambda \lim_{n\to+\infty} \frac{1}{n}\sum_{i=1}^{n}(X_i-\theta)^+, \tag{1}$$

with x^+ the positive part of x.

To analyze the behavior of the failure detector and estimate $\phi_I(\theta)$, the system will be considered to be time homogeneous. Then parameters are constant on a sufficiently large period to ensure stationarity of the random process.

In a first model, we suppose that the heartbeat receiving process may be considered as a renewal process and the impact of variability of the inter-arrivals of beats on suspicion rate is established. A second model focuses on the impact of the latency in the receiver stack on the suspicion rate.

3.2 Variable sending delay

An current implementation of the heartbeat sender is a simple loop of waiting periods.

```
loop forever
     wait(period)
     send(heartbeat)
end loop
```

When this algorithm is perturbated by the operating system or access to the network, variability occurs and heartbeats are not periodic.

In a first approximation, we consider the inter-arrival process as a renewal process. It corresponds to strategies when the receiver estimates the distribution of inter-arrivals and tries to fix the time-out according to some histogram.

Then, because the inter-arrivals of beats are independent with the same probability law, the failure suspicion rate is just

$$\phi_I(\theta) = \lambda \mathbb{E}\left[X - \theta\right]^+. \tag{2}$$

This kind of formula is of high interest because it rapidly gives the order of Φ_I when the shape of the distribution of inter-arrivals distribution of beats is known.

Exponential model In the case when the inter-arrivals are exponentially distributed with rate λ. The arrival process is a Poisson process and

$$\phi_I(\theta) = \lambda \int_0^{+\infty} (x-\theta)^+ \lambda e^{-\lambda x} dx = e^{-\lambda\theta} \tag{3}$$

In figure **??**, the first curve shows exponential decreasing of $\phi_I(\theta)$ depending on time-out. As an example, to achieve a false suspicion rate of 10^{-3} the adequate time-out should be seven times the mean inter-arrival period.

In fact, when the inter-arrival X exhibits an *new better than used in expectation property (NBUE)*, the quantity $\Phi_I(\theta)$ is bounded from above by the exponential model and so

$$\phi_I(\theta) \leq e^{-\lambda\theta}.$$

Moreover, if we need to decrease the false suspicion rate by an adaptative scheme, an additive increment strategy will be sufficient.

Low variance model In many cases the exponential model overestimates the false suspicion rate, typically when the variance of inter-arrivals is small. To obtain finer results, Erlang distributions with parameters $(k, k\lambda)$ and density

$$f_X(x) = \frac{(k\lambda)^k x^{k-1} e^{-k\lambda x}}{(k-1)!},$$

with mean $\frac{1}{\lambda}$ and variance $\frac{1}{k\lambda^2}$. Then

$$\phi_I(\theta) = \lambda \int_0^{+\infty} \frac{(k\lambda)^k x^{k-1} e^{-k\lambda x}}{(k-1)!} (x-\theta)^+ dx \tag{4}$$

It may be shown that

$$\Phi_I(\theta) = e^{-k\lambda\theta} P_k(\lambda\theta),$$

where P_k is a polynomial of degree $k-1$. For small values of k, figure **??** shows the suspicion probability for a mean inter-arrivals of beats equal to 1 and a variance of $\frac{1}{k}$.

For example, with a variance $\frac{1}{5\lambda^2}$, a time-out of three times the inter-arrivals of beats is sufficient to ensure a false suspicion rate of 10^{-3}.

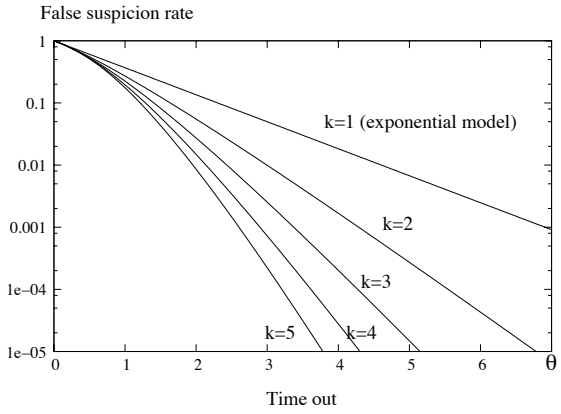

Fig. 4. Suspicion probability related to reactivity for the low variance model

In the case when the inter-arrival could be modelled by a normal distribution with mean $\frac{1}{\lambda}$ and standard deviation σ, we can bound the false suspicion rate by

$$\phi_I(\theta) \leq \frac{\sigma}{\sqrt{2\pi}} e^{-\frac{(\theta-\frac{1}{\lambda})^2}{2\sigma^2}} \tag{5}$$

In this case, figure **??** indicates the false suspicion rate. Naturally, these curves decrease more rapidly than the Erlang model. For a standard deviation of 0.5, taking a time-out of 3 times the period is sufficient to guarantee a false suspicion rate of 10^{-3}.

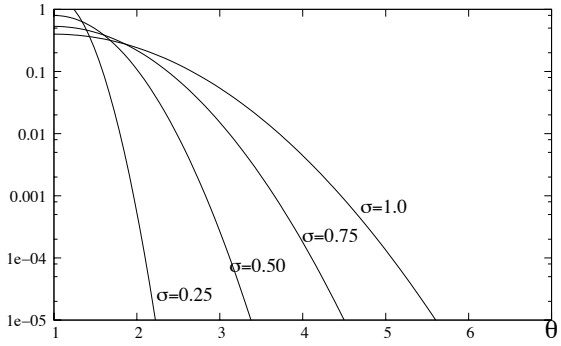

Fig. 5. Suspicion probability related to reactivity for the normal model

High variance model Unfortunately, the observed distribution could exhibit large values and when tail of the distribution is not of a negative exponential form. Then Pareto distribution functions ($\alpha > 2$) could be used

$$f_X(x) = \frac{\alpha - 1}{\alpha - 2} \frac{1}{(1 + \frac{x}{\alpha - 2})^{\alpha}}. \tag{6}$$

For these parameters, the mean has been fixed to 1 and the variance, for $\alpha > 3$, is $\frac{\alpha-1}{\alpha-3}$. The false suspicion rate could easily be computed by

$$\Phi_I(\theta) = \frac{1}{(1+\frac{\theta}{a-2})^{a-2}}.$$

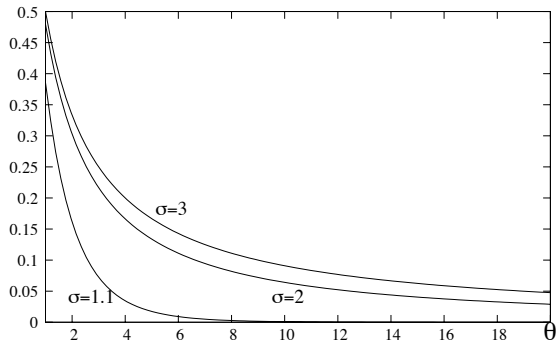

Fig. 6. Suspicion probability related to reactivity for the Pareto model

In this situation, it is clear that the high variability of inter-arrivals of beats produces a very poor quality of service of the failure detector . Even with a standard deviation of 1.1 the time out period should be more then ten times the heartbeat period to achieve a suspicion rate of 10^{-3}.

Synthesis In this table the inter-arrivals of beats is 1 and the time-out function gives the quality of service for false suspicions.

Distribution shape	**Properties**	**Time-out function**
Exponential	Most mixed distribution, bound for *New Better than Used in Expectation* distribution	$e^{-\theta}$
Erlang(k, k)	Exponential tail, low coefficient of variation	$P_k(\theta)e^{-k\theta}$
Gauss(1, σ^2)	White noise model around a deterministic value	$\leq \frac{\sigma}{\sqrt{2\pi}} e^{-\frac{(\theta-1)^2}{2\sigma^2}}$
Pareto(α)	Heavy tail distribution	$\frac{1}{(1+\frac{\theta}{a-2})^{a-2}}$ if $\alpha > 3$

3.3 Queueing of heartbeat messages

General model During experimentations, we observe that the delay between heartbeats mainly depends on the nature of the receiver: laptop or PDA. This suggests that the capability of the receiver introduces variability of inter-beats periods. Moreover, inter-arrivals appear to be correlated and the correlation could be important. Observing the phenomena at the network level by a non-intrusive "sniffer" we establish that heartbeats

are emitted as specified (e.g., periodically). The problem is due to the time spent by the receiver module to get the heartbeat from its own network buffer. Consequently, we have to take into account the contention of heartbeats on the receiver and variability appear when the heartbeat is delivered from the network layer to the heartbeat module at the middleware layer. A queueing model (figure **??**) is used to describe the system.

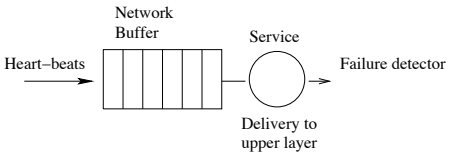

Fig. 7. Model for beats delivery

In such a queueing model denote the arrival process of beats by $\{A_n\}_{n\in\mathbb{N}}$ and the sequence of service delays for delivering the beats by $\{S_n\}_{n\in\mathbb{N}}$.

The interesting process for dimensioning is the inter-output process denoted by $\{Z_n\}_{n\in\mathbb{N}}$. The aim of this section is to compute the stationary distribution of this process. Following the evolution equation approach [**?**] the process $\{Z_n\}$ satisfies

$$Z_{n+1} = S_{n+1} + [A_{n+1} - R_n]^+ , \tag{7}$$

where R_n is the residual service time of clients in the queue just after the n^{th} arrival. This expression is obtained by the study of two cases :
- the server is busy at the arrival of client $n+1$, it begins its service at the end of the preceding client and the inter-output corresponds to the service time of client $n+1$;
- the queue is empty, $A_{n+1} - R_n$ is positive and represent the elapsed time between the last client output and the arrival of client $n+1$.

Provided that arrival and service processes are stationary ergodic, the queueing system is stable if $\mathbb{E}S < \mathbb{E}A$. Thus, the embedded process $\{R_n\}$ is also stationary and consequently, the process $\{Z_n\}$ converges to a stationary distribution denoted by Z.

The $GI/M/1$ case We suppose now that the inter-arrivals are independent with the distribution density $f_A(.)$. The services are considered exponentially distributed with rate μ and independent. The system is modelled by a $GI/M/1$ queue, this queue is stable iff $\frac{1}{\mu\mathbb{E}A} < 1$. The embedded process (number of clients in the queue) at arrival times is a homogeneous Markov chain and the stationary distribution is geometrically distributed with parameter β defined as the unique fixed point of the equation

$$\beta = \mathcal{L}_A(\mu(1-\beta)),$$

where $\mathcal{L}_A(.)$ is the Laplace transform of the inter-arrivals density f_A [**?**].

Moreover, because of the memoryless property of service time, the residual service time R is exponentially distributed with rate $\mu(1-\beta)$. The residual service time is a geometric sum of i.i.d. exponentially distributed random variables.

Given an inter-arrival distribution, it is possible to numerically compute the distribution of

$$Z = S + [A - R]^+ ;$$

and to deduce the false suspicion probability given a reactivity θ as

$$\mathbb{P}(Z > \theta) = \mathbb{P}\left(S + [A - R]^+ > \theta\right). \tag{8}$$

The $D/M/1$ case In the case when failure detectors have periodic heartbeats (period $A = \frac{1}{\lambda}$), the formulation above could be simplified. First, we compute the rate of the exponential distribution of R. Because $\mathcal{L}_A(t) = e^{-At}$, β is the unique solution of

$$\beta = e^{-A\mu(1-\beta)}.$$

Then we compute the distribution of $[A - R]^+$:

$$\mathbb{P}\{(A-R)^+ \leq x\} = \begin{cases} 0 & if\ x < 0; \\ e^{-\mu(1-\beta)A} & if\ x = 0; \\ e^{-\mu(1-\beta)(A-x)} & if\ 0 \leq x \leq A; \\ 1 & if\ x \geq A. \end{cases} \tag{9}$$

Then we form the convolution of the service time distribution and the distribution of $(A - R)^+$:

$$\begin{aligned} \mathbb{P}\{Z \leq x\} &= \mathbb{P}\{(A-T)^+ + S \leq x\} \\ &= \int_0^x \mathbb{P}\{(A-T)^+ \leq x - s\}\mu e^{-\mu s} ds \\ &= \int_0^x P\{(A-T)^+ \leq t\}\mu e^{-\mu(x-t)} dt \\ \mathbb{P}\{Z \leq x\} &= \begin{cases} \frac{1}{2-\beta}(e^{-\mu(1-\beta)(A-x)} - e^{-\mu((1-\beta)A+x)}) & if\ x \leq A; \\ 1 - \frac{e^{-\mu x}}{2-\beta}(e^{-\mu(1-\beta)A} + (1-\beta)e^{\mu A}) & if\ x \geq A. \end{cases} \end{aligned} \tag{10}$$

The density is obtained by differentiation

$$f_Z(x) = \begin{cases} \frac{\mu}{2-\beta} e^{-\mu(1-\beta)A}((1-\beta)e^{\mu(1-\beta)x} + e^{-\mu x}) & if\ x < A; \\ \frac{\mu}{2-\beta} e^{-\mu x}(e^{-\mu(1-\beta)A} + (1-\beta)e^{\mu A}) & if\ x \geq A. \end{cases} \tag{11}$$

For a given θ, the false suspicion rate is

$$\phi_I(\theta) = \frac{1}{A}\mathbb{E}\,[Z - \theta]^+ = \int_0^\infty (x-\theta)^+ f_Z(x)\, dx.$$

After some computation, for $\theta > A$, we obtain

$$\mathbb{E}\,[X - \theta]^+ = \frac{1}{(2-\beta)\mu} e^{-\mu\theta}(e^{-\mu(1-\beta)A} + (1-\beta)e^{\mu A}) \qquad \theta \geq A, \tag{12}$$

and we deduce

$$\phi_I(\theta) = \frac{1}{A(2-\beta)\mu} e^{-\mu\theta}(e^{-\mu(1-\beta)A} + (1-\beta)e^{\mu A}) \qquad \theta \geq A. \tag{13}$$

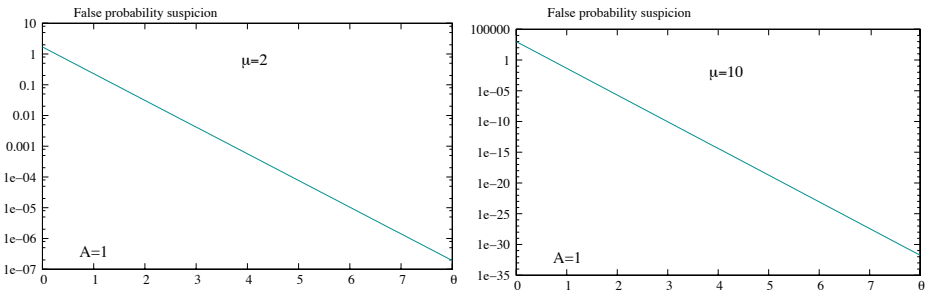

Fig. 8. False suspicion probability, $D/M/1$ model $A = 1$

When the system is loaded, the impact on false suspicion probability is important. For example, if the time to retrieve the heartbeat on the network is about half of the heartbeat period ($\mu = 2\lambda$), we should fix a time-out of four times the inter-arrival period to get a quality of service less than 10^{-3}.

From a practical point of view, this model permits us to adapt the suspicion policy of the failure detector to the architecture of the device. Moreover we may deduce the impact of the size of the network on the queueing system by considering μ as a function of the number of devices in the network.

However, these results are obtained for a deterministic arrival process on the receiver. This hypothesis could be weakened by modifying the arrival law. Using a Poisson process arrival, the queue is a $M/M/1$ and the output process is a Poisson process. In this case, we apply results from the previous section. Moreover, stochastic ordering techniques provides results on arrival processes that could compare with deterministic and Poisson process. This gives stochastic bounds for the dimensioning.

4 Experimentation

In this section we will use a real system to illustrate the relationship between the timeout value and the quality of information provided by the failure detectors.

4.1 First Approach

Experimental design: This first study utilized 2 laptop devices (Linux 800 Mhz) and 2 personal digital assistants (Linux 200 Mhz). The interconnections were based on a 802.11b wireless ad-hoc network. The failure detector modules developed were installed in each device (import module and export module). Thus, each device has an unreliable view of the global system based on the information in its own failure detector's import module. The parameter settings used for the import modules were 100 ms for the heartbeat time period, the timeout value was not fixed (infinite value). During the experiment which lasted approximately 15 minutes, about 10,000 measures were obtained. The system appeared to be stressed. In the experiment, the system's reactivity

was of the order of 1 second. Since the experiment was conducted in a dedicated environment, i.e., no other applications were running, it is to be expected that in a system under standard application load the delay will likely increase.

Results: These graphic representations illustrate the various behaviors existing between a PDA and a laptop. Note that, in all these experiments, the environment was "stressed", the heartbeat losses could be significant (loss rate was around 50% when the receiver was a PDA).

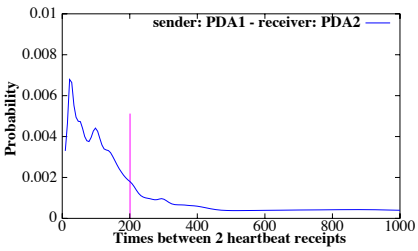

Fig. 9. Distribution of the update times on a PDA, of information concerning a remote PDA.

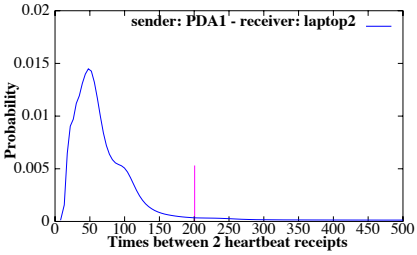

Fig. 10. Distribution of the update times on a laptop concerning information of a remote PDA.

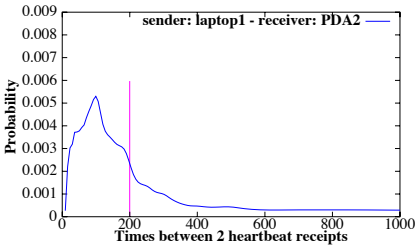

Fig. 11. Distribution of the update times, on a PDA of information concerning a remote laptop.

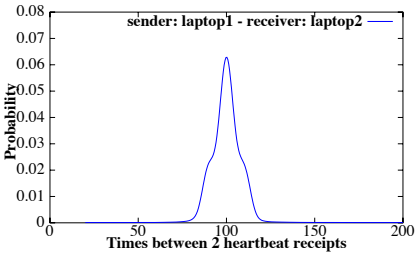

Fig. 12. Distribution of the update times, on a laptop concerning information of a remote laptop.

As we can see in the graphics, if the timeout value had been fixed at 200 ms ($\theta = 2*$ the heartbeat emission time period), it would have been a too small value for some of the devices and would have generated a lot of false suspicions. Indeed, when the sending and receiver device are laptops (see figure **??**), the distribution of heartbeat receipt delays is centered around the emission duration mean value (100 ms). Most of these durations are included between 50 ms and 150 ms. Therefore, a timeout value fixed at 200 ms seems to be appropriate in spite of false suspicions engendered, since the system reactivity is preserved.

On the other hand, when the receiver is a PDA (figures **??** and **??**), the distribution curves show that a timeout value equal to 200 ms is not adapted because it generates too many false suspicions.

Thus, this experiment points out the importance of good parameter setting. According to the kind of devices, a same parameter configuration does not imply a same quality of information:

– A laptop will not wrongly suspect another laptop (figure **??**)
– A PDA often stands a good chance of suspecting a laptop which is present in the system (figure **??**)

4.2 An "ideal setting" Experimentation

Experimental design: The high loss rate observed in the previous experiment denotes that the heartbeat emission frequency is not adequate and it disrupts the system network. Then, the goal of this next experimentation is to obtain a sample of measures which will be used as a reference for the models. The parameters may be adjusted so that the network works correctly, which means there is no voluntary stress or overload.

In this experiment, the system was composed with 6 devices: 3 PDAs (ipaq linux 200 Mhz), 2 laptops (linux 800 Mhz) and 1 laptop device which is used as a network sensor (linux 800 Mhz). The sensor role is to capture network traffic and record all heartbeat packets. This sensor will allow us to get an exterior view of the system behavior during the experimentation.
Parameters setting:

– Heartbeat emission period time: 500 ms
– Timeout : none
– Experimental duration: around 15 minutes

Losses: With this parameter setting, the heartbeat mechanism do not overload the system studied. Essentially due to external disturbances in wireless environment, heartbeat message losses are then limited (approximately 2 out of 1000 messages).

Heartbeat Reception Analysis: Figure **??** represents the distribution of elapsed time between two receipts of messages from the same device. Note that the distributions are slightly different according to the type of emitter/receiver devices.

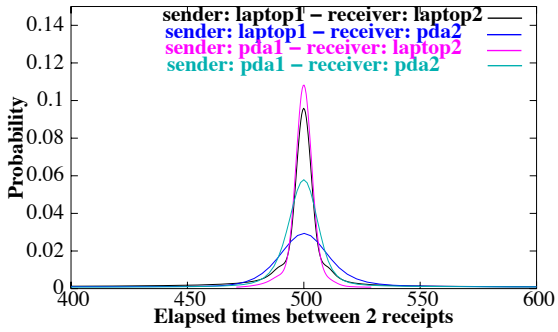

Fig. 13. Distributions of the update times

If the timeout value is fixed at $2 * (heartbeat\ period\ time)$, then the wrong suspicion rate is of the order of 10^{-3} when the receiver devices are laptops, and of 10^{-2} when receivers are PDAs.

4.3 Experimentation in a Disrupted Environment

Experimental Design: For this experiment, the platform configuration is like the previous one. However, contrary to the previous experiment, here we introduced a voluntary disruption. To do this, a laptop is used to generate a data transfer (ping with 4 KB/20ms packets) to an external device during the experiment.

Losses: To compare with the previous experiment, the loss rate is more important in this case (around 15% of the messages are lost). Whereas the loss rate is between approximatively 6% and 9% for all other devices, it is approximatively 44% for the device which generate the network overload.

Heartbeat Reception Analysis: As in the previous section, figure **??** illustrates the elapsed time between two heartbeat messages from a same remote device received by each device.

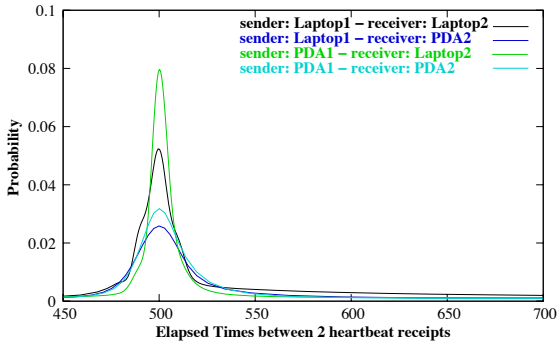

Fig. 14. Distributions of the update times

In this context, it is possible to get durations between two successive receipts which could be more than six times the average of heartbeat emission delays. As we can see in figure **??**, there are many long receipt durations when the receiver is a PDA. Moreover, some delays between heartbeat receipts are very small. This phenomenon may be explained by the fact that after a long waiting time (before receiving the next heartbeat message), because of the heartbeat messages are regularly sent, several messages could arrive closely together.

Thus, it seems that a correlation exists between successive waiting times of two heartbeat receipts and should be used in further modelling.

5 Conclusion

Tuning failure detectors is of great importance for the efficient control of distributed systems. However, the tradeoff between the quality of information and the reactivity of failure detectors should be established clearly. In this paper we demonstrated that stochastic models, taking into account the architecture, can be useful for setting specific time-out delays.

This study should now be extended to a finer analysis of correlation. Auto-regressive approaches could follow the evolution of the network, especially since network delays depend on the load on the network. Another approach could be a finer description of the spatial organization of the network. Distances between devices could affect the reliability of communications. In this case, stochastic geometry techniques could be efficiently used to model the knowledge an entity could built of the whole system.