FREIE UNIVERSITÄT BERLIN

Abort and Blocking Risks of Atomic Transactions

in Mobile Ad-Hoc Networks

Joos-Hendrik Böse

B-08-07
June 2008

FACHBEREICH MATHEMATIK UND INFORMATIK
SERIE B • INFORMATIK

Abort and Blocking Risks of Atomic Transactions in Mobile Ad-Hoc Networks

Joos-Hendrik Böse

Freie Universität Berlin, Department of Computer Science
Takustr. 9, 14195 Berlin, Germany
joos.boese@fu-berlin.de

Abstract

It is generally known that in the case of multiple node or communication failures atomic commit protocols cannot avoid blocking. While in wired networks such situations are rare because of low failure probabilities, mobile ad-hoc networks (MANETs) are considered to be a more challenging. In this technical report I present a probabilistic model to predict the abort and blocking risks of distributed atomic transactions for arbitrary MANET scenarios. The model presented is applied to a standard MANET scenario to demonstrate the blocking risks to be expected.

1 Introduction and Motivation

Thhis report examines the blocking problem of atomic transactions in MANETs. This is motivated by the observation that in practice, networks like the Internet that does not provide delivery guarantees is used to host critical applications, while the theoretical impossibility results on consensus [9, 5] and non-blocking atomic commit [10, 17] seem to have no practical impact. Commit protocols such as 2PC are commonly used in practice although they are susceptible to blocking situations and may hinder progress of applications. However, such situations are rare even at high transaction load and are negligible in practice. One question to be answered by this report is whether the situation is similar in MANETs. For example, I present the expected dimension of transaction abort and blocking probabilities in MANETs for different transaction models. Until now, most research concerned with coordination problems in MANETs simply stated that, due to node mobility and limited resources, "communication is highly unreliable" and thus blocking situations require special attention, e.g. by using more failure-resistant commit protocols. Quantitative statements about the expected number of failures and their impact on transaction abort and blocking have not been published so far.

However, general statements about abort and uncertainty rates in MANETs are hard to derive, as an infinitive number of transaction and MANET scenarios

exist and each combination shows individual failure characteristics. Therefore I am presenting a model to calculate the expected abort and blocking rates of transactions in a certain MANET scenario and show for an example scenario what abort and blocking probabilities have to be expected for different transaction models.

The purpose of the model developed here is twofold; on the one hand, it can be used to evaluate the applicability of a transactional application in a MANET setting a priori. Without such a model it requires vast simulation studies to find out if a transactional system is applicable in a MANET environment. For example, if in a scenario 50 % of all transactions must be expected to abort, then transaction processing is not feasible, and the transaction model must be enhanced to be more failure tolerant. On the other hand, the presented model can be used to optimize transaction processing at runtime, e.g. if the coordinator knows how many participants a transaction has and approximately how long the processing phase will take, it can calculate the probabilities of aborting and blocking. If these probabilities are unacceptable, the transaction can be rejected, or additional schemes like a backup coordinator can be embedded in commit processing to compensate for blocking. In short, the questions answered by the presented calculation model developed below are:

- Given a transaction and a MANET scenario, what is the probability that this transaction will abort.
- What is the probability that a participant of a strict or semantic transaction will encounter a blocking situation caused by a participant failure that cannot be compensated by standard cooperative recovery.
- What is the probability that a participant of a strict or semantic transaction will encounter a blocking situation caused by a node failure of the coordinator.

Part I of this report presents the system-, failure- and transaction models underlying the calculation model presented in Part II. Part I also describes how failure distributions of the system model, such as the reliability of communication paths, are derived for a given MANET scenario. Part II then presents an in-depth investigation of the blocking problem. Note that the calculation models developed are not only applicable to MANETs but to any environment that is modeled by the partially synchronous system model.

Part I
System and Transaction Models

In this part I present the system, failure, and transaction models used throughout this work.

2 System and Failure Models

In this section I will define the system and failure models of this work. The system model is based on the standard partially synchronous system model [8], assuming communication and site failures of nodes and is enhanced with certain assumptions on reachability and availability of mobile nodes in a MANET.

2.1 System Model

The system model considers a MANET $\mathcal{A}$ formed in a single area of a larger network described by the AGB mobility model [4]. The macro view of the AGB model is used here to model the fact that a MANET is not a closed system, but new nodes can join as well as leave $\mathcal{A}$. I assume that the total number of nodes in $\mathcal{A}$, denoted by $n_{\mathcal{A}}$, shows a negligible variation and is assumed to be constant over time, which is feasible if nodes enter and leave $\mathcal{A}$ at equal rates.

The probability that a node disconnects from $\mathcal{A}$ because it moves into another area at time t is described by the probability density function (pdf) $f_L(t)$. Analogously, the probability that a node joins $\mathcal{A}$ after being disconnected at time t is described by the pdf $f_J(t)$.

Nodes in $\mathcal{A}$ are assumed to have the same radio range and to relay messages for each other to provide for multi-hop routing, e.g. using AODV. Although message delays in $\mathcal{A}$ depend on the hop count of communication paths, for the sake of simplicity I assume an average message delay δ_m for $\mathcal{A}$. Note that δ_m is not an upper bound for message delay, as this would contradict the asynchronous system model, but rather a value describing the average delay of messages in case communication path is available.

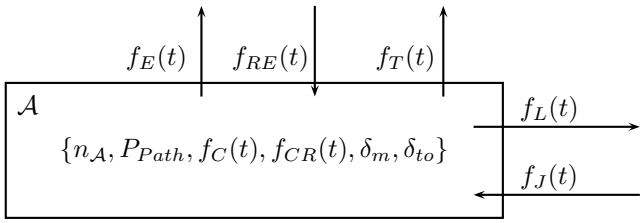


Figure 1: System Model.

Topology dynamics, mainly influenced by the mobility and link models, are captured in my system model by (i) the constant path probability P_{Path} of $\mathcal{A}$, (ii) by the pdf $f_C(t)$, describing the path duration for $\mathcal{A}$, and (iii) the pdf $f_{CR}(t)$ describing the probability that a broken path recovers after t_r. The probability that a broken communication path recovers is a conditional probability presuming that both communication partners remain in $\mathcal{A}$.

In addition to moving to other areas, nodes may disconnect from $\mathcal{A}$ forever if they experience a non-recoverable technical failure. The probability of this

happening is described by the pdf $f_T(t)$. Recoverable failures causing a disconnect from $\mathcal{A}$ represent for example energy-related outages. The probability that a node disconnects due to a recoverable technical failure is described by the pdf $f_E(t)$, and the density of energy-related outage times by the pdf $f_{RE}(t)$. I use the subscripts E and RE here because I assume recoverable technical failures to be energy-related.

Figure 1 depicts the general idea of the system model. While the communication characteristics within $\mathcal{A}$ are described by the parameters $n_{\mathcal{A}}$, P_{Path}, $f_C(t)$, $f_{CR}(t)$, δ_m, and δ_{to}, the leave and rejoin probability of nodes is described by the pdfs $f_{RE}(t)$, $f_E(t)$, $f_T(t)$, $f_L(t)$ and $f_J(t)$.

The described system model is generic in the sense that it describes arbitrary MANET scenarios; to examine a concrete MANET scenario, the pdfs introduced here must be derived for the scenario under consideration. How these probabilities can be derived is shown in Section 4.

2.2 Failure Model

The failure model describes failures from a single node's perspective. Failures lead to situations where a node cannot communicate with another node anymore. A node in the system model described above can generally experience a *node failure* or a *communication failure*. Node and communication failures are defined as follows:

Node failure

A node failure describes any event that causes a node to disconnect from $\mathcal{A}$. Hence, the cumulative density function (cdf) $F_N(t)$, the probability for a node to experience a node failure within time t, is given by the probability that (i) a node leaves $\mathcal{A}$, (ii) exhibits an energy-related failure, or (iii) experiences a technical failure. Given the pdfs $f_L(t)$, $f_E(t)$, and $f_T(t)$ from the system model, $F_N(t)$ can be calculated by considering the complementary probabilities of the cdfs $F_L(t)$, $F_E(t)$ and $F_T(t)$ as

$$F_N(t) = 1 - [(1 - F_L(t)) * (1 - F_E(t)) * (1 - F_T(t))] \tag{1}$$

It is assumed that mobile nodes are equipped with some kind of stable storage that survives node failures. Hence, data written to stable storage is available on reconnection to $\mathcal{A}$.

Communication failure

A communication failure describes any event that leads to an outage of the communication between two nodes that are connected to $\mathcal{A}$. A communication failure causes the break of a previously functional communication path induced by the dynamic network topology. The probability for a communication failure to happen within time t is given by the distribution of path durations described by the cdf $F_C(t)$, which is directly derived from the according pdf $f_C(t)$

provided by the system model. Hence, the probability for a communication failure within time t shall be denoted by $F_C(t)$.

$F(t)$ shall denote the cdf of the *general failure* that either a communication or a node failure occurs until t, derived by considering the complementary probabilities of node and communication failures:

$$F(t) = 1 - [(1 - F_C(t)) * (1 - F_N)] \tag{2}$$

From a single node's perspective, $F(t)$ describes the probability that communication with another node fails because either the communication path breaks or because the other node disconnected from $\mathcal{A}$ within t.

3 Transaction Models

I define two transaction models to analyze commit processing in MANETs: one representing the traditional transaction model providing strict atomicity and another transaction model representing advanced transaction models [1, 6, 18, 19, 20] providing semantic atomicity [12]. The calculation model presented in Part II of this report will consider these two models. Both models are based on a general model of distributed transactions and differ in commit processing only.

3.1 General Distributed Transaction Model

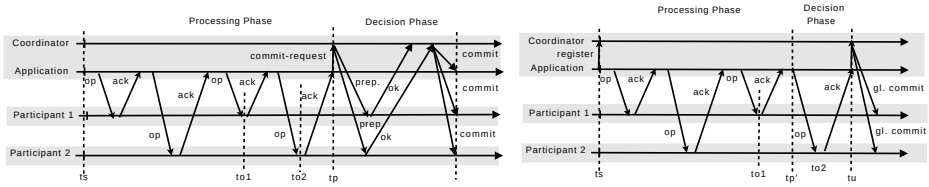
The basic transaction model I consider is the flat ACID transaction model. Following the X/Open DTP model [7], a transaction consists of a set of operations that are issued by an *application*. All operations received by a *participant* constitute a local transaction branch of the global transaction. To avoid the need for initially choosing a *coordinator*, I assume that the application process and the transaction coordinator are colocated. Each execution of an operation is acknowledged by the participant. These acknowledgments are used to detect node and communication failures of participants. If an acknowledgment is not received during a timeout Δ_{op}, the application requests the coordinator to globally abort the transaction. The coordinator will then issue abort messages to all participants.

Generally, I distinguish between the *processing* and the *decision-phase* of a distributed transaction. The processing-phase begins at time t_s when the transaction is initiated, and ends at time t_p when the acknowledgment of the last operation of the global transaction is received by the application.

I assume that a participant i receives the last operation of its transaction branch at some random time t_o. For each participant, the random variable t_o is distributed within the interval $[t_s, t_p]$ according to the pdf $o(t_o)$. The pdf $o(t_o)$ depends on application semantics, i.e. the role of participant i in the transaction. For the sake of simplicity I assume the same pdf for all participants of a transaction. The distribution of operations in $[t_s, t_o]$ for a participant is not

considered. The basic idea of the model is that a failure in the interval $[t_s, t_o]$ is detected at the latest at time t_o, independently of the distribution of operations within this interval.

If no failure is detected during $[t_s, t_p]$, the decision phase is initiated by starting an ACP at time t_p.



(a) Strict model with 2PC.

(b) Semantic model with Early Commit.

Figure 2: General transaction model

Figure 2 depicts the temporal process of a transaction and its spatial distribution in the assumed general transaction model. As already mentioned, the application and the coordinator are assumed to be colocated on the same node, allowing message delays of message exchange between the coordinator and the application to be neglected. Execution delays during the processing-phase are neglected as well. Based on this general model, I differentiate between the strict and semantic transaction model.

3.2 Strict Transaction Model

In the strict model, termination of a local transaction branch is conditionally bound to the termination of the other branches of the same global transaction. A local transaction branch that has already completed all its operations is not allowed to commit until all other remote branches are known to have terminated successfully. To resolve these termination dependencies, a local transaction manager must be able to announce that it is *prepared* to commit its local branch. In the strict model, the standard 2PC protocol as known from [7] is used to terminate the global transaction atomically, thus ensuring *strict atomicity*. Due to its popularity, the 2PC protocol is not described here in detail.

3.3 Semantic Transaction Model

In the semantic model a local transaction branch terminates as soon its last operation is processed. A weaker atomicity notion called *semantic atomicity* [12] is maintained by means of compensating transactions, which semantically undo the effects of a committed branch. A global commit is finalized if all participants have committed, while an abort is issued if at least one local transaction branch has failed. In the case of a local commit conflicting with the global decision, the associated compensation transaction has to be be executed.

As long as a participant is uncertain about the global decision, it has to maintain conditions that allow for compensation. A general problem with this model is that effects of the committed branch are visible to other transactions and that compensation must consider these so-called *dependent transactions*. E.g., the correctness criterion *soundness* [12], requires that the isolated execution of dependent transactions show the same outcome as a schedule that also includes the branch and its compensation. Schedules that violate this condition have to be rejected, thus hindering progress of dependent transactions. Another negative effect is that uncertainty about the global decision possibly leads to non-optimal behavior. For instance, in a disaster mission control application, where rescue units commit themselves to missions, an uncertain unit will move towards the mission site although the mission has possibly been already aborted. Hence, it is not available for other missions. Therefore it is highly desirable also in semantic atomicity to minimize uncertainty about the global decision.

3.4 Blocking in the Strict and Semantic Model

Regarding the type of failure, two blocking situations can be distinguished: (i) a participant suffers from a communication failure with the coordinator, while in its window of uncertainty; and (ii) the coordinator suffers from a node failure while participants are uncertain. In the following the sizes and bound of uncertainty windows in the different transaction models are discussed and notations used within the remainder of this report are presented.

I consider *cooperative-recovery* (CR) [3] for both transaction models to compensate for blocking. Here, a blocked participant tries to retrieve the global decision from other participants by sending them a request at recovery time. A participant can provide the global decision if it has never voted and is therefore free to decide on abort or it has already received the coordinator's decision.

3.4.1 Strict Atomicity

In the strict model, 2PC is started at time t_p. I assume that all *prepare* messages are sent at the same time (t_p). Then the length of the critical window ΔU, where participants are vulnerable to a coordinator's node failure or a failure of the participant, is at minimum $\Delta U_{min} = 2\delta_m$ and at maximum $\Delta U_{max} = 2\delta_m + \Delta_{vo}$. In cases where no undetected participant failure has occurred, the window has length ΔU_{min}, as all participants answer the prepare request within δ_m. Otherwise, the coordinator must await a timeout Δ_{vo}, so that ΔU increases to ΔU_{max}.

3.4.2 Semantic Atomicity

While in strict atomicity ΔU is the same for all participants, in semantic atomicity ΔU is individual for each participant. This is due to the fact that a participant i commits its local transaction branch right after successfully executing its last operation at time t_o, moving into uncertainty afterwards. The ac-

knowledgment of this operation is an implicit *yes* vote to the coordinator. The coordinator later derives the global decision, without requiring an additional vote from this participant. The commit protocol assumed to assure semantic atomicity is straightforward. In fact the last operation of the last participant decides the global transaction. If all other operations have been successful it depends on this operation whether global commit can be decided, otherwise the transaction is already aborted. Hence, in semantic atomicity I define the end of the processing phase as t'_p, which is the time at which the coordinator sends the last operation to the last participant denoted by PA_{last}. Hence, execution and acknowledgment of this operation can be considered as the decision phase. The coordinator derives the global decision at $t_u = t'_p + 2\delta_m + \Delta_{ex}$, where Δ_{ex} is a constant time required for the execution of the last operation.

The size of the uncertainty window is generally wider with semantic atomicity compared to strict atomicity. If a participant i does not experience failure, its individual uncertainty period already starts at time $t_{o,i}$ and ends with $t_u + \delta_m$. If a participant failure is detected at time t_f, two cases have to be distinguished. In case $t_{o,i} \leq t_f$, the phase of uncertainty is described by the interval $[t_{o,i}, t_f + \delta_m]$. For $t_{o,i} > t_f$ the participant is not uncertain, because it receives the coordinator's decision before moving into uncertainty (this causes an abort during the processing-phase).

While this section presents the model underlying this work, these models are related to a concrete MANET scenario.

4 MANET and Transaction Parameter

In this section, I present how the cdfs $F_N(t)$, $F_C(t)$ and $F_{CR}(t)$ describing the probability of node and communication failures in the failure model of this work can be derived for a concrete MANET scenario. Additionally I discuss the application dependent pdf $o(t)$, describing the distribution of operation within $[t_s, t_p]$. As example MANET scenario used within the remainder of this work, I assume the following setting based on a disaster recovery situation:

> 15 mobile recovery units move on a square of 500 m * 500 m according to the Random Way-point (RWP) mobility model at 2.0–5.0 mps, relaying messages for each other using AODV. Batteries of nodes are assumed to deliver 2 h of service, while the mean time to failure due to a technical failure is 500 h. The rescue units form a MANET $\mathcal{A}$ connected to other areas according to the AGB mobility model; each node has an expected sojourn time of 30 min before moving out of $\mathcal{A}$. For example, rescue units salvage injured persons from collapsed buildings in $\mathcal{A}$, and transport them to a rendezvous site outside of $\mathcal{A}$ for medical treatment. Mobile units are assumed to carry PDAs with IEEE 802.11–compliant radio adapters with a radio range of approximately 100 m.

4.1 MANET Parameters

To obtain the probability $F(t)$ that communication between two nodes is possible within t, the cdf for node failures $F_N(t)$ and for communication failures $F_C(t)$ must be determined for this example scenario.

4.1.1 Probability of Node Failures $F_N(t)$

A node failure causes the complete disconnection of a node from $\mathcal{A}$. The probability of this event is derived from the following probabilities: (i) disconnection caused by exhausted energy resources $f_B(t)$; (ii) disconnection due to a technical problem $f_T(t)$; or (iii) because the node moves out of the area of $\mathcal{A}$, given by the pdf $f_L(t)$.

$f_L(t)$ can be directly obtained from the AGB mobility model as defined in [4]. The AGB model takes the probability distribution of how long a node remains within one area as input. In this work, I am assuming an exponentially distributed sojourn time, while any other distribution could be chosen here when more information about the distribution of sojourn times is available. For the example scenario, I am assuming $F_L(t)$ to be an exponential distribution with parameter $\lambda_L = 1/1800$ (the expectation of $F_L(t)$ with λ_L is 30 min). In the real world, sojourn times of participants may differ; for example, a supply team dispatching supplies to rescue workers spends less time in $\mathcal{A}$ than a rescue team working on a mission site with heavy machinery. However, for the sake of simplicity, I assume an average sojourn time for all nodes.

The probability that a node disconnects from $\mathcal{A}$ due to exhausted energy resources until t is denoted by the cdf $F_B(t)$. For a randomly chosen node from $\mathcal{A}$, it is unknown how long it has been operational, and hence how long its energy resources will last. If mobile nodes enter $\mathcal{A}$ with fully charged batteries and have a constant energy consumption, a randomly chosen node from $\mathcal{A}$ can be assumed to have remaining energy resources uniformly distributed in the interval $[0, b]$. b denotes the maximum service time of 7200 s as described above. If nodes are not assumed to enter $\mathcal{A}$ with fully charged batteries, then an exponential distribution with parameter $\lambda_E = 1/b$ is a feasible assumption for $f_E(t)$. It is then modeled that the probability of exhausted energy resources within an infinitesimally small time step is always the same, while the expected service time is b. However, both the uniform distribution over the interval $[0, b]$ as well as an exponential distribution are significant simplifications, because, in reality, the remaining energy is mainly influenced by fluctuating power consumption, which is subject to numerous influences and therefore hard to capture analytically. However, it will be shown later that the probability of a transaction failure due to exhausted energy resources is small and negligible. Therefore, a raw estimate is favored over accurate modeling here. In the following, I will mostly use the uniform distribution over $[0, b]$ for calculations if not stated differently.

The disconnection from $\mathcal{A}$ caused by a technical failure is a rare event. The scenario description states that the mean time to failure is given by 500 h, hence an exponential distribution with $\lambda_T = 1/(18 * 10^4)$ is a meaningful assumption

for $F_T(t)$, which results in a negligible probability of node failures due to technical defects. However, in other scenarios with much cheaper hardware, like sensor nodes, $F_T(t)$ may become more relevant.

If for $f_E(t)$ a uniform distribution over $[0, b]$ is assumed, the probability that a node failure happens within time t is given by:

$$F_N(t) = 1 - [(1 - \frac{t}{b}) * (1 - (1 - e^{\frac{t}{\lambda_L}})) * (1 - (1 - e^{\frac{t}{\lambda_T}}))] \quad (3)$$

Note that in Formula (3), the case $t > b$ is neglected, as I am concerned only with small values of t in the following. In Part II, I will show that transactions with a processing phase larger than 100s are not feasible in the example scenario.

4.1.2 Communication Failures $F_c(t)$

The failure model of this work defines a communication failure as all events that lead to an outage of the communication between two nodes in $\mathcal{A}$, while both communication partners are connected to $\mathcal{A}$. A communication path that was functional before breaks if a direct link between two nodes on the path suddenly becomes unavailable. If no multi-hop routing is used, every link break immediately causes a communication failure, while, with multi-hop routing, the underlying routing scheme possibly provides an alternative route.

The probability that a communication path is available until time t is described by the cdf $F_C(t)$, which is primarily influenced by the *node density*, *radio range of nodes*, *node mobility,* and the *routing scheme* in $\mathcal{A}$. I also show in the following that $F_C(t)$ also depends on the hop count of the path when communication is initiated. As it is complicated to model the numerous dependent events that cause the break of a communication path, most scholars propose statistical analysis of path duration based on simulation studies. For example, in [2, 15] the distribution of path durations for different mobility models is derived by simulation. In [11, 13] an analytical approach is also proposed to approximate the distributions of path durations. However, [11, 13, 2, 15] show that the underlying mobility model impacts path and link durations, but for the most common mobility models, such as RWP, Manhattan Mobility [14], and Freeway Mobility [16], an exponential distribution of path durations for routes with more than 2 hops is a reasonable approximation.

The work cited above solely considers paths with 2+ hop count and derives exponentially distributed path durations. In contrast, I am especially interested in the probability distribution of paths with 1–2 hops. This is due to the fact that the abort rate for transactions initiated in 1–2 hop distances is considerably smaller than for transactions initiated with participants in arbitrary hop distances, as shown in Section 6. In fact, transaction processing with participants in 2+ hops distances mostly shows such a high abort probability that the feasibility of transaction processing must be questioned.

To derive $F_C(t)$ for 1–2 hop paths, as well as for 2+ hop paths in the example scenario, I present a simulation study using the ns2 network simulator. The simulation considers movement in $\mathcal{A}$ only, where 15 nodes move according

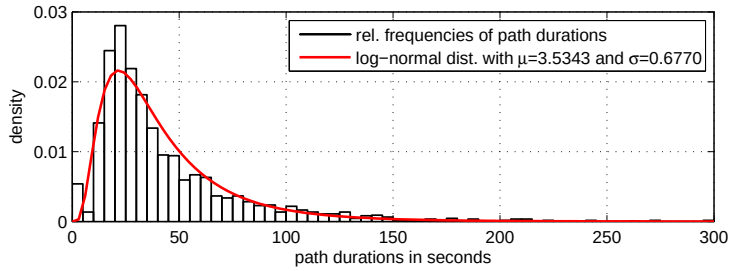
to the RWP mobility model within an area of 500 m times 500 m, as assumed in the example scenario, with speeds of 2.0–5.0 mps and a pause time of 1 s before choosing a new way-point. The following behavior of nodes was simulated in ns2: two nodes in 1–2 or 2+ hop range were randomly chosen and a probe message was exchanged every second between these nodes. A node receiving a probe answered with an acknowledgment message. The time until a communication path breaks, i.e. the time when no acknowledgment for a probe message was received anymore, was measured as well as message delays of all messages exchanged. The derived average message delay δ_m for the example scenario is required later.

The resulting histograms showing the frequencies of measured path durations are shown in Figure 3. Analysis of the durations of paths initiated in 2+ hop range given in Figure 3(b) confirms the results of [11, 13, 2, 15]. For these paths, an exponential distribution of path durations can be presumed. Figure 3(b) shows an exponential distribution with parameter $\lambda = 0.051$ fitting the measured distribution, where λ is derived using the Maximum Likelihood method. An important observation to be made here is the high probability of very short path durations in the exemple scenario, i.e. 22 % of the paths do not survive 5 seconds. I show later that the abort rate in such a setting is not acceptable, even for very short transactions.

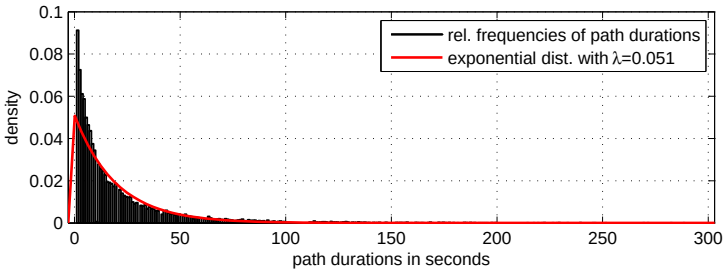
A different picture is obtained for paths that are initiated in 1–2 hop distances. The resulting histogram of path durations in Figure 3(a) shows a very different shape compared to 2+ hop paths. What is important here is the high probability that a path will survive the first seconds after initiation, e.g. for the exemple scenario, 99 % of all links survive the first 5 seconds. After a period with a small risk for a path break right after path initiation, the risk increases quickly, as shown in Figure 3(a), and after 40 s about 60 % of all links must be expected to have suffered from a path break. The path characteristics of 1–2 hop paths is accurately approximated by a log-normal distribution, as shown by the red curve in Figure 3(a) with parameter $\mu = 3.5343$ and $\sigma = 0.677$ for the example scenario. Again, standard techniques such as the Maximum Likelihood method can be used to derive these parameters.

To demonstrate the influence of node speed, Figure 3(c) depicts the results for a simulation assuming the same mobility model as in the example scenario, while node speeds are reduced from 2.0–5.0 mps to 1.0–2.0 mps. Here, path durations are obviously higher, and the distribution of path durations is also accurately modeled by a log-normal distribution.

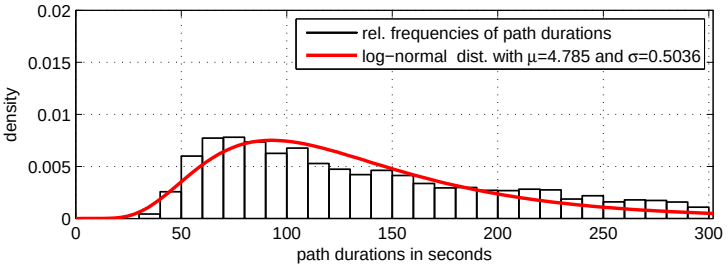
Log-normal and exponential distributions can be assumed for $f_C(t)$ for the common mobility scenarios at moderate and high node mobility, depending on path initiation distance. While assuming exponential distributions eases calculations significantly due to the memory-less property of exponential distributions, I mostly consider a log-normal distribution for $f_C(t)$ as this would be a realistic choice for the example scenario. However, the calculation models presented in Part II can take both distributions as input, as the model does not make any assumptions about the type of the input pdfs.



(a) Relative frequencies of path durations for paths initiated in 1–2 hop distance, measured in the example scenario with AODV multi-hop routing.



(b) Relative frequencies of path durations for paths initiated in 2+ hop distance, measured in the example scenario with AODV multi-hop routing.



(c) Relative frequencies of path duration for paths initiated in 1–2 hop distance, measured in the example scenario at reduced speeds of 1.0–2.0 mps and with AODV multi-hop routing.

Figure 3: Histograms of measured path durations based on 10000 links initiated.

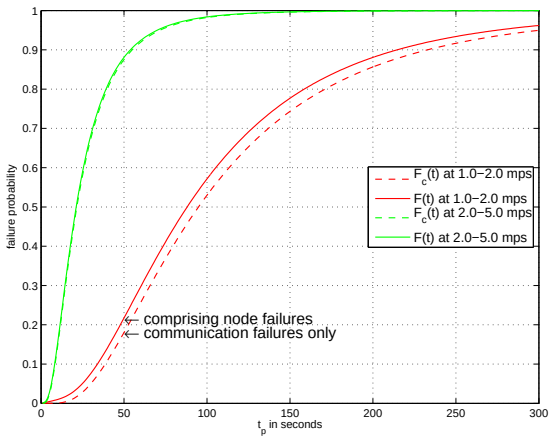


Figure 4: Influence of node failures on $F(t)$.

4.1.3 General Failure $F(t)$

The probability of a general failure happening within time t is given by $F(t)$ and describes the probability that a communication failure between two nodes will occur or that the communication partner will disconnect from $\mathcal{A}$.

Given the cdfs of the example scenario derived above, the general failure probability of the example scenario if 1–2 hop paths are assumed is now given by Formula (4):

$$F(t) = 1 - [(1 - \frac{1}{\sqrt{2\pi} \cdot \sigma} \int_0^t \frac{1}{x} e^{-\frac{(lnx-\mu)^2}{2\sigma^2}} dx) * ((1 - \frac{t}{b}) * (e^{-t(\frac{1}{\lambda_L + \lambda_T})}))] \qquad (4)$$

with values for the parameters σ, μ, λ_T, λ_L, and b as derived in Section 4.1.1.

A plot of $F(t)$ for the example scenario reveals that communication failure is by far the most decisive factor. Figure 4 shows a diagram plotting $F_C(t)$ (dotted lines) and $F(t)$ (solid lines) for the example scenario with node speeds of 1.0–2.0 mps and 2.0–5.0 mps.

Figure 4 shows that, for the example scenario, node failures are almost negligible. Even if nodes move slowly at 1.0–2.0 mps, the influence of node failures on the general failure rate is small compared to communication failures in this setting. For example, at 50 s processing time, considering node failures raises the overall failure probability from 18 % to 22 %, as shown by the red dotted and the red solid curve in Figure 4.

Node failures might have a larger influence in other scenarios, e.g. where the probability $f_L(t)$ to leave the MANET is larger, or more failure-prone hardware like sensors nodes is employed.

One important property of communication failures is that they are assumed to eventually recover if both communication partners remain in $\mathcal{A}$. To under-

stand to what extent failures are transparent to transaction processing, $f_{CR}(t)$ must be derived.

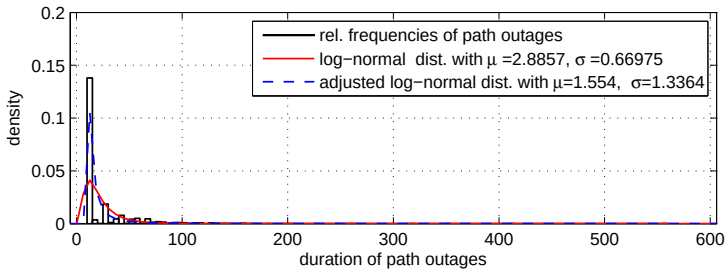
4.1.4 Probability of Path Recovery $F_{CR}(t)$

The time that communication between two nodes is unavailable is influenced by multiple factors. The *node density* and *network size* ($n_{\mathcal{A}}$) influence the probability that an alternative path can be found, while *node mobility* influences the probability that new paths are formed. Additionally, the routing scheme plays an important role, as the time required to detect an invalid route and to initiate discovery of an alternative route differs for multi-hop routing algorithms. Proactive routing like DSDV recognizes broken routes more quickly, because route changes are constantly propagated through the network. DSR maintains multiple paths for one destination, while AODV maintains only one route per destination and has to perform a route discovery whenever a path breaks. As all these factors are hard to grasp analytically, I do a simulation study using the same simulation as in Section 4.1.2, with the difference that exchange of the probe message is continued after the path breaks and the time is measured until the probe message is received again. For the example scenario with AODV routing, Figure 5(a) shows the resulting histogram of path outages. It can be observed that new paths are found with a probability of 14% after 10s. AODV requires some time to recognize that a path is not available anymore and then starts discovery of a new route. A log-normal distribution here fits the distribution of the path outage periods, if the delay δ_{PB} is considered that describes the time AODV requires to detect the path break, as shown by the dotted line in Figure 5(a). $f_{CR}(t)$ is then given by:

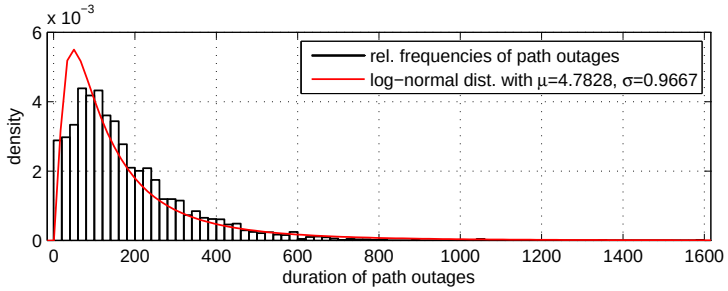
$$f_{CR}(t) = \begin{cases} \frac{1}{(t-\delta_{PB})\sigma_r\sqrt{2\pi}} * e^{\left(-\frac{(ln(t-\delta_{PB})-\mu_r)^2}{2\sigma_r^2}\right)} & for\ t > \delta_{PB} \\ 0 & for\ t \leq \delta_{PB} \end{cases} \quad (5)$$

The influence of the routing scheme and node speeds on path recovery is demonstrated by Figure 5(b) and 5(c). Figure 5(b) shows the distribution of path outages of the example scenario if no multi-hop routing is used, i.e. no messages are relayed and communication is only possible between nodes in direct radio range. Here, long outage periods are more likely than in scenarios where multi-hop routing is used. However, a log-normal distribution with parameters $\mu = 4.78$ and $\sigma = 1.34$ provides a good approximation to these frequencies. Figure 5(c) shows the effect of slow-moving nodes if no multi-hop routing scheme is considered. Here, the duration of path outages is much more widely distributed, and very long outage periods of up to 500 s may occur.

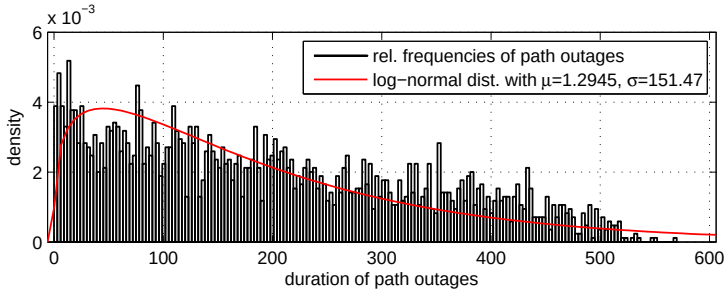
I show in Part II that $f_{CR}(t)$ has a strong influence on abort decisions during the collection phase of transactions and hence has to be considered if multi-hop routing is used.



(a) Relative frequencies of first outage durations with AODV for paths initiated in 1–2 hops distance.



(b) Relative frequencies of first outage durations without multi-hop routing, for link initiated in 1–2 hops distance.



(c) Relative frequencies of first outage durations without multi-hop routing and slow node movement of 1.0–2.0 mps for hops initiated in 1–2 hops distance.

Figure 5: Distribution of path outages measured within 10000 links initiated within the example MANET scenario.

4.2 Transaction Parameters

While above the parameters of the MANET scenario have been derived, I discuss the parameters of the transaction deployed to the MANET scenario in the following. The transaction parameters of a scenario are (i) the transaction model; (ii) the size of processing phase $[t_s, t_p]$; (iii) the number of participants n_p and (iv) the distributions of the last operations given by $o(t_o)$. I do not propose an example transaction scenario, as the main goal of this report is to examine what kinds of transactions are feasible in the example MANET scenario presented at the beginning of this section.

Whether the strict or the semantic transaction model is assumed depends on application semantic, i.e. whether a distributed database application or a non-traditional application is considered.

I consider varying processing periods $[t_s, t_p]$ and $[t_s, t'_p]$ respectively for the examination of transaction scenarios, as one objective of this report is to ascertain what transaction sizes are feasible in MANETs. The size of the processing phase $[t_s, t_p]$ mainly depends on the number of operations issued to participants and the grade of parallelism of operation allocation. Guessing an execution delay, the number of operations issued, and using the given message delay δ_m, it is easy to approximate the size of the processing phase for a concrete transaction. The range of transaction sizes examined in the following will span from 2 s to a maximum of 200 s.

The number of nodes n_p participating in a distributed transaction is assumed to be small, e.g. 2–6 participants. The exact value of n_p depends on the concrete application on hand. In the following, I will mostly assume 3 participants, but I also vary n_p to examine its influence on abort and blocking rates, and extended uncertainty in the semantic model.

The distribution of the time the last operation of a participant's transaction branch is issued, denoted by t_o, has a great influence on abort and blocking rates. Within the interval $[t_s, t_o]$ I assume that messages are constantly exchanged and a failure is detected at last at t_o. Hence, t_o defines the period $[t_o, t_p]$ where a communication failure with a participant or a node failure of the participant is not detected by the coordinator, because no message exchange happens. For example, the coordinator might send some operations to participant i right at the beginning of the transaction, and after receiving the results from i at $t_{o,i}$ only participant j receives operations and i is not contacted again during $[t_{o,i}, t_p]$. Therefore $t_{o,i}$ is the last possibility to detect a failure of participant i before transaction termination starts. Note that t_o is different for every participant, and for the calculation model presented below t_o follows the same probability distribution $o(t_o)$ for all participants. In reality, the distribution of operations during the processing phase depends on the application and the role of each participant. Arbitrary distributions for $o(t_o)$ can be imagined. For simplicity, I assume a uniform distribution of t_o over t_p, i.e.

$$o(t_o) = 1/t_p \tag{6}$$

where t_p is the size of the interval $[t_s, t_p]$ if $t_s = 0$.

In the following the MANET and transaction parameters are used as input for a calculation model predicting abort and blocking risks of transactions.

Part II
Calculation Model

Based on the system and transaction models introduced in Part I of this work, I now develop a calculation model to predict abort and blocking probabilities.

5 Preliminary Considerations

I first present some preliminary calculations, which are frequently used. In many cases, I calculate the probability that an event happens during an interval $[t_1, t_2]$. Given a cdf F, it is computed by the difference $F(t_2) - F(t_1)$. In the following I use the notation $F(t_1..t_2)$ for this probability. For most calculations, I present two variants, one considering a single recovery cycle of communication paths and another neglecting recovery of communication paths. If a cdf F describes the variant that does not consider recovery, then F' denotes the expression considering a single recovery cycle.

Important preliminary results are the probabilities that a transaction actually enters the decision phase of a transaction. The decision phase is not entered if the transaction is aborted within the processing-phase, because the coordinator detects a participant's failure. These probabilities are derived in the following.

5.1 Recognized Failures in the Processing Phase

A participant's node or communication failure is only detected in the processing phase if it happens within the interval $[t_s, t_o]$, since then the coordinator would observe that an operation has not been acknowledged. This event occurs if a participant suffers from a failure at time t_f before the last operation is processed at time t_o, hence if $t_o > t_f$. $P_{o>f}(t_p)$ denotes the probability that a participant's failure happens in the interval $[t_s, t_o]$ and thus is detected by the coordinator. $P_{o>f}(t_p)$ is given by:

$$P_{o>f}(t_p) = \int_0^{t_p} \int_0^{t_o} o(t_o) * f(t_f)\, dt_f dt_o \tag{7}$$

with $t_s = 0$. The bounds of the integrals in $P_{o>f}(t_p)$ are chosen by the following consideration: if $t_o \in [0, t_p]$, then t_f must occur in the interval $[0, t_o]$. I use the subscript of $o > f$ to indicate the failure type considered; i.e. $P_{o>f_N}(t_p)$ denotes the probability that a node failure occurs and is recognized by the coordinator, while $P_{o<f_C}(t_P)$ describes the same for communication failures.

If (i) a log-normal distribution with parameters μ and σ for communication failures, (ii) node failure probabilities as derived above, and (iii) $o(t_p) = 1/t_p$ are applied to $P_{o>f}(t_p)$, then $P_{o>f}(t_p)$ results in the expression given by Formula (8).

$$\begin{aligned} P_{o>f}(t_p) &= \int_0^{t_p} \Bigg[\frac{1}{2 \cdot t_p} * 3 - 2e^{-\lambda_L t_o} - 2e^{-\lambda_T} + \sqrt{\frac{1}{\sigma^2}} \cdot \sigma \\ &\quad + \frac{2 \cdot t_o}{b} * \left(1 - Erf \left[\frac{\mu - ln(t_o)}{\sqrt{2} \cdot \sigma} \right] \right) \Bigg] dt_o \end{aligned} \tag{8}$$

The complementary probability $1 - P_{o>f}(t_p)$ is the probability that a failure occurs and is *not* detected *or* that no failure happens during $[t_s, t_p]$.

5.2 Unrecognized Failures in the Processing Phase

A failure of a participant during $[t_s, t_p]$ is *not* recognized if the failure happens after the last operation was acknowledged. Hence, if $t_o \in [0, t_p]$, then t_f has to be from the interval $[t_o, t_p]$ for this event to happen. The probability that a failure occurs in $[t_s, t_p]$ *and* is *not* recognized by the coordinator is denoted by $P_{o<f}(t_p)$ and is given by:

$$P_{o<f}(t_p) = \int_0^{t_p} \int_{t_o}^{t_p} o(t_o) \cdot f(t_f)\, dt_f dt_o \tag{9}$$

Using a log-normal distribution for $f_C(t)$ as in the example scenario and $f(t)$ as derived in Section 4.1.2, $P_{o<f}(t)$ is given by:

$$\begin{aligned} P_{o<f}(t_p) &= \int_0^{t_p} \Bigg[\frac{2 \cdot \left(e^{-\lambda_L t_o} + e^{-\lambda_T t_o} - e^{-\lambda_L t_p} - e^{-\lambda_T t_p} + \frac{t_p - t_o}{b} \right)}{2 \cdot t_p} \\ &\quad + \frac{Erf \left[\frac{\mu - ln(t_o)}{\sqrt{2}\sigma} \right] - Erf \left[\frac{\mu - ln(t_p)}{\sqrt{2}\sigma} \right]}{2 \cdot t_p} \Bigg] dt_o \end{aligned} \tag{10}$$

The complementary probability $1 - P_{o<f}(t_p)$ describes the probability of the events that either *no* failure occurs during $[t_s, t_p]$ or that a failure occurs *and* is recognized.

5.3 Unrecognized Failure and Recovery

As described in the system model, communication failures are subject to recovery, and the first random outage time of a communication path is described by the pdf $f_{RC}(t)$. A result frequently used is the probability of an agnostic failure in $[t_s, t_p]$. A failure of a participant has no consequences if the failure happens after t_o and recovers by t_p. Given the pdf of path outage $f_{RC}(t_r)$, last operation $o(t_o)$, and communication failure $f_C(t_f)$, the probability of an agnostic communication failure is given by $P_{o<f_C,r}(t_p)$:

$$P_{o<f_C,r}(t_p) = \int_0^{t_p} \int_0^{t_f} \int_0^{t_p - t_f} f_{RC}(t_r) \cdot o(t_o) \cdot f_C(t_f)\, dt_r dt_o dt_f \quad (11)$$

The bounds of the integrals in $P_{o<f_X,r}(t_p)$ are chosen according to the following consideration: if $t_f \in [0, t_p]$ then $t_o \in [0, t_f]$ and $t_r \in [0, t_p - t_f]$.

Note that $P_{o<f_C,r}(t_p)$ assumes only a single failure and recovery cycle, while in reality multiple failure and recovery cycles may occur over time. However, the probability of multiple failure and recovery cycles is negligible for the short transactions considered in this work. Simulation results presented later show that accurate predictions are derived by considering one recovery cycle. If for $f_C(t)$ and $F_{RC}(t)$ exponential distributions are assumed, consideration of multiple failure and recovery cycles is possible, because the memoryless property of exponential distributions can be exploited to model a stochastic process describing the states of a path. Such calculations are omitted here, but their integration in the calculation model presented here is straightforward.

The complementary probability $1 - P_{o<f_C,r}(t_p)$ describes the probability that (i) no failure happens during $[t_s, t_p]$, *or* (ii) that a failure is experienced and recognized, *or* (iii) that a failure is not recognized *and* does not recover until t_p. Note that recovery from node failures is not considered for calculation of abort and blocking risks as motivated in Section 4.1.4. Therefore, node failures are not considered here, while for the calculation of the probability that a failure is not recognized by the coordinator and does not recover before t_p, node failures have to be considered as shown in the following.

5.4 Unrecognized Failure and no Recovery

The event that a participant suffers from an unrecognized failure in the interval $[t_s, t_p]$ and the failure does not recover until t_p may occur in two situations: (i) if the participant suffers from a communication failure that does not recover until t_p, or (ii) if the participant suffers from a node failure.

The probability of the first event is calculated by $P_{o<f_C,nr}(t_p)$

$$P_{o<f_C,nr}(t_p) = \int_0^{t_p} \int_0^{t_f} \int_{t_p - t_f}^{\infty} f_{RC}(t_r) \cdot o(t_o) \cdot f_C(t_f)\, dt_r dt_o dt_f \quad (12)$$

where the bounds of the integrals are chosen as follows: if $t_f \in [0, t_p]$, then $t_o \in [0, t_f]$ and the outage time of the communication path must exceed the remaining processing phase, hence $t_r \in [t_p - t_f, \infty]$. If situation (ii) is considered, the probability that a general failure happens in $[t_s, t_p]$, which is *not* recognized and does *not* recover by t_p, is given by $P_{o<f,nr}(t)$:

$$P_{o<f,nr}(t_p) = 1 - [1 - P_{o<f_N}(t)] * [1 - P_{o<f_C,nr}(t)] \quad (13)$$

Given the preliminary considerations above, calculations to predict abort and blocking probabilities are presented in the following.

6 Abort Probability

The abort probability of a transaction is central to deciding whether atomic transaction processing is feasible at all in a certain MANET scenario with the strict or semantic transaction model. Transaction processing has to be assumed as unfeasible if, for a transactional system deployed in a MANET scenario, a high rate of started transaction must be expected to abort due to node or communication failures. The tolerance of abort rates depends on application semantics, but generally I assume here that an abort rate larger than 20 % is not tolerable for most applications. Since I am concerned with the influence of failures induced by the MANET environment, I assume that the ACID properties for local transaction branches are generally guaranteed and do not cause transaction aborts, i.e. participants always vote for commit.

The abort probability is also important, as it is a major factor in the examination of blocking probabilities. Note that blocking situations can only occur if the transaction has not been aborted before. This effect is especially strong in the strict model, where participants move into uncertainty at t_p.

In the following, I present a calculation model to derive the abort probabilities for the strict and semantic transaction models. The model is then applied to the example MANET scenario of this work.

6.1 Abort Probability in the Strict Model

The duration of the processing phase is determined by the number and distribution of operations and therefore application dependent. In contrast in the strict model, the size of the decision phase denoted by ΔU solely depends on whether all participants receive the prepare message, i.e. if an unrecognized failure of a participant causes the coordinator to await time-out Δ_{vo}.

A transaction can be aborted during the processing phase $[t_s, t_p]$ or during the decision phase $[t_p, t_p + \Delta U]$. Both events are mutually exclusive and considered separately in the following. Abort is decided in $[t_s, t_p]$ if the coordinator misses an acknowledgment and within $[t_p, t_p + \Delta U]$ if a vote is missing. First, I consider the probability of transaction abort in the interval $[t_s, t_p]$ and afterwards for the decision phase $[t_p, t_p + \Delta U]$.

6.1.1 Abort Probability in the Processing Phase

The probability that in a transaction with n_p participants all participants either do not suffer from a failure within $[t_s, t_p]$ or the failure is not recognized is calculated by $([1 - P_{o>f}(t_p)]^{n_p})$. The complement of $([1 - P_{o>f}(t_p)]^{n_p})$ describes the probability that at least one participant suffers from a recognized failure in $[t_s, t_p]$. This is the probability of a transaction to abort in $[t_s, t_p]$ caused by a participant failure. If the coordinator suffers from a node failure within $[t_s, t_p]$, participants will abort unilaterally at $t_p + \Delta U_{max} + \delta_m$. This is safe, as no participant will move into prepared state, because no prepare message can arrive. The probability that a transaction is aborted can now be calculated by

the probability that either a recognized participant failure *or* a node failure of the coordinator happens within $[t_s, t_p]$ denoted by $P_{a_p}(t_p)$.

$$P_{a_p}(t_p) = 1 - [1 - P_{o>f}(t_p)]^{n_p} * [1 - F_n(t_p)] \tag{14}$$

6.1.2 Abort Probability in the Decision Phase

In interval $[t_p, t_p + \Delta U]$, a transaction is aborted if the coordinator misses the vote of a participant after awaiting a timeout Δ_{vo}. This can happen either because a participant has not received a prepare message or its vote message cannot be transmitted due to a communication failure. The prepare message is not received in three events: (A) if a participant suffers an unrecognized failure that does not recover until t_p; (B) if the prepare message is lost due to a communication or node failure of a participant in $[t_p, t_p + \delta_m]$; and (C) if a participant receives the prepare message, but its vote message is lost due to a communication failure within the interval $[t_p + \delta_m, t_p + 2\delta_m]$.

If recovery of communication failures is not considered, the probability that at least one of n_p participants experiences situation A while the coordinator does not suffer a node failure is given by $PA(t_p)$.

$$PA(t_p) = ([1 - P_{o>f}(t_p)]^{n_p} - [1 - F(t_p)]^{n_p}) * [1 - F_n(t_p)] \tag{15}$$

Node failures of the coordinator within $[t_p, t_p + \Delta U]$ are not considered, as this situation causes blocking, which is not considered here but in Section 8. The probability of situation B is given by $F(t_p..t_p + \delta_m)$, that of C by $F_C(t_p + \delta_m..t_p + 2\delta_m)$. Since the events B and C are not independent, the probability for the event that B or C occurs is calculated by $PBC(t_p)$.

$$\begin{aligned} PBC(t_p) &= F(t_p..t_p + \delta_m) + F_c(t_p + \delta_m..t_p + 2\delta_m) \\ &\quad -F(t_p..t_p + \delta_m) * F_c(t_p + \delta_m..t_p + 2\delta_m) \end{aligned} \tag{16}$$

The probability that abort is decided in the decision phase if recovery of communication paths is not considered is now given by

$$P_{a_d}(t_p) = PA(t_p) + [1 - F(t_p)]^{n_p} * PBC(t_p) \tag{17}$$

In the case where recovery of communication paths is considered, an additional event must be regarded for situation A. This is the situation that at least one of n_p participants suffers from an unrecognized failure in $[t_s, t_p]$ that does not recover by t_p, while the other participants do not suffer from a failure in $[t_s, t_p]$ or an unrecognized failure occurs which recovers in time. This probability is calculated by $PA'(t_p)$

$$\begin{aligned} PA'(t_p) &= \Bigg[\sum_{i=1}^{n_p} \binom{n_p}{i} P_{o<f,nr}(t_p)^i \\ &\quad * \sum_{j=0}^{n_p - i} \binom{n_p - i}{j} [1 - F(t_p)]^j P_{o<f,r}(t_p)^{n_p - i - j}\Bigg] \end{aligned} \tag{18}$$

If recovery of communication paths is considered for situation B and C, the probability that at least one participant suffers B or C has to consider that participants may suffer from a failure that recovers by t_p. Hence, the probability that, for at least one of n_p participants, event B or C occurs is given by $PBC'(t_p)$.

$$PBC'(t_p) = \sum_{i=0}^{n_p} [1 - F(t_p)]^i * P_{o<f_C,r}(t_p)^{n_p - i} * (1 - [1 - PBC(t_p)]^i) \quad (19)$$

The probability of abort in the decision phase, considering one recovery cycle of communication failures, is now given by $P'_{a_d}(t_p)$

$$P'_{a_d}(t_p) = PA'(t_p) + PBC'(t_p) \quad (20)$$

The overall risk of a transaction to abort is now simply calculated by considering the probability that abort is decided in interval $[t_s, t_p]$ or $[t_p, t_p + \Delta U]$. If no recovery of communication is considered, $P_a(t_p)$ gives the overall abort probability.

$$P_a(t_p) = P_{a_p}(t_p) + P_{a_d}(t_p) \quad (21)$$

$P'_a(t_p)$ analogously gives the overall abort probability if a single recovery cycle of communication failures is assumed.

$$P'_a(t_p) = P_{a_p}(t_p) + P'_{a_d}(t_p) \quad (22)$$

6.1.3 Abort Predictions and Simulation Results

In the following, the calculation model described so far is applied to the example scenario and compared to measurements obtained from simulation experiments. Figure 6(a) presents the predicted abort rates for the example MANET scenario with three transaction participants and strict transactions for processing phases varying from 1–300 s. Additionally, I will present abort rates measured in experiments. Simulation experiments are done using the ns2 network simulator and the same mobility and radio settings as for the simulations presented in Section 4.1.2 and Section 4.1.4. The following behavior was implemented in ns2 to simulate a transaction of the strict model:

A transaction is initiated by choosing a random node to act as coordinator. Then n_p participants are randomly chosen from all nodes in 1–2 hop distance. For every participant, t_o is calculated using $o(t_p)$. Beginning with transaction start and ending at t_o, the coordinator sends a message representing an operation to every participant that has not reached its t_o. A participant receiving such a message replies with an acknowledgment. If the coordinator does not receive an acknowledgment for an operation message within time-out δ_{to}=1 s, the transaction is aborted. If acknowledgments for all issued operations are received by t_p, the 2PC protocol is initiated. Here, abort is decided if a vote times out after Δ_{vo}=1 s.

Figure 6(a) compares the abort rates predicted by the proposed calculations, with measurements obtained from the simulation study for the example scenario.

Abort in Processing Phase

It can be observed that the predicted abort rate in the processing phase approximates the measured rates accurately, independent of the routing mechanism used. For example, at a processing phase of 40 s, $P_{a_p}(t_p)$ predicts an abort rate of 55.7 %, while the measured rate of transactions aborted in the processing phase is 55.4 % with AODV. If no multi-hop routing is used, the prediction of $P_{a_p}(t_p)$ also meets the measured rate accurately. Generally, the probability of an abort decision in the processing phase monotonically increases over t_p and shows a log-normal like shape, as its major influence is the probability for communication failures during the processing phase.

Abort in Decision Phase

In the decision phase, it is observed that the rate predicted by $P_{a_d}(t_p)$ is higher than the real abort rate of the example scenario. For 40 s processing phase, Formula $P_{a_d}(t_p)$ predicts a 37.4 % abort probability, while the measured rate is only 13.7 %. This deviation is explained by the fact that $P_{a_d}(t_p)$ neglects recovery of failed communication paths, i.e. the event that the communication path between a participant and the coordinator fails after t_o, but recovers by t_p is not included. The probability for this event is high if multi-hop routing is used, while it is negligible without. Hence, in the case that no multi-hop routing is used, $P_{a_d}(t_p)$ approximates the real abort rate accurately as shown in Figure 6(b), e.g. at a processing phase of 40 s, an abort rate of 36.3 % is observed here, while $P_{a_d}(t_p)$ predicts 38.4 %. If $P'_{a_d}(t_p)$ is used for prediction of abort rates, a good approximation of the expected abort rate is also achieved for scenarios with multi-hop routing as shown in Figure 6(a). For small t_p the measured values are slightly smaller than predicted by $P'_{a_d}(t_p)$, while for large t_p the measured abort rate in decision phase is slightly higher than predicted. Higher predicted values for small t_p are explained by the effect that, for small t_p, the message delay is smaller than for large t_p, as the participants and the coordinator remain in closer vicinity and communication is mostly single hop. The prediction calculated by $P'_{a_d}(t_p)$ uses δ_m, which is an average value greater than the real message delay for small t_p. Smaller predictions for large t_p are explained by the fact that $P'_{a_d}(t_p)$ considers only a single failure recovery cycle, while in reality multiple failure and recovery cycles can occur. Especially for large t_p the probability of multiple failure and recovery cycles of communication paths occurring increases. However, it is shown that the calculation model presented above accurately predicts the dimension of the expected abort rate for a given MANET scenario.

Overall Abort Rate

The overall abort rate for the example scenario is high. For example, at t_p=20 s an overall abort rate of 32.7 % is observed. A tolerable abort rate smaller than 20 % for transactions with three participants is found at a processing times smaller 15 s. If no multi-hop routing is used, feasible transaction processing is

only possible for transactions shorter than 13 s in the example scenario. How sensitive this result is to node speeds is shown in Figure 6(c). Here, the effect of node speeds is demonstrated. In Figure 6(c), the example MANET scenario is simulated with lower node speeds of 1.0–2.0 mps. At these lower speeds, the predicted and measured abort rates are significantly smaller than at 2.0–5.0 mps. Here, transaction with a size up to 60 s show an abort probability smaller than 20 %. In Figure 6(e), the influence of multi-hop routing can be observed. If multi-hop routing is used, the measured abort rate decreases from 54.4 % to 32.7 % at 20 s processing time compared to a scenario where no multi-hop routing is used.

Initiation in 1–2 Hop vs. 2+ Hop Distances

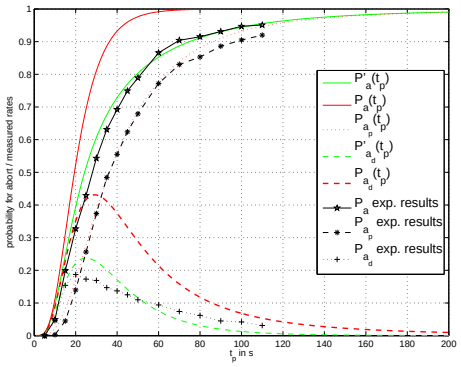
As yet, I have only presented results for transactions initiated in 1–2 hop distances. I stated in Section 4.1.2 that transaction processing in the example scenario is not feasible if transactions are initiated among participants that are in 2+ hop distances. The reason can be observed in Figure 6(d), showing the abort rate in the processing phase for 1–2 and 2+ initiation distances. For 2+ hop distance transactions, the abort probability in the processing phase is larger than 20% for transactions with a processing phase greater than 3s. Here, only very short transactions are feasible at all with three participants. I argue that in such a scenario transaction processing is not feasible, as the abort rate is unacceptable.

6.2 Abort Probability in the Semantic Model

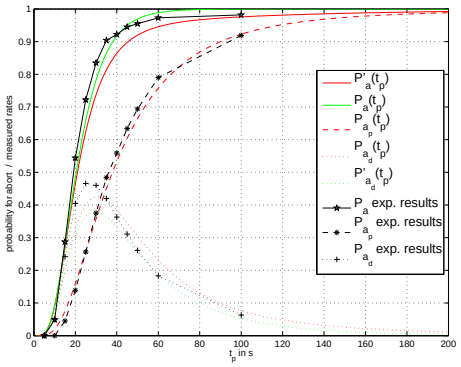
The semantic transaction model allows for temporarily diverse commit decisions of participants. A participant derives a local preliminary decision on abort or commit that is verified later when the final decision is made by the coordinator. The processing phase of a transaction ends at t'_p, when the coordinator issues the last operation to the last participant. Successful acknowledgment of this operation decides the global transaction. Therefore, the global decision is derived at time $t'_P + 2\delta_m + \Delta_{ex}$, where Δ_{ex} is the time required by the last participant to execute its last operation; I denote this point in time as t_u.

In the semantic scheme, the decisive factor for transaction abort is the probability of abort during the processing phase, because the effect that an unrecognized failure that does not recover in time causes an abort decision in the decision phase does not exist. Thus, the abort probability is expected to be lower than in the strict model. Node failures of the coordinator during the processing and decision phase are not considered in the following calculations, because in the semantic model these failures cause an extended uncertainty situation, which is extensively examined in Section 8.

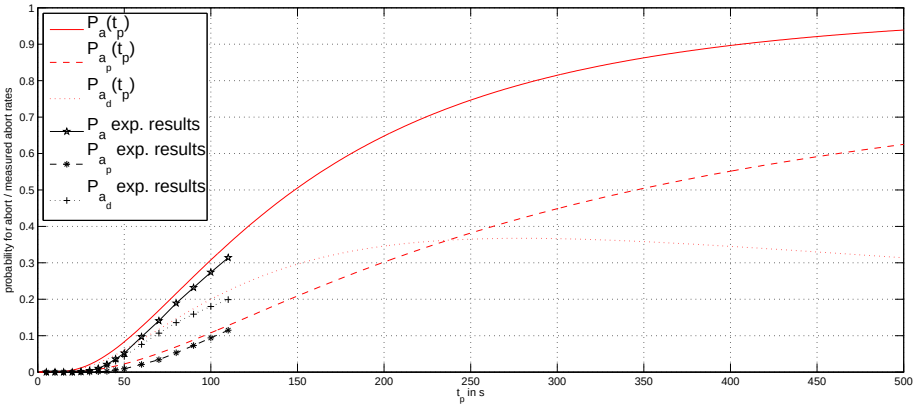
In the semantic model, I denote the probability of abort during the processing phase as $P^*_{a_p}(t'_p)$, which is computed by the complementary probability that neither all nodes in PA_{other} do not cause an abort nor does PA_{last} during



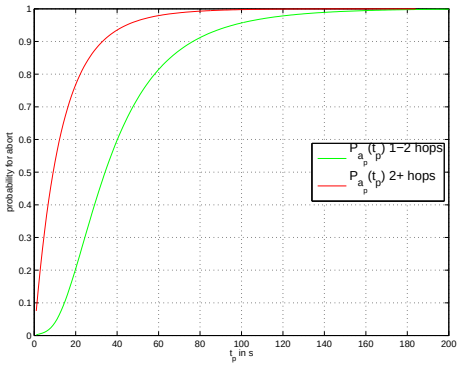
(a) Abort rates in the example MANET scenario with AODV multi-hop routing and the strict transaction model. Simulation and predictions are based on $n_p = 3$. Prediction used an average message delay of $\delta_m = 180ms$.



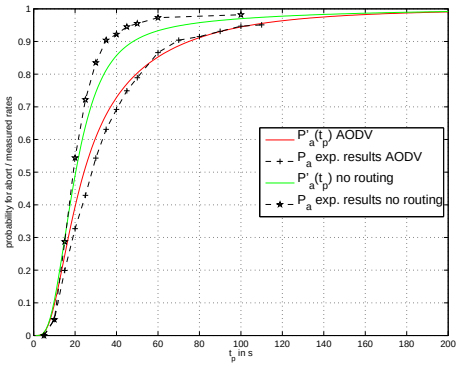
(b) Abort rates for the example MANET scenario *without* multi-hop routing. Simulation and predictions are based on $n_p = 3$. Prediction used an average message delay of $\delta_m = 180ms$.



(c) Abort rates for $n_p = 1$ and reduced speeds of 1.0–2.0 mps in the example MANET scenario and strict transaction model.



(d) Comparison of abort rates in the processing phase for 1–2 and 2+ hops initiation distances.



(e) Abort rates with and without multi-hop routing.

Figure 6: Abort probabilities in the strict transaction model. All measurements are based on 10000 transactions initiated.

$[t_s, t_p]$.

$$P^*_{a_p}(t'_p) = 1 - [1 - P_{o>f}(t'_p)]^{n_p - 1} * [1 - F(t'_p)] \quad (23)$$

In the interval $[t'_p, t_u]$ only a failure of PA_{last} can cause an abort decision denoted by $P^*_{a_d}(t'_p)$. For this event to happen, the transaction should not be aborted during the processing phase and PA_{last} has to suffer from a node or communication failure in the interval $[t'_p, t_u]$.

$$P^*{}_{a_d}(t_p) = [1 - P_{o>f}(t'_p)]^{n_p - 1} * F(t'_p..t_u) \quad (24)$$

The overall probability of a transaction being aborted in the semantic model is given by $P^*_a(t'_p)$.

$$P^*_a(t'_p) = P^*_{a_p}(t'_p) + P^*_{a_d}(t'_p) \quad (25)$$

6.2.1 Predictions and Simulation Results

In Figure 7, the abort rate predicted by $P^*_a(t_p)$ is compared with measurements obtained from an ns2 simulation study. In the simulation study, the message exchange of the semantic model was implemented. Similar to the simulation study of the previous section, coordinators and $n_p = 3$ participants in 1–2 hop distance are randomly chosen. For every participant, t_o was derived by $o(t_p)$ as given in Formula (6). Operation messages are issued to a participant by the coordinator every second until the participant reaches t_o. If acknowledgments for operations are not received within $\delta_{to} = 1s$, abort is decided. The last participant waits for $\Delta_{ex} = 1s$ before answering its last operation. In the simulation, different routing agents were used, first the AODV routing agent and then the ns2 Dumb routing agent.

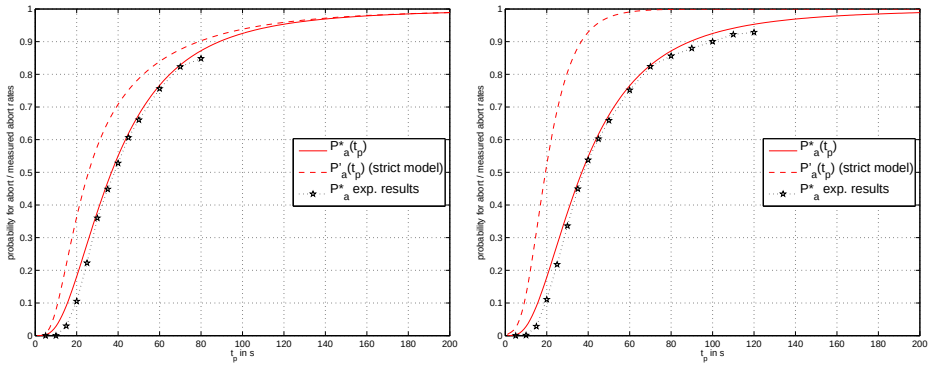
The simulation results validate the predictions of $P^*_a(t_p)$ in both cases. It can be observed that $P^*_a(t_p)$ predicts slightly higher abort probabilities than observed in experiments for small t_p. Similar to the previous section, this is explained with smaller δ_m in reality for short transactions, while $P^*_a(t_p)$ uses an average estimate of δ_m for long and short transactions.

In contrast to the strict transaction model, multi-hop routing has little influence on the abort rate, as recovery of communication links does not influence the abort probability in the semantic model.

Another important result is the validation of the presumption that the abort probability in the semantic model is smaller than in the strict model. It showed that this is especially true in the case where no multi-hop routing is used. In this case, transactions with t_p=20 s show an abort risk of 10.5 % in the semantic model, while in the strict model abort has a probability of 54.4 %. In case multi-hop routing is used, then the decrease in abort probability is smaller, e.g. 10.4 % in the semantic model compared to 32.7 % in the strict model at t_p=20 s.

6.3 Summary and Discussion - Abort Probabilities

In this section, I presented formulae to approximate the abort rate of transactions in the strict and semantic transaction model. The general observation is that the abort rate is high in the example MANET scenario. Abort rates of



(a) Abort probability of semantic transactions in the example scenario with AODV routing.

(b) Abort probability of semantic transactions in the example scenario without multi-hop routing.

Figure 7: Abort probabilities in the example MANET scenario and semantic transaction models. The measured abort rates are based on 10000 initiated transactions.

similar dimensions should be expected in other MANET scenarios that are of the same class as the example scenario, i.e. scenarios that show similar node speeds and network densities. It can be ascertained that only short transactions with values of t_p<10 s in the strict model and t'_p<20 s in the semantic model are feasible if abort rates smaller than 10 % are required. Generally, the semantic transaction model shows a lower susceptibility for abort than the strict model, e.g. 10.4 % at t'_p=20 s, while in the strict model 32.7 % is observed at t_p=20 s. However, the abort probability is mainly influenced by node mobility and node speeds; decreased node speeds drastically reduce the expected abort rate, as shown in Section 6.1.3. Multi-hop routing slightly reduces the number of aborts, as the probability for transient failures that are tolerated in the strict model increases.

I verified the presented calculation model by simulations using the ns2 network simulator. They show that abstractions of the system model accurately describe the real-world behavior of transactions in MANETs. The calculation model presented provides accurate predictions of abort rates in a real world setting.

Although abort rates have been only presented for the example MANET scenario, I argue that this section shows that a primary problem of transaction processing in MANETs are high abort rates. It is important to keep in mind that other failure situations, such as the blocking situations considered in the following, are subsequent problems, as blocking can only occur if a transaction was not aborted before.

7 Probability of Blocking caused by Participant Failures

In the strict and the semantic transaction models, a participant encounters a blocking or extended uncertainty situation if it suffers a communication failure with the coordinator or disconnects from $\mathcal{A}$ while it is uncertain about the global decision. The probability of this event is strongly influenced by the probability that participants enter their uncertainty window and by the extend of the uncertainty period.

In the following, the probability of this blocking situation is analyzed in the example MANET scenario for strict and semantic transactions.

7.1 Probability of Blocking in the Strict Model

In 2PC, a participant failure causes blocking if a communication failure with the coordinator or a disconnection of the participant happens in the interval $[t_p, t_p + \Delta U]$. The aim of this section is to develop calculations that predict the probability of a participant to experience such a situation.

The formulae I present in the following have to be interpreted from a single participant's perspective, i.e. they describe the probability of an individual participant to suffer from blocking. In the following, I denote this participant by PA, the set of the other $n_p - 1$ participants is called PA_{other}.

The probability of blocking is calculated by considering the probability that PA enters its uncertainty window and that a failure occurs while PA is uncertain. As described in Part I, the uncertainty window ΔU in 2PC can be of size $\Delta U_{min} = 2\delta_m$ or of size $\Delta U_{max} = 2\delta_m + \Delta_{vo}$ when the coordinator awaits a time-out Δ_{vo} for a missing vote. The most decisive factors in the computation of the blocking risk of PA are the probabilities for entering the uncertainty window and that the uncertainty window is extended to ΔU_{max}. Thus, the probabilities for entering an uncertainty window of size ΔU_{min} or of size ΔU_{max} are required. A condition for the event that PA enters the uncertainty window is that the receipt of the prepare message by PA is not hindered by a failure of PA. The probability of this condition to be met is given by $[1 - P_{o<f,nr}(t_p)]$ denoted by $P_{AU}(t_p)$.

The probability that ΔU is of size ΔU_{min} is given by the probability that all $n_p - 1$ participants do not suffer from a failure or that all failures recover by t_p. Hence, all nodes in PA_{other} receive the prepare message and answer with a vote message, resulting in ΔU_{min}. This probability is given by $P'_{U_{min}}(t_p)$ if a single recovery cycle of communication failures is assumed:

$$P'_{U_{min}}(t_p) = [1 - P_{o<f,nr}(t_p)]^{n_p - 1} \tag{26}$$

and by $P_{U_{min}}(t_p)$ if no recovery of communication is considered:

$$P_{U_{min}}(t_p) = [1 - F(t_p)]^{n_p - 1} \tag{27}$$

If at least one participant of PA_{other} does not reply with a vote message, the uncertainty window enlarges to size ΔU_{max}. If recovery of communication failures is assumed, the probability of this event is denoted by $P'_{U_{max}}(t_p)$ and given by the probability that at least one node of PA_{other} suffers a failure that does not recover until t_p, while the other nodes in PA_{other} either do not suffer from a failure or the failure recovers by t_p:

$$P'_{U_{max}}(t_p) = \sum_{i=1}^{n_p-1} \left[\binom{n_p-1}{i} * P_{o<f,nr}(t_p)^i * [1 - P_{o<f,nr}(t_p)]^{n_p-1-i} \right] \quad (28)$$

$P_{U_{max}}(t_p)$ describes the probability that the uncertainty window of PA is of size ΔU_{max} in case recovery of communication failures is not considered. $P_{U_{max}}(t_p)$ solely requires that at least one node in PA_{other} suffers from an unrecognized failure given by

$$P_{U_{max}}(t_p) = [1 - P_{o>f}(t_p)]^{n_p-1} - [1 - F(t_p)]^{n_p-1} \quad (29)$$

The probability that PA suffers from a failure within its window of uncertainty is given by $F(t_p..t_p + \Delta U)$. I denote the probability $F(t_p..t_p + \Delta U_{max})$ by $UF_{max}(t_p)$ and define $UF_{min}(t_p)$ analogously.

The risk of PA of suffering from a blocking situation caused by a failure during uncertainty can now be derived as the probability that PA enters an uncertainty window of size ΔU_{min} or ΔU_{max} and that a failure occurs during this period. This probability is computed by $P'_u(t)$ in case recovery of communication is considered:

$$\begin{aligned} P'_u(t_p) &= P_{AU}(t_p) * P'_{U_{max}}(t_p) * UF_{max}(t_p) \\ &\quad + P_{AU}(t_p) * P'_{U_{min}}(t_p) * UF_{min}(t_p) \end{aligned} \quad (30)$$

If recovery of paths is not regarded, the risk of PA of suffering blocking is given by $P_u(t_p)$:

$$\begin{aligned} P_u(t_p) &= P_{AU}(t_p) * P_{U_{max}}(t_p) * UF_{max}(t_p) \\ &\quad + P_{AU}(t_p) * P_{U_{min}}(t_p) * UF_{min}(t_p) \end{aligned} \quad (31)$$

7.1.1 Predictions and Simulation Results

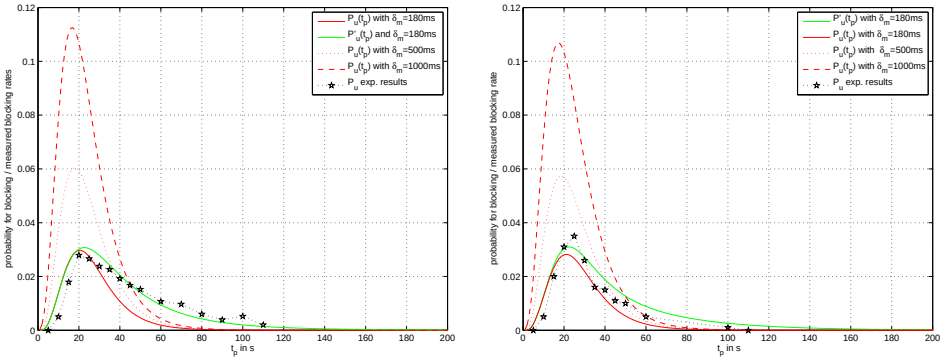
To verify the developed formulae predicting blocking caused by participants, I did a simulation study using ns2. In this study, the message flow of strict transactions with $n_p = 3$ and with 2PC is simulated. Transaction initiation and processing is similar to the simulations of Section 6.1. In the simulation study done here, node failures of the coordinator are anticipated to limit blocking situations to cases caused by participant failures only. In the simulation, participants are informed about the global decision of the coordinator at time $t_p + \Delta U$. The coordinator issues a message with the global decision to all participants at this point in time. To measure the number of blocking situations, all

transaction participants are examined if they have received the global decision or if a participant remains uncertain.

Figure 8 depicts the results of the simulation study and compares the measured blocking rates to predictions of $P_u(t_p)$ and $P'_u(t_p)$. Figure 8(a) shows measurements and predictions for the example MANET scenario with AODV routing, while Figure 8(b) depicts the same without multi-hop routing.

The important observation concerning transaction processing in the example MANET scenario is that the probability of blocking induced by participant failures is low compared to abort probabilities. For example, at processing times smaller than 15s, the blocking probability is at maximum 2%.

Additionally it can be observed that the predictions of $P'_u(t_p)$ meet the measured data better than $P_u(t_p)$ in case AODV is used (see Figure 8(a)). If no multi-hop routing is used, $P_u(t_p)$ and $P'_u(t_p)$ provide accurate approximations of the measured blocking rate (see Figure 8(b)). For multi-hop routing, the probability of multiple sequential path-outages and recovery-cycles increases for large t_p and leads to slightly higher blocking rates than predicted by $P'_u(t_p)$.



(a) Probability of blocking caused by participant failures in the example MANET scenario with AODV.

(b) Probability of blocking caused by participant failures in the example MANET scenario without multi-hop routing.

Figure 8: Probability of a blocking situations caused by a participant failure in the strict model. Transaction parameters are $\Delta_{vo} = 1s$ and $n_p = 3$. The measured blocking rates are based on 10000 initiated transactions.

Figure 8 also show that the message delay δ_m has a major influence on blocking probability. If a message delay of δ_m=1 s is assumed, the blocking probability rises to 10.5 %. Such message delays are imaginable if the traffic load is very high in a MANET scenario.

However, the blocking risk examined here can be further reduced if blocked participants execute a cooperative-recovery scheme. Only if this scheme is unsuccessful is a participant blocked for an undefined period of time. In the following, the probability of this event is examined.

7.1.2 Probability of Blocking with Cooperative-Recovery

A participant suffering a blocking situation caused by a communication failure with the coordinator initiates a cooperative-recovery scheme as described in Part I. The recovery scheme is started at time $t_{cr} = t_p + \Delta U_{max} + \delta_m$, as this is the latest time that the global decision can arrive from the coordinator. The success of the cooperative-recovery scheme depends on the probability that PA can reach one of PA_{other} that is not blocked.

For communication paths between PA and nodes in PA_{other}, I assume the same distribution of path durations as for communication paths between the coordinator and participants. Note that this is a simplification, as in reality participants can be at maximum in 4 hop distance if the transaction is initiated in 1–2 hop distances.

If recovery of paths is disregarded, the probability that PA can reach another node in PA_{other} is given by $F(t_{cr})$. In case recovery of communication paths is considered to derive more realistic predictions if a multi-hop routing scheme is used, the situation that communication between two participants fails and recovers has to be considered. I therefore define $P_{c,nr}(t)$ as the probability of the situation that a communication path between PA and one of PA_{other} breaks and does not recover until t. $P_{c,nr}(t)$, calculated as

$$P_{c,nr}(t) = \int_{0}^{t} \int_{t-t_{f_c}}^{\infty} f_C(t_{f_c}) f_{RC}(t_r) dt_r dt_{f_c} \tag{32}$$

$F(t_{cr})$ converges to 0 and $P_{c,nr}(t_{cr})$ converges towards 1.0 for large t_{cr}. In the real world the reachability of another participant is obviously not certain or impossible for large t_{cr}, because a communication path will most likely experience numerous failure and recovery cycles over time. While for exponentially distributed path durations and recovery probabilities a stochastic process can be used to derive probabilities for the state of the communication path considering multiple failure and recovery cycles, this is not possible for log-normal distributed path durations. One option is to use the path probability P_{path} as an approximation for the probability that two participant nodes can reach each other for cooperative recovery. However, for small t_{cr} the path probability will underestimate the probability that two participants can reach each other and is better approximated by $F(t_{cr})$ and $P_{c,nr}(t_{cr})$, while for larger t_{cr} the real probability is approximated better than by $F(t_{cr})$ and $P_{c,nr}(t_{cr})$. In the following I will present calculations using $F(t_{cr})$, $P_{c,nr}(t_{cr})$ and P_{path}.

To derive the probability that PA experiences a blocking situation and cooperative-recovery is not successful, the state of nodes in PA_{other} has to be considered. If PA is blocked, a node in PA_{other} can experience one of three situations: (i) the node never received a prepare message and therefore never entered uncertainty; (ii) the prepare message is received, the participant voted and also received the global decision; or (iii) the participant is blocked like PA.

Participants in PA_{other} that experienced situation (i) and (ii) are potential cooperation partners for PA. Cooperative-recovery is not successful if PA

cannot reach at least one of these nodes.

To calculate the probability that cooperative-recovery is not successful, I distinguish the two cases that ΔU is either of size ΔU_{min} or of size ΔU_{max}. The uncertainty window is of size ΔU_{max} if at least one unrecognized failure occurs with one of PA_{other} that does not recover until t_p. In the following, I enumerate the probabilities for all combinations of events that lead to a situation where j nodes of PA_{other} encounter situation (i) and cannot be reached, while k of PA_{other} experience situation (ii) and are also unreachable for PA. In the formula presented, I use three nested sums to enumerate the combined events. The outer sum selects subsets X of PA_{other} that do not encounter a node failure. The second sum selects subsets Y with j nodes from X that have suffered from an unrecognized communication failure that does not recover until t_p. These nodes have experienced situation (i). Hence, for unsuccessful cooperative-recovery, PA should not reach any of the j nodes. The innermost sum considers participants that encounter situation (ii). Here, subsets Z with k nodes from Y are selected that have received the global decision but are not reachable by the participant due to a communication failure. The resulting formula is given by $CR1'(t_p)$:

$$\begin{aligned} CR1'(t_p) \quad = \Bigg[& \sum_{i=0}^{n-1} \binom{n-1}{i} P_{o<f_n}(t_p)^i (1 - F_n(t_p))^{n-1-i} \\ & * \sum_{j=0}^{n-1-i} \binom{n-1-i}{j} P_{o<f_c,nr}(t_p)^j (1 - P_{o<f_c,nr}(t_p))^{n-1-i-j} P_{c,nr}(t_{cr})^j \\ & * \sum_{k=0}^{n-1-i-j} \binom{n-1-i-j}{k} (1 - F(t_p..t_p + \Delta U_{max}))^k P_{c,nr}(t_{cr})^k \\ & * F(t_p..t_p + \Delta U_{max})^{n-1-i-j-k} \Bigg] \end{aligned} \tag{33}$$

If recovery of communication paths is not assumed, i.e. $F(t)$ is used to describe the probability that PA can reach a node in PA_{other}, $CR1'$ is reduced to $CR1$:

$$\begin{aligned} CR1(t_p) \quad = \Bigg[& \sum_{i=0}^{n-1} \binom{n-1}{i} P_{o<f_n}(t_p)^i (1 - F_n(t_p))^{n-1-i} \\ & * \sum_{j=0}^{n-1-i} \binom{n-1-i}{j} P_{o<f_c}(t_p)^j (1 - P_{o<f_c}(t_p))^{n-1-i-j} F_C(t_{cr})^j \\ & * \sum_{k=0}^{n-1-i-j} \binom{n-1-i-j}{k} (1 - F(t_p..t_p + \Delta U_{max}))^k F_C(t_{cr})^k \\ & * F(t_p..t_p + \Delta U_{max})^{n-1-i-j-k} \Bigg] \end{aligned} \tag{34}$$

If P_{path} is used to describe the probability that PA can reach a node in PA_{other}, then $CR1(t_p)$ results in $CR1^{pp}(t_p)$ by replacing all occurrences of $F_C(t_{cr})$ with P_{path}.

If $i = j = 0$, Formulae $CR1'(t_p)$,$CR1^{pp}(t_p)$, and $CR1(t_p)$ consider a case where the uncertainty window is of size ΔU_{min}. The probability of this event

has to be subtracted from $CR1(t_p)$, $CR1^{pp}(t_p)$ and $CR1'(t_p)$ respectively, and is given by $CR2(t_p)$.

$$\begin{aligned} CR2(t_p) &= CR1(t_p) - \Big[(1-F_n(t_p))^{n-1}(1-P_{o<f_c}(t_p))^{n-1} \\ &\quad * \sum_{i=0}^{n-1} \Big[\binom{n-1}{i} (1-F(t_p..t_p+\Delta U_{max}))^i F_C(t_p)^i \\ &\quad * F(t_p..t_p+\Delta U_{max})^{n-1-i} \Big] \Big] \end{aligned} \tag{35}$$

$CR2^{pp}(t_p)$ and $CR2'(t_p)$ are derived by replacing all occurrences of $F_C(t_p)$ with P_{path} and $P_{c,nr}(t_p)$ respectively.

If all nodes vote, ΔU_{min} is entered. PA then blocks if suffering from a failure during $[t_p, t_p+\Delta U_{min}]$, described by probability UF_{min} as calculated in Section 7.1. Cooperative-recovery is not successful if all nodes of PA_{other} that received the global decision are not reachable for PA. This probability is denoted by $CR3(t_p)$, $CR3'(t_p)$, and $CR3^{pp}(t_p)$ respectively.

$$\begin{aligned} CR3(t_p) &= (1-F(t_p))^{n-1} * \sum_{i=0}^{n-1} \binom{n-1}{i} (1-F(t_p..t_p+\Delta U_{min}))^i \\ &\quad * F_C(t_{cr})^i F(t_p..t_p+\Delta U_{min})^{n-1-i} \end{aligned} \tag{36}$$

$$\begin{aligned} CR3'(t_p) &= (1-P_{o<f,nr}(t_p))^{n-1} * \sum_{i=0}^{n-1} \binom{n-1}{i} (1-F(t_p..t_p+\Delta U_{min}))^i \\ &\quad * P_{c,nr}(t_{cr})^i F(t_p..t_p+\Delta U_{min})^{n-1-i} \end{aligned} \tag{37}$$

$$\begin{aligned} CR3^{pp}(t_p) &= (1-F(t_p))^{n-1} * \sum_{i=0}^{n-1} \binom{n-1}{i} (1-F(t_p..t_p+\Delta U_{min}))^i \\ &\quad * P_{path}^i F(t_p..t_p+\Delta U_{min})^{n-1-i} \end{aligned} \tag{38}$$

The probability that PA suffers a blocking situation that cannot be recovered immediately is now given by

$$\begin{aligned} P_{u,cr}(t_p) &= P_{AU}(t_p) * UF_{max}(t_p) * CR2(t_p) \\ &\quad + P_{AU}(t_p) * UF_{min}(t_p) * CR3(t_p) \end{aligned} \tag{39}$$

if no recovery of communication paths is considered and by

$$\begin{aligned} P'_{u,cr}(t_p) &= P_{AU}(t_p) * UF_{max}(t_p) * CR2'(t_p) \\ &\quad + P_{AU}(t_p) * UF_{min}(t_p) * CR3'(t_p) \end{aligned} \tag{40}$$

$$\begin{aligned} P^{pp}_{u,cr}(t_p) &= P_{AU}(t_p) * UF_{max}(t_p) * CR2^{pp}(t_p) \\ &\quad + P_{AU}(t_p) * UF_{min}(t_p) * CR3^{pp}(t_p) \end{aligned} \tag{41}$$

if a single recovery cycle is considered or in case P_{path} is used to describe the probability of a communication link between PA and a node from PA_{other}. $P_{u,cr}(t_p)$, $P^{pp}_{u,cr}(t_p)$, and $P'_{u,cr}(t_p)$ calculate the probability that the first request round of cooperative-recovery is not successful. Note that consecutive recovery rounds are possibly successful. $P_{u,cr}(t_p)$, $P^{pp}_{u,cr}(t_p)$, and $P'_{u,cr}(t_p)$ are the relevant probabilities here, because only a participant that experiences blocking and cannot recover immediately must retry recovery for an indefinite period. This is exactly the situation described by blocking and extended uncertainty.

7.1.3 Predictions for the Example Scenario

Figure 9 plots the probabilities of extended uncertainty derived by $P^{'}_{u,cr}(t_p)$, $P^{pp}_{u,cr}(t_p)$ and $P_{u,cr}(t_p)$ for the example MANET scenario, with and without multi-hop routing, for $n_p = 3$ as well as for $n_p = 2$.

The important observation is that the risk of PA suffering indefinite blocking is significantly reduced by cooperative-recovery. For example, without cooperative-recovery and with AODV routing, the probability of blocking is 2 % at a processing time of 15 s with three participants (see Section 7.1.1). If cooperative-recovery is used, this probability decreases to 0.22 %. Only for processing phases greater than 30 s does this probability reach a considerable value of 0.7 %. However, for the example MANET scenario I assumed that only transactions with transaction sizes smaller than 15 s are feasible.

Simulation results obtained from an ns2 simulation study show that the proposed calculation model predicts the real-world blocking rates accurately, as shown in Figure 9(a) and 9(b). The measurements were derived from a simulation similar to the simulation presented in Section 7.1.1, with the difference that a cooperative-recovery scheme is initiated by blocked participants at time t_{cr}. The presented blocking rates are obtained by counting all recovery attempts that have been successful within the first message round of cooperative-recovery. If no routing is used, $P_{u,cr}(t_P)$ and $P^{'}_{u,cr}(t_p)$ provide a good approximation of measured results for all processing times (see Figure 9(b)). In case multi-hop routing is used, $P_{u,cr}(t_P)$ and $P^{'}_{u,cr}(t_p)$ show upper and lower bounds for the real blocking rate (see Figure 9(a)). This is explained by the fact that $P_{u,cr}(t_P)$ does not consider any recovery of failed communication paths, while $P^{'}_{u,cr}(t_p)$ considers exactly one recovery cycle and assumes no subsequent path failures. In contrast, $P^{pp}_{u,cr}(t_p)$ considers the constant P_{path} of $\mathcal{A}$ and therefore meets the real values exactly for large t_p (here, for t_p<50 s). For the transaction sizes where transaction processing is feasible in the example scenario, predictions of $P_{u,cr}(t_P)$ and $P^{'}_{u,cr}(t_p)$ are close together and therefore also meet the simulation results accurately.

Additionally, it is shown in Figure 9(c) and 9(d) that the number of participants has a strong influence on the blocking probability. Figure 9(c) shows that the probability of blocking increases significantly for n_p=2 compared to n_p=3. For n_p=2 the probability of blocking is 0.48 % compared to 0.22 % with n_p=3 if no multi-hop routing is used (see Figure 9(c)). In fact, this result is not surprising, as more participants increase the probability that an unblocked participant

can be reached for cooperative-recovery. The smaller blocking probability with n_p=3 is also explained by the higher abort rate in the processing phase, compared to the case with n_p=2, which leads to fewer transactions entering the commit phase.

The third result I want to present is the influence of multi-hop routing on the blocking probability. Figure 8(b) shows the blocking risk for the example scenario if no multi-hop routing is assumed. The blocking risk here is higher compared to the situation were AODV is used, because communication paths with participants required for recovery are repaired with small probability.

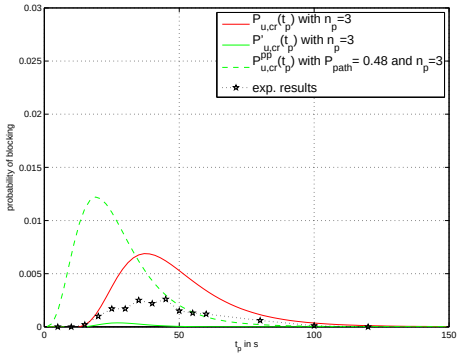
7.2 Probability of Extended Uncertainty in the Semantic Model

Compared to the strict transaction model analyzed above, the uncertainty window in the semantic model is larger, because participants enter uncertainty right after processing their last operation. A participant enters its uncertainty window at time t_o and leaves uncertainty at $t_u + \delta_m$. Recall that $t_u = t'_p + \Delta_{ex} + \delta_m$. In the semantic model, the processing phase of a transaction is given by the interval $[t_s, t'_p]$, while the decision phase is defined by $[t'_p, t_u + \delta_m]$.

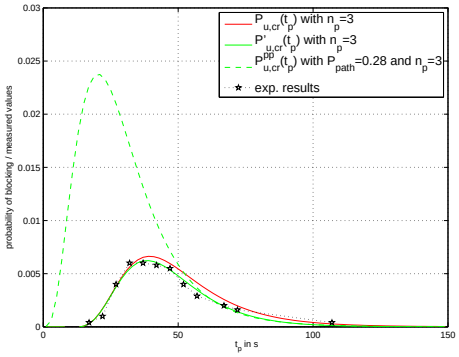
Analogous to blocking caused by a participant failure in the strict model, an extended uncertainty situation in the semantic model is defined as any situation where the global decision is made by the coordinator but cannot be transferred to PA because of communication failure or because PA has disconnected from $\mathcal{A}$. In contrast to the strict model, where blocking can only occur in the decision phase, extended uncertainty can also occur in the processing phase.

In the interval $[t_s, t'_p]$, the coordinator decides on abort if detecting a failure. PA experiences an extended uncertainty situation if a recognized failure with one of PA_{other} or with PA_{last} occurs, while PA has already entered its uncertainty window and cannot receive the global abort decision from the coordinator. The abort decision is not received if PA (i) experiences a communication failure with the coordinator that does not recover in time, or (ii) if PA disconnects from $\mathcal{A}$.

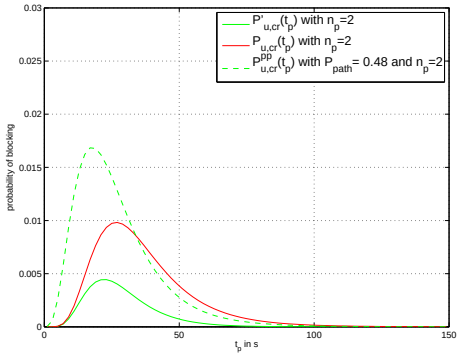
Note that, while in the strict model all participants enter uncertainty at time t_p, the time a participant enters uncertainty in the semantic model varies for every participant and is given by t_o. Hence, for every point in time t_f in $[t_s, t'_p]$, the situation must be considered that a failure occurs causing a global abort of the transaction (I call this situation A) and that PA is in its uncertainty window and cannot receive the global abort decision (I call this event B). The probability of event A is given by $P_{o<f_c,nr}(t_f)$ and $P_{o<f_c}(t_f)$ respectively, while the probability of situation B is computed by $[1-(1-f(t_f)*O(t_f..t'_p))^{n_p-2}*(1-f(t_f))]$. The probability that situation A and B happens in the interval $[t_s, t'_p]$ is the probability of PA to experience extended uncertainty in this interval. I call this probability $Pu_1(t'_p)$. $Pu_1(t'_p)$ is given by:



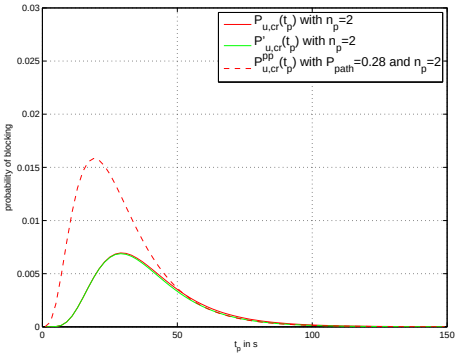
(a) Probability of indefinite blocking caused by a participant failure if cooperative-recovery is used. Predictions for the exemple MANET scenario with AODV multi-hop routing.



(b) Probability of indefinite blocking caused by a participant failure if cooperative-recovery is used. Predictions for the exemple MANET scenario without multi-hop routing.



(c) Probability of blocking if cooperative recovery and AODV is used, with n_p=2.



(d) Probability of blocking with cooperative-recovery, without routing, and n_p=2.

Figure 9: Risk of PA suffering a blocking situation and unsuccessful cooperative-recovery as calculated by $P'_{u,cr}(t_p)$, $P^{pp}_{u,cr}(t_p)$ and $P_{u,cr}(t_p)$. The presented probabilities are based on the example MANET scenario with transaction parameters n_p=3, n_p=2, Δ_{vo}=1 s, and δ_m=180 ms.

$$Pu_1(t'_p) = \int_0^{t'_p} \left[[1 - (1 - f(t_f) * O(t_f..t'_p))^{n-2} * (1 - f(t_f))] * P_{o<f_c}(t_f) \right] dt_f \quad (42)$$

and by

$$P'u_1(t'_p) = \int_0^{t'_p} \left[[1 - (1 - f(t_f) * O(t_f..t'_p))^{n-2} * (1 - f(t_f))] * P_{o<f_c,nr}(t_f) \right] dt_f \quad (43)$$

An extended uncertainty situation of PA is caused in the interval $[t'_p, t_u + \delta_m]$ if the global decision is made by the coordinator at t_u, but PA cannot receive this decision because of a communication failure or disconnection from $\mathcal{A}$. The probability for PA to experience an extended uncertainty situation if the transaction enters the decision phase is given by $Pu_2(t'_p)$:

$$P'u_2(t'_p) = P_{o<f_c,nr}(t_u) * (1 - P_{o>f}(t'_p)^{n-2} * [1 - F(t'_p)])$$

and by

$$Pu_2(t'_p) = P_{o<f}(t_u) * (1 - P_{o>f}(t'_p)^{n-2} * [1 - F(t'_p)]) \quad (44)$$

The probability that PA experiences an extended uncertainty situation in the processing or the decision phase is then given by $P^*_u(t'_p)$:

$$P'^*_u(t'_p) = P'u_1(t'_p) + P'u_2(t'_p) \quad (45)$$

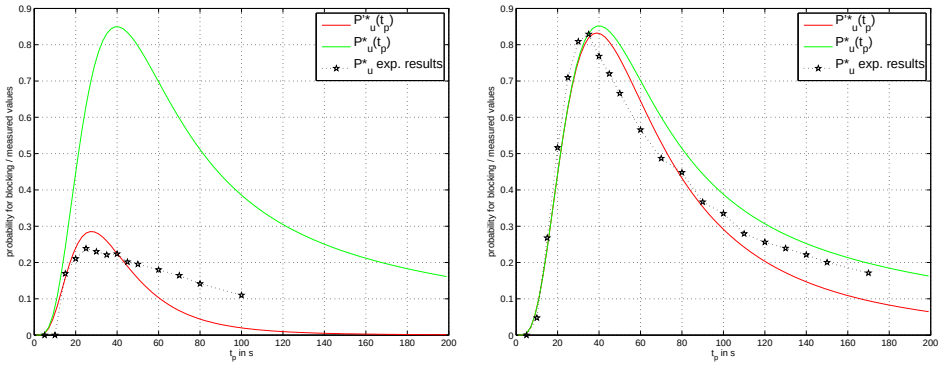
and if no recovery of communication paths is considered, the probability is

$$P^*_u(t'_p) = Pu_1(t'_p) + Pu_2(t'_p) \quad (46)$$

7.2.1 Predictions and Simulation Results

Figure 10 depicts the probability of extended uncertainty caused by participant failures calculated by $P'^*_u(t'_p)$ and $P^*_u(t'_p)$. Predicted uncertainty rates are compared to measurements obtained from an ns2 simulation study. The ns2 simulation study simulates the message flow of transactions in the semantic transaction model. The number of participants that are uncertain about the global decision at time $t_u + \delta_m + \delta_{to}$ is measured by counting all the participants that have entered uncertainty and have not subsequently received the global decision.

The hypothesis that the semantic model shows a higher susceptibility to extended uncertainty situations than the strict model does to blocking is clearly confirmed by analytical predictions as well as by simulation results. Figure 10(a) shows that the probability of extended uncertainty in the example MANET scenario with AODV multi-hop routing is considerably higher, with 16.94 % for 15 s processing time compared to 2 % in the strict model. If no multi-hop routing



(a) Probability for extended uncertainty in the example MANET scenario with AODV multi-hop routing.

(b) Probability for extended uncertainty caused by a participant failure.

Figure 10: Abort and extended uncertainty rates for transactions providing semantic atomicity, with $n_p = 3$ and $\delta_m = 180ms$.

is used, the probability for extended uncertainty is even higher, with 26.77 % at t_p=15 s. Hence, the effect of short path outages in case multi-hop routing is used reduces the probability of uncertainty drastically and again shows the importance of considering recovery of paths if multi-hop routing is used.

Omitting path recovery leads to predictions that are unrealistically high (see Figure 10(a)). The fault made by $P_u^{'*}(t'_p)$ in considering only one failure-and-recovery cycle is reflected by the effect that, for large t_p, $P_u^{'*}(t'_p)$ underestimates the probability of extended uncertainty. However, I argue that this has only a small impact, because at large t_p where the simplified assumption of the calculation model becomes relevant, transaction processing is not feasible due to high abort rates, as discussed in Section 6.2. Recall that only transactions with a processing phase smaller than 20 s are considered feasible in the example MANET scenario and semantic transaction model.

7.2.2 Probability for Extended Uncertainty with Cooperative-Recovery

In the semantic model, cooperative-recovery is started at $t^*_{cr} = t_u + \delta_m$, as this is the latest point in time a participant can expect the global decision. The probability that PA suffers from an extended uncertainty situation that cannot be compensated for immediately by cooperative-recovery at t^*_{cr} depends on the probability that PA suffers from extended uncertainty and that neither a node of PA_{other} nor PA_{last} is certain and reachable for PA at t^*_{cr}.

I first consider recovery with PA_{last} separately. A node failure of PA_{last} within $[t_s, t'_p]$ always causes an abort of the global transaction and also induces PA_{last} unavailability for cooperative-recovery at t^*_{cr}. A communication failure of PA_{last} with the coordinator within the processing phase $[t_s, t'_p]$ also causes

an abort of the global transaction, but PA_{last} is always a potential cooperation partner for PA, as PA_{last} is always certain within $[t_s, t'_p]$. Now, I assume that such a communication failure occurs at time t_f during $[t_s, t'_p]$. PA_{last} is only available for cooperative-recovery with PA if it does not suffer node failure within $[t_f, t^*_{cr}]$.

The probability that at time t_f the transaction is aborted by a communication failure between the coordinator and PA_{last} (given by $f_C(t_f)$), while PA_{last} *does not* suffer from a node failure until recovery of PA is started (given by $[1 - F_n(t_f..t^*_{cr})]$) *and* PA is uncertain at time t_f (given by $O(0..t_f)$) *and* PA cannot reach PA_{last} at t^*_{cr} is given by $CR1^{*'}(t_f)$:

$$\begin{aligned} CR1^{*'}(t_f) &= O(0..t_f) * [f_n(t_f) + f_c(t_f) * [1 - F_n(t_f..t^*_{cr})] \\ &\quad * P_{c,nr}(t^*_{cr})] * P_{o<f_c,nr}(t_f) \end{aligned} \tag{47}$$

If recovery of communication paths is not considered and the reachability of a recovery partner is calculated by $F_C(t_{cr})$, then $CR1^*$ is derived.

$$\begin{aligned} CR1^{*}(t_f) &= O(0..t_f) * [f_n(t_f) + f_c(t_f) * [1 - F_n(t_f..t^*_{cr})] \\ &\quad * F_C(t^*_{cr})] * P_{o<f_c}(t_f) \end{aligned} \tag{48}$$

If the reachability of a participant of PA_{other} is described by P_{path}, $CR1^*$ results in $CR1^{*pp}(t_f)$:

$$\begin{aligned} CR1^{*pp}(t_f) &= O(0..t_f) * [f_n(t_f) + f_c(t_f) * [1 - F_n(t_f..t^*_{cr})] \\ &\quad * P_{path}] * P_{o<f_c}(t_f) \end{aligned} \tag{49}$$

Line (2) of $CR1^{*'}(t_f)$, $CR1^*(t_f)$, and $CR^{*pp}(t_f)$ considers the probability that PA encounters a communication failure with PA_{last} that prevents communication at time t^*_{cr} (given by $P_{c,nr}(t^*_{cr})$ $F_C(t^*_{cr})$ and P_{path} respectively). The factors $P_{o<f_c}(t_f)$ (in Formulae (48) and (49)) and $P_{o<f_c,nr}(t_f)$ (in Formula (47)) respectively, describe the probability that the coordinator cannot reach PA at time t_f, when the global transaction is aborted. PA_{last} is also not available for cooperative-recovery if it encounters node failure at t_f.

To estimate the probability that a node of PA_{other} is a potential partner for successful cooperative-recovery, for every point in time within the interval $[t_s, t'_p]$, the state of each participant must be considered. A participant can remove uncertainty from PA if it encounters one of the two following situations: (i) if it has not suffered from a failure and has not received its last operation; or (ii) if it encounters a communication failure with the coordinator that leads to abort of the transaction, i.e. the communication failure occurs before t_o.

Nodes that experience node failure or are uncertain cannot remove uncertainty from PA. Figure 11 shows the decision tree with paths leading to situations (i) and (ii). The idea of the following calculation is to sum up the probabilities for all nodes of a subset of PA_{other} to traverse this tree in the way

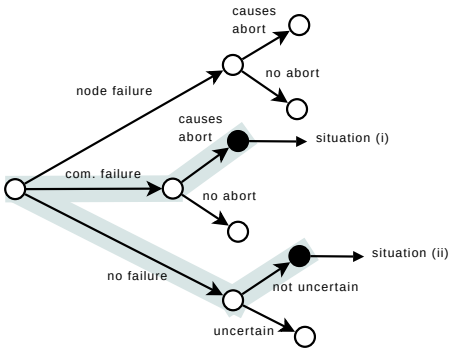


Figure 11: Decision Tree of nodes in PA_{other}

that it encounters situation (i) *or* situation (ii) *and* is *not* reachable for PA for cooperative-recovery at t^*_{cr} due to a communication failure.

To select the relevant probabilities for nodes in PA_{other} to encounter situation (i) or (ii), I use nested sums in $CR2^{*'}(t'_p)$ according to the following considerations: for a point in time t_f, a set A from PA_{other} with i nodes is selected, which encounters a node failure that causes abort, while the other $n-2-i$ nodes are divided in the set B of j nodes, which experience a node failure that does not lead to the abort of the global transaction, and a set C of $n-2-i-j$ nodes, which does not experience a node failure. Set C is further decomposed into sets D with k nodes, which encounter a communication failure with the coordinator that causes abort and E with $n-2-i-j-k$ nodes, which either encounter a communication failure that does not lead to transaction abort or do not suffer from a communication failure with the coordinator at t_f. All nodes in D experience situation (i). Set E is further decomposed into sets F and G, where F contains l nodes, which experience a communication failure that does not cause abort, while G contains $n-2-i-j-k-l$ nodes, which do not experience a communication failure at t_f. Set G now contains m nodes, which have not experienced any failure and are not uncertain as they have not received their last operation at t_f, while $n-2-i-j-k-l-m$ nodes of G are uncertain. Hence, the m nodes of G have experienced situation (ii). Cooperative-recovery is not successful if PA cannot reach nodes in D and m nodes of G. From these considerations, I derive Formula $CR2^{*'}(t'_p)$:

$$
\begin{aligned}
CR2^{*'}(t'_p) \quad = \quad & \int_0^{t'_p} pcr_1(t_f) * \sum_{i=0}^{n-2} \binom{n-2}{i} P'_{o>f_n}(t_f)^i \\
& * \sum_{j=0}^{n-2-i} \binom{n-2-i}{j} P'_{o<f_n}(t_f)^j * [1-f_n(t_f)]^a \\
& * \sum_{k=0}^{a} \binom{a}{k} (P'_{o>f_c}(t'_f) * [1-F_n(t_f..t_u)] * P_{c,nr}(t^*_{cr}))^k \\
& * \sum_{l=0}^{b} \binom{b}{l} P'_{o<f_c}(t_f)^l * [1-f_c(t_f)]^c \\
& * \sum_{m=0}^{c} \binom{c}{m} (O(t_f..t'_p) * [1-F_n(t_f..t^*_{cr})] * P_{c,nr}(t^*_{cr}))^m \\
& * O(0..t_f)^d \, dt_f
\end{aligned} \tag{50}
$$

with

$$
\begin{aligned}
a &= n-2-i-j \\
b &= n-2-i-j-k \\
c &= n-2-i-j-k-l \\
d &= n-2-i-j-k-l-m
\end{aligned}
$$

The variant of $CR2^{*'}(t'_p)$ that does not consider recovery of communication paths is called $CR2^{*}(t'_p)$ and is derived by substituting all occurrences of $P_{f_c,nr}(t^*_{cr})$ with $F_C(t^*_{cr})$. If reachability of a recovery partner is approximated by P_{path}, $CR2^{*pp}(t'_p)$ is derived by substituting all occurrences of $P_{f_c,nr}(t^*_{cr})$ with P_{path} in $CR2^{*'}(t^*_{cr})$.

Formula $CR2^{*'}(t'_p)$ includes a path where the transaction is not aborted in $[t_s, t'_p]$. This happens if $i = k = 0$. I denote this case as $CR3^{*'}(t'_p)$:

$$
\begin{aligned}
CR3^{*'}(t'_p) \quad = \quad & \int_0^{t'_p} pcr_1(t_f) * \sum_{i=0}^{n-2} \binom{n-2}{i} P'_{o<f_n}(t_f)^i * *(1-f_n(t_f))^{n-2-i} \\
& * \sum_{j=0}^{n-2-i} \binom{n-2-i}{j} P'_{o<f_c}(t_f)^j * (1-f_c(t_f))^a \\
& * \sum_{k=0}^{a} \binom{a}{k} (O(t_f..t'_p) * (1-F_n(t_f..t_u)) * P_{f_c,nr}(t^*_{cr}))^k \\
& * O(0..t_f)^d \, dt_f
\end{aligned}
$$

Again, variant $CR3^{*}(t'_p)$ is derived by substituting of $P_{c,nr}(t^*_{cr})$ with $F_C(t^*_{cr})$. $CR3^{*pp}$ is derived similarly by using P_{path} instead of $P_{c,nr}(t^*_{cr})$.

Now, the probability that PA suffers an extended uncertainty situation that cannot be recovered immediately at time t^*_{cr} is given by:

$$
P^{*'}_{u,cr}(t'_p) = CR2^{*'}(t'_p) - CR3^{*'}(t'_p) \tag{51}
$$

$$
P^{*}_{u,cr}(t'_p) = CR2^{*}(t'_p) - CR3^{*}(t'_p) \tag{52}
$$

and

$$P^{*pp}_{u,cr}(t'_p) = CR2^{*pp}(t'_p) - CR3^{*pp}(t'_p) \tag{53}$$

respectively.

7.2.3 Predictions for the Example Scenario

Figure 12 depicts the probabilities calculated by $P^{*}_{u,cr}(t'_p)$, $P^{*'}_{u,cr}(t'_p)$ and $P^{*pp}_{u,cr}(t'_p)$ for the example MANET scenario and semantic transactions with 2–3 participants.

The major result is that, similar to the strict case, cooperative-recovery significantly compensates for extended uncertainty. For example, Figure 12(a) shows a reduction in the probability of extended uncertainty from 21 % to 1.7 % at 20 s processing time and multi-hop routing with n_p=3 (compare Figures 12(a) and 10(a)). If no multi-hop routing is used, the probability for blocking is reduced from 51 % to 1.19 % at 20 s processing time and n_p=3.

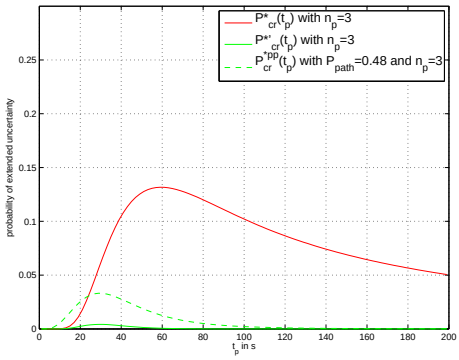
If multi-hop routing is used, recovery of communication paths has a major influence as shown in Figure 12(a). In this case, $P^{*'}_{u,cr}(t'_p)$ underestimates the real uncertainty rate, because only one failure-and-recovery cycle of communication paths is considered by $P^{*'}_{u,cr}(t'_p)$. Hence, once recovered a link is assumed to remain operational. In reality, this is obviously not true, and therefore the observed rate of uncertainty probability is upper bounded by $P^{*}_{u,cr}(t'_p)$ and lower bounded by $P^{*'}_{u,cr}(t'_p)$. $P^{*pp}_{u,cr}(t'_p)$ predicts uncertainty rates considering the constant path probability for communication paths and derives values lying between $P^{*}_{u,cr}(t'_p)$ and $P^{*'}_{u,cr}(t'_p)$ for large t_p. However, for short processing times that are of interest here, the values of all three predictions are close together and provide a good approximation of expected uncertainty rates in the example scenario.

Similar to the strict case, the number of participants is a decisive factor. This can be observed in Figure 12(d) showing the uncertainty rates for the example scenario and transactions with n_p=2. Here, the rate of extended uncertainty situations increases from 1.9 % to 5 %, with n_p=3.

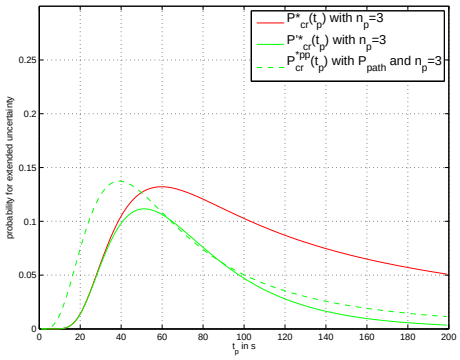
7.3 Summary and Discussion

In this section, I have presented a calculation model to predict the blocking and uncertainty risks induced by participant failures with and without cooperative-recovery. Results for the example MANET scenario show that the risk of PA suffering blocking is very low if multi-hop routing and cooperative-recovery is used. In fact, without cooperative-recovery the risk is below 4 % for reasonable processing times, while cooperative-recovery reduces this risk to less than 1 %. In the semantic model, the probability of extended uncertainty is considerably higher. Generally, the probability for blocking if cooperative-recovery is used is strongly influenced by the number of participants involved in a transaction.

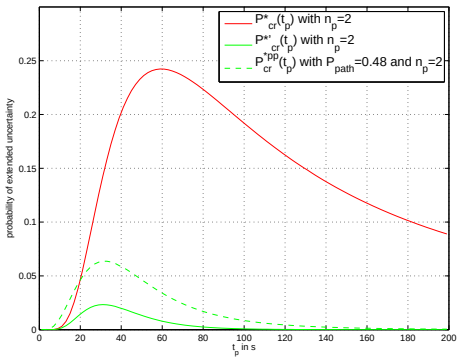
Transactions with just two participants show the highest risk of blocking in the strict model, here cooperative-recovery is less effective and the probability



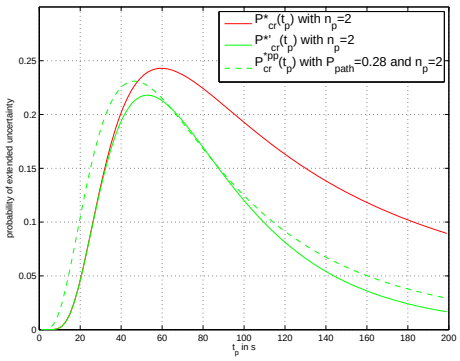
(a) Probability for an extended uncertainty situation in case cooperative-recovery and AODV multi-hop routing is used.



(b) Probability for an extended uncertainty situation in case cooperative-recovery is used without multi-hop routing.



(c) Probability of extended uncertainty with $n_p = 2$ and AODV routing.



(d) Probability of extended uncertainty with $n_p = 2$ and without multi-hop routing.

Figure 12: Probability for extended uncertainty in the example MANET scenario and the semantic transaction model.

that such a transaction reaches the decision phase in the strict model is considerably higher than with more participants. In the semantic model, a transaction with 2 participants shows the highest risk for extended uncertainty measured so far. Hence, semantic transactions with few participants will most likely require recovery schemes in addition to cooperative-recovery to compensate for extended uncertainty situations.

8 Blocking Caused by Coordinator Failures

While the previous section was concerned with blocking situations caused by participant failures, this section examines the probability of blocking caused by a node failure of the transaction coordinator. In the literature, this situation is assumed to be the more severe case, because the failure of the central coordination entity may cause blocking of multiple participants, while failures of a participant results in blocking at the participant only.

I demonstrate in the following that the probability of blocking caused by a node failure of the coordinator is low for the example MANET scenario even if recovery of communication paths is omitted in calculations. The calculations presented in the following therefore provide an upper bound for blocking and uncertainty probabilities. As in Section 7, I first consider the strict transaction model and then the semantic model.

8.1 Strict Transaction Model

In the strict model, the most decisive factors in the computation of the blocking risk of a participant are the probabilities for entering the decision phase and that ΔU is extended to ΔU_{max}. The probabilities for PA to enter an uncertainty window of ΔU_{min} or of ΔU_{max} are given by the probability that the coordinator awaits time-out Δ_{vo} or not. I already calculated these probabilities as $P_{U_{min}}(t_p)$ and $P_{U_{max}}(t_p)$ in Section 7.1 in Formulae (27) and (29). In fact, a time-out Δ_{vo} may also happen if no failure of a node in PA_{other} happens until t_p, but in the interval $[t_p, t_p + 2\delta_m]$. Here, a time-out is caused by a participant if a general failure happens within $[t_p, t_p + \delta_m]$ or a communication failure occurs within $[t_p + \delta_m, t_p + 2\delta_m]$. I do not consider these cases here, because the probability of such an event is negligible, as the intervals are of size δ_m only. Generally, I neglect events that occur in intervals smaller than $2\delta_m$ in the following.

I denote the probability of a node failure of the coordinator within interval $[t_p, t_p + \Delta U_{min}]$ by $CF_{U_{min}}(t_p)$, which is given by $F_n(t_p..t_p + \Delta U_{min})$. $CF_{U_{max}}(t_p)$ is defined analogously. The probability that PA does not encounter any failure until $t_p + \Delta U_{min}$, given by $1 - F(t_p + \Delta U_{min})$, is denoted by $PA_{U_{min}}(t_p)$. Analogously I define $PA_{U_{max}}(t_p)$.

The probability that PA enters an uncertainty window of size ΔU_{max} or of ΔU_{min} and, while uncertain about the global decision, the coordinator suffers

from a node failure and thus PA is blocked is now given by $P_u(t_p)$:

$$\begin{aligned} P_u(t_p) &= PA_{Umax}(t_p) * CF_{Umax}(t_p) * P_{Umax}(t_p) \\ &\quad + PA_{Umin}(t_p) * CF_{Umin}(t_p) * P_{Umin}(t_p) \end{aligned} \tag{54}$$

8.1.1 Blocking Probability with Cooperative-Recovery

If PA suffers from blocking, a cooperative-recovery scheme is initiated. The success of this scheme is given by the probability that PA can reach at least one node in PA_{other} that is not blocked. Recall that here I only consider the case that a coordinator failure during ΔU leads to blocking. Now, if PA is blocked, all nodes of PA_{other} that also received the prepare message are blocked too. Only nodes that encountered an unrecognized communication failure remain unblocked and thus are potential cooperative partners for recovery. Such a partner is reachable for PA if it is still alive and no communication failure between them has happened within $[t_S, t_p + \Delta U + \delta_m]$. As above, I distinguish the two cases that ΔU is either of length ΔU_{min} or ΔU_{max}.

Again, I consider the case that at least one unrecognized failure leads to ΔU_{max}. In Formula (55) I investigate probabilities for all combinations of events that lead to at least one unrecognized failure and additionally let j nodes of PA_{other} remain unblocked. I use two nested sums that enumerate combined events. The outer sum selects subsets X of nodes of PA_{other} that do not encounter a node failure. The inner sum then selects from X the subsets Y of nodes that have suffered from an unrecognized communication failure by t_p. All j nodes in subsets Y are unblocked and potential recovery partners for PA. If all j nodes are unreachable, because of a communication failure with PA, cooperative-recovery is unsuccessful.

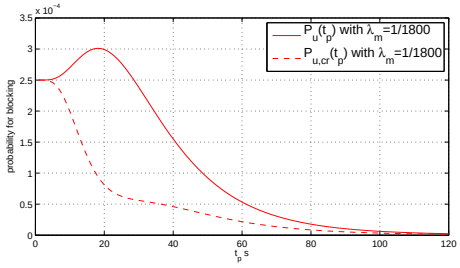
$$\begin{aligned} CR1(t_p) &= \sum_{i=0}^{n_p-1} \Bigg[\binom{n_p - 1}{i} * P_{o<f_n}(t_p)^i [1 - F_n(t_p)]^{n_p-1-i} \\ &\quad * \sum_{j=0}^{n_p-1-i} \binom{n_p - 1 - i}{j} P_{o<f_c}(t_p)^j [1 - F_c(t_p)]^{n_p-1-i-j} \\ &\quad * \left[F_c\left(t_p + \Delta U_{max}\right)\right]^j \Bigg] \end{aligned} \tag{55}$$

As $CR1(t_p)$ also includes the case that no node of PA_{other} encounters a failure ($i = j = 0$), which leads to ΔU_{min}, one needs to subtract the probability of this event leading to $CR2(t_p)$:

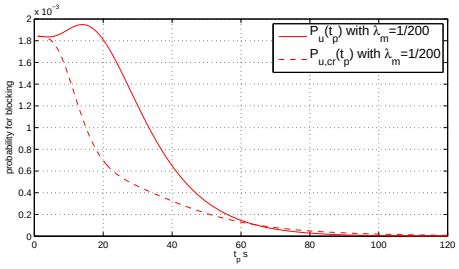
$$CR2(t_p) = CR1(t_p) - (1 - F_n(t_p))^{n_p-1} * (1 - F_c(t_p))^{n_p-1} \tag{56}$$

In the case that ΔU_{min} is entered, all nodes have voted and thus are uncertain. Then no partner for cooperative-recovery exists. The probability that ΔU_{min} is entered is given by $P_{U_{min}}(t_p)$, as calculated in Formula (27). The probability that PA is blocked due to a node failure of the coordinator during ΔU and cannot recover cooperatively is now given by

$$\begin{aligned} P_{u,cr}(t_p) &= PA_{Umax}(t_p) * C_{Umax}(t_p) * CR2(t_p) \\ &\quad + PA_{Umin}(t_p) * C_{Umin}(t_p) * P_{Umin}(t_p) \end{aligned} \tag{57}$$



(a) Probability of blocking caused by a node failure of the coordinator in the example MANET scenario with and without cooperative-recovery.



(b) Probability of blocking caused by a node failure of the coordinator with and without cooperative-recovery and increased node failure probability ($\lambda_m = 1/200$).

Figure 13: Probability of blocking caused by a node failure of the coordinator in the strict model for the example MANET scenario with n_p=3 and δ_m=180 ms.

8.1.2 Predictions for the Example MANET Scenario

Figure (13) presents blocking probabilities calculated by $P_{u,cr}(t_p)$ for the example MANET scenario. The important result here is that the probability of PA suffering blocking due to a coordinator's node failure in ΔU is very small. For the example MANET scenario, this probability is in the 10^{-4} domain and is further reduced by cooperative-recovery (see Figure 13(a) and 13(b)). Even if the probability of a node failure is drastically increased, e.g. by assuming that the expected sojourn time of mobile nodes in $\mathcal{A}$ is only 10min instead of 30min, the probability of blocking caused by a node failure of the coordinator does not leave the 10^{-3} domain, as shown in Figure 13(b).

The main reason for the small blocking risk induced by coordinator node failures is the small size of the vulnerability window in 2PC, where node failure causes blocking. For the example MANET scenario used here, and scenarios where the probabilities for node and communication failures show a similar relation, it can be derived that blocking caused by a coordinator node failure is a rare case that can be neglected for most MANET scenarios.

8.2 Semantic Model

In the semantic model, the end of the processing phase is given by t'_p, which is the time the last operation for PA_{last} is issued by the coordinator. At time $t_u = t'_p + \Delta_{ex} + \delta_m$ the coordinator derives the global decision. Δ_{ex} also serves as time-out, i.e. if the coordinator does not receive an acknowledgment until t_u it suspects PA_{last} to be failed and decides to abort.

In contrast to the strict model, all participants but PA_{last} enter uncertainty already during $[t_s, t'_p]$ with acknowledgment of their last operation at t_o. PA_{last} enters uncertainty at $t'_p + \delta_m + \Delta_{ex}$ and remains uncertain for $2\delta_m$. In the following, I will first consider the risk of extended uncertainty in the interval

$[t_s, t'_p]$ and afterwards in $[t'_p, t_u]$.

In the interval $[t_s, t'_p]$ a coordinator node failure causes an extended uncertainty situation of PA if the failure occurs after t_o and PA did not previously cause transaction abort. This probability is computed by $Pu_1(t_{f_{n,c}})$, where $t_{f_{n,c}}$ denotes the time of the coordinator node failure:

$$Pu_1(t_{f_{n,c}}) = \int_{o}^{t_{f_{n,c}}} o(t_o)[1 - F(t_o)]dt_o \tag{58}$$

The probability that PA is not uncertain and has not caused an abort until $t_{f_{n,c}}$ is given by $Pnu_1(t_{f_{n,c}})$, where $[1 - F(t_{f_{n,c}})]$ is the probability that PA_{last} does not cause an abort of the transaction until $t_{f_{n,c}}$:

$$Pnu_1(t_{f_{n,c}}) = \int_{t_{f_{n,c}}}^{t_p} o(t_o)dt_o[1 - F(t_{f_{n,c}})] \tag{59}$$

The calculation of the probability that PA is uncertain and the coordinator suffers node failure in $[t'_p, t_u]$ has to consider that the transaction has not previously aborted. This probability is given by $Pu_2(t'_p)$:

$$Pu_2(t'_p) = [1 - F(t_u)] * [1 - P_{o>f}(t'_p)]^{pn-1} * F_n(t'_p..t_u) \tag{60}$$

The probability that PA suffers from extended uncertainty in interval $[t'_p, t_u]$ is directly given by $Pu_2(t'_p)$. For extended uncertainty caused in $[t_s, t'_p]$, PA is required to be uncertain when the coordinator's node failure happens (line (1) of Formula (61)), while $n-2$ nodes in PA_{other} are uncertain or not, which is considered in line (2) of Formula (61) by enumerating all possible combinations of i uncertain and $n-2-i$ certain nodes in PA_{other}. The last participant PA_{last} is required not to cause abort of the transaction in $[t_s, t'_p]$. For the probability that PA suffers from extended uncertainty I now derive

$$\begin{aligned} P'_u(t'_p) \quad = \quad & \int_{0}^{t'_p} \Bigg[f_n(t_{f_{n,c}}) * Pu_1(t_{f_{n,c}}) \\ & * \sum_{i=0}^{n-2} \left[\binom{n-2}{i} Pu_1(t_{f_{n,c}})^i Pnu_1(t_{f_{n,c}})^{n-2-i} \right] \\ & * (1 - F(t_{f_{n,c}})) \Bigg] dt_{f_{n,c}} + Pu_2(t'_p) \end{aligned} \tag{61}$$

8.2.1 Blocking Probability with Cooperative-Recovery

If PA does not receive the global decision until $t_u + \delta_m$, it executes a cooperative-recovery scheme. I compute the probability for this scheme to be unsuccessful. The probability that PA cannot reach a participant that is certain depends on the probability that all certain participants have suffered from a node failure after $t_{f_{n,c}}$ or from a communication failure with PA until $t_u + 2\delta_m$. I denote this probability by C'.

$$C'(t_{f_{n,c}}) = F_n(t_{f_{n,c}}..t_u) + F_c(t_u + 2\delta_m) - F_n(t_{f_{n,c}}..t_u) * F_c(t_u + 2\delta_m) \tag{62}$$

In Formula (61) I already distinguished between certain and uncertain participants in PA_{other}. To derive the probability that PA suffers from extended uncertainty and cooperative-recovery is not successful, i.e. PA remains uncertain, I expand Formula (61) with the probability that no certain participant (PA_{last} and $n-2-i$ of PA_{other}) is reachable for PA. I then derive $P'_{u,cr}(t'_p)$.

$$\begin{aligned} P'_{u,cr}(t'_p) \quad = \quad & \int_0^{t'_p} \Bigg[f_n(t_{f_{n,c}}) * Pu_1(t_{f_{n,c}}) \\ & * \sum_{i=0}^{n-2} \left[\binom{n-2}{i} Pu_1(t_{f_{n,c}})^i \left[Pnu_1(t_{f_{n,c}}) * C'(t_{f_{n,c}}) \right]^{n-2-i} \right] \\ & * [1 - F(t_{f_{n,c}})] * C'(t_{f_{n,c}}) \Bigg] dt_{f_{n,c}} + Pu_2(t'_p) \end{aligned} \tag{63}$$

8.2.2 Predictions for the Example MANET Scenario

Figure (14) depicts results computed by $P'_{u,cr}(t'_p)$ for the example MANET scenario. Although the probability for uncertainty caused by a node failure of the coordinator is significantly higher than in the strict case, e.g. at maximum 0.7% at a processing time of 30s, it is still low compared to the uncertainty risk induced by participant failures. For values of t_p with moderate abort rates ($t_p < 20s$) the uncertainty risk is smaller than 0.6% in the example MANET scenario. The increased risk compared to the strict case is caused by the fact that a node failure of the coordinator in $[t_s, t'_p]$ can also cause extended uncertainty in the semantic model, while in the strict case, only node failures in the interval $[t_p, t_p + \Delta U]$ are relevant.

Cooperative-recovery compensates uncertainty situations especially well for small t_p, e.g. for values of t_p with moderate abort probability, the probability for extended uncertainty caused by a coordinator's node failure with cooperative-recovery remains smaller than 0.2%, as shown in Figure (14). Hence, the probability for uncertainty caused by the coordinator is negligible for feasible transaction processing for the example MANET scenario and semantic transactions.

9 Summary and Conclusion

In this report I have presented a probabilistic model to predict the abort and blocking probability of atomic transactions in MANETs. The model calculates the probability of blocking situations (i) caused by a node or communication failure of participants or (ii) caused by a node failure of the transaction coordinator. Probabilities are presented for the strict as well as the semantic transaction model and consider cooperative recovery to compensate for blocking situations. Such a model is useful to determine whether transaction processing in a MANET scenario is feasible, i.e. whether the abort rate is acceptable, and if additional mechanisms are required to compensate for blocking. The model can also be used to implement adaptive transaction processing, i.e. the transaction manager

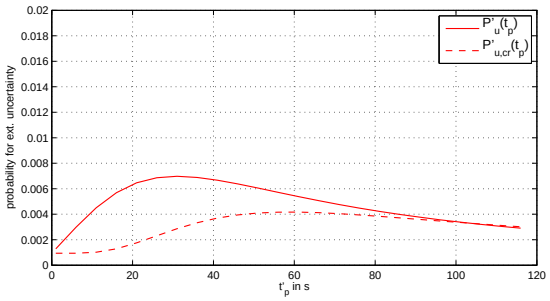


Figure 14: Probability for extended uncertainty caused by a coordinator's node failure for the example MANET scenario and transaction parameters $n_p = 3$, $\delta_m = 180ms$ and $\Delta_{ex} = \delta_{to} = 1s$.

automatically embeds a backup coordinator or chooses a more reliable commit protocol in case a high blocking risk is predicted for a transaction.

I applied the probabilistic model to an example MANET scenario. For strict transactions I showed that the blocking risk induced by node failures of the coordinator is negligible for the considered scenario, while the probability of blocking caused by participant failures is slightly higher but does not exceed 3 %. Cooperative-recovery drastically compensates for blocking, e.g. in the strict model, the blocking risk induced by participant failures is reduced from 3 % to $<$1 % and hence can also be considered to be negligible. Thus, the common hypothesis that higher failure probabilities in MANETs lead to an increased blocking probability is not true for strict transactions.

I argue that a calculation model as presented here is required to identify the MANET and transaction scenarios, where blocking probabilities reach considerably values. MANET and transaction scenarios with high blocking risks are an exception and not the rule. The low blocking risks are generally caused by high abort rates, i.e. transactions rather abort than block in most scenarios, and by effective cooperative-recovery.

Transactions of the semantic model are more susceptible to blocking situations than strict transactions, because uncertainty periods are larger. If no cooperative recovery is used, the risk of extended uncertainty caused by participant failures increases to 25% in the example scenario, while cooperative-recovery reduces this risk to a maximum of 4% in the example scenario. Semantic transactions with two participants are expected to show the highest susceptibility to a blocking situation, because cooperative recovery is less effective and the risk of transaction abort due to participant failures within the processing phase is reduced. Generally, an increased number of participants does not necessarily increase the probability that a blocking situation occurs, because (i) the abort probability is increased, which results in less transactions to enter the decision phase and thus less uncertainty situations can occur, and (ii) more participants are available for cooperative recovery.

References

[1] Gustavo Alonso, Divyakant Agrawal, Amr El Abbadi, Mohan Kamath, Roger Guenthoer, and C. Mohan. Advanced transaction models in workflow contexts. In *ICDE '96: Proceedings of the Twelfth International Conference on Data Engineering*, pages 574–581, Washington, DC, USA, 1996. IEEE Computer Society.

[2] F. Bai, N. Sadagopan, and A. Helmy. Important: a framework to systematically analyze the impact of mobility on performance of routing protocols for adhoc networks, 2003.

[3] Philip A. Bernstein, Vassco Hadzilacos, and Nathan Goodman. *Concurrency control and recovery in database systems.* Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1987.

[4] Sven Bittner, Wolf-Ulrich Raffel, and Manuel Scholz. The area graph-based mobility model and its impact on data dissemination. In *PERCOMW '05: Proceedings of the Third IEEE International Conference on Pervasive Computing and Communications Workshops*, pages 268–272, Washington, DC, USA, 2005. IEEE Computer Society.

[5] Tushar Deepak Chandra and Sam Toueg. Unreliable failure detectors for reliable distributed systems. *Journal of the ACM*, 43(2):225–267, 1996.

[6] Panos K. Chrysanthis and Krithi Ramamritham. Acta: the saga continues. pages 349–397, 1992.

[7] X/Open Company. Distributed transaction processing: The tx (transaction demarcation) specification. http://www.opengroup.org/bookstore/catalog/c504.htm, April 1995.

[8] Cynthia Dwork, Nancy Lynch, and Larry Stockmeyer. Consensus in the presence of partial synchrony. *J. ACM*, 35(2):288–323, 1988.

[9] Michael J. Fischer, Nancy A. Lynch, and Michael S. Paterson. Impossibility of distributed consensus with one faulty process. *J. ACM*, 32(2):374–382, 1985.

[10] R. Guerraoui, M. Larrea, and A. Schiper. Non blocking atomic commitment with an unreliable failure detector. In *SRDS '95: Proceedings of the 14TH Symposium on Reliable Distributed Systems*, page 41, Washington, DC, USA, 1995. IEEE Computer Society.

[11] Yijie Han, Richard J. La, Armand M. Makowski, and Seungjoon Lee. Distribution of path durations in mobile ad-hoc networks: Palm's theorem to the rescue. *Comput. Networks*, 50(12):1887–1900, 2006.

[12] Henry F. Korth, Eliezer Levy, and Abraham Silberschatz. A formal approach to recovery by compensating transactions. In Dennis McLeod, Ron Sacks-Davis, and Hans-Jörg Schek, editors, *16th International Conference on Very Large Data Bases, August 13-16, 1990, Brisbane, Queensland, Australia, Proceedings*, pages 95–106. Morgan Kaufmann, 1990.

[13] Richard J. La and Yijie Han. Distribution of path durations in mobile ad hoc networks and path selection. *IEEE/ACM Trans. Netw.*, 15(5):993–1006, 2007.

[14] Yenliang Lu, Huier Lin, Yajuan Gu, and Helmy A. Towards mobility-rich analysis in ad hoc networks: using contraction, expansion and hybrid models. In *IEEE International Conference on Communications (ICC)*, volume 7, pages 4346– 4351, 2004.

[15] N. Sadagopan, F. Bai, B. Krishnamachari, and A. Helmy. Paths: analysis of path duration statistics and their impact on reactive manet routing protocols, 2003.

[16] Amit Kumar Saha and David B. Johnson. Modeling mobility for vehicular ad-hoc networks. In *VANET '04: Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks*, pages 91–92, New York, NY, USA, 2004. ACM.

[17] Dale Skeen and Michael Stonebraker. A formal model of crash recovery in a distributed system. *IEEE Trans. Software Eng.*, 9(3):219–228, 1983.

[18] Jari Veijalainen, Frank Eliassen, and Berhard Holtkamp. The s-transaction model. pages 467–513, 1992.

[19] Helmut Wächter and Andreas Reuter. The contract model. pages 219–263, 1992.

[20] Gerhard Weikum and Hans-Jorg Schek. Concepts and applications of multilevel transactions and open nested transactions. In *Database Transaction Models for Advanced Applications*, pages 515–553. 1992.