Semantic Analysis of Gossip Protocols for Wireless Sensor Networks

Ruggero Lanotte[1] and Massimo Merro[2]

[1] Dipartimento di Informatica e Comunicazione, Università dell'Insubria, Italy
[2] Dipartimento di Informatica, Università degli Studi di Verona, Italy

Abstract. Gossip protocols have been proposed as a robust and efficient method for disseminating information throughout large-scale networks. In this paper, we propose a compositional analysis technique to study formal probabilistic models of gossip protocols in the context of wireless sensor networks. We introduce a simple probabilistic timed process calculus for modelling wireless sensor networks. A simulation theory is developed to compare probabilistic protocols that have similar behaviour up to a certain probability. This theory is used to prove a number of algebraic laws which revealed to be very effective to evaluate the performances of gossip networks with and without communication collisions.

1 Introduction

Wireless sensor networks (WSNs) are (possibly large-scale) networks of sensor nodes deployed in strategic areas to gather data. Sensor nodes collaborate using wireless communications with an asymmetric many-to-one data transfer model. Typically, they send their sensed data to a sink node which collects the relevant information. WSNs are primarily designed for monitoring environments that humans cannot easily reach (e.g., motion, target tracking, fire detection, chemicals, temperature); they are used as embedded systems (e.g., biomedical sensor engineering, smart homes) or mobile applications (e.g., when attached to robots, soldiers, or vehicles). In wireless sensor networks, sensor nodes are usually battery-powered, and the energy expenditure of sensors has to be wisely managed by their architectures and protocols to prolong the overall network lifetime. Energy conservation is thus one of the major issues in sensor networks.

Flooding is a traditional robust algorithm that delivers data packets in a network from a source to a destination. In WSNs, each node that receives a message propagates it to all its neighbours by broadcast. This causes unnecessary retransmissions increasing the number of collisions, together depriving sensors of valuable battery power. Therefore, flooding algorithms may not be suitable in the context of dense networks like wireless sensor networks.

Gossipping [9] addresses some critical problems of flooding overhead. The goal of gossip protocols is to reduce the number of retransmissions by making some of the nodes discard the message instead of forwarding it. Gossip protocols exhibit both *nondeterministic* and *probabilistic* behaviour. Nondeterminism arises

J.-P. Katoen and B. König (Eds.): CONCUR 2011, LNCS 6901, pp. 156–170, 2011.

as they deal with distributed networks in which the activities of individual nodes occur nondeterministically. As to the probabilistic behaviour, nodes are required to forward packets with a pre-specified gossip probability p_{gsp}. When a node receives a message, rather than immediately retransmitting it as in flooding, it relies on the probability p_{gsp} to determine whether or not to retransmit. The main benefit is that when p_{gsp} is sufficiently large, the entire network receives the broadcast message with very high probability, even though only a nondeterministic subset of nodes has forwarded the message.

Most of the analyses of protocols for large-scale WSNs are usually based on discrete-event simulators (e.g., ns-2, Opnet and Glomosim). However, different simulators often support different models of the MAC physical-layer yielding different results, even for simple systems. In principle, as noticed in [2], owing to their often relatively simple structure, gossip protocols lend themselves very well to formal analysis, in order to predict their behaviour with high confidence. Formal analysis techniques are supported by (semi-)automated tools. For instance, *probabilistic model checking* [6,10] provides both an exhaustive search of all possible behaviours of the system, and exact, rather than approximate, quantitative results. Of course, model checking suffers from the so-called state explosion problem whereas simulation-based approaches are scalable to much larger systems, at the expense of exhaustiveness and numerical accuracy.

Contribution. In this paper, we propose a compositional analysis technique to study probabilistic models of gossip protocols in the context of WSNs. We introduce a simple probabilistic timed process calculus, called `pTCWS`, for modelling wireless sensor networks. We then develop a compositional simulation theory, denoted $\sqsubseteq_p$, to compare probabilistic protocols that have similar behaviour up to a certain probability p. Intuitively, we write $M \sqsubseteq_p N$ if M is simulated by N with a probability (at least) p. Compositionality is crucial when reasoning about large-scale protocols where all nodes run the same probabilistic (simple) code as in gossip protocols. For instance, it allows us to join and sometime merge the behaviour of different components of a network. In particular, for a gossip network $\text{GSP}_{p_{\text{gsp}}}$, which transmits with gossip probability p_{gsp}, we can estimate the probability p_{ok} to simulate a non-probabilistic network GSP_OK whose target nodes successfully receive the message:

$$\text{GSP\_OK} \quad \sqsubseteq_{p_{\text{ok}}} \quad \text{GSP}_{p_{\text{gsp}}} \ .$$

For this purpose, we prove and apply a number of algebraic laws, whose application can be *mechanised*, to evaluate the performances of gossip networks.

The paper uses the gossip protocol described above as baseline. That description, however, is incomplete. It does not specify, for instance, what happens in case of a collision, i.e. when a node receives two messages at the same time. We start our analysis by assuming no collision. Then, we study gossip protocols in the presence of *communication collision*, to determine its effect on the performance results.

In this paper proofs are sketched or omitted; full proofs can be found in [12].

Table 1 Syntax

Networks:		
M, N	$::= \mathbf{0}$	empty network
	$\mid M_1 \mid M_2$	parallel composition
	$\mid n[P]^{\nu}$	node
Processes:		
P, Q	$::= \mathsf{nil}$	stuck
	$\mid !\langle u\rangle.C$	broadcast
	$\mid \lfloor ?(x).C \rfloor D$	receiver with timeout
	$\mid \lfloor \tau.C \rfloor D$	internal with timeout
	$\mid \sigma.C$	sleep
	$\mid X$	process variable
	$\mid \mathsf{fix}\, X.P$	recursion
Probabilistic Choice:		
C, D	$::= \bigoplus_{i\in I} p_i{:}P_i$	

2 A Probabilistic Timed Process Calculus

In Table 1, we define the syntax of pTCWS in a two-level structure, a lower one for *processes* and an upper one for *networks*. We use letters $m, n, \ldots$ for logical names, x, y, z for *variables*, u for *values*, and v and w for *closed values*, i.e. values that do not contain variables.

A network in pTCWS is a (possibly empty) collection of nodes (which represent devices) running in parallel and using a unique common radio channel to communicate with each other. All nodes are assumed to have the same transmission range (this is a quite common assumption in models for ad hoc networks).The communication paradigm is *local broadcast*; only nodes located in the range of the transmitter may receive data. We write $n[P]^{\nu}$ for a node named n (the device network address) executing the sequential process P. The tag ν contains (the names of) the neighbours of n. Said in other words, ν contains all nodes laying in the transmission cell of n (except n). In this manner, we model the network topology.[1] Our wireless networks have a fixed topology as node mobility is not relevant to sensor networks. Moreover, nodes cannot be created or destroyed.

Processes are sequential and live within the nodes. The symbol nil denotes the stuck process. The sender process $!\langle v\rangle.C$ broadcasts the value v, the continuation being C. The process $\lfloor ?(x).C \rfloor D$ denotes a receiver with timeout. Intuitively, this process either receives a value v, in the current time interval, and then continues as C where the variable x is instantiated with v, or it idles for one time unit, and then continues as D. Similarly, the process $\lfloor \tau.C \rfloor D$ either performs an internal action, in the current time interval, or it idles for one time unit and then continues

[1] We could have represented the topology in terms of a restriction operator à la CCS on node names; we preferred our notation to keep at hand the neighbours of a node.

as D. The process $\sigma.C$ models sleeping for one time unit. In sub-terms of the form $\sigma.D$, $\lfloor \tau.C \rfloor D$ and $\lfloor ?(x).C \rfloor D$ the occurrence of D is said to be *time-guarded*. The process $\mathsf{fix}\, X.P$ denotes *time-guarded recursion*, as all occurrences of the process variable X may only occur time-guarded in P.

Remark 1. In the remainder of the paper, with an abuse of notation, we will write $?(x).C$ to denote a persistent listener, defined as $\mathsf{fix}\, X.\lfloor ?(x).C \rfloor X$. Similarly, we will write $\tau.C$ as an abbreviation for $\mathsf{fix}\, X.\lfloor \tau.C \rfloor X$.

The construct $\bigoplus_{i\in I} p_i{:}P_i$ denotes probabilistic choice, where I is an indexing finite set and $p_i \in (0,1]$ denotes the probability to execute the process P_i, with $\sum_{i\in I} p_i = 1$. In process $\lfloor ?(x).C \rfloor D$ the variable x is bound in C. Similarly, in process $\mathsf{fix}\, X.P$ the process variable X is bound in P. This gives rise to the standard notions of *free (process) variables* and *bound (process) variables* and *α-conversion*. We identify processes and networks up to α-conversion. A term is said to be *closed* if it does not contain free (process) variables. We always work with closed networks: The absence of free variables is trivially maintained at run-time. We write $\{^{v}/_{x}\}T$ for the substitution of the variable x with the value v in the term T. Similarly, we write $\{^{P}/_{X}\}T$ for the substitution of the process variable X with the process P in T.

We report some notational *conventions*. $\prod_{i\in I} M_i$ denotes the parallel composition of all M_i, for $i \in I$. We identify $\prod_{i\in I} M_i = \mathbf{0}$ if $I = \emptyset$. We write $P_1 \oplus_p P_2$ to denote the probabilistic process $p{:}P_1 \oplus (1{-}p){:}P_2$. We identify the probabilistic process $1{:}P$ with P. We write $!\langle v \rangle$ as an abbreviation for $!\langle v \rangle.1{:}\mathsf{nil}$. For $k > 0$ we write $\sigma^k.P$ as an abbreviation for $\sigma.\sigma.\ldots.\sigma.P$, where prefix σ appears k times.

Here are some definitions that will be useful in the remainder of the paper. Given a network M, $\mathsf{nds}(M)$ returns the names of M. If $m \in \mathsf{nds}(M)$, the function $\mathsf{ngh}(m, M)$ returns the set of the neighbours of m in M. Thus, for $M = M_1 \mid m[P]^{\nu} \mid M_2$ it holds that $\mathsf{ngh}(m, M) = \nu$. We write $\mathsf{ngh}(M)$ for $\bigcup_{m\in\mathsf{nds}(M)} \mathsf{ngh}(m, M)$.

Definition 1. Structural congruence *over* ***pTCWS****, written* $\equiv$*, is defined as the smallest equivalence relation, preserved by parallel composition, which is a commutative monoid with respect to parallel composition and for which* $n[\mathsf{fix}\, X.P]^{\nu} \equiv n[P\{^{\mathsf{fix}\, X.P}/_{X}\}]^{\nu}$.

The syntax presented in Table 1 allows to derive networks which are somehow ill-formed. With the following definition we rule out networks containing two nodes with the same name. Moreover, as all nodes have the same transmission range, the neighbouring relation must be symmetric. Finally, in order to guarantee clock synchronisation, we impose network connectivity.

Definition 2 (Well-formedness). *M is said to be* well-formed *if*

- *whenever* $M \equiv M_1 \mid m_1[P_1]^{\nu_1} \mid m_2[P_2]^{\nu_2}$ *it holds that* $m_1 \neq m_2$*;*
- *whenever* $M \equiv N \mid m_1[P_1]^{\nu_1} \mid m_2[P_2]^{\nu_2}$ *with* $m_1 \in \nu_2$ *it holds that* $m_2 \in \nu_1$*;*
- *for all* $m, n \in \mathsf{nds}(M)$ *there are* $m_1, \ldots, m_k \in \mathsf{nds}(M)$*, such that* $m{=}m_1$*,* $n{=}m_k$*,* $\nu_j = \mathsf{ngh}(m_j, M)$*, for* $1{\leq}j{\leq}k$*, and* $m_i \in \nu_{i+1}$*, for* $1{\leq}i{\leq}k{-}1$*.*

Henceforth we will always work with well-formed networks.

2.1 Probabilistic Labelled Transition Semantics

Along the lines of [5,11], we propose an *operational semantics* for pTCWS associating with each network a graph-like structure representing its possible reactions: We use a generalisation of labelled transition system that includes probabilities.

Below, we report the mathematical machinery for doing that.

Definition 3 (Deng et al. [5]). *A (discrete)* probability sub-distribution *over a countable set S is a function $\Delta : S \to [0,1]$ such that $\sum_{s\in S} \Delta(s) \in (0..1]$. The* support *of a probability sub-distribution Δ is given by $\lceil \Delta \rceil = \{s \in S \mid \Delta(s) > 0\}$. We write $\mathcal{D}_{\rm sub}(S)$, ranged over Δ, Θ, Φ, for the set of all probability sub-distributions over S with finite support. For any $s \in S$, the* point distribution at *s, written $\overline{s}$, assigns probability 1 to s and 0 to all others elements of S.*

If $p_i \geq 0$ and Δ_i is a sub-distribution for each i in some finite index set I, and $\sum_{i\in I} p_i \in (0,1]$, then the probability sub-distribution $\sum_{i\in I} p_i \cdot \Delta_i$ is given by

$$(\sum_{i\in I} p_i \cdot \Delta_i)(s) \stackrel{\rm def}{=} \sum_{i\in I} p_i \cdot \Delta_i(s) \ .$$

We write a sub-distribution as $p_1 \cdot \Delta_1 + ... + p_n \cdot \Delta_n$, when the index set I is $\{1,\ldots,n\}$. Sometimes, with an abuse of notation, in the previous decomposition, the terms Δ_i are not necessarily distinct (for instance $1 \cdot \Delta$ may be rewritten as $p \cdot \Delta + (1-p) \cdot \Delta$, for any $p \in [0..1]$). A probability sub-distribution $\Delta \in \mathcal{D}_{\rm sub}(S)$ is said to be a *probability distribution* if $\sum_{s\in S} \Delta(s) = 1$. With $\mathcal{D}(S)$ we denote the set of all probability distributions over S with finite support.

Definition 1 and Definition 2 generalise to sub-distributions in $\mathcal{D}_{\rm sub}(\texttt{pTCWS})$. Given two probability sub-distributions Δ and Θ, we write $\Delta \equiv \Theta$ whenever $\Delta([M]_\equiv) = \Theta([M]_\equiv)$ for all equivalence classes $[M]_\equiv \subseteq \texttt{pTCWS}$ of $\equiv$. Moreover, a probability sub-distribution $\Delta \in \mathcal{D}_{\rm sub}(\texttt{pTCWS})$ is said to be well-formed if its support contains only well-formed networks.

We now give the probabilistic generalisation of labelled transition system:

Definition 4 (Deng et al. [5]). *A* probabilistic labelled transition system[2] *(pLTS) is a triple $\langle S, \mathcal{L}, \rightarrow\rangle$ where i) S is a set of states; ii) $\mathcal{L}$ is a set of transition labels; iii) $\rightarrow$ is a labelled transition relation contained in $S \times \mathcal{L} \times \mathcal{D}(S)$.*

The operational semantics of pTCWS is given by a particular pLTS $\langle \texttt{pTCWS}, \mathcal{L}, \rightarrow\rangle$, where $\mathcal{L} = \{m!v\triangleright\mu, m?v, \tau, \sigma\}$ contains the labels denoting broadcasting, reception, internal actions and time passing, respectively. As regards the labelled transition relation, we need to formalise the interpretation of nodes containing probabilistic processes as probability distributions.

Definition 5. *For any probabilistic choice $\bigoplus_{i\in I} p_i{:}P_i$ over a finite indexing set I, $[\![n[\bigoplus_{i\in I} p_i{:}P_i]^\nu]\!]$ denotes the probability distribution defined as follows:*

- *if $I \neq \emptyset$ then for any $M \in \texttt{pTCWS}$: $[\![n[\bigoplus_{i\in I} p_i{:}P_i]^\nu]\!](M) \stackrel{\rm def}{=} \sum_{i\in I \wedge n[P_i]^\nu = M} p_i$*

[2] Essentially the same model has appeared in the literature under different names such as, for instance, *NP-systems* [8] or *simple probabilistic automata* [15].

Table 2 Probabilistic Labelled Transition System

$$\text{(Snd)}\ \frac{-}{m[!\langle v\rangle.C]^{\nu} \xrightarrow{m!v\triangleright\nu} [\![m[C]^{\nu}]\!]} \qquad \text{(Rcv)}\ \frac{m \in \nu}{n[\lfloor ?(x).C \rfloor D]^{\nu} \xrightarrow{m?v} [\![n[\{^{v}/_{x}\}C]^{\nu}]\!]}$$

$$\text{(Rcv-}\mathbf{0}\text{)}\ \frac{-}{\mathbf{0} \xrightarrow{m?v} \overline{\mathbf{0}}} \qquad \text{(RcvEnb)}\ \frac{\neg(m \in \nu \ \wedge\ \mathsf{rcv}(P)) \ \wedge\ m \neq n}{n[P]^{\nu} \xrightarrow{m?v} \overline{n[P]^{\nu}}}$$

$$\text{(RcvPar)}\ \frac{M \xrightarrow{m?v} \Delta \quad N \xrightarrow{m?v} \Theta}{M \mid N \xrightarrow{m?v} \Delta \mid \Theta} \qquad \text{(Bcast)}\ \frac{M \xrightarrow{m!v\triangleright\nu} \Delta \quad N \xrightarrow{m?v} \Theta \quad {\scriptstyle \mu:=\nu\setminus \mathsf{nds}(N)}}{M \mid N \xrightarrow{m!v\triangleright\mu} \Delta \mid \Theta}$$

$$\text{(Tau)}\ \frac{-}{m[\lfloor \tau.C \rfloor D]^{\nu} \xrightarrow{\tau} [\![m[C]^{\nu}]\!]} \qquad \text{(TauPar)}\ \frac{M \xrightarrow{\tau} \Delta}{M \mid N \xrightarrow{\tau} \Delta \mid \overline{N}}$$

$$(\sigma\text{-}\mathbf{0})\ \frac{-}{\mathbf{0} \xrightarrow{\sigma} \overline{\mathbf{0}}} \qquad (\sigma\text{-nil})\ \frac{-}{n[\mathsf{nil}]^{\nu} \xrightarrow{\sigma} \overline{n[\mathsf{nil}]^{\nu}}}$$

$$\text{(Timeout)}\ \frac{-}{n[\lfloor \ldots \rfloor D]^{\nu} \xrightarrow{\sigma} [\![n[D]^{\nu}]\!]} \qquad \text{(Sleep)}\ \frac{-}{n[\sigma.C]^{\nu} \xrightarrow{\sigma} [\![n[C]^{\nu}]\!]}$$

$$(\sigma\text{-Par})\ \frac{M \xrightarrow{\sigma} \Delta \quad N \xrightarrow{\sigma} \Theta}{M \mid N \xrightarrow{\sigma} \Delta \mid \Theta} \qquad \text{(Rec)}\ \frac{n[\{^{\mathsf{fix}\,X.P}/_{X}\}P]^{\nu} \xrightarrow{\lambda} \Delta}{n[\mathsf{fix}\,X.P]^{\nu} \xrightarrow{\lambda} \Delta}$$

– *if* $I = \emptyset$ *then* $[\![n[\bigoplus_{i\in I} p_i{:}P_i]^{\nu}]\!] \stackrel{\text{def}}{=} \overline{n[\mathsf{nil}]^{\nu}}$.

The definition of the relations $\xrightarrow{\lambda}$, for $\lambda \in \mathcal{L}$, is given in Table 2. Some of these rules use an obvious notation for distributing parallel composition over a (sub-)distribution:

$$(\Delta \mid \Theta)(M) \stackrel{\text{def}}{=} \begin{cases} \Delta(M_1) \cdot \Theta(M_2) & \text{if } M = M_1 \mid M_2 \\ \mathbf{0} & \text{otherwise.} \end{cases}$$

In rule (Snd) a sender m broadcasts a message v to its neighbours ν, and then continues as C. In the label $m!v\triangleright\nu$ the set ν contains the neighbours of m which may receive the message v. In rule (Rcv) a receiver gets a message v from a neighbour node m, and then evolves as $\{^{v}/_{x}\}C$. If no message is received in the current time interval the node n will continue with process D, as specified in rule (Timeout). Rules (Rcv-0) and (RcvEnb) serve to model reception enabling for synchronisation purposes. For instance, rule (RcvEnb) regards nodes which are not involved in transmissions originating from m. This may happen either because the two nodes are out of range (i.e. $m \notin \nu$) or because n is not willing to receive ($\mathsf{rcv}(P)$ is a boolean predicate that returns true if $n[P]^{\nu} \equiv n[\lfloor ?(x).C \rfloor D]^{\nu}$, for some x, C, D). In both cases, node n is not affected by the transmission. In rule (RcvPar) we model the composition of two networks receiving the

same message from the same transmitter. Rule (Bcast) models the propagation of messages on the broadcast channel. Note that we loose track of those transmitter's neighbours that are in N. Rule (Tau) models internal computations in a single node. Rule (TauPar) propagates internal computations on parallel components. Rules (σ-nil) and (σ-$\mathbf{0}$) are straightforward as both terms $\mathbf{0}$ and $n[\mathsf{nil}]^\nu$ do not prevent time-passing. Rule (Sleep) models sleeping for one time unit. Rule (σ-Par) models time synchronisation between parallel components. Rule (Rec) is standard. Rules (Bcast) and (TauPar) have their symmetric counterparts.

Below, we report a number of basic properties of our LTS.

Proposition 1. *Let M, M_1 and M_2 be well-formed networks.*

1. *$m \notin \mathsf{nds}(M)$ if and only if $M \xrightarrow{m?v} \Delta$, for some distribution Δ.*
2. *If $M_1 \mid M_2 \xrightarrow{m?v} \Delta$ if and only if there are Δ_1 and Δ_2 such that $M_1 \xrightarrow{m?v} \Delta_1$, $M_2 \xrightarrow{m?v} \Delta_2$ with $\Delta = \Delta_1 \mid \Delta_2$.*
3. *If $M \xrightarrow{m!v \triangleright \mu} \Delta$ then $M \equiv m[!\langle v\rangle.C]^\nu \mid N$, for some m, ν, C and N such that $m[!\langle v\rangle.C]^\nu \xrightarrow{m!v \triangleright \nu} [\![m[C]^\nu]\!]$, $N \xrightarrow{m?v} \Theta$, $\Delta \equiv [\![m[C]^\nu]\!] \mid \Theta$ and $\mu = \nu \setminus \mathsf{nds}(N)$.*
4. *If $M \xrightarrow{\tau} \Delta$ then $M \equiv m[\lfloor \tau.C \rfloor D]^\nu \mid N$, for some m, ν, C, D and N such that $m[\lfloor \tau.C \rfloor D]^\nu \xrightarrow{\tau} [\![m[C]^\nu]\!]$ and $\Delta \equiv [\![m[C]^\nu]\!] \mid \overline{N}$.*
5. *$M_1 \mid M_2 \xrightarrow{\sigma} \Delta$ if and only if there are Δ_1 and Δ_2 such that $M_1 \xrightarrow{\sigma} \Delta_1$, $M_2 \xrightarrow{\sigma} \Delta_2$ and $\Delta = \Delta_1 \mid \Delta_2$.*

As the topology of our networks is static and nodes cannot be created or destroyed, it is easy to prove the following result.

Proposition 2 (Well-formedness preservation). *Let M be a well-formed network. If $M \xrightarrow{\lambda} \Theta$ then Θ is a well-formed distribution.*

2.2 Time Properties

Our calculus enjoys a number of desirable time properties. Proposition 3 formalises the determinism nature of time passing: a network can reach at most one new state by executing the action σ.

Proposition 3 (Time Determinism). *Let M be a well-formed network. If $M \xrightarrow{\sigma} \Delta$ and $M \xrightarrow{\sigma} \Theta$ then Δ and Θ are syntactically the same.*

The maximal progress property says that sender nodes transmit immediately. Said in other words, the passage of time cannot block transmissions.

Proposition 4 (Maximal Progress). *Let M be a well-formed network. If $M \equiv m[!\langle v\rangle.C]^\nu \mid N$ then $M \xrightarrow{\sigma} \Delta$ for no distribution Δ.*

Patience guarantees that a process will wait indefinitely until it can communicate [7]. In our setting, this means that if no transmissions can start then it must be possible to execute a σ-action to let time pass.

Proposition 5 (Patience). *Let $M = \prod_{i\in I} m_i[P_i]^{\nu_i}$ be a well-formed network, such that for all $i \in I$ it holds that $P_i \neq !\langle v\rangle.C$, then there is a distribution Δ such that $M \xrightarrow{\sigma} \Delta$.*

Finally, as recursion is time-guarded, our networks satisfy the *well-timedness* (or *finite variability*) property [14]. Intuitively, only a finite number of instantaneous actions can fire between two contiguous σ-actions.

Proposition 6 (Well-Timedness). *For any well-formed network M there is an upper bound $k \in \mathbb{N}$ such that whenever $M \xrightarrow{\alpha_1} \cdots \xrightarrow{\alpha_h} \Delta$, $\alpha_j \neq \sigma$ for $1 \leq j \leq h$, then $h \leq k$.*

3 Simulation Up to Probability

In this section, we use our pLTS to define an appropriate *probabilistic timed simulation theory* for `pTCWS`. Our focus is on weak similarities which abstract away non-observable actions. To this end, we extend the set of rules of Table 2 with the following two rules:

$$\text{(Shh)}\ \frac{M \xrightarrow{m!v\triangleright\emptyset} \Delta}{M \xrightarrow{\tau} \Delta} \qquad\qquad \text{(Obs)}\ \frac{M \xrightarrow{m!v\triangleright\nu} \Delta \quad \nu \neq \emptyset}{M \xrightarrow{!v\triangleright\nu} \Delta}$$

Rule (Shh) models transmissions that cannot be observed because none of the potential receivers is in the environment. Rule (Obs) models transmissions that can be observed by those nodes of the environment contained in ν. Notice that the name of the transmitter is removed from the label. This is motivated by the fact that nodes may refuse to reveal their identities, e.g. for security reasons or limited sensory capabilities in perceiving these identities. Notice also that in a derivation tree the rule (Obs) can only be applied at top-level.

In the rest of the paper, the metavariable α ranges over the following actions: $!v\triangleright\nu$, $m?v$, τ, and σ.

Let us provide the definition of weak transition. In a probabilistic setting, this definition is somewhat complicated by the fact that transitions go from processes (in our case networks) to distributions; consequently if we use weak transitions $\stackrel{\alpha}{\Longrightarrow}$, which abstract from sequences of internal actions, then we need to generalise transitions, so that they go from (sub-)distributions to (sub-)distributions. We write $M \xrightarrow{\hat{\tau}} \Delta$ if either $M \xrightarrow{\tau} \Delta$ or $\Delta = \overline{M}$, and $M \xrightarrow{\hat{\alpha}} \Delta$ if $M \xrightarrow{\alpha} \Delta$, for $\alpha \neq \tau$. Let $\Delta = \sum_{i\in I} p_i \cdot \overline{M_i}$ be a sub-distribution; we write $\Delta \xrightarrow{\hat{\alpha}} \Theta$ whenever $\Theta = \sum_{j\in J} p_i \cdot \Theta_j$, with $J \subseteq I$, and $M_j \xrightarrow{\hat{\alpha}} \Theta_j$, for any $j \in J$. We define the weak transition relation $\stackrel{\hat{\tau}}{\Longrightarrow}$ as the transitive and reflexive closure of $\xrightarrow{\hat{\tau}}$, i.e. $(\xrightarrow{\hat{\tau}})^*$, while for $\alpha \neq \tau$ we let $\stackrel{\hat{\alpha}}{\Longrightarrow}$ to denote $\stackrel{\hat{\tau}}{\Longrightarrow}\xrightarrow{\alpha}\stackrel{\hat{\tau}}{\Longrightarrow}$. Finally, independently whether α is τ or not, we write $\stackrel{\alpha}{\Longrightarrow}$ to denote $\stackrel{\hat{\tau}}{\Longrightarrow}\xrightarrow{\alpha}\stackrel{\hat{\tau}}{\Longrightarrow}$. Proposition 6 ensures that weak transitions always contain a bounded number of $\xrightarrow{\tau}$ actions.

Since transitions go from networks to distributions we need to lift our relations over networks to sub-distribution. Let $\mathcal{R} \subseteq \mathtt{pTCWS} \times \mathtt{pTCWS}$ be a binary relation over networks. We lift it to a relation $\overline{\mathcal{R}} \subseteq \mathcal{D}_{\rm sub}(\mathtt{pTCWS}) \times \mathcal{D}_{\rm sub}(\mathtt{pTCWS})$ by letting $\Delta\ \overline{\mathcal{R}}\ \Theta$ whenever:

- $\Delta = \sum_{i\in I} p_i \cdot \overline{M_i}$, where I is a finite index set
- for each $i \in I$ there is a network N_i such that $M_i\ \mathcal{R}\ N_i$ and $\Theta = \sum_{i\in I} p_i \cdot \overline{N_i}$.

Definition 6 (Simulation up to probability). *Let $p \in (0..1]$ be a probability. A parameterised relation $\mathcal{R}_p \subseteq \mathtt{pTCWS} \times \mathtt{pTCWS}$ is said to be a* simulation up to probability *p if whenever $(M,N) \in \mathcal{R}_p$ and $M \xrightarrow{\alpha} \Delta$, there are a probability q, with $\frac{p}{q} \in (0..1]$, and a distribution Θ such that $N \stackrel{\hat{\alpha}}{\Longrightarrow} q \cdot \Theta$ and $\Delta\ \overline{\mathcal{R}_{\frac{p}{q}}}\ \Theta$. We write $M \sqsubseteq_p N$ if $(M,N) \in \mathcal{R}_p$ for some simulation up to probability $\mathcal{R}_p$. The equivalence induced by $\sqsubseteq_p$ is denoted $\simeq_p$.*

Intuitively, if $M \sqsubseteq_p N$ then M is simulated by N up to a probability (at least) p. Within the remaining probability $1-p$, the network N might still simulate M. That is why the probability p is a lower-bound, i.e. $\sqsubseteq_q \subseteq \sqsubseteq_p$, for any $q \leq p$.

Example 1. 1. $n[P]^\nu \sqsubseteq_p n[\tau.(P \oplus_p Q)]^\nu$
2. $n[P]^\nu \sqsubseteq_q n[\tau.(P \oplus_p Q)]^\nu$ with $0 \leq q \leq p$
3. $n[Q]^\nu \sqsubseteq_{p(1-q)} n[\tau.(\tau.(P \oplus_q Q) \oplus_p R)]^\nu$
4. $n[!\langle v\rangle.!\langle w\rangle]^\nu \sqsubseteq_{pq} n[\tau.(!\langle v\rangle.\tau.(!\langle w\rangle \oplus_q P) \oplus_p Q)]^\nu$.

From these examples one can realise that when $M \sqsubseteq_p N$ the network N may contain a number of probabilistic choices which are resolved in M with a probability (at least) p. Unfortunately, this notion of similarity is not transitive, in the sense that is it not true that $\sqsubseteq_p \sqsubseteq_q = \sqsubseteq_{pq}$.[3] This would be a highly desirable property to algebraically reason on our networks. However, as one may have noticed from the first three algebraic laws of the previous example, the probability p is often manifested when executing the first action. So, to recover transitivity we add a root condition and replace weak transitions $\stackrel{\hat{\alpha}}{\Longrightarrow}$ with $\stackrel{\alpha}{\Longrightarrow}$.

Definition 7 (Rooted simulation up to probability). *Let $p \in (0..1]$ be a probability. A parameterised relation $\mathcal{R}_p \subseteq \mathtt{pTCWS} \times \mathtt{pTCWS}$ is said to be a* rooted simulation up to probability *p if whenever $(M,N) \in \mathcal{R}_p$ and $M \xrightarrow{\alpha} \Delta$ there is a distribution θ such that $N \stackrel{\alpha}{\Longrightarrow} p \cdot \Theta$ and $\Delta\ \overline{\mathcal{R}_1}\ \Theta$. We write $M \sqsubseteq^{\mathbf{1}}_p N$ if $(M,N) \in \mathcal{R}_p$ for some rooted simulation up to probability $\mathcal{R}_p$. The equivalence induced by $\sqsubseteq^{\mathbf{1}}_p$ is denoted $\simeq^{\mathbf{1}}_p$.*

Proposition 7. *$M \sqsubseteq^{\mathbf{1}}_p N$ implies $M \sqsubseteq_p N$.*

Proposition 8. *If $M \sqsubseteq^{\mathbf{1}}_p N$ and $N \sqsubseteq^{\mathbf{1}}_q O$ then $M \sqsubseteq^{\mathbf{1}}_{pq} O$.*

Here comes a crucial result on the compositionality of our simulation theory.

[3] For details the reader is deferred to [12].

Theorem 1. *Let M, N and O be well-formed networks such that both $M \mid O$ and $N \mid O$ are well-formed as well. Then,*

1. *$M \sqsubseteq_p^{\mathbf{1}} N$ implies $M \mid O \sqsubseteq_p^{\mathbf{1}} N \mid O$*
2. *$M \sqsubseteq_p N$ implies $M \mid O \sqsubseteq_p N \mid O$.*

Below, we report a number of algebraic laws that will be useful in the next section when analysing gossip protocols.

Theorem 2 (Some algebraic laws).

1. $n[\sigma.\mathsf{nil}]^\nu \simeq_1^{\mathbf{1}} n[\mathsf{nil}]^\nu$
2. $\prod_{i\in I} m_i[P_i]^{\nu_{m_i}} \simeq_1^{\mathbf{1}} \prod_{j\in J} n_j[Q_j]^{\nu_{n_j}}$ *iff* $\prod_{i\in I} m_i[\sigma.P_i]^{\nu_{m_i}} \simeq_1^{\mathbf{1}} \prod_{j\in J} n_j[\sigma.Q_j]^{\nu_{n_j}}$
3. $n[\lfloor ?(x).P \rfloor Q]^\nu \simeq_1^{\mathbf{1}} n[\sigma.Q]^\nu$ *if no nodes in ν send in the current time interval.*
4. $n[?(x).P]^\nu \simeq_1^{\mathbf{1}} n[\mathsf{nil}]^\nu$ *if no nodes in ν contain sender processes*
5. $n[?(x).P]^\nu \simeq_1^{\mathbf{1}} n[\sigma.?(x).P]^\nu$ *if no nodes in ν send in the current time unit.*
6. $m[\tau.(!\langle v\rangle \oplus_p \mathsf{nil})]^\nu \mid \prod_{i\in I} n_i[P_i]^{\nu_i} \,{}^{\mathbf{1}}_{1}\!\sqsupseteq\; m[\mathsf{nil}]^\nu \mid \prod_{i\in I} n_i[P_i]^{\nu_i}$ *if $\nu = \bigcup_{i\in I} n_i$, and for all $i \in I$ either $P_i = \mathsf{nil}$ or $P_i = \sigma.Q_i$, for some Q_i.*

4 Gossipping without Collisions

The baseline model for our study is gossipping without collisions where all nodes are perfectly synchronised. For the sake of clarity, communication proceeds in synchronous rounds: A node can transmit or receive one message per round. In our implementation rounds are separated by σ-actions.

The processes involved in the protocol are the following:

$$\mathsf{snd}\langle u\rangle_{p_\mathrm{g}} \stackrel{\text{def}}{=} \tau.(!\langle u\rangle \oplus_{p_\mathrm{g}} \mathsf{nil}) \quad \mathsf{resnd}\langle u\rangle_{p_\mathrm{g}} \stackrel{\text{def}}{=} \sigma.\mathsf{snd}\langle u\rangle_{p_\mathrm{g}} \quad \mathsf{fwd}_{p_\mathrm{g}} \stackrel{\text{def}}{=} ?(x).\mathsf{resnd}\langle x\rangle_{p_\mathrm{g}} \ .$$

Here, a sender broadcasts a value u with gossip probability $p_\mathrm{g} \in (0..1]$, and a forwarder gossips the received value, in the next round, with the same probability.

Now, we can apply our simulation theory to prove algebraic laws on message propagation. For instance, consider a fragment of a network with a sender m and two forwarder neighbours n_1 and n_2. Then, the following holds:

$$m[\mathsf{snd}\langle v\rangle_p]^\nu \mid n_1[\mathsf{fwd}_q]^{\nu_1} \mid n_2[\mathsf{fwd}_r]^{\nu_2} \;{}^{\mathbf{1}}_{p}\!\sqsupseteq\; m[\mathsf{nil}]^\nu \mid n_1[\mathsf{resnd}\langle v\rangle_q]^{\nu_1} \mid n_2[\mathsf{resnd}\langle v\rangle_r]^{\nu_2}$$

whenever $\nu = \{n_1, n_2\}$ and the nodes in $\nu_1 \cup \nu_2 \setminus \{m\}$ cannot transmit in the current instant of time. A complementary law is

$$m_1[\mathsf{snd}\langle v\rangle_{p_1}]^n \mid m_2[\mathsf{snd}\langle v\rangle_{p_2}]^n \mid n[\mathsf{fwd}_q]^\nu \;{}^{\mathbf{1}}_{p}\!\sqsupseteq\; m_1[\mathsf{nil}]^n \mid m_2[\mathsf{nil}]^n \mid n[\mathsf{resnd}\langle v\rangle_q]^\nu$$

with $p = 1-(1-p_1)(1-p_2)$, whenever the nodes in $\nu \setminus \{m_1, m_2\}$ cannot transmit in the current instant of time. More generally, the following result holds.

Theorem 3 (Message propagation). *Let K, I and J be pairwise disjoint subsets of $\mathbb{N}$. Let M be a well-formed network defined as*

$$M \equiv \prod_{k\in K} m_k[\mathsf{nil}]^{\nu_{m_k}} \mid \prod_{i\in I} m_i[\mathsf{snd}\langle v\rangle_{p_i}]^{\nu_{m_i}} \mid \prod_{j\in J} n_j[\mathsf{fwd}_{q_j}]^{\nu_{n_j}}$$

such that for all $i \in I$ it holds that $\bigcup_{j\in J} n_j \subseteq \nu_{m_i} \subseteq \bigcup_{j\in J} n_j \cup \bigcup_{k\in K\cup I} m_k$. Then,

$$M \quad \overset{\mathbf{1}}{r}\sqsupseteq \prod_{h\in K\cup I} m_h[\mathsf{nil}]^{\nu_{m_h}} \mid \prod_{j\in J} n_j[\mathsf{resnd}\langle v\rangle_{q_j}]^{\nu_{n_j}}$$

with $r = 1 - \prod_{i\in I}(1-p_i)$.

The previous theorem is a powerful tool to reason on gossip networks. However, it requires that all senders transmit to all subsequent forwarders. This may represent a limitation. Consider, for example, a simple gossip network GSP_1, with gossip probability p, composed by two source nodes s_1 and s_2, a destination node d and three intermediate nodes n_1, n_2 and n_3:

$$\mathrm{GSP}_1 \overset{\text{def}}{=} \prod_{i=1}^{2} s_i[\mathsf{snd}\langle v\rangle_p]^{\nu_{s_i}} \mid \prod_{i=1}^{3} n_i[\mathsf{fwd}_p]^{\nu_{n_i}} \mid d[\mathsf{fwd}_1]^{\nu_d}$$

with $\nu_{s_1}{=}\{n_1\}$, $\nu_{s_2}{=}\{n_1, n_2\}$, $\nu_{n_1}{=}\{s_1, n_3\}$, $\nu_{n_2}{=}\{s_1, s_2, n_3\}$, $\nu_{n_3}{=}\{n_1, n_2, d\}$.

The reader should notice that we cannot directly apply Theorem 3 to GSP_1. This is because node s_1, unlike s_2, can transmit to n_1 but not to n_2. Theorem 3 becomes much more effective when used together with Theorem 4 which allows us to compose estimates concerning partial networks. Roughly speaking, Theorem 4 allows us to consider in our calculation the probability that a sender transmits as well as the probability that the same sender does not transmit.

Theorem 4 (Composing networks).

$$M \mid m[\mathsf{snd}\langle v\rangle_p]^{\nu_m} \mid \prod_{j\in J} n_j[\lfloor ?(x_j).P_j\rfloor Q_j]^{\nu_{n_j}} \quad \overset{\mathbf{1}}{ps_1+(1-p)s_2}\sqsupseteq \quad N$$

whenever

- $M \mid m[\mathsf{nil}]^{\nu_m} \mid \prod_{j\in J} n_j[\{{}^{v}/_{x_j}\}P_j]^{\nu_{n_j}} \overset{\mathbf{1}}{s_1}\sqsupseteq N$
- $M \mid m[\mathsf{nil}]^{\nu_m} \mid \prod_{j\in J} n_j[\lfloor ?(x_j).P_j\rfloor Q_j]^{\nu_{n_j}} \overset{\mathbf{1}}{s_2}\sqsupseteq N$
- $\bigcup_{j\in J} n_j \subseteq \nu_m \subseteq \bigcup_{j\in J} n_j \cup \mathsf{nds}(M)$
- *nodes in $\nu_m \cap \mathsf{nds}(M)$ cannot receive in the current instant of time.*[4]

Let us compute an estimate of success for the network GSP_1 previously defined. For verification reasons we assume that the environment contains a fresh node *test*, close to the destination, i.e. $\nu_d = \{n_3, test\}$, to test successful gossipping. For simplicity, we assume that the *test* node can receive messages but it cannot transmit.

[4] We could generalise the result to take into account more senders at the same time. This would not add expressivity, it would just speed up the reduction process.

We start proving the following chain of similarities by applying, in sequence, Theorem 2(6), Theorem 2(5), Theorem 3 together with Theorem 2(2), with $q = 1-(1-p)^2$, Theorem 2(5) together with Theorem 2(1), again Theorem 3 together with Theorem 2(2), and Theorem 2(1):

$$
\begin{aligned}
& s_1[\mathsf{snd}\langle v\rangle_p]^{\nu_{s_1}} \mid s_2[\mathsf{nil}]^{\nu_{s_2}} \mid \textstyle\prod_{i=1}^{2} n_i[\mathsf{resnd}\langle v\rangle_p]^{\nu_{n_i}} \mid n_3[\mathsf{fwd}_p]^{\nu_{n_3}} \mid d[\mathsf{fwd}_1]^{\nu_d} \\
\overset{\mathbf{1}}{1}\sqsupseteq\ & \textstyle\prod_{i=1}^{2} s_i[\mathsf{nil}]^{\nu_{s_i}} \mid \textstyle\prod_{i=1}^{2} n_i[\mathsf{resnd}\langle v\rangle_p]^{\nu_{n_i}} \mid n_3[\mathsf{fwd}_p]^{\nu_{n_3}} \mid d[\mathsf{fwd}_1]^{\nu_d} \\
\overset{\mathbf{1}}{1}\sqsupseteq\ & \textstyle\prod_{i=1}^{2} s_i[\mathsf{nil}]^{\nu_{s_i}} \mid \textstyle\prod_{i=1}^{2} n_i[\sigma.\mathsf{snd}\langle v\rangle_p]^{\nu_{n_i}} \mid n_3[\sigma.\mathsf{fwd}_p]^{\nu_{n_3}} \mid d[\sigma.\mathsf{fwd}_1]^{\nu_d} \\
\overset{\mathbf{1}}{q}\sqsupseteq\ & \textstyle\prod_{i=1}^{2} s_i[\mathsf{nil}]^{\nu_{s_i}} \mid \textstyle\prod_{i=1}^{2} n_i[\sigma.\mathsf{nil}]^{\nu_{n_i}} \mid n_3[\sigma.\mathsf{resnd}\langle v\rangle_p]^{\nu_{n_3}} \mid d[\sigma.\mathsf{fwd}_1]^{\nu_d} \\
\overset{\mathbf{1}}{1}\sqsupseteq\ & \textstyle\prod_{i=1}^{2} s_i[\mathsf{nil}]^{\nu_{s_i}} \mid \textstyle\prod_{i=1}^{2} n_i[\mathsf{nil}]^{\nu_{n_i}} \mid n_3[\sigma^2.\mathsf{snd}\langle v\rangle_p]^{\nu_{n_3}} \mid d[\sigma^2.\mathsf{fwd}_1]^{\nu_d} \\
\overset{\mathbf{1}}{p}\sqsupseteq\ & \textstyle\prod_{i=1}^{2} s_i[\mathsf{nil}]^{\nu_{s_i}} \mid \textstyle\prod_{i=1}^{2} n_i[\mathsf{nil}]^{\nu_{n_i}} \mid n_3[\sigma^2.\mathsf{nil}]^{\nu_{n_3}} \mid d[\sigma^2.\mathsf{resnd}\langle v\rangle_1]^{\nu_d} \\
\overset{\mathbf{1}}{1}\sqsupseteq\ & \textstyle\prod_{i=1}^{2} s_i[\mathsf{nil}]^{\nu_{s_i}} \mid \textstyle\prod_{i=1}^{3} n_i[\mathsf{nil}]^{\nu_{n_i}} \mid d[\sigma^3.\mathsf{snd}\langle v\rangle_1]^{\nu_d}.
\end{aligned}
$$

By Proposition 8 it follows that

$$
\begin{aligned}
& s_1[\mathsf{snd}\langle v\rangle_p]^{\nu_{s_1}} \mid s_2[\mathsf{nil}]^{\nu_{s_2}} \mid \textstyle\prod_{i=1}^{2} n_i[\mathsf{resnd}\langle v\rangle_p]^{\nu_{n_i}} \mid n_3[\mathsf{fwd}_p]^{\nu_{n_3}} \mid d[\mathsf{fwd}_1]^{\nu_d} \\
\overset{\mathbf{1}}{p^2(2-p)}\sqsupseteq\ & \textstyle\prod_{i=1}^{2} s_i[\mathsf{nil}]^{\nu_{s_i}} \mid \textstyle\prod_{i=1}^{3} n_i[\mathsf{nil}]^{\nu_{n_i}} \mid d[\sigma^3.\mathsf{snd}\langle v\rangle_1]^{\nu_d}\ .
\end{aligned}
$$

Similarly, by applying in sequence, Theorem 3, Theorem 2(5), Theorem 3 together with Theorem 2(2), Theorem 2(5) together Theorem 2(1), again Theorem 3 together with Theorem 2(2), and finally Theorem 2(6) together with Theorem 2(1) we get:

$$
\begin{aligned}
& s_1[\mathsf{snd}\langle v\rangle_p]^{\nu_{s_1}} \mid s_2[\mathsf{nil}]^{\nu_{s_2}} \mid \textstyle\prod_{i=1}^{3} n_i[\mathsf{fwd}_p]^{\nu_{n_i}} \mid d[\mathsf{fwd}_1]^{\nu_d} \\
\overset{\mathbf{1}}{p}\sqsupseteq\ & \textstyle\prod_{i=1}^{2} s_i[\mathsf{nil}]^{\nu_{s_i}} \mid n_1[\mathsf{resnd}\langle v\rangle_p]^{\nu_{n_1}} \mid \textstyle\prod_{i=2}^{3} n_i[\mathsf{fwd}_p]^{\nu_{n_i}} \mid d[\mathsf{fwd}_1]^{\nu_d} \\
\overset{\mathbf{1}}{1}\sqsupseteq\ & \textstyle\prod_{i=1}^{2} s_i[\mathsf{nil}]^{\nu_{s_i}} \mid n_1[\sigma.\mathsf{snd}\langle v\rangle_p]^{\nu_{n_1}} \mid \textstyle\prod_{i=2}^{3} n_i[\sigma.\mathsf{fwd}_p]^{\nu_{n_i}} \mid d[\sigma.\mathsf{fwd}_1]^{\nu_d} \\
\overset{\mathbf{1}}{p}\sqsupseteq\ & \textstyle\prod_{i=1}^{2} s_i[\mathsf{nil}]^{\nu_{s_i}} \mid n_1[\sigma.\mathsf{nil}]^{\nu_{n_1}} \mid n_2[\sigma.\mathsf{fwd}_p]^{\nu_{n_2}} \mid n_3[\sigma.\mathsf{resnd}\langle v\rangle_p]^{\nu_{n_3}} \mid d[\sigma.\mathsf{fwd}_1]^{\nu_d} \\
\overset{\mathbf{1}}{1}\sqsupseteq\ & \textstyle\prod_{i=1}^{2} s_i[\mathsf{nil}]^{\nu_{s_i}} \mid n_1[\mathsf{nil}]^{\nu_{n_1}} \mid n_2[\sigma^2.\mathsf{fwd}_p]^{\nu_{n_2}} \mid n_3[\sigma^2.\mathsf{snd}\langle v\rangle_p]^{\nu_{n_3}} \mid d[\sigma^2.\mathsf{fwd}_1]^{\nu_d} \\
\overset{\mathbf{1}}{p}\sqsupseteq\ & \textstyle\prod_{i=1}^{2} s_i[\mathsf{nil}]^{\nu_{s_i}} \mid n_1[\mathsf{nil}]^{\nu_{n_1}} \mid n_2[\sigma^3.\mathsf{snd}\langle v\rangle_p]^{\nu_{n_2}} \mid n_3[\sigma^2.\mathsf{nil}]^{\nu_{n_3}} \mid d[\sigma^2.\mathsf{resnd}\langle v\rangle_1]^{\nu_d} \\
\overset{\mathbf{1}}{1}\sqsupseteq\ & \textstyle\prod_{i=1}^{2} s_i[\mathsf{nil}]^{\nu_{s_i}} \mid \textstyle\prod_{i=1}^{3} n_i[\mathsf{nil}]^{\nu_{n_i}} \mid d[\sigma^3.\mathsf{snd}\langle v\rangle_1]^{\nu_d}.
\end{aligned}
$$

By Proposition 8 it follows that

$$
\begin{aligned}
& s_1[\mathsf{snd}\langle v\rangle_p]^{\nu_{s_1}} \mid s_2[\mathsf{nil}]^{\nu_{s_2}} \mid \textstyle\prod_{i=1}^{3} n_i[\mathsf{fwd}_p]^{\nu_{n_i}} \mid d[\mathsf{fwd}_1]^{\nu_d} \quad \overset{\mathbf{1}}{p^3}\sqsupseteq \\
& \textstyle\prod_{i=1}^{2} s_i[\mathsf{nil}]^{\nu_{s_i}} \mid \textstyle\prod_{i=1}^{3} n_i[\mathsf{nil}]^{\nu_{n_i}} \mid d[\sigma^3.\mathsf{snd}\langle v\rangle_1]^{\nu_d}\ .
\end{aligned}
$$

Finally, we can apply Theorem 4 and Proposition 7 to derive:

$$
\mathrm{GSP}_1 \quad \overset{\mathbf{1}}{p^3(3-2p)}\sqsupseteq \quad \prod_{i=1}^{2} s_i[\mathsf{nil}]^{\nu_{s_i}} \mid \prod_{i=1}^{3} n_i[\mathsf{nil}]^{\nu_{n_i}} \mid d[\sigma^3.\mathsf{snd}\langle v\rangle_1]^{\nu_d}\ .
$$

This result essentially says that the gossip network GSP_1 succeeds in transmitting the message v to the destination d, after three rounds, with probability

(at least) $p^3(3-2p)$. Thus, for a gossip probability $p = 0.8$ the destination will receive the message with probability 0.72, with a margin of 10%. For $p = 0.85$ the probability at the destination increases to 0.8, with a margin of 6%; while for $p = 0.9$ the probability at destination rises to 0.88, with a difference of only 2%. So, $p = 0.9$ can be considered the threshold of our small network.[5]

5 Gossipping with Collisions

An important characteristic of the wireless domain is that transmissions are prone to collisions due to the well-known *hidden-terminal problem*. In the previous section we have reasoned assuming no collisions. In this section, we formally demonstrate that, as expected, the presence of communication collisions deteriorates the performances of gossip protocols.

A receiver node faces a collision if it is exposed to more than one transmission in the same round and as a result drops some of these transmissions. We can model this behaviour in pTCWS as follows:

$$\mathsf{resndc}\langle u\rangle_{p_{\mathrm{g}}} \stackrel{\text{def}}{=} \lfloor ?(x).\mathsf{nil} \rfloor \mathsf{snd}\langle u\rangle_{p_{\mathrm{g}}} \qquad\qquad \mathsf{fwdc}_{p_{\mathrm{g}}} \stackrel{\text{def}}{=} ?(x).\mathsf{resndc}\langle x\rangle_{p_{\mathrm{g}}} \ .$$

Here, the forwarder process waits for a message in the current instant of time. If it receives a second message in the same round then it is doomed to fail. Otherwise, it moves to the next round and broadcasts the received message with gossip probability p_{g}. Thus, for example, the first law of the previous section becomes:

$$m_1[\mathsf{snd}\langle v\rangle_{p_1}]^n \mid m_2[\mathsf{snd}\langle v\rangle_{p_2}]^n \mid n[\mathsf{fwdc}_q]^\nu \stackrel{\mathbf{1}}{p} \sqsupseteq m_1[\mathsf{nil}]^n \mid m_2[\mathsf{nil}]^n \mid n[\mathsf{resndc}\langle v\rangle_q]^\nu$$

with $p = p_1(1-p_2)+p_2(1-p_1)$ which is definitely smaller than $1-(1-p_1)(1-p_2)$, the lower bound seen in the previous section without collisions.

More generally, if collisions are taken into account Theorem 3 needs to be slightly changed as follows.

Theorem 5 (Message propagation with collision). *The same as Theorem 3 except for processes* fwd_{q_j} *and* $\mathsf{resnd}\langle v\rangle_{q_j}$ *which are replaced by* fwdc_{q_j} *and* $\mathsf{resndc}\langle v\rangle_{q_j}$*, respectively; the probability* r *is* $\sum_{i\in I} p_i \prod_{j\in I\setminus\{i\}}(1-p_j)$.

Here, the probability changes with respect to Theorem 3 because a forwarder successfully receives the value v only if exactly one sender transmits.

Let us apply our theorems to compute the probability of successful gossipping in the presence of collisions. Let us define:

$$\mathrm{GSP}_2 \stackrel{\text{def}}{=} \prod_{i=1}^{2} s_i[\mathsf{snd}\langle v\rangle_p]^{\nu_{s_i}} \mid \prod_{i=1}^{3} n_i[\mathsf{fwdc}_p]^{\nu_{n_i}} \mid d[\mathsf{fwdc}_1]^{\nu_d}$$

with the same network topology as GSP_1.

[5] Had we considered a larger network, with more senders, we would have obtained a more significant threshold.

By applying Theorem 5 and Theorem 2 to compute estimates, Theorem 4 to compose such estimates, and Proposition 7, we obtain:

$$\mathrm{GSP}_2 \quad {}_q\sqsupseteq \quad \prod_{i=1}^{2} s_i[\mathsf{nil}]^{\nu_{s_i}} \mid \prod_{i=1}^{3} n_i[\mathsf{nil}]^{\nu_{n_i}} \mid d[\sigma^3.\mathsf{snd}\langle v\rangle_1]^{\nu_d}$$

with $q = p(2p^2(1-p)^2 + p^3) + (1-p)p^3 = p^3(3 - 4p + 2p^2)$. This probability is definitely smaller than that computed for GSP_1, demonstrating that collisions degrade the performances of gossip protocols. Thus, for instance, for a gossip probability $p = 0.8$ the destination in GSP_2 will receive the message with probability 0.55 while in GSP_1 this probability is 0.72; similarly for $p = 0.9$ the probability of success in GSP_2 is about 0.74 while in GSP_1 it is 0.88.

6 Conclusions, Future and Related Work

We have proposed a probabilistic simulation theory to compare the performances of gossip protocols for wireless sensor networks. This theory is used to prove a number of algebraic laws which revealed to be very effective to evaluate the performances of gossip networks with and without communication collisions. Our simulation theory provides lower-bound probabilities. However, due to the inherent structure of gossip networks, the probabilities of our algebraic laws are actually *precise* (see [12] for details). As future work, we will study gossip networks with *random delays* and *lossy channels*. Moreover, we intend to mechanise the application of our laws to deal with large-scale gossip networks.

A nice survey of formal verification techniques for the analysis of gossip protocols is presented in [2]. Probabilistic model-checking has been used in [6] to study the influence of different modelling choices on message propagation in flooding and gossip protocols. It has been used also in [10] to investigate the expected rounds of gossipping required to form a connected network. However, the analysis of gossip protocols in large-scale networks remains beyond the capabilities of current probabilistic model-checking tools. For this reason, the paper [3] suggests to apply mean-field analysis for a formal evaluation of gossip protocols.

Several process calculi for wireless systems have been proposed in the last five years. Our calculus is a probabilistic variant of [4] which takes inspiration from [5,11]. The paper [17] contains the first probabilistic untimed calculus for wireless systems, where connections are established with a given probability.

Our notion of simulation up to probability may remind one of the idea of simulation with a fixed precision. A first version of probabilistic bisimulation with ϵ precision appeared in [1] to relax security constrains. Indeed their simulation is able to tolerate local fluctuations by allowing small differences in the probability of occurrence of weak actions. In [13] a theory of approximate equivalence for task-structured Probabilistic I/O Automata is proposed. In this case, the distance between probabilities is based on trace distributions. Afterwards, in order to study simulations in cryptographic protocols [16] proposed a notion of simulation where distances may grow by a negligible value at each step.

Acknowledgements. The referees provided useful suggestions.

References

1. Aldini, A., Bravetti, M., Gorrieri, R.: A process-algebraic approach for the analysis of probabilistic noninterference. Journal of Computer Security 12, 191–245 (2004)
2. Bakhshi, R., Bonnet, F., Fokkink, W., Haverkort, B.: Formal analysis techniques for gossiping protocols. Operating Systems Review 41(5), 28–36 (2007)
3. Bakhshi, R., Cloth, L., Fokkink, W., Haverkort, B.: Mean-field analysis for the evaluation of gossip protocols. In: Huth, M., Nicol, D. (eds.) QEST, pp. 247–256. IEEE Computer Society, Budapest (2009)
4. Ballardin, F., Merro, M.: A calculus for the analysis of wireless network security protocols. In: Degano, P., Etalle, S., Guttman, J. (eds.) FAST 2010. LNCS, vol. 6561, pp. 206–222. Springer, Heidelberg (2011)
5. Deng, Y., van Glabbeek, R., Hennessy, M., Morgan, C.: Characterising testing preorders for finite probabilistic processes. Logical Methods in Computer Science 4(4) (2008)
6. Fehnker, A., Gao, P.: Formal verification and simulation for performance analysis for probabilistic broadcast protocols. In: Kunz, T., Ravi, S. (eds.) ADHOC-NOW 2006. LNCS, vol. 4104, pp. 128–141. Springer, Heidelberg (2006)
7. Hennessy, M., Regan, T.: A process algebra for timed systems. Information and Computation 117(2), 221–239 (1995)
8. Jonsson, B., Ho-Stuart, C., Yi, W.: Testing and refinement for nondeterministic and probabilistic processes. In: Langmaack, H., de Roever, W., Vytopil, J. (eds.) FTRTFT 1994 and ProCoS 1994. LNCS, vol. 863, pp. 418–430. Springer, Heidelberg (1994)
9. Kermarrec, A., van Steen, M.: Gossiping in distributed systems. Operating Systems Review 41(5), 2–7 (2007)
10. Kwiatkowska, M., Norman, G., Parker, D.: Analysis of a gossip protocol in prism. SIGMETRICS Performance Evaluation Review 36(3), 17–22 (2008)
11. Kwiatkowska, M., Norman, G., Parker, D., Vigliotti, G.M.: Probabilistic mobile ambients. Theoretical Computuer Science 410(12-13), 1272–1303 (2009)
12. Lanotte, R., Merro, M.: Semantic analysis of gossip protocols for wireless sensor networks, Forthcoming Technical Report, Dept. Computer Science, Verona (2011)
13. Mitra, S., Lynch, N.: Proving approximate implementations for probabilistic I/O automata. Electronic Notes in Theoret. Comput. Science 174(8), 71–93 (2007)
14. Nicollin, X., Sifakis, J.: An overview and synthesis on timed process algebras. In: Larsen, K.G., Skou, A. (eds.) CAV 1991. LNCS, vol. 575, pp. 376–398. Springer, Heidelberg (1992)
15. Segala, R.: Modeling and Verification of Randomized Distributed Real-Time Systems. Ph.D. thesis, Laboratory for Computer Science, MIT (1995)
16. Segala, R., Turrini, A.: Approximated computationally bounded simulation relations for probabilistic automata. In: Sabelfeld, A. (ed.) CSF, pp. 140–156. IEEE Computer Society, Venice (2007)
17. Song, L., Godskesen, J.: Probabilistic mobility models for mobile and wireless networks. In: Calude, C.S., Sassone, V. (eds.) TCS 2010. IFIP AICT, vol. 323, pp. 86–100. Springer, Heidelberg (2010)