Proc. of *FCT'09*, LNCS 5966, pages 50–61. © Springer, 2009.

Depletable Channels: Dynamics and Behaviour

Pietro Cenciarelli, Daniele Gorla and Ivano Salvo

University of Rome, "La Sapienza", Dpt. of Computer Science
{cenciarelli,gorla,salvo}@di.uniroma1.it

Abstract A simple model of multi-hop communication in ad-hoc networks is considered. Similar models are often adopted for studying energy efficiency and load balancing of different routing protocols. We address an orthogonal question never considered by the networking community: whether, regardless of specific protocols, two networks may be considered as equivalent from the viewpoint of the communication service they provide. In particular, we consider equivalent two networks with identical maximum and minimum inhibiting flow, and prove that this notion of equivalence coincides with a standard trace-based notion of equivalence borrowed from the theory of concurrency. We finally study the computational complexity of the proposed equivalence and discuss possible alternatives.

1 Introduction

In recent years, much attention has been devoted to research in the area of ad hoc networking. Many complex theoretical problems are at stake and a variety of efficient routing protocols have been studied for exchanging information across a network without using centralized control [25,24,6].

Ad hoc networks are typically wireless, and *multi-hop* communication is adopted because of limited wireless transmission range. Moreover, they usually exhibit *dynamic* behaviour in that their topology may vary over time as a result of mobility or resource consumption. In particular, a crucial kind of resource in most sensor network applications is *energy* [3,4].

In this paper we study the dynamics of ad hoc communication in a rather simple, and yet significant network model. Dynamics is meant in the sense of *change of state* and is induced by energy consumption. Similar models have been adopted for studying energy efficiency and load balancing of different routing protocols [10,18]. Here we address an orthogonal question which has not received attention in the literature on computer networks as yet: whether, regardless of specific protocols, two networks may be considered as *equivalent* from the viewpoint of the communication service they provide.

In our framework, a network is a (possibly cyclic) oriented graph equipped with a function associating with each node a non-negative integer representing *depletable charge*. We are interested in networks as channels for transmitting information. Thus, we consider *communication channels*, i.e. networks with a

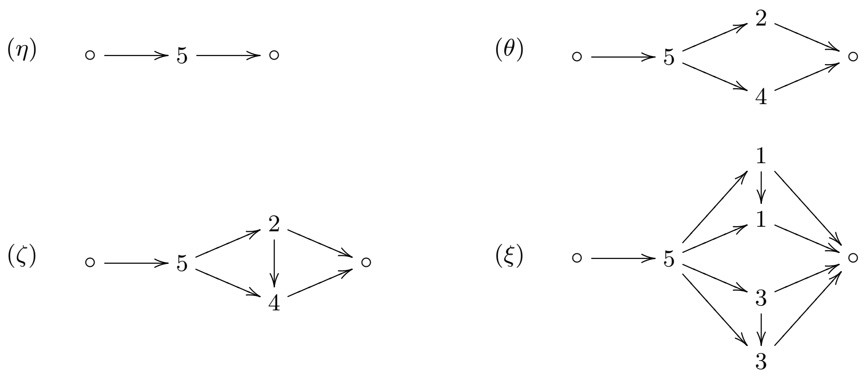
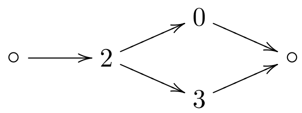


Figure 1. Four channels

chosen pair of nodes called *source* and *target*. At a given time a number of atomic items are fed to the source and instantly flow to the target. Charges may change as result of information passing through the net. Each item passing through a node consumes one unit of the node's charge, thus leaving the channel in a state of lower energy.

In drawing channels, we let n stand for a node of charge n. Circles ($\circ$) are used to denote nodes whose charge is large enough to be irrelevant. Source and target are drawn respectively as the leftmost and rightmost node in the picture. Four channels are depicted in Figure 1. When three items are transmitted along channel θ, node charges may change as shown below.



In particular, this result is obtained by routing two items along the northern path, $\circ\, 5\, 2\, \circ$, and one along the southern, $\circ\, 5\, 4\, \circ$. In this new state the channel is still capable of transmitting two more items, after which the channel is *dead*, i.e. any transmission from source to target is inhibited by some exhausted node. Different routings of three items are also possible in θ: for example, letting them all pass south. On the other hand, not all of them may choose the northern path, as its capacity is limited to 2 by the upper node. For the same reason θ can only transmit up to five items, which is the value of its minimum cut. Indeed, all four channels in the picture support a maximum flow of 5; but are they all *equivalent*?

In our model, where fault tolerance is not at stake, we may intuitively agree that η and θ are indistinguishable in the source-to-target communication service they provide. For example, we could view η as the *specification* of a communication service and θ as a possible implementation. However, would ζ implement η correctly? We argue that the two channels may *behave* differently: while η and θ are always alive after *any* transmission of four items, not so for ζ, where the 4-valued flow sending two items along the path $\circ\, 5\, 2\, 4\, \circ$ and two along $\circ\, 5\, 4\, \circ$

yields a dead channel. Similarly, channel ξ may be killed by a flow of just four items. Then again: can ζ and ξ be considered as different implementations of the same communication service?

The present paper moves a first step towards a formal study of energy-sensitive network behaviour. We study a natural notion of network equivalence which equates η with θ of Figure 1, but not with ζ and ξ. In particular, we shall equate two nets with identical maximum flow and minimum inhibiting flow (i.e., the minimum number of items whose transmission leads to a dead channel). Such an equivalence has a well known corresponding notion in the theory of concurrency, i.e. it corresponds to *(complete) trace equivalence* [1] built up over the labeled transition system arising from all the possible transmissions a channel can be engaged in. More refined notions of behavioural equivalence are studied in concurrency; most notably *bisimulation* [19,16]. We show that, in spite of its simplicity, our model exhibits a variety of natural notions of behavioural equivalences, whose richness is comparable with that of process calculi. In particular, we shall reveal in Section 5 that, although trace equivalent, ζ and ξ in Figure 1 do exhibit different behaviour and can in fact be distinguished in terms of bisimulation.

We believe that a theory of behavioural equivalence relating different network topologies and charge distributions may provide guidance in solving optimization problems and a better understanding of protocol properties, such as invariance with respect to sameness of behaviour.

The paper is structured as follows. First, we present our model of energy-sensitive network channels in Section 2, by describing a simple graph-based model and the associated dynamic behaviours expressed in terms of labeled transitions from a channel to another one. Then, in Section 3, we define a notion of channel equivalence by means of *intrinsic* features of channels; we then relate such an equivalence to a standard trace-based equivalence built up over the transitions previously defined. This is the main result of the paper. Section 4 tackles complexity issues and shows that trace equivalence is a computationally hard problem, even when restricting to acyclic networks. Section 5 relates trace equivalence with bisimulation. Section 6 discusses related work and draws conclusions. Because of limited space, proofs do not include full details; full proofs can be found in the on-line technical report available at `http://www.dsi.uniroma1.it/~gorla/papers/CGS-nets-full.pdf`.

2 The Model

An *oriented graph* (V, E) consists of a set V of *vertices* (or *nodes*) and a set $E \subseteq V \times V$ of ordered pairs of vertices, called *edges*. A *walk* is a sequence $u_1 \dots u_n$ of nodes such that (u_i, u_{i+1}) is an edge, for all $i < n$. We call *network* a finite oriented graph equipped with a function η associating with each node a non-negative integer representing *depletable charge*. We write just η to denote a network (V, E, η) when its underlying graph, called *topology*, is understood.

We study networks as means to transmit information. Once fixed a *source* node, written s, and a *target* node, written t, we call the network a *communication channel* (*channel* for short). A *path* in a channel is a (possibly cyclic) directed walk from s to t. The set of paths of a channel η is denoted by $P(\eta)$.

Paths are often defined in the literature as *acyclic* walks, and path-oriented definitions of *flow* associate numerical values separately to paths *and* cycles [2, Section 3.5]. In our framework, this would amount to allowing *spontaneous* flow, not originating in the source and depleting the network by cycling in it without ever reaching the target. Since we are interested in modeling information traveling the net as result of a communication act by s, we restrict our attention to flows *from* s *to* t. Formally: let r_{vp} denote the number of times in which a node v is repeated in a path p (zero if $v \notin p$); A *flow* for η is a function $\phi : P(\eta) \to \mathbb{N}$ such that $\phi(v) \leq \eta(v)$, for every $v \in V$, where $\phi(v)$ denotes the amount of v's charge consumed by ϕ, that is $\phi(v) = \sum_{p \in P(\eta)} r_{vp} \cdot \phi(p)$. The *value* of ϕ is $\sum_{p \in P(\eta)} \phi(p)$. We denote by max_η the value of the maximum flow for η. We call η a *dead* channel if $max_\eta = 0$.

To capture the notion of channel dynamics, we introduce a labeled *transition* relation $\xrightarrow{\cdot}$ over channels of identical topology, where $(V, E, \eta) \xrightarrow{n} (V, E, \theta)$ is defined to hold when there exists a flow ϕ of value n in η such that $\theta(v) = \eta(v) - \phi(v)$ for all nodes v. The flow ϕ is said to *witness* the transition. A transition $\eta \xrightarrow{n} \theta$, and likewise any witness of it, is said to *inhibit* η if θ is dead. We denote by min_η the smallest value of an inhibiting flow in η.

To conclude, we now give two simple properties of the labeled transition system just defined, namely *composition* and *decomposition* of transitions.

Proposition 1. *If* $\eta \xrightarrow{n} \theta \xrightarrow{m} \zeta$ *are transitions, so is also* $\eta \xrightarrow{n+m} \zeta$.

Proof. Let ϕ and ψ witness the two transitions above. It is easy to check that the function assigning $\phi(p) + \psi(p)$ to each path p of η is indeed a flow of value $n + m$. □

Proposition 2. *If* $\eta \xrightarrow{n+m} \zeta$ *is a transition, so are* $\eta \xrightarrow{n} \theta \xrightarrow{m} \zeta$, *for some* θ.

Proof. It is sufficient to show it for $m = 1$; the general result follows from Proposition 1. Let ϕ witness the $n + 1$ transition and let p be a path with $\phi(p) \geq 1$. The function assigning $\phi(q)$ to all paths $q \neq p$ and $\phi(p) - 1$ to p is clearly a flow witnessing a transition $\eta \xrightarrow{n} \theta$, while $\theta \xrightarrow{1} \zeta$ is obtained by the flow assigning 1 to p and 0 to all other paths. □

3 Behavioural Equivalence

Two channels may be indistinguishable in the service they provide; such are η and θ of Figure 1. This statement can be made precise by equipping our model with a notion of channel *behaviour*, so that channels exhibiting identical behaviour may be considered as different implementations of the same communication service. To that effect, we first identify the *observations* an external user is allowed to make on a channel. This establishes the level of abstraction at which channels may be distinguished.

The very first attempt one can do in this direction is to equate all channels with the same maximum flow. In this way, we would equate two channels by only considering an intrinsic (or structural) property of the equated channels, without looking at their dynamic behaviour that arises from the transitions defined for our model. However, it is possible to bridge the structural view put forward by the maximum flow and the dynamic behaviour arising from the transitions. Indeed, as a first theoretical result of this paper, we prove that this structural property of the channel has a well-known counterpart in concurrency theory: it corresponds to what is usually called *general trace equivalence* for labeled transition systems [1]. By straightforwardly adapting the standard definitions to our framework, a *general trace* for a channel η is a sequence $\langle n_1 \dots n_k \rangle$ such that there exist transitions $\eta_0 \xrightarrow{n_1} \eta_1 \dots \xrightarrow{n_k} \eta_k$ where $\eta_0 = \eta$.

Lemma 1. *For every η and $n \leq max_\eta$, there exists η' such that $\eta \xrightarrow{n} \eta'$.*

Theorem 1. *Two channels have identical maximum flow if and only if they have identical sets of general traces.*

Proof. (If) By contradiction: assume, e.g., that $max_\eta < max_\zeta = n$. Then, there exists ζ' such that $\zeta \xrightarrow{n} \zeta'$. However, there exists no η' such that $\eta \xrightarrow{n} \eta'$; contradiction.

(Only if) Let $max_\eta = max_\zeta$ and let $\langle n_1 \dots n_k \rangle$ be a general trace of η. By Proposition 1, $\eta \xrightarrow{n} \eta'$, for some η' and $n = n_1 + \dots + n_k$. Since $n \leq max_\eta = max_\zeta$, by Lemma 1 $\zeta \xrightarrow{n} \zeta'$, for some ζ'. By Proposition 2, we conclude that $\langle n_1 \dots n_k \rangle$ is a general trace of ζ. □

In this way, we would equate all the channels in Figure 1: they all have a maximum flow of value 5. In particular, every net η with $n = max_\eta$ is equivalent to the net

$$\circ \rightarrow n \rightarrow \circ$$

However, as noticed in the introduction, ζ and ξ can be distinguished from η and θ by observing death. Since users *do* notice when channels are dead, we seek a more refined notion of equivalence capable of distinguishing ζ and ξ from η and θ.

To this aim, we can also consider the smallest value of an inhibiting flow, viz. min_η. We can now equate two channels that have the same maximum and minimum inhibiting flow value. In this way, channels η and θ of Figure 1 would be equated (since $max_\eta = max_\theta = min_\eta = min_\theta = 5$), channels ζ and ξ would be equated (since $max_\zeta = max_\xi = 5$ and $min_\zeta = min_\xi = 4$), but the last two ones would not be equivalent to the first two ones, as desired.

Also in this case, this refined notion of equivalence has a well-known counterpart in concurrency theory: it corresponds to what is usually called *(complete) trace equivalence* [1]. A *complete trace* (or, simply, a trace) for a channel η is a sequence $\langle n_1 \dots n_k \rangle$ such that there exist transitions $\eta_0 \xrightarrow{n_1} \eta_1 \dots \xrightarrow{n_k} \eta_k$ where

$\eta_0 = \eta$ and η_k is dead. We denote by $tr(\eta)$ the set of complete traces of a channel η. Two channels are *complete trace equivalent* (or, simply, trace equivalent) if they have identical sets of complete traces.

To prove this characterization (that is the main theoretical result of our paper), we use some classical definitions and results from the theory of network flows (e.g., residual net and augmenting path); we refer the reader to [2,8] or to our on-line full version of this paper for all the details. A *cut* of a channel is a subset S of the vertices such that $s \in S$ and $t \notin S$. We denote by $S^{\rightarrow}$ the set of edges (u, v) such that $u \in S$ and $v \notin S$. We write $u \overset{p}{\rightsquigarrow} v$ to specify that the first and last nodes of a walk p are u and v respectively; $u \rightsquigarrow v$ denotes such a walk when the name p is not relevant. If p is a walk of the form $u \rightsquigarrow v \rightsquigarrow v' \rightsquigarrow w$, we denote by $v \overset{p}{\rightsquigarrow} v'$ the portion of p from v to v'. Given a node u and a set K of edges, we write $u \triangleleft K$ to mean that every path $u \rightsquigarrow t$ includes at least one edge of K.

Lemma 2. *Let η be a channel and ϕ an inhibiting flow of value $n < max_\eta$; then, there exists an inhibiting flow of value $n + 1$.*

Proof. To prove this result, we find it useful to work in a framework where values are associated with edges and flows are expressed by assigning a flow to every edge (and not to every path). Graphs where vertices are weighted can be easily transformed in graphs where edges are weighted by applying a well-known *node splitting* technique [2, Section 2.4]. Moreover, the edge-oriented presentation of flows is less abstract than the path-oriented one, in that there may be more path-oriented flows corresponding to one edge-oriented [2, Theorem 3.5].

Since ϕ inhibits η, we have that ϕ saturates at least one cut of η, i.e. $\phi(e) = \eta(e)$, for every $e \in S^{\rightarrow}$; let us consider all such cuts and let S be a maximal cut (w.r.t. to '$\subseteq$'). Since the value of ϕ is smaller than max_η, by standard results [2] there exists an augmenting path for η after ϕ.

We now prove that there exists an augmenting path p' that crosses S exactly once, where an augmenting path *crosses* a cut if it includes at least one edge (u, v) such that, within η, it holds that $u \triangleleft S^{\rightarrow}$ and $v \not\triangleleft S^{\rightarrow}$. It is easy to show that every augmenting path crosses S at least once. Let us fix one of them, say p, and let (u, v) be the first edge in p that crosses S. There must be a path $v \overset{q}{\rightsquigarrow} t$ in η after ϕ, otherwise S would not be maximal. Indeed, we can prove the following

Technical lemma: Let η be a channel and ϕ a flow that saturates one of its cuts S. Assume that there exists a $v \notin S$ and a non-empty $K \subseteq E$ such that $v \triangleleft K$ and all the edges in K are saturated by ϕ. Then, there exists a cut of η greater than S still saturated by ϕ.

Hence, we have that $p' \triangleq s \overset{p}{\rightsquigarrow} u,v \overset{q}{\rightsquigarrow} t$ is an augmenting path with exactly one edge crossing S, viz. (u, v).

Let ϕ' be the flow obtained by updating ϕ with p' as follows:

$$\phi'(u, v) \triangleq \begin{cases} \phi(u, v) + 1 & \text{if } (u, v) \in p' \\ \phi(u, v) - 1 & \text{if } (v, u) \in p' \\ \phi(u, v) & \text{otherwise} \end{cases}$$

It can be proved [8] that ϕ' is a flow for η of value $n+1$; if we prove that ϕ' inhibits the channel, we have done. To this aim, it suffices to prove that it saturates S. If it was not the case, p' would include (v,u), for some $(u,v) \in S^{\rightarrow}$. Since $u \in S$, $u \lhd S^{\rightarrow}$; hence, (v,u) cannot be the edge of p' that crosses S. Then, it can either be $v \lhd S^{\rightarrow}$ or $v \not\lhd S^{\rightarrow}$; however, both these possibilities lead to a contradiction:

$v \not\lhd S^{\rightarrow}$: since $s \lhd S^{\rightarrow}$, $v \not\lhd S^{\rightarrow}$ implies that there must be an edge crossing S *before* (v,u) in p'; since $t \not\lhd S^{\rightarrow}$, $u \lhd S^{\rightarrow}$ implies that there must be an edge crossing S *after* (v,u) in p'; since p' has only one edge crossing S, this case is not possible.

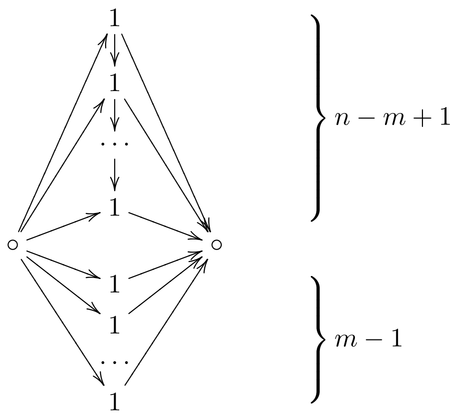
$v \lhd S^{\rightarrow}$: in this case, by the technical lemma, we could exhibit a cut saturated by ϕ greater than S. □

Theorem 2. *Two channels are trace equivalent if and only if they have identical maximum and minimum inhibiting flow.*

Proof. (If) Let $\langle n_1 \ldots n_k \rangle \in tr(\eta)$; because of Proposition 1, there exists an inhibiting transition for η with value $n = n_1 + \ldots + n_k$. If $n \in \{min_\eta, max_\eta\}$, by hypothesis we have that there exists an inhibiting transition for θ with value n; otherwise, we can start from a minimum inhibiting flow and use Lemma 2 for $n - min_\eta$ times to obtain an inhibiting flow for θ with value n. In both cases, by Proposition 2 we have that $\langle n_1 \ldots n_k \rangle \in tr(\theta)$, as desired.

(Only if) Let us consider the traces $\langle min_\eta \rangle$ and $\langle max_\eta \rangle$, both belonging to $tr(\eta)$; by hypothesis, they also belong to $tr(\theta)$. If by contradiction were $max_\theta > max_\eta$ (it cannot be '$<$' because $\langle max_\eta \rangle \in tr(\theta)$), we would have that $\langle max_\theta \rangle \in tr(\theta)$ but $\langle max_\theta \rangle \notin tr(\eta)$, in contradiction with $tr(\eta) = tr(\theta)$. If $min_\theta < min_\eta$, the proof is similar. Thus, $max_\theta = max_\eta$ and $min_\theta = min_\eta$, as desired. □

Notice that every channel is trace equivalent to a channel, that we call *canonical*, with a very simple topology (in particular, it has no cycles). Let us define the channel $\gamma_{m,n}$ as:



It is easy to check that $min_{\gamma_{m,n}} = m$ and $max_{\gamma_{m,n}} = n$. Thus, $\gamma_{m,n}$ can be considered the standard representative of the trace-equivalence class of all the nets with minimum inhibiting flow m and maximum flow n.

4 Complexity Issues

Theorem 2 characterizes trace equivalence in terms of maximum flow and minimum inhibiting flow. It is well-known that there exist polynomial time algorithms for finding the maximum flow in a net. We are left with studying the complexity of the following problem, that we call *minimum inhibiting flow* (MIF, for short):

MIF: Given a network η, find the value of the minimum inhibiting flow for η.

MIF can be turned into a decisional problem:

DMIF: Given a network η and an integer k, is there an inhibiting flow for η with value at most k?

Theorem 3. *MIF is NP-complete.*

Proof. Clearly, DMIF is in NP; by standard techniques, we can exploit this fact to also prove that MIF is in NP.

To show that MIF is NP-hard, we reduce the problem of finding a maximal matching of a given cardinality in a bipartite graph to DMIF. We recall that a maximal matching in a graph (V, E) is a set of edges $F \subseteq E$ such that:

- $\forall e, e' \in F$ it holds that $e \cap e' = \varnothing$;
- $\forall e \in E \exists e' \in F$ such that $e \cap e' \neq \varnothing$.

Let $G = (V_1, V_2, E)$ be a bipartite undirected graph. We consider the channel (V', E', η), where

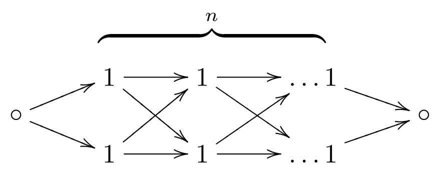
- $V' = V_1 \cup V_2 \cup \{s, t\}$, for $\{s, t\} \cap (V_1 \cup V_2) = \varnothing$;
- $E' = \{(u, v) : \{u, v\} \in E \wedge u \in V_1 \wedge v \in V_2\} \cup \bigcup_{u \in V_1} \{(s, u)\} \cup \bigcup_{u \in V_2} \{(u, t)\}$;
- $\eta(v) = 1$ for every $v \in V_1 \cup V_2$.

It is easy to show that G has a maximal matching of cardinality k if and only if η has an inhibiting flow of value k. $\square$

We observe that, in the reduction just shown, we need to consider acyclic unitary networks only, i.e. networks in which depletable charge of each node is 1. This implies that MIF is an intractable problem even in this restricted case.

It is now worth noting that in concurrency theory complexity measures are usually expressed in terms of the size of the labeled transition system (LTS, for short) resulting from all the labeled transitions of a given process (in our case, a channel). This is because the definitions and characterizations of process equivalences are usually given on the LTSs of the equated processes, and not on the processes themselves. Even for simple models like CCS, trace equivalence is exponential in the size of the LTS [23], while other equivalences (like, e.g., *bisimilarity* [19,16]) are polynomial [12]. However, if expressed in terms of the size of the process, all these equivalences become (at least) exponential, since the number of states of a LTS is exponential in the size of its originating process.

Thanks to Theorem 2, we could have directly checked equivalences on the LTSs resulting from the equated channels. However, also in our case we would have an exponential blow up of the number of states. For example, consider the channel:

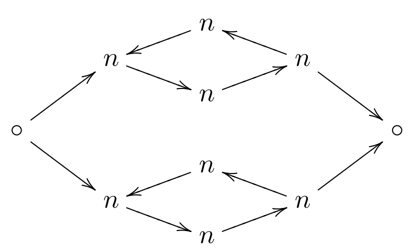


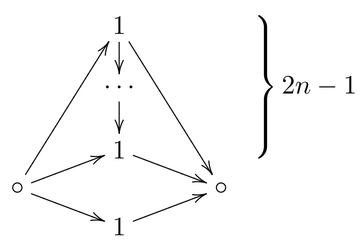
It has $2n + 2$ vertices and it produces a LTS with 2^n states: there are 2^n paths along which a unitary flow can be sent.

We have instead defined our behavioural equivalence by relying on properties of the equated channels, and not of their LTSs. Nevertheless, as we have just shown, trace equivalence seems not verifiable in polynomial time (w.r.t. the size of the equated channels); this should not be surprising. On one hand, this agrees with the usual hardness of trace equivalence in concurrency theory mentioned above; on the other hand, this stimulates future work on more efficiently verifiable, but still properly discriminating, equivalences.

5 Beyond Trace Equivalence

To conclude our presentation of trace equivalence, let us pinpoint some of its limitations; such issues are standard in concurrency theory and scales to our model too. The main issue is that trace equivalence is not preserved by transitions. Indeed, consider the nets η and the canonical net $\gamma_{2,2n}$, for any $n > 1$:





Clearly, they are trace equivalent. Then, consider the unitary transition of η that inhibits its norther cycle by taking n times such a cycle; it turns η into the channel η' having 0 on every node of its northern cycle. There is no unitary transition of $\gamma_{2,2n}$ that leads to a channel that is trace equivalent to η'. Indeed, if the unitary flow passes through the bottom vertex, the resulting channel has a maximum flow of $2n-1$ (whereas $max_{\eta'} = n$); if the flow passes through (some of) the top $2n-1$ vertices, the resulting channel has a minimum inhibiting flow greater than 1 (whereas $min_{\eta'} = 1$).

In concurrency theory, a classical notion of equivalence that is more finely grained than those based on traces relies on the notion of *bisimulation*. In our framework, this a symmetric relation $\Re$ on channels such that $\eta \Re \theta$ and $\eta \xrightarrow{n} \eta'$ imply $\theta \xrightarrow{n} \theta'$, for some θ' such that $\eta' \Re \theta'$. Two channels η and θ are called *bisimulation equivalent* if they are related by a bisimulation. In view of Theorem 2, it follows immediately from the definition that bisimulation equivalence is included in trace equivalence. Moreover, by what we have just said, the inclusion is strict: the channels η and $\gamma_{2,2n}$ depicted above are *not* bisimulation equivalent. Another example is given by channels ζ and ξ of Figure 1: after sending two items along $\circ\, 5\, 2\, 4\, \circ$ in ζ we have a net with maximum flow at 2; on the contrary, every 2-valued flow in ξ yields a channel with maximum flow greater than 2.

A challenging issue for future work is finding a characterization of bisimulation equivalence in terms of structural properties of channels, in the same spirit as the characterizations we have provided for trace equivalence in this paper.

6 Conclusions and Related Work

We presented a simple model of communication networks, called channels. The communication infrastructure is modeled by a graph connecting a sender s with a receiver t. Nodes have a depletable charge. Labeled transitions are used to describe the dynamics of channels, where states of the LTS are channels of identical topology and labels are the number of information units transmitted in a communication from s to t via a legal network flow. We equated channels by means of intrinsic channel properties (that is, their maximum flow and minimum inhibiting flow) and studied their complexity. Finally, we showed that such equivalence coincides with a natural notion of equivalence borrowed from concurrency theory.

There are several research lines that can be pursued to develop the framework presented in this paper. First of all, we assume that source and target are *fixed* during a channel evolution. More realistic models include scenarios where only the target is fixed (e.g., sensor networks) or where both source and target can be any node of the net. Moreover, our model assumes that the network topology does never change during the computation. This is clearly a simplifying assumption and makes our model unsuited for MANETs. It would be challenging to introduce in the model such advanced features and study the resulting equivalences.

Related work. In the last years, network scenarios have been modeled and studied by means of process algebraic techniques. In such papers, the authors usually first give a syntax for writing nets, featuring some distinguishing issues of the modeled applications; then, they give an operational semantics and a behavioural equivalence to reason over nets; finally, the theory is used in some concrete application, e.g. to verify the correctness of some network protocol or to equate different networks with the same behaviour. According to the kind of network modeled, we mention: [11,13,22], where mobile ad hoc networks are considered; [15,17,14], where wireless systems are considered; [5], where peer-to-peer overlay networks are considered. Our approach clearly follows this research line. However, we do not have a process syntax and just write networks via their physical topology, assuming that some suitable software is hardcoded into every node of the net to properly implement some routing strategy. A somehow similar approach has been followed by some of the authors in a previous paper [7], where the framework was based on (hyper)graph rewriting. There, apart from functional equivalence, other network measures (e.g., robustness) were related to bisimulation in the model.

It is worth saying that our MIF problem somehow resembles the Network Inhibition Problem (NIP) [20]. There, every edge of a flow net is equipped with a destruction cost; the problem is to find a flow that leaves the net in the worst possible condition (i.e., with the minimum max flow) and whose cost is smaller than a given quantity. In *loc. cit.*, it is proved that NIP is NP-complete for several class of graphs, but polynomially approximable for most of them (e.g., planar or grid).

A related paper is [21], where a network model (somehow similar to ours) is used to study the complexity of finding optimal flow subnetworks. A challenging issue for future research is the understanding of how the two approaches relate to each other.

To conclude, we have proposed a usage of formal models different from those usually exploited in the network community. There, formal models are often used [9] for model checking and simulations to study, e.g., correctness of network protocols, optimal schedulings, network measures or power consumption.

Acknowledgements Thanks to Flavio Chierichetti for his valuable support in the proof of Theorem 3.

References

1. L. Aceto, W. Fokkink, and C. Verhoef. Structural operational semantics. In J. Bergstra, A. Ponse, and S. Smolka, editors, *Handobook of Process Algebra*, pages 197–292. North-Holland, 2001.
2. R. Ahuja, T. Magnanti, and J. Orlin. *Network Flows, theory, algorithms, and applications.* Prentice-Hall, New Jersey, 1993.
3. K. Akkaya and M. F. Younis. A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, 3(3):325–349, 2005.
4. I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *IEEE Communications Magazine*, 40(8):102–116, 2002.
5. J. Borgström, U. Nestmann, L. O. Alima, and D. Gurov. Verifying a structured peer-to-peer overlay network: The static case. In *Global Computing*, volume 3267 of *LNCS*, pages 250–265. Springer, 2005.

6. A. Boukerche, editor. *Algorithms and Protocols for Wireless Sensor Networks.* Wiley & Sons, 2009.
7. P. Cenciarelli, D. Gorla, and E. Tuosto. Network applications of graph bisimulation. In R. Heckel and G. Taentzer, editors, *Proceedings of the 4th International Conference on Graph Transformation (ICGT'08)*, volume 5214 of *LNCS*, pages 131–146. Springer, 2008.
8. T. Cormen, C. Leiserson, and R. Rivest. *Introduction to Algorithms.* MIT Press, 1990.
9. A. Fehnker and A. McIver. Formal techniques for the analysis of wireless networks. In *2nd International Symposium on Leveraging of Formal Maethods, Verification and Validation (IEEE-ISOLA).* IEEE, 2006.
10. Y. Ganjali and A. Keshavarzian. Load balancing in ad hoc networks: single-path routing vs. multi-path routing. In *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 2, pages 1120–1125, 2004.
11. J. C. Godskesen. A calculus for mobile ad hoc networks. In *Proc. of COORDINATION*, volume 4467 of *LNCS*, pages 132–150. Springer, 2007.
12. P. Kanellakis and S. Smolka. CCS Expressions, Finite State Processes and Three Problems of Equivalence. *Information and Conputation*, 86(1):43–68, 1990. An extended abstract appeared in *Proc. of ACM PODC'83.*
13. M. Merro. An observational theory for mobile ad hoc networks. In *Proc. of MFPS*, volume 173 of *ENTCS*, pages 275–293, 2007. Full version to appear in *Information and Computation.*
14. M. Merro and E. Sibilio. A timed calculus for wireless networks. In *Proc. of FSEN*, 2009. To appear.
15. N. Mezzetti and D. Sangiorgi. Towards a calculus for wireless systems. In *Proc. of MFPS*, volume 158 of *ENTCS*, pages 331–353, 2006.
16. R. Milner. *Communication and Concurrency.* Prentice Hall, 1989.
17. S. Nanz and C. Hankin. A framework for security analysis of mobile wireless networks. *Theoretical Computer Science*, 367(1-2):203–227, 2006.
18. N. Nehra, R.B.Patel, and V.K.Bhat. Routing with load balancing in ad hoc network: A mobile agent approach. In *6th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2007)*, 2007.
19. D. Park. Concurrency and automata on infinite sequences. In *Theoretical Computer Science*, volume 104 of *LNCS*, pages 167–183. Springer, 1981.
20. C. A. Phillips. The network inhibition problem. In *Proc. of STOC*, pages 776–785. ACM Press, 1993.
21. T. Roughgarden. On the severity of braess's paradox: Designing networks for selfish users is hard. *Journal of Computer and System Science*, 72(5):922–953, 2006.
22. A. Singh, C. Ramakrishnan, and S. A. Smolka. A process calculus for mobile ad hoc networks. In *Proc. of COORDINATION*, volume 5052 of *LNCS*, pages 296–314. Springer, 2008.
23. L. Stockmeyer and A. Meyer. Word Problems Requiring Exponential Time. In *Proc. of 5th Symp. on Theory of Computing (STOC)*, pages 1–9. ACM, 1973.
24. C. Toh. *Ad Hoc Mobile Wireless Networks: Protocols and Systems.* Prentice Hall, 2002.
25. O. K. Tonguz and G. Ferrari. *Ad Hoc Wireless Networks: A Communication-Theoretic Perspective.* John Wiley & Sons, 2006.