3

A temporal reachability analysis

L. Cacciari and O. Rafiq
Laboratoire TASC / Informatique, Université de Pau,
Avenue de l'université, 64000 Pau — France,
tel.: (+33) 59 92 31 99, fax: (+33) 59 80 83 74,
e-mails: {leo.cacciari,omar.rafiq}@univ-pau.fr

Abstract

Reachability analysis is the most popular and the most used method in protocol validation. It consists in constructing a graph called reachability graph, describing communication of machines exchanging messages through FIFO channels. The states and structure of this graph are then analysed according to given properties to validate the related communication protocol. In this paper, we go from communicating machines used in reachability analysis, to design temporal communicating machines allowing one to specify quantitative temporal aspects of communication protocols. A temporal reachability graph describing the global behavior of temporal communicating machines, is then defined. After that, we show how this graph can be used to analyse general properties of communication protocols submitted to temporal constraints.

Keywords

Protocols, Modeling, Transition models, Temporal constraints, Temporal communicating systems, Temporal reachability analysis, Communication properties, Validation.

I. INTRODUCTION

In this paper, we deal with validation of protocols with temporal constraints, by using reachability analysis principles (West, 1978). Among the time semantics proposed in the literature, we have chosen an interval semantics, to design temporal communicating machines taking into account the quantitative temporal aspects of communication protocols. A global graph describing the communication between the temporal machines is then constructed. This graph may be locally infinite. However, it is possible to deduce from it, a locally finite graph that we call temporal reachability graph, by using the approach developed by Alur and Dill to group the global states into regions (Dill, 1989; Alur and Dill, 1994). Beyond the classical state explosion problem, our approach is adapted to study asynchronous communicating systems with temporal constraints and specially to validate new general properties of protocols submitted to temporal

P. Dembiński et al. (eds.), *Protocol Specification, Testing and Verification XV*

constraints: delayed message, blocking delayed message, delayed reception and blocking delayed reception.

II. TRANSITION MODELS AND TEMPORAL CONSTRAINTS: A SURVEY

Several aspects of computer systems in general and of communication protocols in particular: reliability, recoverability, performance,..., are highly dependent on time in all its forms: qualitative and quantitative. Therefore, modeling and analysis of temporal constraints in computer systems is a fundamental subject of study. In this paper, we deal with quantitative temporal aspects i.e. explicit values of time.

To integrate quantitative temporal constraints into systems specification, several extensions of basic formal description techniques are proposed: *transition models* (Ramchandani, 1974; Merlin and Faber, 1976; Dill, 1989; Alur and Dill 1990 and 1994; Berthomieu and Diaz, 1991; Courtiat and Diaz, 1991; Henzinger, Manna and Pnueli, 1991a), *algebraic languages* (Bergstra and Klop, 1984; Nicollin and Sifakis, 1991; Courtiat et al., 1993; Leduc and Leonard, 1993) and *temporal logics* (Alur and Henzinger, 1991 and 1993; Henzinger, Manna and Pnueli, 1991a and 1991b). Our contribution being based on a transition model, this section gives a brief survey of the main temporal extensions of such models.

II.1. Timed Petri nets and time Petri nets

Two basic temporal extensions have been proposed for *Petri nets*: *timed Petri nets* by Ramchandani (1974) and *time Petri nets* by Merlin (Merlin and Faber, 1976). In timed Petri nets, a finite duration is associated with each transition which must fire as soon as it is enabled. In time Petri nets two real values a_t and b_t such that: $0 \leq a_t \leq b_t \leq \infty$, are associated with each transition t. Suppose that t is enabled at time τ, then t cannot fire before $\tau+a_t$ and must fire before or at $\tau+b_t$, unless it is disabled before its firing by the firing of another transition. It is clear that Merlin model is more general than Ramchandani model. Berthomieu and Diaz (1991) have developed an interesting approach using Merlin model for specifying and verifying temporal constraints in communication protocols. All these models present the same limitations than pure Petri nets in describing asynchronous communication and infinite communication channels.

II.2. Timed automata

Timed automata have been introduced by Alur and Dill (Dill, 1989; Alur and Dill, 1990 and 1994) for modeling real-time systems. A timed automaton is a finite automaton with a finite set of *real-valued clocks*. The clocks can be reset to 0 (independently of each other) with the transitions of the automaton, and keep track of the time elapsed since the last reset. The transitions of the automaton put certain constraints on the clock values: a transition may be taken only if the current values of the clocks satisfy the associated constraints. A state of a given real-time system is then represented by the pair $<s, t>$ where s is a state of the automaton and t is a set of real values corresponding to the clocks.

Alur and Dill work deals specifically with real-time systems considered as a whole and not with separate systems communicating through channels with given properties.

The work of Alur and Dill has the merit to have introduced the *regions method* (Dill, 1989; Alur and Dill, 1994) for analyzing temporal systems. Idea is to group the states of a given temporal system into sets of states called regions and having the same first componant s and different clocks values t. They prove that the number of these sets is finite and each one could be re-

presented by one state of the system and a subset of $\mathbb{R}^n$ described by a set of inequalities with no more than two variables per inequality. Using solving principles proposed by Aspavall and Shiloach (1979), one can design simple algorithms to establish non trivial properties of temporal systems.

Another interesting aspect of Alur and Dill work is the design of a model checking approach for temporal systems (Alur, Courcoubetis and Dill 1990 and 1993; Alur, Dill et al. 1992; Alur and Dill, 1994).

II.3. Timed transition diagrams

Timed transition diagrams have been defined by Henzinger, Manna and Pnueli (1991a) for modeling real-time processes. They are finite directed graphs whose vertices are called locations. A location is distinguished for starting the control at each process. Each diagram has a finite set of variables. Each edge (l, l') in the graph is labeled by three elements: a *guarded instruction* $c \rightarrow x:=e$, a *minimal delay* $i \in \mathbb{N}$ and a *maximal delay* $s \in \mathbb{N} \cup \{\infty\}$ such that $s \geq i$. The intended operational meaning of the given edge is as follows. The minimal delay i guarantees that whenever the control of the corresponding process P has resided at the location l for at least i time units during which the guard c has been continuously true, then P may proceed to l'. The maximal delay s ensures that whenever the control of P has resided at l for s time units during which the guard c has been continuously true, then P must proceed to l'. In other words, time cannot advance before either the guard c becomes false, which may be caused by a process parallel to P, or the process P proceeds. In doing so, the control of P moves to l' "instantaneously", and the current values of $\mathbf{e}$ are assigned to the variables $\boldsymbol{x}$.

Processes communicate through shared variables and authors deal specially with synchronous communication by using a CSP rendez-vous. However, they notice that communication through channels could be modelised by shared variables. It then remains to specify the transit delay in the channels.

II.4. Communicating real-time state machines

Communicating real-time state machines (CRSM) introduced by Shaw (1992), are similar to timed transition diagrams with some differences. A CRSM M has a finite number of states, with one start state and zero or more halt states. Each state transition is described by a *guarded command*: <guard>→<command> where a <guard> is a *Boolean expression* over the local variables of M and the <command> can be an *input*, an *output*, or an *internal command*. A transition can only be executed if its guard is evaluated to true. Internal commands can specify either a computation or a physical activity. Input and output (IO) are synchronous and modeled directly after CSP.

An ideal global real-time clock is assumed. A transition with an internal command C is ready at the same time that its "from" state is entered. The corresponding execution time is given by the pair: $[t_{min}(C), t_{max}(C)]$ indicating that the duration d of C is somewhere in the interval $0 \leq t_{min}(C) \leq d \leq t_{max}(C) \leq \infty$.

IO times are also represented by pairs, only in this case denoting the earliest and latest times that the IO can occur after entering a given state. The idea is that a machine is ready to perform IO at its earliest time and remains ready throughout the interval. The intersection of a sender and a receiver interval gives the time of possible communications. It is assumed that IO happens as soon as possible. Dinstinction between commands together with synchronous communication make the Shaw model very difficult to use in modeling communication protocols.

II.5. Temporal communicating machines

Our approach which is described in details in the following sections, is based on *temporal communicating machines* (Rafiq and Cacciari, 1995) wich are finite state machines like in Shaw work. However, these machines communicate through infinite channels in which the messages spend a certain period of time. Another difference with Shaw work, is that we adopt the same semantics for executing any kind of commands: the Merlin semantics. Finally, to validate communication properties of such machines, we use the regions approach proposed by Dill (1989) which has also been used by Berthomieu and Diaz (1991).

III. REACHABILITY ANALYSIS

This section introduces basic notions and principles of reachability analysis of communicating systems.

III.1. Communicating systems

A *communicating finite state machine* (CFSM) is a 4-tuple CM = (Q,A,q^o,σ) where Q is a finite set of *states*, A is a finite set of *actions*, q^o is a distiguished state of Q, the *initial state* and σ is a partial function $\sigma : A \times Q \rightarrow Q$, the *transition function*. The set A of actions is the disjoint union of three sets: $A = A^{in} \cup A^{out} \cup A^{int}$, A^{out} is the set of *sending actions*; its elements are in the form $-a$, where a denotes the *sent message*. The set $M^- = \{a \mid -a \in A^{out}\}$ is the *set of outgoing messages*. A^{in} is the set of *receiving actions*; its elements are in the form $+a$ where a denotes the *received message*. The set $M^+ = \{a \mid +a \in A^{in}\}$ is the set of *incoming messages*. A^{int} is the set of *internal actions*, i.e. actions which do not imply neither sending nor receiving of a message. These actions are denoted by lowercase latin letters without prefix + nor –.

A *communicating system* is a set of communicating machines exchanging messages through FIFO channels. In this paper, we will use only communicating systems made up of two communicating machines. Let us notice, however, that most of the estabilished results can be generalized to systems composed of more than two communicating machines.

A *communicating system* is then defined as a 4-tuple $S = (CM_1,CM_2,C_{1,2},C_{2,1})$ where $CM_i = (Q_i,A_i,q_i^o,\sigma_i)$ is a CFSM for $i = 1, 2$, and $C_{i,j}$ is a FIFO channel connecting CM_i to CM_j, $1 \le i \ne j \le 2$. All messages used in emission by one of the two CFSM are considered as incoming messages in the other one. $M_{i,j} = M^-_i = M^+_j$, $1 \le i \ne j \le 2$ is then the set of messages that CM_i may send to CM_j through Ci,j.

III.2. Reachability graph

The behavior of S is described by a graph, called *reachability graph*, whose verticies are *global states* of S and whose edges are labelled by *global transitions* of S.

A *global state* of S is a 4-tuple $g = (q_1,q_2,c_{1,2},c_{2,1})$ where $q_i \in Q_i$ is the current state of CM_i and $c_{i,j} \in (M_{i,j})^*$ is the current content of $C_{i,j}$.

A *global transition* is a pair (i,α) where $\alpha \in A_i$. It is said to be *firable* in $g = (q_1,q_2,c_{1,2},c_{2,1})$, if and only if $\sigma_i(\alpha,q_i)$ is defined and either $\alpha \in A_i^{int} \cup A_i^{out}$ or $\alpha = +a \in A_i^{in}$ and $c_{j,i} = aw$ with $w \in (M_{j,i})^*$.

When a global transition $t = (i,\alpha)$ is firable in a global state $g = (q_1,q_2,c_{1,2},c_{2,1})$, its firing makes the system go to the global state $g' = (q'_1,q'_2,c'_{1,2},c'_{2,1})$ with $q'_i = \sigma_i(\alpha,q_i)$, $q'_j = q_j$, $1 \leq i \neq j \leq 2$ and:

1. if $\alpha \in A_i^{int}$, then $c'_{1,2} = c_{1,2}$ and $c'_{2,1} = c_{2,1}$;
2. if $\alpha = -a \in A_i^{out}$, then $c'_{i,j} = c_{i,j}a$ and $c'_{j,i} = c_{j,i}$;
3. if $\alpha = +a \in A_i^{in}$, then $c'_{j,i} = w$ and $c'_{i,j} = c_{i,j}$.

g' is said to be *directly reachable* from g *by* t and one can write $g \rightarrow_t g'$ or simply $g \rightarrow g'$. A global state f is said to be *reachable from the global state* g, if there exists a sequence $g_0 = g, g_1, \ldots, g_n = f$, $n \geq 0$, of global states such that $g_{i-1} \rightarrow g_i$, for $i = 1, 2, \ldots, n$. If f is reachable from g, one writes $g \rightarrow^* f$.

The global state $g^o = (q^o_1,q^o_2,\varepsilon,\varepsilon)$, where both machines are in their initial state and both channels are empty, is the *initial state* of S. A global state g is said to be *reachable*, if it is reachable from the initial state, i.e. if $g^o \rightarrow^* g$.

Reachability graph of S is defined as the directed and edge-labelled graph $\mathcal{G}_S$ whose vertices are the reachable global states of S. In $\mathcal{G}_S$ there exists a directed edge going from the state g to the state f and labelled by the global transition (i,α), if and only if $g \rightarrow_{(i,\alpha)} f$.

Since the size of communication channels is not *a priori* bounded, it follows that $\mathcal{G}_S$ my be infinite. The finiteness problem of $\mathcal{G}_S$ is undecidable (Brand and Zafiropulo, 1983). However, for each vertex of $\mathcal{G}_S$ the number of outgoing and incoming edges is finite and bounded, i.e. $\mathcal{G}_S$ is *locally finite*. Let us recall that a graph is said to be *finite*, if it has a finite number of verticies and a finite number of edges and it is said to be *locally finite*, if every vertex is the endpoint of a finite number of edges. One may remark that a finite graph is locally finite but the converse may be false.

Reachability graph is used to validate given properties of communication protocols described as communicating systems. Among properties taken into account one may notice deadlocks, unspecified receptions, blocking unspecified receptions and blocking cycles. These properties are said to be *general*, as they are independent of the provided service. Nevertheless, the reachability graph can also be used to verify service properties which are generally described in a suitable modal or temporal logic. This approach is called *model checking* in the literature.

IV. COMMUNICATING SYSTEMS AND TEMPORAL CONSTRAINTS

Starting from the concept of communicating systems, we introduce the notion of temporal communicating system, to take into account quantitative temporal constraints of communication protocols. Such constraints may correspond to response delay of requests, to firability conditions of internal and external actions and to waiting delay of messages.

The behavior of a temporal communicating system is then described by using a graph, which can be locally infinite. Nevertheless, it is possible to deduce from this behavior graph a locally finite one, which is called temporal reachability graph allowing one to study several general properties of communication protocols subject to temporal constraints.

Let $\mathbb{T} = \mathbb{Q}^+$. $\mathbb{Q}^+$ denotes the set of non-negative rational number and corresponds to the set of *observable instants*. An *interval* $[\tau,\rho]$ on $\mathbb{T}$ is an interval in the classical sense, with $\tau \in \mathbb{N}$ and $\rho \in \mathbb{N} \cup \{+\infty\}$.

IV.1. Temporal communicating machines

A *temporal communicating machine* is a 4-tuple TCM = (Q,A,q^o,π) where Q, A and q^o have the same definition as in a communicating finite state machines and π, the *transition function*, is a partial function $\pi : A\times\mathbb{T}\times Q\rightarrow Q$. For every action α and every state q, we define **fire**(α,q) as the set of instants in which the action α can be executed startig from the state q:

fire$(\alpha,q) = \{t\in \mathbb{T} \mid \pi(\alpha,t,q)$ is defined$\}$.

We make the technical hypothesis that $\pi(\alpha,t,q)$ has the same value $\sigma(\alpha,q)$ for every $t\in$ **fire**(α,q) (*determinism hypothesis*). The CFSM undelying TCM is then defined as $MC = (Q,A,q^o,\sigma)$.

To allow analysis of temporal communicating systems we make the hypothesis that for every action $\alpha\in A$ and every state $q\in Q$, the set **fire**(α,q) is either empty or it is an interval of $\mathbb{T}$ (*continuity hypothesis*). The notations **min**(α,q) and **max**(α,q) denote respectively lower and upper bound of **fire**(α,q).

IV.2. Temporal communicating systems

A *temporal communicating system* is a 4-tuple **TS** = $(TCM_1,TCM_2,TC_{1,2},TC_{2,1})$ where $TC_{i,j}$ are *temporal FIFO channels* through which the messages from TCM_i to TCM_j are exchanged. A temporal FIFO channel TC is an ordinary FIFO channel augmented whith an interval $[min(TC),max(TC)]$ of $\mathbb{T}$. This interval allows one to specify the transit delay of messages through TC: a message should not stay more than $max(TC)$ time units in TC. and should remain at least $min(TC)$ time units in it.

The communicating system underlying to **TS** is **S** = $(CM_1,CM_2,C_{1,2},C_{2,1})$, where CM_i is the CFSM underlying to TCM_i and $C_{i,j}$ is the FIFO channel undelying to $TC_{i,j}$.

Before tackling the behavior of **TS**, let us define the notion of temporal word. Let Σ be a finite alphabet, a *temporal word* on Σ is a word $w = a_1t_1a_2t_2\ldots a_nt_n$ of $(\Sigma\mathbb{T})^*$ such that $t_1\geq t_2\geq\ldots\geq t_n$. The set of temporal words on Σ is denoted by $\mathcal{T}(\Sigma)$. For $w = a_1t_1a_2t_2\ldots a_nt_n\in\mathcal{T}(\Sigma)$, $\|w\| = n$. For $w\in\mathcal{T}(\Sigma)$ and $t\in\mathbb{T}$,

$$w\oplus t = \begin{cases} a_1(t_1+t)a_2(t_2+t)\ldots a_n(t_n+t) \text{ if } w = a_1t_1a_2t_2\ldots a_nt_n\neq\varepsilon, \\ \varepsilon \text{ otherwise.} \end{cases}$$

IV.3. Behavior of temporal communicating system

A global state of **TS** specifies the time spent by messages in the temporal channels as well as the time spent by each machine in its current state. Therefore, a temporal machine may execute an action, if this one is firable in the underlying communicating system and the related temporal constraints are fulfilled i.e. a machine TCM_i may execute an action α in a state q, if it has spent within q at least **min**(α,q) time units and no more than **max**(α,q) time units. Furthermore, if

$\alpha = +a$, the message a has to be at the head of $TC_{j,i}$ within which it has spent at least $min(TC_{j,i})$ time units and no more than $max(TC_{j,i})$ time units. If TCM_i has spent $\mathbf{max}(\alpha,q)$ within q, it has imperatively to execute either α or another possible action, in order to avoid blocking.

Going from these considerations, in order to simplify the behavior definition of a **TS** as well as related validation, we make the following technical hypothesis: if $q \in Q$ and $\alpha \in A$ are such that $\mathbf{fire}(\alpha,q) \neq \emptyset$, then $\mathbf{min}(\alpha,q) \leq \mathrm{Max}(q)$ where $\mathrm{Max}(q)$ is the lowest upper bound of firability intervals of actions firable in q, i.e. $\mathrm{Max}(q)=\min \{\mathbf{max}(\alpha,q) \mid \mathbf{fire}(\alpha,q) \neq \emptyset\}$ (*limitation hypothesis*).

IV.3.1. Global states

A *global state* of **TS** is a 6-tuple $tg = (q_1,t_1,q_2,t_2,tc_{1,2},tc_{2,1})$ where $q_i \in Q_i$ is a state of TCM_i, $t_i \in \mathbb{T}$ is the time elapsed since TCM_i is within q_i with $t_i \leq Max(q_i)$ and $tc_{i,j} \in \mathbf{T}(M_{i,j})$ is the content of $TC_{i,j}$ where if $tc_{i,j} = a\tau w \neq \varepsilon$, then $\tau \leq max(TC_{i,j})$.

Let $tg = (q_1,t_1,q_2,t_2,tc_{1,2},tc_{2,1})$ be a global state of **TS** with $tc_{1,2} = a_1\tau_1a_2\tau_2\ldots a_k\tau_k$ and $tc_{2,1} = b_1\rho_1b_2\rho_2\ldots b_h\rho_h$. tg can be identified to the pair $<g,t>$ where the global state $g = (q_1,q_2,a_1a_2\ldots a_k,b_1b_2\ldots b_h)$ of $\mathbf{S} = (CM_1,CM_2,C_{1,2},C_{2,1})$, is the *underlying state* of tg and $t = (t_1,t_2,\tau_1,\tau_2,\ldots,\tau_k,\rho_1,\rho_2,\ldots,\rho_h) \in \mathbb{T}^n$ is the *temporal characteristics* of tg, $n = \|tc_{1,2}\| + \|tc_{2,1}\| + 2$ is the *size* of tg.

IV.3.2. Global transitions

Two kinds of transition are defined in **TS**: simple transition and temporal transition. A *simple transition* is a pair (i,α) where $\alpha \in A_i$ and $i = 1, 2$. It is *firable* in the global state $tg = (q_1,t_1,q_2,t_2,tc_{1,2},tc_{2,1})$, if and only if $\pi_i(\alpha,t_i,q_i)$ is defined and either $\alpha \in A_i^{int} \cup A_i^{out}$ or $\alpha = +a \in A_i^{in}$ and $tc_{j,i} = a\tau w$ for $w \in \mathbf{T}(\mathrm{M_{j,i}})$ with $\tau \geq min(TC_{j,i})$.

When a simple transition (i,α) is firable in $tg = (q_1,t_1,q_2,t_2,tc_{1,2},tc_{2,1})$, its firing leads **TS** to $tg' = (q'_1,t'_1,q'_2,t'_2,tc'_{1,2},tc'_{2,1})$ where $q'_i = \pi_i(\alpha,q_i,t_i)$, $t'_i = 0$, $q'_j = q_j$, $t'_j = t_j$ and:

1. if $\alpha \in A_i^{int}$ then $ct'_{i,j} = ct_{i,j}$ and $ct'_{j,i} = ct_{j,i}$;
2. if $\alpha = -a \in A_i^{out}$, then $ct'_{i,j} = ct_{i,j}a\tau$, where $\tau = 0$ and $ct'_{j,i} = ct_{j,i}$;
3. if $\alpha = +a \in A_i^{in}$, then $ct'_{i,j} = ct_{i,j}$ and $ct'_{j,i} = w$, where $ct_{j,i} = a\tau w$.

One may then write $tg \rightarrow_{(i,\alpha)} tg'$ or simply $tg \rightarrow tg'$.

Let us notice that if a simple transition (i,α) is firable in a global state $tg = <g,t>$ of **TS**, then it is also firable in the global state g of the underlying communicating system **S**. On the other side, if (i,α) is firable in a global state g of **S**, it is not sure that the corresponding transition is also firable in $tg = <g,t>$. It is firable, if and only if the temporal characteristics of tg fulfill the previous conditions.

A temporal transition t is an element of $\mathbb{T}$ and it is firable in tg if and only $tg \oplus t.=(q_1,t_1+t,q_2,t_2+t,tc_{1,2} \oplus t,tc_{2,1} \oplus t)$ is a global state, i.e. if $t_i+t \leq \mathrm{Max}(q_i)$ for $i = 1, 2$ and $\tau_{i,j} + t \leq max(TC_{i,j})$ where $tc_{i,j} = a\tau_{i,j}w$, $i, j = 1, 2$, $i \neq j$. $tg' = tg \oplus t$ is said to be an *evolution* of tg. The set of evolutions of tg is denoted by $tg^{\oplus}$.

Using the previous definitions, one can prove lemma 1.

Lemma 1

If a simple transition (i,α) is firable in a global state tg, then (i,α) is firable in every evolution of tg, i.e. in every state $tf \in tg^{\oplus}$ ♦.

A global state tf is *reachable from the global state* tg if there exists a sequence $tg_0 = tg, tg_1, \ldots, tg_k = tf$ with $k \geq 0$, such that for $i = 1, 2, \ldots, k$, either $tg_{i-1} \rightarrow tg_i$ or tg_i is an evolution of tg_{i-1}.

The *initial state* of **TS** is $tg^o = (q^o_1, 0, q^o_2, 0, \varepsilon, \varepsilon)$ where both machines are in their initial state, the value of the corresponding timers is equal to zero and both channels are empty. A global state is said to be *reachable*, if it is reachable from the initial state.

IV.3.3. Behavior graph

The behavior of **TS** can be described by a directed and edge-labelled graph $\mathfrak{B}_{TS}$, called *behavior graph* of **TS**. The verticies of $\mathfrak{B}_{TS}$ are the reachable global states of **TS**. In $\mathfrak{B}_{TS}$ there exists an edge labelled by a simple transition (i,α) and going from tg to tf, if and only if $tg \rightarrow_{(i,\alpha)} tf$. There exists an edge labelled by a temporal transition t and going from tg to tf, if and only if $tf = tg \oplus t$. In general, $\mathfrak{B}_{TS}$ is not locally finite and therefore it is not tractable by software tools. To cope with this problem, we can deduce a locally finite graph from $\mathfrak{B}_{TS}$. This graph is called temporal reachability graph and allows one, among other things, to analyse several properties of communication protocols subject to temporal constraints.

IV.4. Communication properties

The communication properties considered in this paper are general properties, i.e. independent of the provided service: unspecified receptions, blocking unspecified receptions, deadlocks, delayed messages, blocking delayed messages, delayed receptions and blocking delayed receptions. They may be analysed by examining just the global states and the structure of the behavior graph.

Let $tg = (q_1, t_1, q_2, t_2, tc_{1,2}, tc_{2,1})$ be a reachable global state of **TS**.

TCM_i has an *unspecified reception* in tg, if and only if $tc_{j,i} = a\tau w$ and $\mathbf{fire}(+a,q_i) = \emptyset$. In other words, the reception of the message at the head of the incoming channel of TCM_i is impossible in tg, independently of temporal constraints. An unspecified reception is said *blocking unspecified reception* if, moreover, $\mathbf{fire}(\alpha,q_i) = \emptyset$ for every action $\alpha \in A_i^{int} \cup A_i^{out}$.

tg is a *deadlock*, if and only if $ct_{1,2} = ct_{2,1} = \varepsilon$ and for every evolution $tg' \in tg^{\oplus}$ of tg, no simple transition is firable in gt'. If $tg = (q_1, t_1, q_2, t_2, \varepsilon, \varepsilon)$ is a deadlock, no progress of the system is possible and the protocol is therefore definitely blocked.

There exists a *delayed message* for TCM_i in tg, if and only if $tc_{j,i} = a\tau w$, $\mathbf{fire}(+a,q_i) \neq \emptyset$, $\tau < min(TC_{j,i})$ and $t_i = \mathrm{Max}(q_i)$. In other words, there exists a delayed message in tg, if and only if the message a which is at the head of $TC_{j,i}$ can not yet be delivred, but a possible transition of TCM_i must be fired since $t_i = \mathrm{Max}(q_i)$. A delayed message is said *blocking delayed message* if, moreover, $\pi_i(\alpha,t_i,q_i)$ is undefined for every action $\alpha \in A^{int} \cup A^{out}$. In other words, if there exists a blocking delayed message for TCM_i in tg, then TCM_i is definitely blocked.

TCM_i has a *delayed reception* in tg, if and only if $tc_{j,i} = a\tau w$, $\mathbf{fire}(+a,q_i) \neq \emptyset$, $t_i < \mathbf{min}(+a,q_i)$ and $\tau = max(TC_{j,i})$. In other words, TCM_i has a delayed reception in tg, if and

only if the reception of the leading message of $TC_{j,i}$ is not defined at the current time, but this message must be delivred by the channel since $\tau = max(TC_{j,i})$. A delayed reception is said *blocking delayed reception* if, moreover, $\pi_i(\alpha,t_i,q_i)$ is undefined for every action $\alpha \in A^{int} \cup A^{out}$. In other words, if TCM_i has a blocking delayed reception in tg, it is definitely blocked.

V. TEMPORAL REACHABILITY GRAPH

As we said in section IV, the behavior graph $\mathfrak{B}_{TS}$ of **TS** can be locally infinite. In this section, we will show how to derive from $\mathfrak{B}_{TS}$ a temporal reachability graph (TRG), which is locally finite and which allows one to validate the previous properties.

V.1. Regions and firability

Every vertex of TRG is a set of global states. These sets, which are called *regions* (Dill, 1989, Alur and Dill, 1994, Alur, Dill et al. 1992), are closed under temporal transitions. The edges connecting the regions in the temporal reachability graph are simple transitions of **TS**.

Definition 1

A *region* is a set R of global states such that if $<g,t>$ and $<h,p> \in R$ then $g = h$. Moreover, if $tg \in R$, then $tg^{\oplus} \subseteq R$.

If R is a region, we set $R = <g,\mathcal{R}>$, where g is the global state underlying to the states of R and $\mathcal{R} \subseteq \mathbb{T}^n$ is the set of the temporal characteristics of states of R.

Firing of a simple transition in a region is defined as follows. Let $R = <g,\mathcal{R}>$ be a region and let (i,α) be a simple transition. Let us take the set $F = <g,\mathcal{F}>$, where $\mathcal{F} = \{t \in \mathcal{R} \mid (i,\alpha)$ is firable in $<g,t>\}$. It follows from lemma 1 that F is a region. It is called the *firability region* of (i,α) in R. If F is not empty, (i,α) is said to be *firable* in R and the consequence of firing (i,α) in R is $R' = \{tg \mid \exists\, tf \in <g,\mathcal{F}>$ s.t. $tf \rightarrow_{(i,\alpha)} tg\}^{\oplus}$, i.e. the closure of the set of global states reached by (i,α) from a state of R. If R' is the result of firing (i,α) in R, one writes $R \rightarrow_{(i,\alpha)} R'$ or simply $R \rightarrow R'$.

V.2. Reachability and inequalities

The *initial region* R^o is the set of the evolutions of the initial state of **TS**: $R^o = <(q^o_1,q^o_2,\varepsilon,\varepsilon),\mathcal{R}^o>$, where $\mathcal{R}^o$ is the set of pairs $(t_1, t_2) \in \mathbb{T}^2$ fulfilling the following system of inequalities:

$$\begin{cases} t_1 - t_2 = 0 \\ t_1 \leq \mathrm{Max}(q^o_1) \\ t_2 \leq \mathrm{Max}(q^o_2). \end{cases} \qquad (1)$$

A region R is said to be *reachable*, if there exist regions $R_0, R_1, \ldots, R_k$, $k \geq 0$, with $R_0 = R^o$, $R_k = R$ and $R_{i-1} \rightarrow R_i$ for $i = 1, 2, \ldots, k$.

Theorem 1
If R is a reachable region and *tg* is a global state belonging to R, then *tg* is reachable. Conversely, every reachable state belongs to a reachable region.

Sketch of the proof
This theorem is proved by induction on the number of simple transitions used to reach R from the initial region R^o

The converse is proved by induction on the number of simple transitions leading from the initial state tg^o to the state *tg* ♦.

Definition 2
$\mathcal{TRG}_{TS}$, the temporal reachabilityt graph of **TS** has as verticies the set of reachable regions. There exists an edge labelled by the simple transition (i,α) and going from the vertex R to the vertex R', if and only if $R \rightarrow_{(i,\alpha)} R'$.

One may notice that TRG is locally finite, since the set of actions is supposed to be finite. Moreover, it may contain several regions having the same underlying state.

One interesting aspect of TRG is that the reachable regions may be specified easely by systems of inequalities.

Theorem 2
Let $R = <g,\mathcal{R}>$ be a reachable region and let (i,α) be a simple transition firable in R and $F = <g,\mathcal{F}>$ the firability region of (i,α) in R. $\mathcal{R}$ and $\mathcal{F}$ can be defined by a system of linear inequalities with at most two variables per inequality

Proof
Let us remark at first that if $R = <g,\mathcal{R}>$ is a reachable region and (i,α) is a simple transition, one can specify the fact that (i,α) is firable in R by one or two inequalities. One writes $t_i \geq \mathbf{min}(\alpha,q_i)$ and if $\alpha = +a$, one takes a second inequality $\tau \geq min(TC_{j,i})$ to describe the fact that the message *a* must spend at least $min(TC_{j,i})$ time units in $TC_{j,i}$. This shows that if $\mathcal{R}$ can be described by a system of inequalities with at most two variables per inequality so does $\mathcal{F}$.

Let now $R_0 = R^o, R_1, \ldots, R_k = R$ be a regions sequence such that $R_{i-1} \rightarrow R_i$ for i = 1, 2, ..., k. If k = 0, R is the initial region and it is defined by system (1), which is in the required form. Let then be k > 0. By induction on *k*, one may suppose that R_{k-1} satisfies the theorem. Let $R_{k-1} \rightarrow_{(i,\alpha)} R_k$. By the induction hypothesis, the firability region of (i,α) in R is $F = <g_{k-1},\mathcal{F}>$ where $\mathcal{F}$ is defined by a system of inequalities with at most two variables per inequality. It is easy to see that the system of inequalities defining $\mathcal{F}$ can be modified to give a system defining $\mathcal{R}$ and still having at most two variables per inequalitiy ♦.

Like the classical reachability graph, TRG can be infinite and its finiteness is undecidable. One can see that by setting all the temporal constraints to $[0,\infty]$ and, therefore, TRG of such a temporal communicating system is equivalent to its related reachability graph. When TRG is finite, it can be constructed in a time polynomial in its size (Rafiq and Cacciari, 1995).

VI. TEMPORAL REACHABILITY ANALYSIS

In this section, we will show how one can use TRG to validate the communication properties defined in section IV. After that, we will illustrate this approach through a simple tranfer protocol.

VI.1. Principles

The basic idea is: if R = <g,$\mathfrak{R}$> is a reachable region, one can verify, in a time polynomial in the size of R, if R contains or not states having one of the properties defined in section IV.

R contains respectively an unspecified reception, a blocking unspecified reception or a deadlock, if and only if the underllying state in the corresponding communicating system is respectively an unspecified reception, a blocking unspecified reception or a deadlock. Moreover, all the states of R have the same property.

Let $tg = (q_1,t_1,q_2,t_2,tc_{1,2},tc_{2,1})$ be a global state where $tc_{1,2} = a_1\tau_1 w$. a_1 is a delayed message in tg, if and only if the following conditions hold: $\mathbf{fire}(+a,q_1) \neq \emptyset$, $t_1 = \mathrm{Max}(q_1)$ and $\tau_1 < min(TC_{2,1})$. It then follows that, in order to find out the set of $MR_1 \subseteq R$ of states in which TCM_1 has a delayed message, one has to establish that $\mathbf{fire}(+a,q_1) \neq \emptyset$ and, in this case, to construct the set $\mathfrak{D}$ of the elements of $\mathfrak{R}$ such that $t_1 = \mathrm{Max}(q_1)$ and $\tau_1 < min(TC_{2,1})$. The construction of $\mathfrak{D}$ can be realized in a time quadratic in the size of R.

The same approach is used for dealing with the other properties.

VI.2. Example of a simple transfer protocol

In this example we consider a sender sending data (*Dt*) to a receiver by using a reliable channel. Before sending, the sender has to be sure, by using messages *Cr*, *Ca* and *Cn* that the receiver is ready (*Ca*) or not (*Cn*) to receive data. 5 time unities have to be elapsed between two successive sending sequences (*-Cd,+Ca,-Dt*). The related protocol is described in Figure 1.

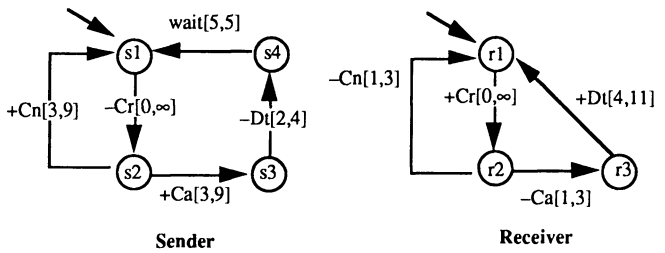

Figure 1: a simple transfer protocol

Let us assume that $min(TC_{i,j}) = 1$ and $max(TC_{i,j}) = 3$. The related TRG is given in Figure 2.

Let us now consider the case with $min(TC_{1,2}) = 1$, $max(TC_{1,2}) = 5$, $min(TC_{2,1}) = 2$ and $max(TC_{2,1}) = 3$. One can see that the transition *(1,-Cr)* is firable in R^o and $R^o \rightarrow_{(1,-Cd)} R^1$, where:

$$R^1 = <(s2,r1,Cr,\varepsilon);\begin{cases}0 \leq t_1 = \tau \leq 5\\ t_2 - t_1 \geq 0\end{cases}>.$$

Similarly, $R^1 \rightarrow_{(2,+Cr)} R^2$ where:

$$R^2 = \langle (s2,r2,\varepsilon,\varepsilon); \begin{cases} 1 \le t_1 \le 8 \\ 0 \le t_2 \le 3 \\ 1 \le t_1 - t_2 \le 5 \end{cases} \rangle$$

and $R^2 \rightarrow_{(2,-Ca)} R^3$ where:

$$R^3 = \langle (s2,r3,\varepsilon,Ca); \begin{cases} 2 \le t_1 \le 9 \\ 0 \le t_2 = \tau \le 3 \\ 2 \le t_1 - t_2 \le 8 \end{cases} \rangle.$$

R^3 contains global states in which the message *Ca* is delayed. To see that, it is sufficient to notice that the states of R^3 in which Ca is delayed are those belonging to $\langle (s2,c,\varepsilon,Ca), \mathcal{T} \rangle$, where $\mathcal{T}$ is the polyhedron of points of $\mathcal{R}^3$ such that $t_1 = 9 = \mathrm{Max}(s2)$ and $\tau < 2 = min(TC_{2,1})$. Now, one can see that this set is not empty and it is defined by:

$$\begin{cases} t_1 = 9 \\ 1 \le t_2 = \tau < 2 \end{cases}.$$

Finally, let us consider another case defined by $min(TC_{i,j}) = 0$ and $max(TC_{i,j}) = 1$. In this case $R^o \rightarrow_{(1,-Cr)} R^1$, where:

$$R^1 = \langle (s2,r1,Cr,\varepsilon); \begin{cases} 0 \le t_1 = \tau \le 1 \\ t_2 - t_1 \ge 0 \end{cases} \rangle.$$

Similarly, $R^1 \rightarrow_{(2,+Cr)} R^2$ where:

$$R^2 = \langle (s2,r2,\varepsilon,\varepsilon); \begin{cases} 0 \le t_1 \le 4 \\ 0 \le t_2 \le 3 \\ 0 \le t_1 - t_2 \le 1 \end{cases} \rangle$$

and $R^2 \rightarrow_{(2,-Ca)} R^3$ where:

$$R^3 = \langle (s2,r3,\varepsilon,Ca); \begin{cases} 1 \le t_1 \le 5 \\ 0 \le t_2 = \tau \le 1 \\ 1 \le t_1 - t_2 \le 4 \end{cases} \rangle.$$

The states of R^3 for which $\tau = 1$ and $t_1 < 3$ are those in which the sender has a delayed reception. Now, one can see that the set of such states is not empty and it is given by:

$$DR^3 = \langle (s2,r3,\varepsilon,Ca); \begin{cases} 2 \le t_1 < 3 \\ t_2 = \tau = 1 \end{cases} \rangle.$$

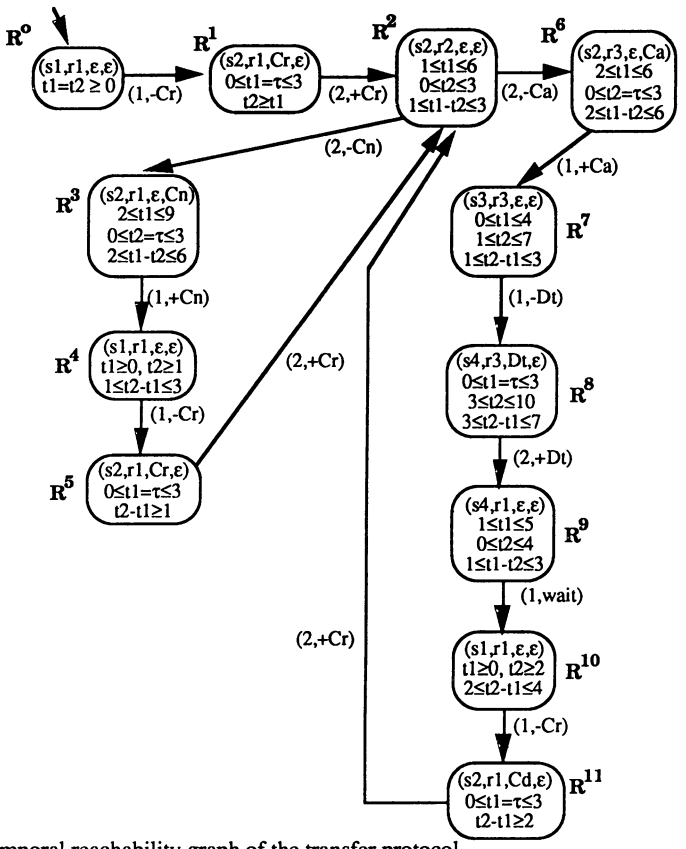

Figure 2: temporal reachability graph of the transfer protocol

VII. CONCLUSION

In this paper, we have designed a temporal communicating machine model, to specify quantitative temporal aspects of communication protocols. After that, we have defined a behavior graph describing the communication between temporal machines, but this graph is locally infinite. We have then showed how to deduce from this graph, a locally finite one allowing one to validate new general properties, dealing with temporal constraints in communication protocols. This graph is called temporal reachability graph. Our work is now oriented to design an approach, to verify properties of the service provided by communication protocols with temporal constraints, using temporal reachability graph and an adapted temporal logic.

ACKNOWLEDGMENTS

This work has been supported by CNET-France Télécom under Grant 92 1B 178 as part of the CNET-CNRS Cesame project on the design of high-speed multimedia cooperative systems.

REFERENCES

Alur R., Courcoubetis C. and Dill D. L. (1990), Model-cheking for real-time systems, in *Proc. 5th IEEE Symp. on Logic in Computer Science*, pp. 414-425.

Alur R., Courcoubetis C. and Dill D. L. (1993), Model-cheking in dense real-time, *Information and Computation* **104**, pp. 2-34.

Alur R. and Dill D. L. (1990), Automata for modelling real-time systems, in *Proc. ICALP'90* , LNCS 443, Springer-Verlag , pp. 323-335.

Alur R. and Dill D. L. (1994), A theory of timed automata, *Theoretical Computer Science*, **126**, pp. 183-235.

Alur R. and Henzinger T. A. (1991), Logics and models of real-time: a survey, in *Proc. REX Workshop*, LNCS 600, Springer-Verlag, pp. 74-106.

Alur R. and Henzinger T. A. (1993), Real-time logics: complexity and expressiveness, *Information and Computation*, **104**, pp. 35-77.

Alur R., Dill D. L., Wong-Toi H., Courcoubetis C. and Halbwachs N. (1992), Minimizing of timed transition systems. in *Proc. CONCUR'92*, LNCS 630, Springer-Verlag, pp. 340-354.

Aspavall B. and Shiloach Y. (1979), A Polynomial time algorithm for solving systems of linear inequalities with two variables per inequality, in *Proc. 20th Ann. Symp. on Foundation of Computer Sciences*, IEEE, pp. 205-217.

Bergstra J. A. and Klop J. W. (1984), Process algebra for synchronous communication, *Information and Control*, **60**.

Berthomieu B. and Diaz M. (1991), Modeling and verification of time dependent systems using time Petri nets, *IEEE Trans. Soft. Eng*, **17**, pp. 259-273.

Brand D. and Zafiropulo P. (1983), On communicating finite state machines, *Journal of ACM*, **30**, pp. 361-371.

Courtiat J.-P. and Diaz M. (1991), Time in state-based formal description techniques for distributed systems, in *Proc. REX Workshop*, LNCS 600, Springer Verlag, pp. 149-175.

Courtiat J.-P., De Camargo M. S. and Saidouni D. E. (1993), RT_LOTOS : LOTOS temporisé pour la spécification de systèmes temps réel, in *Actes de CFIP'93*, Hermès, Paris, France, pp.427-441.

Dill D. L. (1989), Timing assumptions and verification of finite-state concurrent systems, in *Proc. Workshop on Automatic Verification Méthods for Finite State Systems*, LNCS 407, Springer Verlag, pp. 197-212.

Henzinger T. A., Manna Z. and Pnueli A. (1991a), Timed transition systems, in *Proc. REX Workshop*, LNCS 600, Springer Verlag, pp. 226-251.

Henzinger T. A., Manna Z. and Pnueli A. (1991b), Temporal proof methodologies for real-time systems, in *Proc. 18th Annual ACM Symp. on Programming Languages*, pp. 353-366.

Leduc G. and Leonard L. (1993), Comment rendre LOTOS apte à spécifier des systèmes temps réel, in *Actes de CFIP'93*, Hermès, Paris, France, pp.407-426.

Merlin P. and Faber D. J. (1976), Recoverability of communication protocols, *IEEE Trans. Comm*, **24**.

Nicollin X. and Sifakis J. (1991), An overview and synthesis on timed process algebras, in *Proc. CAV'91*, LNCS 575, Springer Verlag, pp. 376-398.

Ostroff J. S. (1989), Automated verification of timed transition models, in *Proc. Workshop on Automatic Verification Méthods for Finite State Systems*, LNCS 407, Springer Verlag, pp. 247-256.

Rafiq O. and Cacciari L. (1995), Protocoles, contraintes temporelles et validation, in *Proc. of CFIP'95*, Hermès, Paris, France, pp. 257-292.

Ramchandani C. (1974), *Analysis of asynchronous concurrent systems by timed Petri nets*, Tech. Rep. 120, Project MAC, MIT.

Shaw A. C. (1992), Communicating real-time state machines, *IEEE Tans. Soft. Eng.*, **18**, pp. 805-816.

West C. H. (1978), An automated technique of communications Protocol Validation, *IEEE Trans. Comm.*, **26**, pp. 1271-1275.

BIOGRAPHIES

Leo CACCIARI has got the Laurea in mathematics of the university La Sapienza (Roma, Italy) in 1985. He received the Doctor degree in computer science from the university of Bordeaux-I in 1989. Since 1990, he is an associate professor at the university of Pau, France. He is a member of the French national research project CESAME (Formal Design of High Speed Multimedia Cooperative Systems) that is supported by the CNET and CNRS. His research interests include computer networks, protocol specification, verification, implementation and testing.

Omar RAFIQ received the Doctor ès-Sciences degree (1983) in computer science from the university of Bordeaux-I where he was an assistant professor from 1974 to 1978. He spent 10 years (1978-1987) in banking (BNP), research (INRIA and ADI) and industry (RENAULT and BULL), before joining the university of Pau in 1987 as a professor of computer science. He participated to national and international projects on networks and protocols (RHIN, CESAME, ESPRIT and COST). He served as a Chairman of CFIP'91 and IWPTS'93 conferences, and as a co-Chairman of CFIP'88 and FORTE'95. He was the Editor-in-Chief of the Networking and Distributed Computing Journal (1990-1994) and he is the Editor-in-Chief of the Electronic Journal on Networks and Distributed Processing. He has chaired sessions at conferences and served on program committees. He is an expert member of AFNOR and ISO. His research interests include computer networks, protocol engineering and distributed processing.