Uma abordagem para especificação e verificação de sistemas dependentes do tempo

Murilo S. de Camargo & Jean-Marie Farines
INE - CTC LCMI - EEL - CTC
Universidade Federal de Santa Catarina
Caixa Postal 476 - 88.040-900 - Florianópolis - S.C.
Tél.: (+55) 482 31-9202 - Fax: (+55) 482 34-9770
E-mail: murilo@inf.ufsc.br & farines@lcmi.ufsc.br

Resumo

Neste artigo apresenta-se uma abordagem para especificação e verificação de *sistemas dependentes do tempo*; ou seja, sistemas nos quais o tempo intervém direta e explicitamente (ex. sistemas tempo-real, protocolos de comunicação e aplicações multimídia). Inicialmente, apresenta-se e justifica-se o uso de uma *álgebra de processos temporizada, RT-LOTOS* (uma extensão temporal de LOTOS) para especificação de sistemas dependentes do tempo. Em seguida, abordam-se os métodos para análise/verificação para esse tipo de sistemas e apresenta-se a proposta de uso dos *Grafos Temporizados* como modelo subjacente para facilitar a verificação utilizando uma técnica de *verificação de modelos* a partir da *lógica temporal tempo-real TCTL*. Por fim, descreve-se um ambiente para auxílio à verificação que encontra-se atualmente em fase de desenvolvimento.

Abstract

In this paper, we present an approach to specify and to verify time dependent systems; in this kind of systems (e.g. real-time systems, communication protocols and multimedia applications), time appears in a direct and explicit way. First, we present and justify the use of a timed process algebra, called RT-LOTOS which is a temporal extension of LOTOS, to specify time dependent systems. In the following we discuss about analysis and verification methods to be used in the systems and we present a proposal based on model checking which uses the real-time temporal logic TCTL on a timed graph obtained from the RT-LOTOS specification; this translation aims to improve and to facilitate the proposed method. At last, we describe a specification and verification environment which is being presently developed in our laboratory.

1 Introdução

Correção, confiabilidade e bom desempenho são requisitos exigidos normalmente no desenvolvimento de sistemas computacionais em geral, podendo ter no caso dos dois últimos requisitos citados um grau maior ou menor de importância dependendo da aplicação. Para sistemas computacionais dependentes do tempo e particularmente para aplicações altamente críticas, o próprio conceito de correção pois ela depende não só dos resultados lógicos computados, mas também do tempo no qual esses resultados são produzidos.

A utilização de técnicas formais nas diversas fases do ciclo de vida (especificação, projeto e implementação), apresenta-se atualmente como uma solução necessária para este tipo de sistemas. Máquinas de Estados Finitas, Lógicas Temporais, Redes de Petri e Álgebras de Processos são alguns dos modelos formais mais utilizados.

Neste trabalho o interesse é centrado no desenvolvimento de uma abordagem para especificação e verificação de sistemas dependentes do tempo. Nesta abordagem, propõe-se utilizar como linguagem de especificação a Álgebra de Processos Temporizada RT-LOTOS [5, 6, 7] que é uma extensão temporal da linguagem de especificação LOTOS [9]. Para verificação dos sistemas dependentes do tempo especificados em RT-LOTOS propõe-se a utilização de uma técnica baseada em *verificação de modelos* na Lógica Temporal Tempo-Real TCTL [1]. Inicialmente, representa-se a propriedade em lógica TCTL e a seguir verifica-se se o modelo que representa o comportamento do sistema satisfaz ou não a fórmula TCTL que representa esta propriedade. Na abordagem proposta, para facilitar a verificação, as especificações do sistema em RT-LOTOS são traduzidas em termos de grafos temporizados. A passagem de uma representação em RT-LOTOS para uma em grafos temporizados é um dos objetivos principais deste artigo e será detalhado nas próximas seções.

Este artigo está organizado da seguinte maneira. A seção 2 apresenta uma breve revisão sobre os principais formalismos para representação de sistemas dependentes do tempo e uma justificativa nas escolhas feitas na nossa abordagem. A seção 3 descreve sucintamente o formalismo RT-LOTOS e apresenta alguns exemplos de processos dependentes do tempo descritos por este. A seção 4 introduz a abordagem proposta para verificação de sistemas dependentes do tempo. A seção 5 apresenta a definição formal de RT-LOTOS em Grafos Temporizados. A seção 6 apresenta uma visão geral do ambiente para verificação em desenvolvimento na UFSC. E finalmente, a seção 7 apresenta as conclusões finais sobre o trabalho apresentado.

2 Escolha do formalismo para representação

Nesta seção são apresentados três formalismos utilizados em sistemas dependentes do tempo: as redes de petri com tempo, as lógicas para tempo-real e as álgebras de processos tempo-real; ainda é discutida a escolha do formalismo a ser usado neste trabalho: álgebra de processos tempo-real.

As redes de Petri têm como características principais a legibilidade e a capacidade de expressar as diversas relações de causalidade: paralelismo, seqüenciamento, não determinismo e sincronização. Entretanto, este formalismo apresenta dificuldades para tratar com estruturas de dados e com mecanismos de estruturação como os operadores de composição paralela por exemplo. Diversas formas de expressar o tempo se encontram na literatura, associando-o aos lugares, arcos e transições [4]. As redes de petri têm sido usadas com bons resultados para o desenvolvimento de sistemas dependentes do tempo, entretanto as limitações citadas não nos levaram a considerá-las como o formalismo a ser escolhido na nossa abordagem.

As lógicas temporais designam lógicas modais cujos operadores são interpretados de maneira temporal. Além dos operadores da lógica proposicional, os operadores "*em seguida*",

"doravante", *"eventualmente"* e *"até"* são os operadores básicos deste tipo de lógica. Elas podem ainda ser estendidas incorporando a representação e o tratamento de propriedades temporais quantitativas como periodicidade, *deadline*, e atraso. Existem uma variedade grande de lógicas deste tipo que se diferenciam pelas características do relógio (explícito ou implícito), da semântica (de intervalo ou de ponto; linear ou arborescente) [13]. As lógicas temporais, de uma forma geral, são excelentes formalismos para se descrever as propriedades de um sistema mas apresentam dificuldades para representar especificações complexas pela sua falta de estruturação, além da sua maior complexidade de uso.

As álgebras de processos alem de possuir um poder de expressão que lhe permite expressar as diversas relações de causalidade apresentam facilidades para representar comportamentos complexos a partir da composição de especificações; além deste poder de expressão as álgebras de processos possuem uma semântica que as tornam extremamente interessantes para a verificação e teste de sistemas distribuídos. Entretanto, é necessário, em geral, extender as álgebras de processos com o tempo para poder modelar comportamentos tempo-real, pois com as álgebras de processos não temporizadas não é possível especificar o tempo exato da ocorrência de um evento, nem forçar a ocorrência de alguns deles. A nossa escolha neste trabalho recai então sobre uma álgebra de processos temporizada, pelas razões acima levantadas. LOTOS sendo a mais completa álgebra de processos definida na atualidade [2], apesar de algumas limitações (não representação do paralelismo verdadeiro, impossibilidade de reconfiguração dinâmica, por exemplo), além de ser padronizada pela ISO [9]. Escolhemos então utilizá-la em nossa abordagem para sistemas dependentes do tempo, dando-lhe uma extensão temporal adequada para representação das características deste tipo de sistemas.

3 A especificação de sistemas dependentes do tempo: RT-LOTOS

3.1 Uma visão intuitiva de RT-LOTOS

RT-LOTOS [5, 6, 7] é uma extensão temporal da linguagem LOTOS que tem como fundamento básico a associação de intervalos temporais às ações observáveis. Um intervalo $[t_{\min}, t_{\max}]$ associado a uma ação a determina a janela de tempo na qual esta pode ser oferecida. O intervalo $[0, \infty]$ é associado por hipótese a cada ação observável sem intervalo definido explicitamente. De maneira análoga a LOTOS Básica, se denotará o conjunto das ações observáveis por *Act*.

De acordo com o paradigma das álgebras de processos onde toda ação observável depende de seu ambiente para se realizar, na linguagem RT-LOTOS, uma ação só poderá ser realizada durante o intervalo de tempo especificado; no entanto o limite superior deste, t_{max}, não implica absolutamente na realização forçada e urgente da ação temporizada.

Uma ação especial a^* é associada à não realização de uma ação temporizada $[t_{min}, t_{max}]a$ caracterizando assim uma violação temporal. O conjunto destas ações especiais é chamado Act^*. Com estas ações especiais é possível exprimir com simplicidade, no caso de violações temporais, os tratamentos de exceções a partir de um novo operador chamado de preempção temporal.

Além disso, toda ação observável temporizada pode ser ocultada na linguagem RT-LOTOS. Por questões de coerência, é também possível associar restrições temporais às ações internas i; no entanto, existem duas diferenças que caracterizam a urgência da ação interna:

- na falta de uma indicação, considera-se a ação interna i como temporizada pelo intervalo $[0, 0]$, o que implica na ocorrência imediata e incontrolável da ação i quando ela é oferecida;

- nenhuma violação temporal pode ser associada à ação interna $[t_{min}, t_{max}]i$, sendo que ela deve necessariamente ser realizada no intervalo $[t_{min}, t_{max}]$.

3.2 A sintaxe e os operadores de RT-LOTOS

Uma expressão em RT-LOTOS se escreve a partir de uma extensão da sintaxe da linguagem LOTOS Básica. Os termos de RT-LOTOS são gerados pela sintaxe apresentada a seguir.

$$E ::= stop \mid exit \mid [t_{min}, t_{max}]a; E \mid [t_{min}, t_{max}]i; E \mid E[\,]E' \mid E|[L]|E' \mid hide\ L\ in\ E \mid$$
$$E >> F \mid E [> F \mid E < L] \{a_1{:}Q_1, \cdots, a_n{:}Q_n\} \mid P[a_1, \cdots, a_n]$$

3.2.1 Descrição informal dos operadores de RT-LOTOS

"*stop*" representa um processo que não faz nada, exceto deixar o tempo progredir.

"*exit*" representa o processo de terminação com sucesso clássica de LOTOS. A ação δ da semântica de *exit* não é urgente.

"$[t_{min}, t_{max}]a; E$" representa um processo no qual a ação oferecida a só pode ser realizada dentro do intervalo de tempo $[t_{min}, t_{max}]$; antes de t_{min} a ação sofre um atraso deixando o tempo apenas progredir, e depois de t_{max} o oferecimento da ação deixa de existir. Se no intervalo de tempo a ação a é realizada, então $[t_{min}, t_{max}]a; E$ passa a se comportar como E. Se a ação a ficou oferecida até o instante t_{max} e não foi realizada, então ocorre a violação temporal a^* neste instante e o processo passa a se comportar como o processo *stop*.

"$E[\,]F$" representa a escolha entre os processos E e F, sendo que ela é realizada quando ocorre uma primeira ação $a \in Act \cup \{i, \delta\}$; isto implica que a realização de uma ação de violação temporal a^* não decide uma escolha.

"$E|[L]|F$" representa o operador de composição paralela de dois processos. Cada um dos dois processos pode realizar, de modo independente, as ações que não pertençam ao conjunto L, sendo que para as ações no conjunto L os processos devem se sincronizar. As sincronizações ocorrem desde que elas sejam possíveis, isto é, se ambos os processos oferecem uma ação para se sincronizar, então esta ação é realizada imediatamente. Ou seja, estamos atribuindo uma semântica de progresso máximo às *ações sincronizáveis*. No caso de um dos dois processos não oferecer a ação sincronizável até a expiração do intervalo de tempo da ação oferecida ocorrerá uma ação de violação temporal associada a ação cujo intervalo temporal expirou; o comportamento subseqüente continuará sendo a composição paralela do processo que segue a violação temporal com o processo que não ofereceu a ação à sincronizar.

"hide L in E" representa um processo que transforma as ações em L em ações invisíveis (e urgentes).

"$E >> F$" representa a composição seqüencial de dois processos E e F. Sua interpretação informal é que se o processo E termina com sucesso, então a execução do processo F é possível.

"$E [> F$" representa um processo que se comporta como E, mas que pode ser interrompido a qualquer instante pela realização de uma ação de F. Se E termina sem que o processo F comece, então "$E [> F$" também termina. Mas, se uma ação de F ocorre antes do término de E, então a execução do processo E é abandonada e o controle passa para o processo F.

"$E < L] \{a_1{:}Q_1, \cdots, a_n{:}Q_n\}$" representa o operador de preempção temporal. Este operador possui aridade $n+1$ onde $n < \infty$ é a cardinalidade do conjunto L, i.e. $|L| = n$. "a_i" é o rótulo identificador, "Q_i" é o processo e "$a_i : Q_i$" indica que o processo "Q_i" tratará uma eventual violação temporal "a_i^*". Se durante a execução de E ocorrer uma violação temporal a_i^* com $a_i \in L$, então a execução do processo E é abandonada e o processo Q_i é iniciado no seu lugar.

$P[a_1/a'_1, \cdots, a_n/a'_n]$ representa uma instanciação do processo $P[a'_1, \cdots, a'_n]$ em que as ações a'_i têm seu nome trocado para a_i, para $i = 1, \cdots, n$.

A álgebra de processos RT-LOTOS, incluindo-se aí a sua semântica operacional transicional (*à la Plotkin*), encontra-se completamente descrita em [5, 6].

3.3 Alguns exemplos descritos em RT-LOTOS

São apresentados a seguir alguns exemplos de representações em RT-LOTOS.

Retardo ("delay") Um retardo de $t \in D^{\omega}$ (D^{ω} é um domínio de tempo com um elemento infinito ω) imposto a um processo P pode ser representado por:

```
Q:=[t]i;P
```

Processo periódico Sejam Q um processo e $t \in D^{\omega}$, então um processo P que representa o lançamento do processo Q a cada período de tempo t pode ser representado por:

```
P:=Q|||[t]i;P
```

Timeout Sejam P e Q dois comportamentos, e $t \in D^{\omega}$. Um *timeout* é um mecanismo dependente do tempo que se comporta como P, se uma ação inicial de P ocorre até o instante t, ou como Q após o tempo t. No nosso formalismo, um mecanismo de *timeout* pode ser modelado como:

```
P [] [t]i;Q
```

Watchdog Em RT-LOTOS um mecanismo de *watchdog* pode ser modelado utilizando o operador de preempção, da maneira que segue:

```
P [> [t]i;Q
```

onde P e Q representam respectivamente o comportamento normal e o de exceção e $t \in D^{\omega}$. O processo resultante se comporta como P até o instante t, após o que P é abortado, e Q é iniciado.

Recuperação após sinalização de falha temporal: Considere-se um processo P representando um sistema tempo-real em que uma dada ação a seja considerada imprescindível e devendo satisfazer restrições de tempo do tipo $[t_{min}, t_{max}]$ em cada uma das suas ocorrências no processo. Deseja-se representar o seguinte comportamento: uma eventual não realização da ação a por violação da restrição de tempo no processo P deve provocar a realização das três etapas seguintes:

1. a instância do processo P, em execução, deve ser abortada,
2. um processo de recuperação Q deve ser lançado, e em seguida, após a recuperação deste
3. uma outra instância do processo P deve ter início.

De forma genérica, a especificação de um tal comportamento em RT-LOTOS pode ser facilmente representada por:

```
P <a] {a:(Q >> P)}
```

Assim, na primeira vez que ocorrer uma violação temporal (ação a^*) no processo P o processo "`(Q >> P)`" será iniciado. No caso da não ocorrência de a^* em P o processo "`(Q >> P)`" nunca terá inicio, e se P terminar, então o processo "`P <a] {a:(Q >> P)}`" terminará também.

Fluxo de vídeo com dependência entre unidades de informações: Deseja-se representar um fluxo-vídeo no qual existe uma trama de referência representada pela ação `m` no instante `d`

e por exemplo duas outras tramas que dependem da trama (*frame*) de referência e que serão representadas pelas ações a e b respectivamente nos instantes 2d e 3d. Além disso, quando a trama de referência m ou uma das outras tramas a e b não é fornecida, a execução do processo em curso é abandonada por um tempo equivalente ao tempo necessário à chegada da proxima trama de referência (respectivamente 2d, d ou 0).

A especificação RT-LOTOS é dada por:

```
video := ([d]m;[d]a;[d]b;video)
         <m,a,b>
           {m:[2d]i;video, a:  [d]i;video, b:video }
```

3.4 Considerações Gerais

Em conclusão, pode-se afirmar que além de oferecer como em LOTOS não temporizado, as facilidades de representação de comportamentos com seqüencialidade, paralelismo e sincronização nas ações, RT-LOTOS permite um conjunto de mecanismos temporais que direta ente ou por composição garantem a representação do comportamento temporal dos sistemas. Em particular, a ação de violação temporal (a^*) e o operador de preempção temporal "$E < L\}\{a_1{:}Q_1, \cdots, a_n{:}Q_n\}$" têm um papel muito importante no tratamento de falhas temporais e no oferecimento de um comportamento de exceção quando ocorrem, como o mostra os dois últimos exemplos anteriores.

Além dos exemplos simples apresentados anteriormente, RT-LOTOS foi utilizado com sucesso na especificação de diversos sistemas dependentes do tempo mais complexos, e entre outros, aplicações multimídias; em particular, uma especificação completa do proble a da sincronização de lábios (vídeo e audio) se encontra em [5].

Apresentamos neste artigo a parte comportamental de RT-LOTOS. Entretanto, esta versão básica foi estendida, introduzindo a passagem de parâmetros na comunicação, um operador de contagem do tempo similar ao existente em [15] e uma representação de dados, com o intuito de poder representar completamente os sistemas dependent s do tempo. Detalhes sobre a versão completa de RT-LOTOS podem ser encontrados em [5].

4 A verificação de sistemas dependentes do tempo

Uma vez definido o modelo para representação de sistemas dependentes do tempo a ser utilizada na abordagem proposta neste artigo, apresentaremos a parte relacionada a verificação desta. Após uma discussão inicial sobre métodos de verificação em sistemas dependentes do tempo, adota-se o método de verificação de modelos a partir de uma lógica temporal temporizada. A necessidade de utilizar uma representação reduzida dos sistemas a serem analizados, nos leva a utilizar modelos do tipo Grafos Temporizados oriundos das especificações escritas em RT-LOTOS; esta transformação será amplamente apresentada e discutida em itens subseqüentes

Quando se utiliza uma álgebra de processos para especificar sistemas como é o caso neste artigo, os métodos de verificação de sistemas geralmente utilizados são os baseados em *verificação de equivalências* e os baseados em *verificação de modelos*.

Os métodos de *verificação de equivalências* são caracterizados pelo confronto de duas especificações do sistema a verificar: uma primeira descrevendo os detalhes de realização dos procedimentos necessários para implementar um determinado comportamento do sistema; e uma segunda, mais simples, que descreve uma representação mais abstrata do comportamento desejado. Geralmente, utilizam-se os Sistemas de Transições Rotuladas como formalismo de base para comparar estas duas especificações que podem originalmente serem representados a

partir de outros formalismos. Desta comparação, pode-se determinar se os comportamentos são equivalentes.

Na *Verificação de modelos* ("*model-checking*"), o método consiste em provar a satisfação ou não de asserções sobre um modelo de do comportamento do sistema. Usa-se um primeiro formalismo para poder descrever o sistema (em geral máquinas de estados finitos, redes de Petri, ou autômatos); deste formalismo são gerados grafos que representam os estados alcançáveis pelo sistema. Um segundo que é formalismo é uma lógica modal, em geral, permite expressar asserções sobre propriedades do sistema que podem ser provadas ou refutadas.

A aplicação desses métodos num contexto onde o tempo intervém direta e explicitamente não é trivial. A representação de sistemas dependentes do tempo na forma de Sistemas de Transições Rotuladas implica em adicionar um número muito elevado de transições para denotar a progressão do tempo. Por exemplo, para representar um mecanismo de timeout onde o tempo de espera é de 2 unidades de tempo (ex.: $a; E[][2]i; F$), teríamos que ter três ramificações representando as possibilidades de realização de a nos instantes 0,1 e 2 no caso do domínio de tempo ser o dos números naturais; ou teríamos um número infinito de ramificações para representar as possibilidades de ocorrência de transições de progressão do tempo caso o domínio de tempo fosse o dos números racionais positivos. Este exemplo mostra a possibilidade de crescimento do número de estados, que a introdução do tempo provoca nos Sistemas de Transições Rotuladas, sobretudo quando deseja-se utilizar um modelo de tempo denso.

A partir da discussão acima, pode-se concluir que o uso de Sistemas de Transições Rotuladas como formalismo de base para os métodos de verificação citados não é, em geral, de grande interesse prático para tratamento de sistemas dependentes do tempo. Em particular, apesar da sua importância na definição completa de uma Álgebra de Processo Temporizada as bissimulações com tempo que podem ser definidas para uma álgebra de processos temporizadas parecem difíceis de serem verificadas quando o domínio de tempo for denso.

A técnica de verificação descrita em [1] tem se mostrado como uma alternativa promissora para a verificação de sistemas dependentes do tempo. Nessa referência, os comportamentos dos sistemas a verificar são descritos em grafos temporizados e as propriedades dos sistemas são especificadas usando a lógica TCTL; então um algoritmo verifica se os modelos (grafos temporizados dos sistemas) satisfazem ou não as fórmulas TCTL.

A abordagem apresentada em [1] tem ainda uma importância diferenciada no contexto de RT-LOTOS, pois em [12] mostra-se que é possível transladar uma álgebra de processos temporizada (ATP) para grafos temporizados. Da mesma maneira, a abordagem apresentada neste artigo tem como ponto de partida a representação das especificações de sistemas dependentes do tempo a partir da linguagem RT-LOTOS. O processo de verificação inicia pela transformação dessas especificações em RT-LOTOS para grafos temporizados e pela descrição das propriedades a verificar na forma de fórmulas da lógica TCTL. A verificação da satisfação ou não dessas fórmulas sobre o modelo grafos temporizados do sistema permite concluir a respeito das propriedades do sistema representado.

5 Definindo RT-LOTOS como Grafos Temporizados

Nesta seção é apresentada a definição formal da semântica de RT-LOTOS no formalismo Grafo Temporizado que servirá de base para a transformação de um formalismo no outro. Inicialmente, os grafos temporizados são definidos formalmente. Em seguida, define-se a semântica operacional deste formalismo. Depois, a semântica de RT-LOTOS é definida em grafos temporizados. Finalmente, apresentam-se alguns pequenos exemplos para ilustrar a nova representação subjacente à RT-LOTOS.

5.1 Grafos Temporizados

No formalismo básico *Grafo Temporizado* [1] as mudanças de estado são realizadas por ações instantâneas e não existem transições relativas ao progresso do tempo.

Para modelar um sistema usando grafos temporizados, atribui-se a ele um conjunto finito de *nodos* que correspondem aos estados do sistema e um conjunto finito de *relógios* com valores tomados num dado domínio de tempo D arbitrário. As transições que um sistema pode realizar dependem dos valores dos relógios e são habilitadas a partir da verificação de condições sobre os relógios que expressam restrições de tempo do sistema. Simultaneamente à realização de uma transição, alguns dos relógios podem ser reinicializados. Em qualquer instante, a leitura de um relógio é igual ao tempo passado desde a última vez que ele foi reinicializado.

5.1.1 Definições

Definição 1 Seja A um vocabulário de ações no qual a denota um elemento de A. Uma tupla $T = (t_1, \cdots, t_n)$ de "*relógios*" é um conjunto de variáveis tomando seus valores em D. Representa-se por $\mathcal{B}(T)$ o conjunto dos predicados b sobre T, isto é, o conjunto das aplicações de D^n em $\{tt, ff\}$. Além disso, $\mathcal{F}(T)$ é o conjunto das aplicações f de D^n em D^n.

Um *Grafo Temporizado* é uma estrutura $G = (N, T, n_0, \mathtt{urg}, \rightarrow)$, onde

- N é um conjunto finito de *nodos*
- T é uma tupla de *relógios* (*timers*)
- $n_0 \in N$ é o *nodo inicial*
- $\mathtt{urg}$ é uma aplicação de N em $\{tt, ff\}$, chamada de *função de urgência*
- $\rightarrow \subseteq N \times A \times \mathcal{B}(T) \times \mathcal{F}(T) \times N$ é a relação de transição

A seguir, escreve-se $n \xrightarrow{a,b,f} n'$ ao invés de $(n, a, b, f, n') \in \rightarrow$. □

Os relógios de um grafo temporizado são variáveis que podem ser incrementadas "continuamente" e na mesma velocidade. Inicialmente, todos os relógios têm valor 0. Os nodos representam os estados de controle. De um nodo n, se a condição de sensibilização b de uma transição $n \xrightarrow{a,b,f} n'$ é satisfeita pelos valores dos relógios, então a transição *pode* ser executada. Isto é, a ação a pode ser realizada, e o estado de controle torna-se n' após a modificação dos valores dos relógios de acordo com a função f.

Definição 2 Define-se a notação $A(G)$ para representar o conjunto que define o vocabulário de ações de um dado grafo temporizado $G = (N, T, n_0, \mathtt{urg}, \rightarrow)$. Formalmente,

$$A(G) = \{a | \forall n, n' \in N,\ n \xrightarrow{a,b,f} n' \in \longrightarrow\}$$ □

Definição 3 Para qualquer predicado $b \in \mathcal{B}(T)$, o predicado $\hat{b}$ é definido por

$$\hat{b}(T) = (\exists d \in D)\ \ b(T + d)$$

onde $T + d = (t_1 + d, \cdots, t_n + d)$. Note-se que $\hat{b}$ é um predicado significando "eventualmente b", isto é, "b é verdade ou deverá sê-lo no futuro". □

Definição 4 Define-se a *condição de sensibilização de um nodo* $n \in N$, denotado por $\mathbf{en}(n)$:

$$\mathbf{en}(n) \stackrel{def}{=} \bigvee \left\{ b \mid n \xrightarrow{a,b,f} \right\}$$ □

Note-se que $\mathtt{en}(n)$ caracteriza os valores dos relógios para os quais uma transição pode ser executada a partir do nodo n.

Definição 5 A *condição de atividade* de um nodo n, denotada por $\mathtt{act}(n)$, é definida como:

$$\mathtt{act}(n) \stackrel{def}{=} \widetilde{\mathtt{en}(n)} \wedge \neg \mathtt{urg}(n) \qquad \square$$

O predicado $\mathtt{act}(n)$ é quem dita se uma dada transição *pode, não pode*, ou *deve* ser realizada para um dado valor de uma tupla de relógios.

5.1.2 Semântica Operacional de Grafos Temporizados

O predicado act é definido de maneira que o sistema possa permanecer no nó n no instante definido pelo valor de T somente se $\mathtt{act}(n)(T) = tt$. Se a condição de urgência $\mathtt{urg}(n)$ é ff, então não se exige realizações urgentes de transições no nodo n. Inversamente, $\mathtt{urg}(n) = tt$ fará com que o predicado $\mathtt{act}(n)$ seja ff e isto forçará a realização de uma transição de n, provocada pela supressão da progressão do tempo no modelo operacional de grafos temporizados.

Formalmente, o modelo operacional de um grafo temporizado, $G = (N, T, n_0, \mathtt{urg}, \rightarrow)$ é um sistema de transições rotuladas,

$$STR_G = \langle N \times T, A \cup D_{\bullet}, \{\stackrel{a}{\rightarrow} : a \in A(G)\}\rangle$$

Ou seja, os estados são pares (n, T), os rótulos são elementos de $A \cup D_{\bullet}$ (onde $D_{\bullet} \stackrel{def}{=} D - \{0\}$), e o estado inicial é $(n_0, \vec{0})$, onde n_0 é o nodo inicial do grafo temporizado. A relação de transição é definida pelas regras seguintes.

$$\frac{n \stackrel{a,b,f}{\longrightarrow} n' \ , \ b(T)}{(n,T) \stackrel{a}{\rightarrow} (n', f(T))} \qquad \frac{\mathtt{act}(n)(T+d)}{(n,T) \stackrel{d}{\rightarrow} (n, T+d)}$$

A primeira regra diz que uma transição é executável se o seu predicado de sensibilização é avaliado como sendo verdadeiro. A segunda regra formaliza a condição à satisfazer para permanecer em um nodo esperando d unidades de tempo.

5.2 Definindo RT-LOTOS como Grafos Temporizados

Nesta seção define-se como construir grafos temporizados $G[\![E]\!]$ para expressões de comportamento E de RT-LOTOS de maneira que $E \sim G[\![E]\!]$ (onde $\sim$ representa a relação equivalência forte). Mais precisamente, define-se uma álgebra de grafos temporizados sobre os operadores de RT-LOTOS.

5.2.1 Descrição informal do mapeamento RT-LOTOS/Grafos Temporizados

A seguir é apresentada uma rápida descrição informal de como a semântica de RT-LOTOS deve ser mapeada em grafos temporizados.

- Os nodos n de um grafo temporizado representam expressões de comportamento definidas em RT-LOTOS, nos seus vários estágios de seu processo de execução. Abstrai-se do tempo para esta representação.
- A tupla de relógios T associada a um grafo temporizado representam os tempos locais relativos às várias ações que estão sendo oferecidas no grafo da especificação RT-LOTOS.
- O nodo n_0 representa a especificação inicial em RT-LOTOS.
- O conjunto dos predicados $B(T)$ representa as diversas restrições de tempo impostas para a realização das transições.

5.2.2 As limitações do mapeamento RT-LOTOS/Grafos Temporizados

Deve-se observar que o mapeamento de RT-LOTOS em grafos temporizados não é total. Isto deve-se ao fato de que grafos temporizados só são capazes de representar sistemas com um número finito de estados (mesmo sendo este formalismo capaz de absorver todos os estados intermediários devidos às transições temporais). Em outras palavras um sistema só poderá ter uma representação em grafos temporizados se a especificação RT-LOTOS do sistema sem a sua dimensão temporal levar a um sistema de transições finito. Assim, existem expressões de comportamento RT-LOTOS que apesar de válidas não têm representação definida em grafos temporizados. Um exemplo de tal situação é o processo

$$P = Q|||i; P$$

Este processo não possui representação por um sistema de transições finito, e assim não é possível representá-lo como um grafo temporizado. Observe-se que esta é uma limitação que vale tanto para RT-LOTOS quanto para LOTOS. Em vista disto, as expressões de comportamento RT-LOTOS que podem ser definidas em grafos temporizados constituem-se num subconjunto próprio de RT-LOTOS que leva em conta as duas condições seguintes: toda expressão recursiva deve ser guardada; e toda sub-expressão da forma $A|[L]|B$ é tal que todas as derivações de A e de B não contenham $A|[L]|B$ como sub-expressão.

Estas condições são suficientes para excluir os comportamentos infinitos de LOTOS (e também de RT-LOTOS) e são as mesmas que foram estabelecidas e adotadas em [?, 11, 8] em mapeamentos de LOTOS (e extensões temporais de LOTOS) em outros sistemas de estados finitos.

5.2.3 Representação dos operadores RT-LOTOS em Grafos Temporizados

Para facilitar e padronizar as definições desta seção considerem-se $G_E = (N_E, T_E, n_E^0, \mathtt{urg}_E, \rightarrow_E)$ e $G_F = (N_F, T_F, n_F^0, \mathtt{urg}_F, \rightarrow_F)$ como dois grafos temporizados com o vocabulário $Act^{t,\delta} \cup Act^*$ e tal que $N_E \cap N_F = \emptyset$ e $T_E \cap T_F = \emptyset$. Também, $\mathcal{R}(T)$ denota a função de reinicialização do conjunto de *relógios* T; $\mathcal{R}(\emptyset)$ significa que nenhum relógio é reinicializado. Para simplificar a notação, representar-se-á por f a função que reinicializa todos os relógios de um grafo numa dada transição.

Devido as limitações de espaço e por ser similar a dos outros operadores, a translação dos operadores *ocultação*, *composição seqüencial*, *preempção* e *instanciação de processos* será omitida neste artigo. No entanto, ela é descrita det lhadamente em [5].

Inação: $G[\![stop]\!] = (N, T, n_0, \mathtt{urg}, \emptyset)$ com $N = \{n_0\}$, um único relógio $T = t$, $\mathtt{urg}(n_0) = ff$; e, evidentemente, não existe nenhuma relação de transição associada com $G[\![stop]\!]$.

Terminação com sucesso: $G[\![exit]\!] = (N, T, n_0, \mathtt{urg}, \rightarrow)$ com $N = \{n_0, n_1\}$, $T = t$ e $\mathtt{urg}(n_0) = \mathtt{urg}(n_1) = ff$ onde n_1 é um nodo sumidouro. A relação de transição $\rightarrow$ é definida como o menor conjunto que satisfaz: $n_0 \xrightarrow{\delta, t \geq 0} n_1$

Prefixação: Em termos de Grafos Temporizados para RT-LOTOS, as restrições de tempo são definidas a partir dos intervalos temporais associados a cada transição. Por exemplo, a prefixação $[t_1, t_2]a; E$ define o estabelecimento de um predicado do tipo $t_1 \leq t \leq t_2$ e teria-se desta maneira duas transições possíveis: uma representando a realização da ação a, isto é, $n \xrightarrow{a, t_1 \leq t < t_2, \{t\}} n'$ (para n' representando o grafo de E); e a transição $n \xrightarrow{a^*, t = t_2, \{t\}} n^*$ que representa a realização da ação a^* (para n^* representando um estado do tipo *sumidouro*, equivalente ao processo *stop* de RT-LOTOS).

Observe-se que as condições lógicas b que devem ser satisfeitas para realização de uma transição em um grafo temporizado são definidas a partir dos intervalos temporais associados às ações em operações de prefixação. Assim, os b's usados nas regras de transição, a serem apresentadas a seguir, correspondem exatamente às restrições de tempo impostas a realização de uma ação a através do intervalo temporal associado a esta ação.

Cada operação de prefixação possui um relógio t associado a ela. Este relógio é inicializado em zero no instante inicial do oferecimento da ação sendo prefixada, e é incrementado de acordo com a evolução do tempo. Este relógio é usado para definir formalmente os valores do predicado b associado a uma ação, e dos predicados $\mathtt{en}(n)$ e $\mathtt{urg}(n)$ associados aos nodos do grafo que representa a prefixação (como pode ser visto na tabela 1).

$[t_1,t_2]\alpha \equiv$	$\mathtt{urg}(n_0) =$	$\mathtt{en}(n_0) =$	$\widehat{\mathtt{en}(n_0)} =$	$\mathtt{act}(n_0) =$
$[0,0]i \equiv i$ ou $[0,0]a$	tt, para $t \geq 0$	tt, em $t = 0$ ff, para $t > 0$	tt	ff, para $t \geq 0$
$[0,\omega]i$ ou $[0,\omega]a \equiv a$	ff, para $t \geq 0$	tt, para $t \geq 0$	tt	tt, para $t \geq 0$
$[t_1,\omega]i$ ou $[t_1,\omega]a$	ff, para $t \geq 0$	ff, para $t < t_1$ tt, para $t \geq t_1$	tt	tt, para $t \geq 0$
$[t_1,t_2]i$ ou $[t_1,t_2]a$	ff, para $t < t_2$ tt, para $t \geq t_2$	ff, para $t < t_1$ tt, para $t_1 \leq t \leq t_2$ ff, para $t > t_2$	tt	tt, para $t < t_2$ ff, para $t \geq t_2$

Tabela 1: Valores de $\mathtt{urg}(n_0)$, $\mathtt{en}(n_0)$, $\widehat{\mathtt{en}(n_0)}$ e $\mathtt{Act}(n_0)$ para vários tipos de intervalos e ações

O Grafo Temporizado para a ação prefixada $\alpha \in Act^i$ é:
$G[\![[t_1,t_2]\alpha; E]\!] = (N, T, n_0, \mathtt{urg}, \rightarrow)$ com:
$N = N_E \cup \{n_0, n^*\}$ $(n_0, n^* \notin N_E)$
$T = T_E \cup \{t\}$, onde t é o relógio que registra a idade do oferecimento da ação α, e $t \notin T_E$.
$\mathtt{urg}(n) = \mathtt{urg}_E(n)$ se $n \in N_E$
$\mathtt{urg}(n_0) = ff$, $\mathtt{urg}(n^*) = ff$
A relação transição $\rightarrow$ é definida como o menor conjunto que satisfaz:

$$n_0 \xrightarrow{a,t_1 \leq t < t_2, \mathcal{R}(T_E)} n_E^0 \qquad n_0 \xrightarrow{a^*, t = t_2, \mathcal{R}(T_E)} n^* \text{ (para } a \neq i) \qquad \frac{n \in N_E \ , \ n \xrightarrow{a,b,f}_E n'}{n \xrightarrow{a,b,f} n'}$$

Escolha: O grafo temporizado é: $G[\![E[\,]F]\!] = (N, T, n_0, \mathtt{urg}, \rightarrow)$ com
$N = N_E \cup N_F \cup (N_E \times N_F)$
$T = T_E \cup T_F$
$n_0 = (n_E^0, n_F^0)$
$\mathtt{urg}(n) = \mathtt{urg}_j(n)$, onde $j \in \{E, F\}$ se $n \in N_j$
$\mathtt{urg}(n_E, n_F) = \mathtt{urg}(n_E) \vee \mathtt{urg}(n_F)$
A relação de transição $\rightarrow$ é o menor conjunto que satisfaz:

$$\frac{n \xrightarrow{a,b,f}_j n' \ , \ a \in Act^{i,\delta} \cup Act^*}{n \xrightarrow{a,b,f} n'} \qquad \frac{n_E \xrightarrow{a,b,f}_E n'_E \ , \ a \in Act^{i,\delta}}{(n_E^0, n_F^0) \xrightarrow{a,b \wedge \mathtt{act}(n_F^0),f} n'_E} \qquad \frac{n_F \xrightarrow{a,b,f}_F n'_F \ , \ a \in Act^{i,\delta}}{(n_E^0, n_F^0) \xrightarrow{a,b \wedge \mathtt{act}(n_E^0),f} n'_F}$$

$$\frac{n_E \xrightarrow{a^*,b,f}_E n'_E}{(n_E, n_F) \xrightarrow{a^*, b \wedge \mathtt{act}(n_F), \mathcal{R}(\emptyset)} (n'_E, n_F)} \qquad \frac{n_F \xrightarrow{a^*,b,f}_F n'_F}{(n_E, n_F) \xrightarrow{a^*, b \wedge \mathtt{act}(n_E), \mathcal{R}(\emptyset)} (n_E, n'_F)}$$

Comentário: Observe-se que nas duas últimas regras (as que correspondem a evolução do sistema pela ação a^*) as funções de reinicialização de relógios são ambas iguais a $\mathcal{R}(\emptyset)$ o que corresponde a não reinicializar nenhum relógio. Isto foi feito para permitir que a semântica da realização de uma ação a^* no operador de escolha ficasse bem definida, isto é, a^* não decide uma escolha. Observe-se também que não existe necessidade de reinicializar os relógios do processo que realiza a ação a^* visto que o comportamento que a segue é sempre equivalente funcionalmente ao processo *stop*.

Composição Paralela: O grafo temporizado é: $G = [E][L][F] = (N, T, n_0, \mathtt{urg}, \rightarrow)$ com:

$N = N_E \times N_F$

$T = T_E \cup T_F$

$n_0 = (n_E^0, n_F^0)$

$\mathtt{urg}(n_E, n_F) = (\{a | n_E \xrightarrow{a,b,f}_E n'_E\} \cap \{a | n_F \xrightarrow{a,b,f}_F n'_F\} \subseteq L)$

Comentário: Observe-se que a definição do predicado urg acima impõe a urgência requerida às ações a sincronizar em um grafo temporizado somente quando ambos os nodos do grafo estão prontos para isto.

A relação de transição $\rightarrow$ é definida como o menor conjunto que satisfaz as regras apresentadas a seguir:

- Caso da ausência de sincronização:

$$\frac{n_E \xrightarrow{a,b,f}_E n'_E \ , \ a \in Act^i \cup Act^* - L}{(n_E, n_F) \xrightarrow{a,b \wedge \mathtt{act}(n_F),f} (n'_E, n_F)} \qquad \frac{n_F \xrightarrow{a,b,f}_F n'_F \ , \ a \in Act^i \cup Act^* - L}{(n_E, n_F) \xrightarrow{a,b \wedge \mathtt{act}(n_E),f} (n_E, n'_F)}$$

- Caso da presença de sincronização:

  - Ações normais:

$$\frac{n_E \xrightarrow{a,b_E,f_E}_E n'_E \ , \ n_F \xrightarrow{a,b_F,f_F}_F n'_F \ , \ a \in L \cup \{\delta\}}{(n_E, n_F) \xrightarrow{a,b_E \wedge b_F, f_E * f_F} (n'_E, n'_F)}$$

  - Violações temporais:

$$\frac{n_E \xrightarrow{a^*,b_E,f_E}_E n'_E \ , \ n_F \xrightarrow{a^*,b_F,f_F}_F n'_F \ , \ a \in L}{(n_E, n_F) \xrightarrow{a^*,b_E \wedge b_F, f_E * f_F} (n'_E, n'_F)}$$

$$\frac{n_E \xrightarrow{a^*,b_E,f_E}_E n'_E \ , \ \neg(\mathtt{en}(n_F/a) \vee \mathtt{en}(n_F/a^*)) \ , \ a \in L}{(n_E, n_F) \xrightarrow{a^*,b_E \wedge \mathtt{act}(n_F),f_E} (n'_E, n_F)}$$

$$\frac{n_F \xrightarrow{a^*,b_F,f_F}_F n'_F \ , \ \neg(\mathtt{en}(n_E/a) \vee \mathtt{en}(n_E/a^*)) \ , \ a \in L}{(n_E, n_F) \xrightarrow{a^*,b_F \wedge \mathtt{act}(n_E),f_F} (n_E, n'_F)}$$

Nas regras acima, $f_E \star f_F$ denota uma função que age como f_E em T_E e como f_F em T_F.

Preempção temporal: O grafo temporizado é: $G = [E < L]\{\bigcup_j (a_j : Q_j)\}] = (N, T, n_0, \mathtt{urg}, \rightarrow)$ com:

$N = N_E \cup (\bigcup_j N_j) \cup (N_E \times N_1 \times \cdots N_k)$ onde $j \in \{1, \cdots, k\}$

$T = T_E \cup (\bigcup_j T_j) \cup \{t\}$
$n_0 = (n_E^0, n_1^0, \cdots, n_k^0)$
$\mathbf{urg}(n) = \mathbf{urg}_l(n)$ se $n \in N_l$ e onde $l \in \{E\} \cup \{1, \cdots, k\}$
$\mathbf{urg}(n_E, n_1^0, \cdots, n_k^0) = \mathbf{urg}(n_E)$, onde $|L| = k$

Note-se que o grafo temporizado correspondente ao processo Q_j é chamado $G[\![Q_j]\!] = G_j = (N_j, T_j, n_j^0, \mathbf{urg}_j, \rightarrow_j)$ onde $j \in \{1, \cdots, k\}$.

A relação de transição $\rightarrow$ é definida como o menor conjunto que satisfaz:

$$\frac{n \xrightarrow{a,b,f}_l n' \ , \ a \neq \delta}{n \xrightarrow{a,b,f} n'} \qquad \frac{n_E \xrightarrow{a,b,f}_E n'_E \ , \ a \notin Act^*}{(n_E, n_1^0, \cdots, n_k^0) \xrightarrow{a,b,f} (n'_E, n_1^0, \cdots, n_k^0)}$$

$$\frac{n_E \xrightarrow{a_j^*,b,f}_E n'_E}{(n_E, n_1^0, \cdots, n_k^0) \xrightarrow{a_j^*,b,f} n_j^0} \qquad \frac{n_E \xrightarrow{\delta,b,f}_E n'_E}{(n_E, n_1^0, \cdots, n_k^0) \xrightarrow{\delta,b,f} n'_E}$$

Comentário: Na penúltima regra, o a_j^* denota a ocorrência de uma violação temporal em uma ação $a_j \in L$. Desta maneira, o comportamento que segue essa violação temporal é o abandono da execução do grafo $G[\![E]\!]$ e o início da execução do grafo $G[\![Q_j]\!]$ que está associado à ação a_j.

Embora seja longa e detalhada, a prova da consistência da tradução de RT-LOTOS para Grafos Temporizados não é difícil visto que não foi necessário introduzir nenhum elemento semântico adicional para a definição da translação. Assim, pode-se provar (apesar desta prova não ser apresentada aqui) que um processo E descrito em RT-LOTOS e seu grafo temporizado (com os relógios inicializados em 0) são fortemente equivalentes, isto é $E \sim (G[\![E]\!], \vec{0})$

5.3 Alguns exemplos ilustrativos

Login Para iniciar o procedimento, o sistema envia um *prompt* de *login p*. Após este evento, o usuário tem l unidades de tempo para entrar com um nome válido v para terminar a tarefa e iniciar a fase sessão S. Quando o usuário fornece uma resposta inválida n, ou quando o tempo l expira, um novo procedimento de login é inicializado. Além disso, o sistema limita a duração global do procedimento de login em um tempo máximo de d unidades de tempo e força uma exceção E a realizar-se após este tempo. A especificação em RT-LOTOS do procedimento *login*, chamada de `Login`, é apresentada a seguir. Utiliza-se nesta especificação o operador de preempção de RT-LOTOS juntamente com o processo `Login2`. O processo `Login2` é bastante semelhante ao processo `Login`, a única diferença está na maneira com que o processo S é acionado após a realização da ação v. Em `Login` (definido anteriormente) isto é realizado por uma prefixação simples, enquanto que em `Login2` utiliza-se o processo `exit` associado com o operador de composição seqüencial. `Login2` é definido assim para permitir que a realização da ação δ do processo `exit` possa "cancelar" o operador de preempção.

```
Login:= (Login2(p,n,v) [> [d]i; E) >> S
where
   process  Login2(p,n,v):= p; (v; exit [] [0,l]n; Login2(p,n,v))
                             <n] Login2(p,n,v)
   endproc
```

Na figura 1 é mostrado o grafo temporizado do procedimento de login completo correspondendo à especificação RT-LOTOS anterior. Como o processo `Login` não é seqüencial, o grafo não pôde ser construído com um único relógio. São necessários dois relógios para sua construção: t_1, com um limite superior igual a d, que limita a duração global do processo `Login`; e o relógio t do processo `Login2`. Como a função f é sempre a função de reinicialização, escreve-se o conjunto dos relógios a reinicializar no lugar da função.

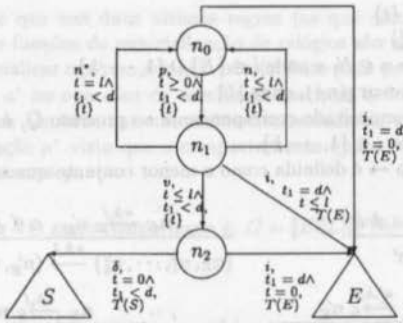

Figura 1: Grafo temporizado de um procedimento de *login* completo

Fluxo de vídeo com dependência entre unidades de informação A especificação RT-LOTOS foi apresentada na seção 3.3. O processo "`video`" apresentado anteriormente pode ser transformado a partir das regras anteriores no grafo temporizado da figura 2.

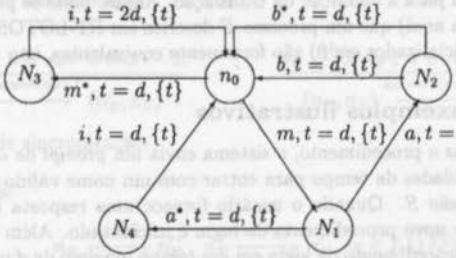

Figura 2: Grafo temporizado do processo `video`

Um processo com dois relógios Apresenta-se por último um terceiro exemplo para ilustrar a composição paralela: o processo P resulta da composição paralela de dois processos elementares e se escrevem:

```
P := a;[0,3]b;E |[b]| c;[0,5]b;F
```

Os processos elementares são representados pelos grafos temporizados da figura 3 ao passo que o resultado da sua composição aparece na figura 4. Pode-se notar que tem-se dois relógios t_1 et t_2 correspondendo a cada processo elementar da composição paralela. Além disso, pode-se observar que a partir do nodo (n_1^1, n_1^2), a ação b é imediata a partir do instante em que as condições sobre os dois relógios sejam satisfeitas, pois neste instante a função $\mathbf{urg}(n_1^1, n_1^2)$ tem o valor tt; esta representação corresponde bem à semântica de RT-LOTOS.

5.4 O ambiente em desenvolvimento

A definição de um esquema de tradução de RT-LOTOS para grafos temporizados possibilitará na abordagem proposta utilizar o método de verificação de modelos que permitem testar a satisfação ou não de propriedades descritas em Lógica Temporal Tempo-Real TCTL sobre os grafos temporizados obtidos a partir das especificações RT-LOTOS. Para suportar o ambiente proposto, um ambiente de especificação e verificação para sistemas dependentes do tempo está

sendo construído. Ele contém as seguintes ferramentas: editor de especificação de sistemas em RT- LOTOS,tradutor em Grafos Temporizados destas especificações, verificador das propriedades temporais especificadas em TCTL e um simulador para obtenção de cenários.

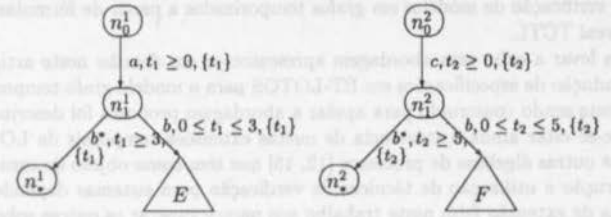

Figura 3: Grafos temporizados dos processos elementares

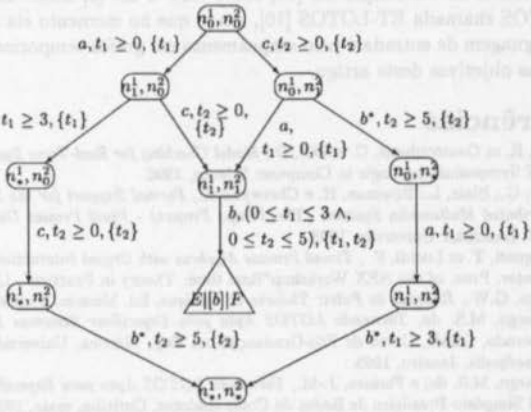

Figura 4: Grafo temporizado do processo resultante

Recentemente, foi desenvolvido pelo laboratório Verimag na França uma ferramenta chamada KRONOS [8] que implementa algoritmos para verificação de modelos de TCTL em grafos temporizados. Esta ferramenta está sendo incluída no ambiente que estamos desenvolvendo. O tradutor que permite transformar as especificações RT-LOTOS em grafos temporizados está sendo implementado neste trabalho, sendo que o desenvolvimento do simulador o será numa etapa posterior.

O objetivo deste ambiente é de formar junto com uma abordagem de especificação/verificação, um conjunto de ferramentas automatizadas que permitem prover o tratamento das especificações de aplicações reais com dependências temporais.

6 Conclusão

Foi apresentada neste artigo uma proposta de abordagem para especificação e verificação de sistemas dependentes do tempo. Esta abordagem consiste em utilizar uma linguagem de especificação formal com alta expressividade para descrever o comportamento de sistemas e utilizar um método de verificação de modelos de uma lógica temporal tempo-real com, também, alta

expressividade e poder de análise. A linguagem de especificação usada foi uma extensão temporal de LOTOS, chamada RT-LOTOS, cujos aspectos sintáticos e semânticos foram brevemente apresentados. Como método para verificação de modelos optou-se pela utilização da abordagem de verificação de modelos em grafos temporizados a partir de fórmulas de lógica temporal tempo-real TCTL.

Para levar a cabo esta abordagem apresentou-se em detalhe neste artigo uma sistemática para tradução de especificações em RT-LOTOS para o modelo grafo temporizado. Finalmente, o ambiente sendo construído para apoiar a abordagem proposta foi descrito sucintamente

Deve-se citar ainda a existência de outras extensões temporais de LOTOS [3, 10, 14] ou ainda de outras álgebras de processos [12, 15] que têm como objeto de estudo a representação, a construção e utilização de técnicas de verificação para sistemas dependentes do tempo. A proposta de extensão feita neste trabalho nos parece superar as outras sobres alguns aspectos, em particular no que diz respeito a descrição de restrições temporais e ao seu tratamento.

No que se refere a verificação das especificações pelo método proposto, ele se encaixa na linha dos trabalhos de verificação em [12] sobre ATP e em [8] sobre uma outra extensão temporal de LOTOS chamada ET-LOTOS [10], sendo que no momento ela se diferencia principalmente pela linguagem de entrada e pelo mapeamento em grafos temporizados proposto cuja descrição é um dos objetivos deste artigo.

Referências

[1] Alur, R. et Courcoubetis, C. et Dill, D., *Model Checking for Real-Time Systems*, In Proceedings of the Fifth IEEE Symposium on Logic in Computer Science, 1990.

[2] Blair, G.; Blair, L.; Bowman, H. e Chetwynd A., *Formal Support for the Specification and Construction of Distributed Multimedia Systems (The Tempo Project) - Final Project Deliverable*, Internal Report MPG-93-23, Lancaster University, 1993.

[3] Bolognesi, T. et Lucidi, F. , *Timed Process Algebras with Urgent Interactions and a Unique Powerful Binary Operator*, Proc. of the REX Workshop"Real time: Theory in Practice", LNCS 600, Springer Verlag, 1991.

[4] Brams, G.W., *Réseaux de Petri: Théorie et Pratique*, Ed. Masson, Tomos 1 e 2, Paris, 1983.

[5] Camargo, M.S. de, *Tornando LOTOS Apta para Especificar Sistemas Dependentes do Tempo*, Tese de Doutorado, LCMI, Curso de Pós-Graduação em Eng. Elétrica, Universidade Federal de Santa Catarina, Florianópolis, Janeiro, 1995.

[6] Camargo, M.S. de; e Farines, J.-M., *Tornando LOTOS Apta para Especificar Sistemas Tempo-Real*, Anais do 12 Simpósio Brasileiro de Redes de Computadores, Curitiba, maio, 1994.

[7] Courtiat, J.P. et de Camargo, M.S. et Saidouni, D.E. *RT.LOTOS: LOTOS Temporisé pour la Specification de Systèmes Temps Réel*, CFIP'95 - Ingénierie des Protocoles, Eds.: R. Dssouli et G. von Bochmann, Montréal, Setembro, 1993, HERMES, Paris, 1993, pags. 427-441.

[8] Daws, C.; Olivero, A. e Yovine, S., *Verifying ET-LOTOS programs with KRONOS*, In Proc. of FORTE'94, Berne, Outubro, 1994.

[9] *LOTOS, A Formal Description Technique Based on the Ordering of Observational Behaviour*, ISO IS 8807, Novembro, 1988.

[10] Leduc, G. et Léonard, L. *A Timed LOTOS Supporting a Dense Time Domain and Including New Timed Operators*, Proc. of the 5th International Conference on Formal Description Techniques, FORTE '92, North-Holland, 1993.

[11] Marsan, M.A. et ali., *A LOTOS Extension for the Performance Analysis of Distributed Systems*, IEEE/ACM Transactions on Networking, Vol.2, No.2, Abril, 1994, pp. 151-165.

[12] Nicollin, X. et Sifakis, J. et Yovine, S., *From ATP to Timed Graphs and Hibrid Systems*, Proceedings of the REX Workshop, Mook, The Netherlands, Lecture Notes in Computer Science No. 600, Springer-Verlag, 1992, pp. 549-572.

[13] Ostroff, J.S., *Formal Methods for the Specification and Design of Real-Time Safety Critical Systems*, Journal Systems Software, Vol. 18, 1992, pp. 33-60.

[14] Quemada, J., et Azcorra, A., et Frutos, D., *A Timed Calculus for LOTOS*, Proc. of the 2nd International Conference on Formal Description Techniques, FORTE '89, North-Holland, 1990.

[15] Yi, Wang, *A Calculus of Real Time Systems*, Ph.D. Thesis, Department of Computer Sciences, Chalmers University of Technology, Göteborg, Sweden, 1991.

[16] Yovine, S., *Méthodes et Outils pour la Vérification de Systèmes Temporisés*, Thèse de Docteur de l'Institut National Polytecnique de Grenoble, França, Maio, 1993.