



Análise Estado Estacionário ISPN

Sérgio Galdino

Departamento de Estatística e Informática, Universidade Católica de Pernambuco
Rua do Príncipe, 526 - Boa Vista, CEP: 50.050-900 - Recife/PE - Brasil
E-mail: segal@dei.unicap.br

Paulo Maciel

Centro de Informática, Universidade Federal de Pernambuco
50.732-970 PO Box 7851, Recife, Brasil
E-mail: prmm@cin.ufpe.br

Resumo Este artigo discute aspectos de modelos intervalares Markovianos escondidos. Um modelo intervalar é um espaço, família ou classe de modelos no qual há parâmetros representados por intervalos em vez de números reais. Nós descrevemos ISPN (*Interval Stochastic Petri Net*) como ferramenta de modelagem de alto-nível compondo o *framework* para modelagem e análise. A análise de modelos ISPN leva em conta os efeitos de variabilidades nas taxas de transições exponenciais e nos pesos das transições imediatos no cálculo de índices. O estudo de casos apresentados serve para descrever os modelos e apresentar os métodos de análise estado estacionário. ISPN pode ser útil em avaliação de desempenho e dependabilidade de sistema em fase inicial de projeto para análise simultânea de variabilidades de múltiplos parâmetros do sistema.

Palavras-chave Modelo intervalar, avaliação de desempenho, rede de Petri estocástica.

Introdução

Muitos sistemas do mundo real são complexos e mal-definidos para serem modelados por procedimentos determinísticos convencionais. Eles frequentemente contêm informações e características que são vagas, imprecisas, qualitativas, linguísticas ou incompletas [39, 17, 22, 31].

Técnicas analíticas são frequentemente usadas na análise de desempenho de sistemas de eventos discretos. Os modelos analíticos usam parâmetros de valores simples (tal como demanda média do recurso) e produz um único ponto medido para cada índice de desempenho de interesse (tal como tempo médio de resposta e utilização média do recurso). Contudo, esta caracterização pontual de parâmetros é inadequada quando incertezas e variabilidades estão associadas com parâmetros do sistema. Um exemplo é engenharia de avaliação de desempenho de *software* que integra modelagem de desempenho em várias fases do projeto do *software*, como também na implementação e está recebendo grande atenção [7, 40]. Mesmo que incertezas e variabilidades são associadas a um ou mais parâmetros do sistema nos estágios primários do projeto do sistema, o projetista experiente deve ter uma boa idéia sobre o intervalo de valores associados com estes parâmetros devido a experiência

anteriores com sistemas similares.

Modelos intervalares aplicados a engenharia estrutural [8] e análise de circuitos [18] estão entre as muitas aplicações de modelos intervalares. A home-page do MiceLab <http://mice.udg.es/> (25 setembro 2007) é dedicada a modelos intervalares. Existem referências a muitos trabalhos publicados. Destaca-se ainda o projeto COPRIN : www.sop.inria.fr/coprin/ (25 setembro 2007)

O livro de Jaulin et al [13] com subtítulo; '*with Examples in Parameter and State Estimation, Robust Control and Robotics*' apresenta análise intervalar no contexto de suas aplicações. Enquanto existe um bom número de livros [11, 14, 9, 10, 15, 30, 28, 26] caracterizando intervalos no contexto de computações numéricas e otimização, todos eles permanecem dentro dos limites estritos da matemática: a única exceção é o livro de Kolev [18] sobre métodos intervalares em análise de circuitos. Portanto, o livro [13] é uma contribuição valiosa para atravessar o fosso que existe entre teoria, ferramentas e aplicações.

A disponibilidade de boas ferramentas de *software* para avaliação de desempenho que permita esconder a cadeia de Markov que está sendo usado é fundamental para o usuário final que trabalha com modelagem de sistemas de eventos discretos. O usuário especifica seu modelo usando uma linguagem de modelagem de alto nível, tal como rede de filas, redes de Petri estocásticas ou álgebra de processo estocástica da qual a cadeia de Markov subjacente ao modelo é automaticamente gerada e analisada. Nós propomos o uso de intervalo para representar as incertezas nos parâmetros dos modelos ISPN (Interval Stochastic Petri Nets) [4, 5, 6] na análise de sistemas de eventos discretos. A aritmética intervalar é usada na solução numérica dos modelos. As taxas das transições exponenciais e os pesos das transições imediatas podem ter qualquer valor dentro de um intervalo.

1 Aritmética Intervalar

A aritmética intervalar é uma alternativa para determinar limites para o erro absoluto de um algoritmo, considerando todos erros de dados e arredondamento [27]. A aritmética intervalar será discutida com mais detalhes nas subseções seguintes. A aritmética intervalar faz cálculos sistemáticos através de intervalos

$[x] = [\underline{x}, \bar{x}]$ limitados de números de máquinas $\underline{x}, \bar{x} \in \mathbb{F}$, em vez de números reais simples x . As operações aritméticas $+$, $-$, \times , \div são definidas através de intervalos. Algoritmos intervalares são implementados em computadores e produzem resultados intervalares garantindo conterem a solução desejada.

A aritmética intervalar deve ser usada com bastante cuidado, caso contrário os resultados confiáveis de limites de erro serão obidos, na maioria das vezes, com muito pessimismo.

1.1 Notação

Neste artigo todas matrizes são denotadas por letras maiúsculas em negrito (**A**), vetores por letras minúsculas em negrito (**a**), e escalar e variáveis por letras minúsculas (*a*). Variáveis intervalares são delimitadas por colchetes (**[A]**, **[a]**, **[a]**). Sublinhado e barra superior denotam limites inferior e superior, respectivamente. Colchetes angulares $\langle \cdot, \cdot \rangle$ são usados por intervalos definidos por um ponto médio e um raio.

Um intervalo real $[x]$ é um conjunto não vazio de números reais

$$[x] = [\underline{x}, \bar{x}] = \{x \in \mathbb{R} : \underline{x} \leq x \leq \bar{x}\} \quad (1)$$

onde \underline{x} e \bar{x} são chamados de *infimum* e *sepreum*, respectivamente.

O conjunto de todos intervalos \mathbb{R} é denotado por $I(\mathbb{R})$ onde

$$I(\mathbb{R}) = \{[\underline{x}, \bar{x}] : \underline{x}, \bar{x} \in \mathbb{R} : \underline{x} \leq \bar{x}\} \quad (2)$$

O ponto médio de $[x]$ e o raio de x são definidos como

$$mid[x] = \hat{x} = \frac{1}{2}(\underline{x} + \bar{x}) \text{ e } rad[x] = \Delta x = \frac{1}{2}(\bar{x} - \underline{x}). \quad (3)$$

O raio pode ser usado para definir um intervalo $[x] \in I(\mathbb{R})$. Então, $\langle m, r \rangle$ denota um intervalo com ponto médio m e raio r . $[x, x] \equiv x$ é chamado intervalo pontual ou intervalo fino. Um intervalo pontual ou fino tem raio zero, enquanto um intervalo espesso tem um raio maior que zero.

1.2 Operações Básicas da Aritmética Intervalar

As operações elementares sobre intervalos $[a] = [\underline{a}, \bar{a}]$ e $[b] = [\underline{b}, \bar{b}]$ são calculadas explicitamente como:

1. $[a, \bar{a}] + [\underline{b}, \bar{b}] = \left[\underline{a} \overset{\nabla}{+} \underline{b}, \bar{a} \overset{\Delta}{+} \bar{b} \right]$
2. $[a, \bar{a}] - [\underline{b}, \bar{b}] = \left[\underline{a} \overset{\nabla}{-} \bar{b}, \bar{a} \overset{\Delta}{-} \underline{b} \right]$
3. $[a, \bar{a}] \cdot [\underline{b}, \bar{b}] = \left[\min(\underline{a} \overset{\nabla}{\cdot} \underline{b}, \underline{a} \overset{\nabla}{\cdot} \bar{b}, \bar{a} \overset{\nabla}{\cdot} \underline{b}, \bar{a} \overset{\nabla}{\cdot} \bar{b}), \max(\underline{a} \overset{\Delta}{\cdot} \underline{b}, \underline{a} \overset{\Delta}{\cdot} \bar{b}, \bar{a} \overset{\Delta}{\cdot} \underline{b}, \bar{a} \overset{\Delta}{\cdot} \bar{b}) \right]$

$$4. [\underline{b}, \bar{b}]^{-1} = \left[1 \overset{\nabla}{/} \bar{b}, 1 \overset{\Delta}{/} \underline{b} \right] \text{ if } 0 \notin [\underline{b}, \bar{b}]$$

$$5. [\underline{a}, \bar{a}] / [\underline{b}, \bar{b}] = [\underline{a}, \bar{a}] \cdot [\underline{b}, \bar{b}]^{-1}$$

Os símbolos ∇ e Δ sobre os operadores indicam operações aritméticas arredondadas dirigidas para cima e para baixo, respectivamente. Estas definições contêm cada possível número que pode ser formado como resultado de cada operação ($+$, $-$, \times , \div) para todo $\underline{a} \in [\underline{a}, \bar{a}]$ e para todo $\bar{b} \in [\underline{b}, \bar{b}]$. Além destas operações, funções intervalares (\exp , \log , \sin , \cos , ...) retornam limites superiores e inferiores.

As operações da aritmética intervalar são definidas para cálculos exatos [27]. Computações de máquinas são afetados por erro de arredondamento. Contudo, as formulas estão modificadas de maneira a considerar o chamado arredondamento direto [20].

1.3 Teorema Fundamental

Um importante resultado é a propriedade de inclusão, freqüentemente chamado de teorema fundamental da análise intervalar [27].

Teorema 1 (Teorema Fundamental) Se a função $f([x]_1, [x]_2, \dots, [x]_n)$ é uma expressão com um número finito de intervalos $[x]_1, [x]_2, \dots, [x]_n \in I(\mathbb{R})$ e operações intervalares ($+$, $-$, \times , \div), e se $[w]_1 \subseteq [x]_1, \dots, [w]_n \subseteq [x]_n$ então $f([w]_1, [w]_2, \dots, [w]_n) \subseteq f([x]_1, [x]_2, \dots, [x]_n)$.

1.4 Métodos de Solução para Sistemas de Equações Intervalares

A solução de sistemas de equações lineares é sujeita a erros devido a precisão finita da aritmética da máquina e a propagação de erros relacionados aos dados iniciais. Se o dado inicial é conhecido como pertencente a intervalos de variações especificadas, então a aritmética intervalar estabelece o cálculo de intervalos que contém o valor exato [12].

Um sistema de equações lineares é da forma

$$[\mathbf{A}][\mathbf{x}] = [\mathbf{b}] \quad (4)$$

onde $[\mathbf{A}] \in I(\mathbb{R})^{n \times n}$ e $[\mathbf{b}] \in I(\mathbb{R})^n$.

Quando a matrix- $n \times n$, $[\mathbf{A}] = [\underline{\mathbf{A}}, \bar{\mathbf{A}}]$, e o vetor- n , $[\mathbf{b}] = [\underline{\mathbf{b}}, \bar{\mathbf{b}}]$, no sistema $[\mathbf{A}] \cdot [\mathbf{x}] = [\mathbf{b}]$ são intervalos, ele é usualmente ainda chamado de um sistema de equações lineares intervalar [30], [32]. Falando precisamente, eles não são usualmente considerados um sistema de equações. O espaço de um intervalo real não é um espaço linear, mas o sistema tem a forma de uma expressão linear. O uso da palavra *equação* é somente justificado quando se considera a solução algébrica do sistema, o qual é raramente o caso. Em muitos casos, outras definições são consideradas, usualmente conjuntos de vetores reais (não necessariamente intervalos),

definidos como segue (ver e.g. [36]):

Conjunto solução unida USS (United Solution Set) = $\Sigma_{\exists\exists}([A], [b])$:

$$USS = \left\{ \tilde{x} \in \mathbb{R}^n \mid (\exists \tilde{A} \in [A]) (\exists \tilde{b} \in [b]) (\tilde{A} \cdot \tilde{x} = \tilde{b}) \right\} \quad (5)$$

este conjunto solução é feito de todas soluções de cada sistema pontual $\tilde{A} \cdot \tilde{x} = \tilde{b}$, onde $=$ é um símbolo para $\tilde{A} \cdot \tilde{x} \cap \tilde{b} \neq \emptyset$ com $\tilde{A} \in [A]$ e $\tilde{b} \in [b]$, isto é, calculando $\min \{ \tilde{x}_k \mid \tilde{x} \in \Sigma_{\exists\exists}([A], [b]) \}$ e $\max \{ \tilde{x}_k \mid \tilde{x} \in \Sigma_{\exists\exists}([A], [b]) \}$, $k = 1, 2, \dots, n$.

Conjunto solução controle CSS (Control Solution Set) = $\Sigma_{\exists\forall}([A], [b])$:

$$CSS = \left\{ \tilde{x} \in \mathbb{R}^n \mid (\forall \tilde{b} \in [b]) (\exists \tilde{A} \in [A]) (\tilde{A} \cdot \tilde{x} = \tilde{b}) \right\} \quad (6)$$

Conjunto solução tolerância TSS (Tolerance Solution Set) = $\Sigma_{\forall\exists}([A], [b])$:

$$TSS = \left\{ \tilde{x} \in \mathbb{R}^n \mid (\forall \tilde{A} \in [A]) (\exists \tilde{b} \in [b]) (\tilde{A} \cdot \tilde{x} = \tilde{b}) \right\} \quad (7)$$

Estes conjuntos soluções generalizados originam-se na existência de muitas aplicações significantes e interessantes. As soluções acima não são soluções do sistema original. Elas são conjuntos de soluções reais de expressões relacionais intervalares, com diferentes relações colocadas no lugar do sinal de igualdade. Usando esta convenção a equação $[A] \cdot [x] = [b]$ tem um conjunto solução $\Sigma_{=}([A], [b])$ igual a $\Sigma_{\exists\forall}([A], [b]) \cap \Sigma_{\forall\exists}([A], [b])$ diferente da solução algébrica. Destas definições segue também que $\Sigma_{\forall\exists}([A], [b]) \subseteq \Sigma_{\exists\exists}([A], [b])$ e $\Sigma_{\exists\forall}([A], [b]) \subseteq \Sigma_{\exists\exists}([A], [b])$.

Não existe um nome abreviado para este tipo de fórmula no qual não somente uma igualdade, mas qualquer outra relação pode estar de acordo como um conectivo principal. O termo “desigualdade” não é apropriado, pois não queremos excluir igualdade. Kulpa [21] usou a frase “expressão relacional”, ou algumas vezes simplesmente “relação”. Contudo, será mantido aqui o uso da frase sistema de equações lineares intervalares.

1.4.1 Conjunto solução unida USS

O conjunto solução unida USS forma um politopo convexo com $2(n^2 - n + 1)$ vértices. O cálculo do intervalo involutório (“hull interval”) necessita de um algoritmo “NP-hard” [19]. Soluções aproximadas requerem usualmente algoritmos com complexidade polinomial somente (ver Figura 1) [30, 12, 11].

A forma da solução de um sistema linear intervalar pode ser bastante complexa, mas desde que a aritmética intervalar é usada, só é possível calcularmos um vetor intervalar que contém a solução verdadeira. O ótimo é conseguir um vetor intervalar solução que é próxima ao menor vetor intervalar contendo a solução verdadeira.

A solução ótima de um Sistema Linear de Equações Algébricas Intervalares é tema constante em pesquisas

Aproximações da USS

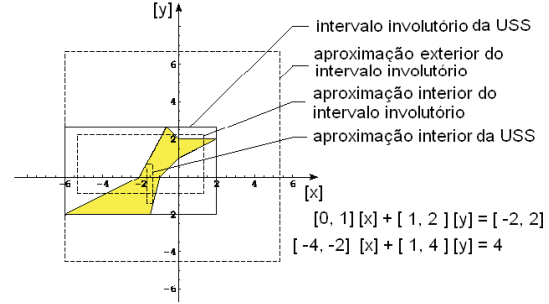


Figura 1: Soluções aproximadas da USS de um sistema intervalar de equações lineares.

recentes [16]. O chamado *problema externo (outer problem)* para um sistema de equações algébricas intervalares [38, 37]: encontrar um vetor intervalar com largura mínima contendo o conjunto $\Sigma([A], [b])$.

O vetor intervalar $[x]^H = ([x]_1^H, [x]_2^H, \dots, [x]_n^H)^T$ de largura mínima contendo $\Sigma([A], [b])$ é chamado intervalo involutório (*the interval hull*) do conjunto $\Sigma([A], [b])$ ou a solução intervalar ótima:

$$[x]_i^H = [x_i^H, \bar{x}_i^H], \quad \begin{aligned} x_i^H &= \min \{x_i \mid x \in \Sigma([A], [b])\} \\ \bar{x}_i^H &= \max \{x_i \mid x \in \Sigma([A], [b])\} \\ i &= 1, 2, \dots, n. \end{aligned}$$

Duas fontes importantes de algoritmos com técnicas de solução para sistemas de equações lineares intervalares são o livro de Neumaier [30] e o relatório de Hargreaves [12]. Hargreaves destaca a importância de um software eficiente para a aritmética intervalar, tal como o MATLAB toolbox INTLAB que foi considerado pelo mesmo na implementação de algoritmos tais como: Eliminação Gaussiana Intervalar, Método de Krawczyk e Método de Hansen-Bliek-Rohn-Ning-Kearfott-Neumaier. INTLAB fornece a função **verifylss** para solução de sistemas lineares de equações intervalares ou para dar soluções com limites verificados de um sistema pontual. O INTLAB desenvolvido pelo grupo de S.M. Rump [34, 35] é disponível livremente em: <http://www.ti3.tu-harburg.de/~rump/intlab/index.html> (25 setembro 2007).

2 Redes de Petri

Redes de Petri é atualmente um nome para uma classe de modelos que podem ser divididos em três níveis principais. O primeiro nível é o nível de pesquisa fundamental- um modelo representativo é o do sistemas de redes elementares. Para modelar sistemas da vida real, sistemas de redes elementares podem crescer em dimensão e tornar-se muito grande para ser manuseado efetivamente (o problema clássico da explosão de estados). O segundo nível permite-nos representar as car-

acterísticas repetitivas de sistemas de redes de Petri elementares [33] de maneira a obter-se modelos mais compactos. O modelo básico do segundo nível é sistemas local/transição (*place/transition systems*) [3]. O terceiro nível é o das redes de alto nível, onde é usada lógica e álgebra para produzir redes compactas adequadas a aplicações da vida real [33]. Redes de Petri coloridas e Redes Transição/Predicado (*predicate/transition nets*) são os melhores modelos de alto nível conhecidos [33]. Em muitas aplicações práticas é necessário considerar o tempo de execução e/ou processos estocásticos. Isto conduz a redes de Petri temporizada e estocástica [25]. Um texto introdutório sobre redes de Petri em geral que é muito usado como referência é o artigo de Murata [29].

2.1 ISPN

ISPN é uma extensão do modelo GSPN (*Generalized Stochastic Petri Nets*) de maneira a introduzir a análise intervalar [4] e já está sendo usada para avaliação de desempenho e dependabilidade (*dependability*) [4, 5, 6]. A GSPN é uma PN (**Petri Net**) temporizada particular que incorpora transições estocásticas temporizadas (traçadas como caixas brancas) e (traçadas como caixas pretas ou barras pretas finas). As transições temporizadas têm tempo de disparo exponencialmente distribuído e transições imediatas disparam em tempo zero. GSPNs foram originalmente definida em [23] e foi melhorada posteriormente [2]. Uma ISPN é também definida como uma óctupla como descrito a seguir:

$$ISPN = (P, T, I(\cdot), O(\cdot), H(\cdot), M_0, \pi(\cdot), [W](\cdot))$$

onde

- P é um conjunto finito de lugares,
- T é um conjunto finito de transições,
- $I, O, H : T \rightarrow \mathbb{N}$ representa as funções de entrada, saída e inibidoras,
- $M_0 : P \rightarrow \mathbb{N}$ é a marcação inicial (i.e., o número inicial de *tokens* em cada lugar),
- Adicionalmente, a definição ISPN compreende a função de prioridade $\pi(\cdot) : T \rightarrow \mathbb{N}$ a qual associa a menor prioridade(0) para as transições temporizadas e prioridades superiores (≥ 1) para as transições imediatas:

$$\pi(t) = \begin{cases} 0 & \text{se } T \text{ temporizada} \\ \geq 1 & \text{se } T \text{ imediata} \end{cases}, \quad (8)$$

- Finalmente, o último item da definição ISPN é a função $[W](\cdot) : T \rightarrow I(\mathbb{R})$ que associa um intervalo real à transições, onde $[w](t)$ é:

- um parâmetro intervalar da fdp do retardo de disparo da transição exponencial negativa, se t é uma transição temporizada,

- um peso intervalar usado para a computação da probabilidade de disparo da transição imediata, se t é uma transição imediata.

Se somente intervalos degenerados (pontos) estão definidos para as transições imediatas e exponenciais, nós teremos um modelo GSPN.

2.1.1 Análise de Desempenho ISPN

Uma ISPN é uma extensão da GSPN onde parâmetros intervalares devem ser considerados para obtenção do gerador infinitesimal intervalar do ICTMC (Interval Continuous-Time Markov Chain) [6]. Os algoritmos clássicos encontrados na literatura [1] são adaptados naturalmente para considerar os coeficientes intervalares do modelo de ISPN. A análise desempenho ISPN é realizada através de quatro sub-tarefas:

- geração do IERG (*Interval Extended Reachability Graph* - Grafo de alcançabilidade estendido intervalar),
- eliminação das marcações voláteis (*vanishing*) e as correspondentes transições de estado,
- análise estado estacionário intervalar,
- computação das métricas. Métricas tais quais o número médio de *tokens* em cada lugar ou conjunto de lugares e o *throughput* de transições temporizada são calculadas pela aritmética intervalar.

As ISPNs contêm transições imediatas e transições com tempos de disparo exponencialmente distribuídos, e pelo menos um intervalo espesso está presente em uma transição exponencial ou imediata. Isto resulta em várias possibilidades ao determinar quais transições são habilitadas. As marcações nas quais pelo menos uma transição imediata é habilitada são chamadas marcações voláteis. Por outro lado, marcações nas quais estão habilitadas só transições exponenciais são chamados marcações tangíveis da mesma forma que nas GSPN. Uma ISPN que contém marcações tangíveis e voláteis é equivalente a uma ICTMC. Neste caso, é chamada uma cadeia de Markov intervalar embutida. Se nós removermos as marcações voláteis, então teremos uma cadeia de Markov intervalar reduzida embutida. Esta cadeia que só contém marcações tangíveis é usada para computar as probabilidades intervalares de estado estacionário de um lugar que é marcado ou não.

De uma dada ISPN, um Grafo de Alcançabilidade Intervalar IERG é gerado contendo marcações como nós e informações estocásticas anotadas aos arcos, desta maneira, relacionado as marcações com as demais. O grafo de alcançabilidade ISPN é um grafo dirigido $RG(ISPN) = (V, E)$, onde $V = RS(ISPN)$ e

$$E = \{ \langle \mathbf{m}, t, \mathbf{m}' \rangle \mid \mathbf{m}, \mathbf{m}' \in RS(ISPN) \text{ e } \mathbf{m} \xrightarrow{t} \mathbf{m}', t \in T \}$$

são o conjunto de nós e arestas, respectivamente.

Se o sistema de rede ISPN é limitado, o $RG(ISPN)$ é finito e pode ser construído, por exemplo, baseado

no **Algoritmo 1** de [7]. A atividade definida no Passo 2.1 assegura que nenhuma marcação é visitada mais que uma vez. Cada marcação visitada é etiquetada (Passo 2.1), e o Passo 2.2.3 assegura que somente as únicas marcações adicionadas a V são aquelas que não foram previamente adicionadas. Quando uma marcação é visitada, somente aquelas arestas que representam o disparo de uma transição habilitada são adicionadas ao conjunto E (Passo 2.2.4).

Algoritmo1 (Geração do IERG)

Entrada - Um modelo ISPN livre de confusão.

Saída - Um grafo dirigido $RG(ISPN) = (V, E)$ de um sistema de rede limitado.

1. Inicializar $RG(ISPN) = (\{m_0\}, \emptyset)$; m_0 está não etiquetada;
2. **enquanto** existe um nó não etiquetado m em V **faça**
 - 2.1 Selecione um nó não etiquetado $m \in V$ e etiquete-o
 - 2.2 **para** cada transição habilitada t em m **faça**
 - 2.2.1 Calcule m' tal que $m \xrightarrow{t} m'$;
 - 2.2.2 **se** existe $m'' \in V$ tal que $m'' \xrightarrow{\sigma} m'$ e $m'' \leq m'$ **então** o algoritmo falha e termina;
(a condição de não limitação foi detectada)
 - 2.2.3 **se** não há nenhum $m'' \in V$ tal que $m'' = m'$ **então** $V := V \cup \{m'\}$; (m' é um nó não etiquetado)
 - 2.2.4 $E := E \cup \{(m, t, m')\}$
3. O algoritmo tem sucesso e $RG(ISPN)$ é o grafo de alcançabilidade estendido intervalar.

2.1.2 Eliminação de Marcações Voláteis

A eliminação de marcações voláteis é um passo para geração de um ICTMC de uma determinada ISPN. Uma vez que o IERG foi gerado, é transformado em uma ICTMC pelo uso de algoritmos matriciais eficientes [1].

As marcações $\mathcal{M} = \mathcal{V} \cup \mathcal{T}$ de um conjunto de alcançabilidade de uma ISPN é dividido em dois conjuntos, as marcações voláteis \mathcal{V} e as marcações tangíveis \mathcal{T} . Considere:

$$[P]^{\mathcal{V}} = [P]^{\mathcal{V}\mathcal{V}} \mid [P]^{\mathcal{V}\mathcal{T}} \quad (9)$$

denote a matriz intervalar onde

- $[P]^{\mathcal{V}\mathcal{V}}$ - probabilidades de transição intervalar entre marcações voláteis;
- $[P]^{\mathcal{V}\mathcal{T}}$ - probabilidades de transição intervalar de marcações voláteis para marcações tangíveis.

Além disso, considere:

$$[U]^{\mathcal{T}} = [U]^{\mathcal{T}\mathcal{V}} \mid [U]^{\mathcal{T}\mathcal{T}} \quad (10)$$

denotar uma matriz intervalar onde

- $[U]^{\mathcal{T}\mathcal{V}}$ - taxa de transição intervalar de marcações tangíveis para marcações voláteis;

- $[U]^{\mathcal{T}\mathcal{T}}$ - taxa de transição intervalar entre marcações tangíveis.

A mesma informação, como contido no IERG, é fornecido por $[P]^{\mathcal{V}}$ junto com $[U]^{\mathcal{T}}$.

Agora obtemos a matriz de taxa intervalar $[U]$. Esta matriz tem dimensão $|\mathcal{T}| \times |\mathcal{T}|$, onde \mathcal{T} denota o conjunto de marcações tangíveis.

$$[U] = [U]^{\mathcal{T}\mathcal{T}} + [U]^{\mathcal{T}\mathcal{V}}(1 - [P]^{\mathcal{V}\mathcal{V}})^{-1}[P]^{\mathcal{V}\mathcal{T}} \quad (11)$$

A matriz geradora infinitesimal intervalar é $[Q] = [q]_{ij}$, onde suas entradas são determinadas por:

$$[q]_{ij} = \begin{cases} [u]_{ij} & \text{if } i \neq j \\ -\sum_{\substack{k \in \mathcal{T} \\ k \neq i}} [u]_{ik} & \text{if } i = j \end{cases} \quad (12)$$

onde \mathcal{T} denota o conjunto de marcações tangíveis.

2.1.3 Análise estado estacionário ISPN

A análise estado estacionário do modelo ISPN requer a solução de um sistema de equações lineares intervalares com tantas equações quanto o número de marcações tangíveis. Uma vez a matriz geradora intervalar $[Q]$ do ICTMC associado com um modelo ISPN tenha sido derivada, nós podemos começar a análise estado estacionário ISPN para produzir medidas intervalares.

3 Estudo de Casos

Nesta seção apresenta-se um estudo de caso da análise de um modelo ISPN de um Sistema Paralelo. O estudo de caso tem o número marcações tangíveis+voláteis que varia entre 20+18 até 825+1620, respectivamente. As análises são realizadas com a ferramenta protótipo ISPN que utiliza a caixa de ferramentas INTLAB do MATLAB. As variabilidades das transições exponenciais, representadas por taxas intervalares, são introduzidas para produzir modelos ISPN não degenerados. A análise ISPN mostra os efeitos das variabilidades nos parâmetros das taxas de transição exponenciais sobre os índices de desempenho.

3.1 Análise de Modelos ISPN de um Sistema Paralelo

O modelo ISPN da Figura 2 é uma descrição do bem conhecido sistema paralelo simples [24] de computação. O ISPN inclui nove lugares e nove transições. As características das transições, deste modelo são resumidos na Tabela 1. As marcações iniciais são $(p_1 = N, N = 2, \dots, 8)$.

Na Tabela 2 é exibida a cardinalidade dos modelos gerados em função da marcação inicial. Os resultados apresentados na Tabela 3 mostram a propagação dos efeitos dos parâmetros intervalares com o crescimento do número de *tokens* no lugar P_1 no Processamento médio = número médio de *tokens* no lugar P_1 e no Processamento do sistema = $Prob\{(m(P_9) > 0)\}$. Os sistemas

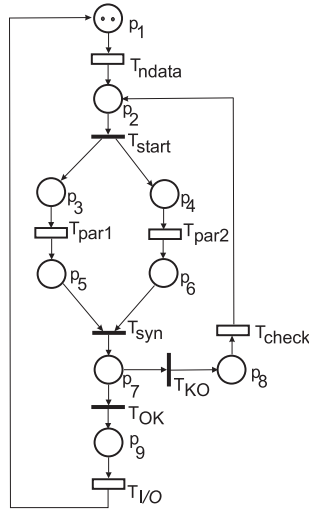


Figura 2: A descrição ISPN de um sistema paralelo

intervalares foram resolvidos pela função VERIFYLSS do INTLAB.

A Tabela 4 mostra várias funções (M-files) para métodos de solução de sistemas de equações lineares intervalares. Observa-se que os intervalos resultantes não diferem significativamente, porém as funções VERIFYLSS, HSOLVE e MKRAW têm tempos de solução semelhantes. Neste caso, com 196 equações lineares intervalares, os métodos VERENCLINTHULL, VERINTHULL e INTGAUSSP não têm tempo de solução competitivos com as funções VERIFYLSS, HSOLVE e MKRAW. A função MKRAW é uma versão modificada da KRAW onde os valores iniciais de cada componente do vetor são consideradas como $[0, 1]$. A função INTGAUSSP é uma versão modificada da função INTGAUSS onde é feito o pré-condicionamento da equação $[A][x] = [b]$ pela inversa do centro da matriz. O centro da matriz é dado por $\tilde{A}_{ij} = \frac{1}{2}(\underline{A}_{ij} + \bar{A}_{ij})$. Então resolve-se o sistema $[C][x] = [d]$ onde $[C] = \tilde{A}^{-1}[A]$ e $[d] = \tilde{A}^{-1}[b]$. O conjunto solução do sistema pré-condicionado $[C][x] = [d]$ contém o conjunto solução do sistema original $[A][x] = [b]$ [11].

A Tabela 5 mostra o tempo de processamento de algumas tarefas que são potencialmente críticas: a geração do espaço de estado/geração da matriz $[Q]$, a matriz inversa $(1 - [P]^{r/r})^{-1}$ e a solução do sistema linear intervalar. A Tabela 5 mostra que tempo de geração do espaço de estado/matriz $[Q]$ é muito crítico na ferramenta protótipo ISPN.

4 Conclusões

Neste artigo, a aritmética intervalar foi adotada para análise ISPN onde as taxas exponenciais e pesos de transições imediatos pertencem a alguns determinados intervalos. O ISPN é principalmente aplicado para mode-

Tabela 1: Especificações dos parâmetros das transições intervalares da ISPN da Figura 2 (SS = Servidor-Simples, SI = Servidor-Infinito)

Transição	Taxa Intervalar (h^{-1})	Semântica
t_{ndata}	$0.2 \pm 1e-4$	SI
t_{par1}	$2.0 \pm 1e-3$	SS
t_{par2}	$1.0 \pm 1e-3$	SS
t_{check}	$0.1 \pm 1e-4$	SS
$t_{I/O}$	$5.0 \pm 1e-3$	SS

Transição	Peso	Prioridade
t_{start}	1	1
t_{syn}	1	1
t_{KO}	0.01	1
t_{OK}	0.99	1

Tabela 2: Cardinalidade do espaço de estado para vários modelos ISPN do sistema paralelo de Figura 2. N é o número de *tokens* no lugar P_1 . A notação de T+V (cardinalidade) é o número de marcações *Tangíveis* = T e *Voláteis* = V.

N	2	3	4	5	6	7	8
T	20	50	105	196	336	540	825
V	18	60	150	315	588	1008	1620
T+V	38	110	255	511	924	1548	2445

lar situações nas quais dados de entrada é conhecido com certo nível de precisão. As taxas e os pesos podem ser subjetivamente especificados como intervalos. Este *framework* fornece um modo para formalizar e estudar problemas relacionados à presença de tais incertezas que podem incluir erros de dados ocorridos durante medidas e erros arredondando gerados durante cálculos. O modelo proposto e o método de análise relacionado permitem análise de performabilidade (*performability*) que considera variabilidade simultânea de parâmetros. Uma consequência imediata é que a análise ISPN pode ser útil para os engenheiros e técnicos como uma ferramenta para tomada de decisão. Como trabalhos futuros, métodos intervalares para análise de transiente serão consideradas e também simulação de Monte Carlo.

Referências

- [1] G. Bolch; S. Greiner and K.S. Trivedi: Queueing networks and Markov chains: modeling and performance evaluation with computer science applications. A Wiley-Inerscience publication, John Wiley and Sons, Inc, Second Edition,(2006)
- [2] Ed Brinksma, editor: Lectures on formal methods and performance analysis: revised lectures/ First EEF Summer School on Trends in Computer Science. Berg en Dal, The Netherlands, July 3 - 7, 2000. 9ed. Springer (2001)
- [3] J. Desel and W. Reisig: Place/Transition Petri Nets. In: Lectures on Petri Nets I: Basic Models. Advances in Petri Nets. Wolfgang Reisig and Grze-

Tabela 3: Avaliação de Desempenho e cardinalidade do espaço de estado para vários modelos ISPN do sistema paralelo de Figura 2. N é o número de *tokens* no lugar P_1 . Processamento médio = $E \{m(P_1 > 0)\}$ = número médio de *tokens* no lugar P_1 . Processamento do sistema = $Prob \{(m(P_9 > 0)\} \cdot \lambda_{I/O}$.

N	Processamento médio	Processamento do sistema
2	< 1.50555067214211, 0.00295064291426 >	< 0.30111056237629, 0.00219996135411 >
3	< 2.19169821603979, 0.00936230661157 >	< 0.43834084369878, 0.00609012205930 >
4	< 2.81824419811506, 0.02118586818893 >	< 0.56365121324037, 0.01198081656274 >
5	< 3.37218634785193, 0.04010364791493 >	< 0.67444122568113, 0.01991544118533 >
6	< 3.84125571003933, 0.06741766653902 >	< 0.76825707379898, 0.02981260579384 >
7	< 4.21692321527027, 0.10394270993835 >	< 0.84339293965107, 0.04165166201173 >
8	< 4.49775685335994, 0.14960261938225 >	< 0.89956237993883, 0.05522624449375 >

Tabela 4: **P4 2.4 GHz 256 MB RAM / Windows XP** Comparando algumas funções (*M-files*) para métodos de solução exterior de sistemas de equações lineares intervalares. O modelo do sistema paralelo da Figura 2, com $N=5$, tem 196 equações lineares intervalares (= 196 marcações tangíveis). Todas as funções são implementadas no MATLAB com a *toolbox* INTALAB.

Função	Tempo de Solução ⁵ (s)	Processamento médio/ Processamento do sistema
VERIFYLSS ¹	0.141±0.059	< 3.37218634785193, 0.04010364791493 > < 0.67444122568113, 0.01991544118533 >
VERENCLINTHULL ²	6.656±1.086	< 3.37222978038842, 0.04000713228195 > < 0.67444687057441, 0.01988753329495 >
VERINTHULL ²	1.939E+03±7.257E+01	< 3.37218671189105, 0.03933690766144 > < 0.67444101116486, 0.01960977054537 >
HSOLVE ³	0.128±0.038	< 3.37222978038857, 0.04000713228667 > < 0.67444687057445, 0.01988753330621 >
MKRAW ⁴	0.160±0.014	< 3.37245777796249, 0.03977913470763 > < 0.67734694762011, 0.01698745624921 >
INTGAUSSP ⁴	1.089E+02±6.086E+00	< 3.37222937457557, 0.04000753809456 > < 0.67444686136312, 0.01988754250620 >

gorg Rozenberg(Eds.). Springer, Berlin (1998) pp. 122-173

- [4] S.M.L. Galdino and P.R.M. MACIEL: Interval Generalized Stochastic Petri Net Models in Performance Evaluation In: IEEE International Conference on Systems, Man, and Cybernetics, 2006, Taiwan. Proceedings of 2006 IEEE International Conference on Systems, Man, and Cybernetics.
- [5] S.M.L. Galdino, P.R.M. MACIEL and N. S. Rosa: Interval Generalized Stochastic Petri Net Models in Dependability Evaluation. In: IEEE International Conference on Systems, Man, and Cybernetics, 2007, Montreal. Proceedings of 2007 IEEE International Conference on Systems, Man, and Cybernetics.
- [6] S.M.L. Galdino, P.R.M. MACIEL and N. S. Rosa: Interval Markovian Models in Dependability Evaluation. IJPAM accepted paper (2007)
- [7] C. Girault and R. Valk: Petri Nets for Systems Engineering: a guide to modeling, verification, and applications. Springer, (2003)
- [8] A. Del Grosso and A. Zucchini: Applications of Interval Arithmetic to the Analysis and Control of Structural Systems. Smart Materials and Structures, **2**, (1993) pp. 103-112
- [9] R. Hammer, M. Hocks, U. Kulisch, and D. Ratz: Toolbox for Verified Computing. Springer, Berlin, Heidelberg, (1993)
- [10] R. Hammer, M. Hocks, U. Kulisch, and D. Ratz: C++ Toolbox for Verified Computing, Springer, Berlin, Heidelberg, (1995)
- [11] E. R. Hansen & G. W. Walster. *Global Optimization Using Interval Analysis*. Second Edition, Revised and Expanded, Marcel Dekker, Inc., New York, 2004.
- [12] G.I. Hargreaves: Interval analysis in matlab. Numerical Analysis Report No. 416, University of Manchester, Reports available from: Department of Mathematics <http://www.ma.man.ac.uk/MCCM>, <http://www.ma.man.ac.uk/~nareports>. Manchester M13 9PL, England, December (2002)
- [13] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter: Applied Interval Analysis: with Examples in Parameter and State Estimation, Robust Control and Robotics. Springer-Verlag, (2001)
- [14] P.R. Kearfott and V. Kreinovich: Applications of interval computations. Kluwer, (1996)
- [15] R. Klatte, U. Kulisch, A. Wiethoff, C. Lawo and M. Rauch: C-XSC, Springer Verlag, (1992)
- [16] M. Lyashko: The optimal solution of an interval system of linear algebraic equations. Reliable Computing, 11:105-127, (2005)
- [17] G. J. Klir and B. Yuan: Fuzzy Sets and Fuzzy Logic: Theory and Applications. Prentice-Hall, PTR, Upper Saddle River, NJ, (1995)
- [18] L.V. Koley: Interval Methods for Circuit Analysis. World Scientific Publishing Co., River Edge, New Jersey (1993)

1 - <http://www.ti3.tu-harburg.de/rump/intlab/> .
2 - <http://www.uivt.cas.cz/~rohrn/matlab/index.html> .
3 - <http://www.maths.manchester.ac.uk/~hargreaves/downloadint.html> .
4 - Adaptado de (3) acima .
5 - média e desvio padrão de 10 repetições .

Tabela 5: **P4 2.4 GHz 256 MB RAM / Windows XP**
Tempos decorrido e tempos parciais para avaliação de desempenho com a ferramenta protótipo ISPN para o sistema paralelo da Figura 2. $N = m(P1)$.

Tempo	N	
	7	8
Tempo decorrido (hh:mm:ss)	00 : 05 : 58	00 : 34 : 28
Geração do Espaço de Estado/matriz Q (s)	$2.70e+02$	$1.63e+03$
Cálculo de $(1 - [P]^{N,N})^{-1}$ (s)	53.78	$3.46e+02$
Solução do sistema linear intervalar (s)	1.625000	4.219000

- [19] V. Kreinovich, G. Heindl and A. Lakeyev. Solving linear interval is np-hard even if we exclude overflow and underflow. *Reliable Computing*, 4:383–388, 1998.
- [20] U.W. Kulish and W.L. Miranker: The Arithmetic of the Digital Computers: A New Approach. *SIAM Review* **28**, 1 (1986)
- [21] Z. Kulpa: Diagrammatic analysis of interval linear equations. *Reliable Computing*, 9:1-20, (2003)
- [22] S. Majumdar; J. Lüthi; G. Haring; and R. Ramadoss: Characterization and Analysis of Software and Computer Systems with Uncertainties and Variabilities. In R. Dumke et al, editor, *Performance Engineering*. LNCS 2047, Springer-Verlag Berlin Heidelberg pp 202-221(2001)
- [23] M.A. Marsan, G. Balbo and G. Conte: A class of generalized stochastic Petri nets for the performance evaluation of multiprocessor systems. *ACM Transactions on Computer Systems*, 2:93-122 (1984)
- [24] M. Ajmone Marsan; G. Balbo; G. Conte; S. Donatelli and G. Franceschinis: *Modelling with Generalized Stochastic Petri Nets*. Wiley Series in Parallel Computing, John Wiley and Sons, (1995)
- [25] M.A. Marsan, A. Bobbio and S. Donatelli: Petri Nets in Performance Analysis: An Introduction. In: *Lectures on Petri Nets I: Basic Models*. Advances in Petri Nets. Wolfgang Reisig and Grzegorz Rozenberg(Eds). Springer, Berlin (1998) pp. 211-256
- [26] R.E. Moore: Interval Arithmetic and automatic error analysis in digital computing. Phd thesis, Stanford University, USA (1962)
- [27] R.E. Moore: *Interval Analysis*. Prentice Hall, Englewood Cliffs, NJ, USA (1966)
- [28] R.E. Moore: *Methods and applications of interval analysis*. Studies in Applied Mathematics (SIAM), (1979)
- [29] T. Murata: Petri net: properties, analysis and applications. *Proceedings of the IEE* 77(4) (1989) pp 541-580
- [30] A. Neumaier: *Interval methods for systems of equations*. Cambridge University Press, Cambridge, (1990)
- [31] S. S. Rao and L. Berke: Analysis of Uncertain Structural Systems Using Interval Analysis. *AiAA Journal*, **35**, No. 4, pp 727-735 (1997).
- [32] J. Rhon: Systems of linear interval equations. *Linear Algebra Appl.*, 126:39-78, (1989)
- [33] Reisig, W and Rozenberg (Eds.): *Elementary Net Systems*. In: *Lectures on Petri Nets I: Basic Models*. Advances in Petri Nets. Wolfgang Reisig and Grzegorz Rozenberg(Eds.). Springer, Berlin, vol 1 (1998) pp. 1-681
- [34] S.M. Rump: Fast and parallel interval arithmetic. *BIT*, 39(3):534-554, (1999)
- [35] S.M. Rump: INTval LABatory. In Tibor Csends, editor, *Developments in Reliable Computing*, Kluwer, Dordrecht, Netherlands, pages 77-105, (1999)
- [36] S.P. Shary: Algebraic approach of the interval linear static identification, tolerance, and control problems, or one more application of Kaucher arithmetic. *Reliable Computing*, 2(1):3-33, (1996)
- [37] S.P. Shary: Algebraic approach in the "outer problem" for interval linear equations. *Reliable Computing*, 3:103–135, (1997)
- [38] S.P. Shary: Outer estimation of generalized solution sets to interval linear systems. *Reliable Computing*, 5:323–335, (1999)
- [39] W. Siler and J. J. Buckley: *Fuzzy Expert Systems and Fuzzy Reasoning* Wiley-Interscience, John Wiley and Sons, Inc (2005)
- [40] C. U. Smith: *Performance Engineering of Software Systems*. Addison-Wesley, Reading, MA, (1990)