

---

**Modelagem e Simulação de Sistemas de  
Gerenciamento de Processos de Negócios  
baseadas em *Workflow net* Temporais com  
Mecanismos de Alocação de Recursos Híbridos  
*Fuzzy***

---

**Joslaine Cristina Jeske de Freitas**



**UNIVERSIDADE FEDERAL DE UBERLÂNDIA  
FACULDADE DE COMPUTAÇÃO  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO**



**Joslaine Cristina Jeske de Freitas**

**Modelagem e Simulação de Sistemas de  
Gerenciamento de Processos de Negócios  
baseadas em *Workflow net* Temporais com  
Mecanismos de Alocação de Recursos Híbridos  
*Fuzzy***

Tese de doutorado apresentada ao Programa de Pós-graduação da Faculdade de Computação da Universidade Federal de Uberlândia como parte dos requisitos para a obtenção do título de Doutor em Ciência da Computação.

Área de concentração: Engenharia de Software

Orientador: Dr. Stéphane Julia

Uberlândia  
2017

Dados Internacionais de Catalogação na Publicação (CIP)  
Sistema de Bibliotecas da UFU, MG, Brasil.

---

F866m      Freitas, Joslaine Cristina Jeske de, 1973-  
2017      Modelagem e simulação de sistemas de gerenciamento de processos  
de negócios baseadas em workflow net temporais com mecanismos de  
alocação de recursos híbridos fuzzy / Joslaine Cristina Jeske de Freitas. -  
2017.  
159 f. : il.

Orientador: Stéphane Julia.  
Tese (doutorado) - Universidade Federal de Uberlândia, Programa  
de Pós-Graduação em Ciência da Computação.  
Inclui bibliografia.

1. Computação - Teses. 2. Sistemas de informação gerencial - Teses.  
3. Redes de Petri - Teses. 4. Conjuntos difusos - Teses. I. Julia, Stéphane.  
II. Universidade Federal de Uberlândia. Programa de Pós-Graduação em  
Ciência da Computação. III. Título.

---

CDU: 681.3



UNIVERSIDADE FEDERAL DE UBERLÂNDIA  
FACULDADE DE COMPUTAÇÃO  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Os abaixo assinados, por meio deste, certicam que leram e recomendam para a Faculdade de Computação a aceitação da tese intitulada **“Modelagem e Simulação de Sistemas de Gerenciamento de Processos de Negócios baseadas em *Workow nets* Temporais com Mecanismos de Alocação de Recursos Híbridos *Fuzzy*”** por **Joslaine Cristina Jeske de Freitas** como parte dos requisitos exigidos para a obtenção do título de **Doutor em Ciência da Computação**.

Uberlândia, 22 de março de 2017.

Orientador: \_\_\_\_\_  
Prof. Dr. Stéphane  
Universidade Federal de Uberlândia/UFU

Banca Examinadora:

\_\_\_\_\_  
Prof. Dr. Ricardo Lüders  
Universidade Tecnológica Federal doParaná/UFTPR

\_\_\_\_\_  
Prof(a). Dr(a). Lígia Maria Soares Passos  
Universidade Federal Rural do Rio de Janeiro/UFRRJ

\_\_\_\_\_  
Prof. Dr(a). Fernanda Francielle de Oliveira Malaquias  
Universidade Federal de Uberlândia/UFU

\_\_\_\_\_  
Prof(a). Dr(a). Márcia Aparecida Fernandes  
Universidade Federal de Uberlândia/UFU



*...ao meu esposo, meu grande companheiro e amor.*



---

## Agradecimentos

Tenho muito o que agradecer. Sou grata a Deus por tudo que tem me dado.

A minha família querida. Meus pais, por tudo que fizeram por mim. Meu esposo, que me apoia em todos os momentos. Minha filha amada, mesmo sem saber o significado de estudar, no seu jeito inocente, durante o meu doutorado só vinha ao escritório para me abraçar e dizer “não vou te atrapalhar mamãe, só vim para dizer que te amo”.

Aos meus amigos queridos! Pessoas especiais que estiveram sempre ao meu lado e em momentos de necessidade, não me desamparam. Luciana (tia Lu), Leiliane, Franciny, Clarimundo, Waldiney vocês fizeram a diferença! Obrigada por todo apoio.

Meu orientador Stéphanie Julia, meu mais profundo agradecimento. Você merece o título de orientador com todas as honras possíveis. A sua capacidade em direcionar e dirigir uma pesquisa é indiscutível.

Agradeço a todos os meus professores. Obrigada por me ensinarem. Minha singela homenagem para alguns professores que fizeram a diferença na minha vida. Minha mãe: minha primeira professora da vida, primeira professora de piano e professora de português e inglês. “Tia” Branca, por me ensinar a ler e escrever, e principalmente, por ter demonstrado tanto carinho. Ao professor Maurosan, o craque em exatas. Zelman, o professor que sabe tudo decor. João Ângelo Martini, o mundo perdeu uma grande pessoa e um excelente professor. Carlos Sica, a pessoa mais bem humorada no mundo. Márcia Aparecida, a melhor professora que já tive. Waldemar Nogueira e Maria Célia, os melhores professores de piano.

Por fim, um agradecimento muito especial ao Erisvaldo, secretário do curso da pós-graduação. Seu profissionalismo, competência e amizade fizeram meus dias na FACOM mais suaves.



*“Sonhos determinam o que você quer. Ação determina o que você conquista.”*

*Aldo Novak*





---

## Resumo

O principal objetivo desta pesquisa de doutorado é propor um novo modelo de processo de negócios para uma gestão eficiente (e realista) dos recursos e do tempo em *Workflow nets*. Esta pesquisa teve como ponto de partida a validação do modelo de alocação de recursos *fuzzy* apresentado em (JESKE, 2006). Em particular, um mecanismo de inferência especializado do tipo *token player* adaptado a um modelo de alocação de recurso híbrido *fuzzy* foi implementado na ferramenta de modelagem, análise e simulação de redes de Petri coloridas chamada CPN Tools. Tais simulações mostraram que o novo modelo em relação a um mecanismo simples de alocação de recurso com uma política de disparo mais cedo (FIFO) permite uma melhoria da proporção de casos que respeitam os prazos estabelecidos. Para resolver o problema do escalonamento em Sistemas de Gerenciamento de *Workflow* deverão ser consideradas características importantes inerentes a estes sistemas. Em particular, quando muitos casos são executados simultaneamente, situações de conflito por recursos compartilhados podem ocorrer e devem ser resolvidas em tempo real (sem um mecanismo de retrocesso). Também, deverão ser consideradas restrições temporais relativas aos prazos de entrega de casos específicos. Para tanto, definiu-se um tempo *fuzzy* associado às atividades dos processos bem como um mecanismo de propagação de restrições cujo papel é garantir a consistência das restrições de tempo para realização das atividades respeitando os prazos de entrega previstos. Com o objetivo de tornar o comportamento do novo modelo o mais próximo e consistente com o comportamento de funcionários humanos envolvidos na execução de atividades de processos de negócios, a noção de possibilidade conjunta que envolve o caráter incerto e nebuloso tanto dos tempos de execução das atividades dos processos quanto das intensidades de carga dos recursos utilizados para a realização de atividades foi inserida no modelo. Novas regras de disparo de transições foram então definidas baseadas no conceito apresentado. Jogadores de rede de Petri do tipo *token player* baseados no novo modelo apresentado foram definidos, implementados, simulados e validados na ferramenta CPN Tools, cujo papel principal é mostrar a execução dos novos modelos apresentados. Com o uso da ferramenta CPN Tools foi possível avaliar o comportamento do sistema em funcionamento esperado ou

em situação mais crítica. É importante salientar que através dos resultados de simulação é possível prever como o sistema se comportará, isso a fim de avaliar diversos cenários possíveis de distribuição de recursos para poder, no futuro, realizar o dimensionamento do sistema baseado numa ferramenta de modelagem e simulação próxima da forma de trabalhar dos funcionários em ambiente de escritório.

**Palavras-chave:** *Workflow net*, Conjuntos *Fuzzy*, Teoria das Possibilidades, CPN Tools.

.

---

# Abstract

The main objective of this doctoral research is to propose a new business process model for efficient (and realistic) resources and time management in Workflow nets. This research had as its starting point the validation of the fuzzy resource allocation model presented in (JESKE, 2006). In particular, a specialized inference mechanism of the token player type adapted to a fuzzy hybrid resource allocation model was implemented in the modeling, analysis and simulation tool of colored Petri nets called CPN Tools. Such simulations have shown that the new model in relation to a simple resource allocation mechanism with an earlier firing policy (FIFO) allows an improvement in the proportion of cases that meet the established deadlines. To solve the scheduling problem in Workflow Management Systems, important characteristics inherent to these systems should be considered. In particular, when many cases are run concurrently, conflict situations due to shared resource can occur and must be resolved in real time (without a backward mechanism). Also, temporary restrictions regarding the deadlines for delivery of specific cases should be considered. For that, a fuzzy time associated to the activities of the processes was defined as well as a mechanism of propagation of restrictions whose role is to guarantee the consistency of the time constraints to carry out the activities respecting the expected delivery times. With the objective of making the behavior of the new model the closest and consistent with the behavior of the human collaborators involved in the execution of business process activities, the notion of joint possibility was inserted in the model. This notion involves the uncertain and nebulous character of the execution times of the activities of the processes and the intensity of the resources used to carry out the activities. New firing rules for transitions were then defined based on the concept presented. Token players based on the new model presented were defined, implemented, simulated and validated in the CPN Tools, whose main role is to show the execution of the new models presented. Using the CPN Tools, it was possible to evaluate the system behavior considering the expected or most critical operating. It is important to point out that through the simulation results it is possible to predict how the system will behave and, from this, to evaluate several possible resource distribution scenarios in order to be able, in the future, to carry out the

sizing of the system based on a modeling and simulation tool that approximates of the way employees work in an office environment.

**Keywords:** Workflow net, Fuzzy Set, Possibility Theory, CPN Tools.

---

## Lista de ilustrações

Figura 1 – Representação de um Conjunto <i>Fuzzy</i> para Tamanhos Médios. . . . .	34
Figura 2 – Representação de um Conjunto <i>Fuzzy</i> . . . . .	34
Figura 3 – Exemplo de Soma <i>Fuzzy</i> . . . . .	35
Figura 4 – Exemplo de Subtração <i>Fuzzy</i> . . . . .	35
Figura 5 – Exemplo de Multiplicação <i>Fuzzy</i> . . . . .	36
Figura 6 – Distribuição de Possibilidade de Passageiros em um Vagão de Metrô . .	37
Figura 7 – Cálculo da Possibilidade para que $a \leq b$ . . . . .	38
Figura 8 – Cálculo da Necessidade para que $a \leq b$ . . . . .	38
Figura 9 – Cálculo de Possibilidade entre dois Eventos. . . . .	38
Figura 10 – Cálculo de Necessidade entre dois Eventos. . . . .	38
Figura 11 – Possivelmente/Necessariamente depois de $a$ . . . . .	39
Figura 12 – Possivelmente/Necessariamente antes de $a$ . . . . .	39
Figura 13 – Possivelmente depois de $A$ e Possivelmente antes de $B$ . . . . .	40
Figura 14 – Necessariamente depois de $A$ e Necessariamente antes de $B$ . . . . .	40
Figura 15 – Rede de Petri. . . . .	42
Figura 16 – Exemplo de uma Rede de Petri Marcada. . . . .	43
Figura 17 – Exemplo de Disparo de uma Transição. . . . .	43
Figura 18 – Exemplo de uma Rede de Petri Marcada. . . . .	45
Figura 19 – Atributos de um Lugar de uma Rede de Petri Colorida. . . . .	50
Figura 20 – Expressões nos Arcos de Entrada e de Saída de uma Transição. . . . .	50
Figura 21 – Expressão Temporizada em um Arco de Saída de uma Transição. . . . .	51
Figura 22 – Inscrições que Podem ser Associadas a uma Transição. . . . .	51
Figura 23 – Diferença da Temporização de uma Transição para a Temporização de um Arco de Saída da Transição. . . . .	52
Figura 24 – Expressão de Guarda em uma Transição Restringe o Disparo para quando a Expressão for Verdadeira. . . . .	53
Figura 25 – Exemplo de aplicação da expressão de guarda de uma transição. . . . .	53
Figura 26 – Exemplo de Segmento de Código Associado a uma Transição. . . . .	54

Figura 27 – Um conjunto de lugares de fusão representa um único lugar. . . . .	56
Figura 28 – Transição de substituição $t_2$ e sua subrede associada secundária. . . . .	56
Figura 29 – Estrutura de um Jogador de Rede de Petri. . . . .	57
Figura 30 – <i>Workflow net</i> para o “Processo de Tratamento de Reclamações”. . . . .	59
Figura 31 – Distribuição Exponencial com Parâmetros $\lambda = 1/2, 1$ e $3/2$ . . . . .	64
Figura 32 – “ <i>WorkFlow net</i> para o Processo de Tratamento de Reclamações”. . . . .	70
Figura 33 – Representação de uma Atividade. . . . .	70
Figura 34 – <i>Workflow net</i> com Representação Explícita das Atividades. . . . .	72
Figura 35 – Alocação de Recurso Discreto. . . . .	73
Figura 36 – Alocação de Recurso Contínuo. . . . .	75
Figura 37 – Alocação de Recurso Contínuo <i>Fuzzy</i> . . . . .	77
Figura 38 – Cálculo de Possibilidade Associado a $t_3$ . . . . .	77
Figura 39 – Marcação após Disparo de $t_3$ . . . . .	79
Figura 40 – Cálculo de Possibilidade Associado a $t_4$ . . . . .	79
Figura 41 – Marcação após o Disparo de $t_4$ . . . . .	79
Figura 42 – Cálculo de Possibilidade Associado a $t_{14}$ . . . . .	80
Figura 43 – Marcação após o Disparo de $t_{14}$ . . . . .	80
Figura 44 – Cálculo de Possibilidade Associado a $t_5$ . . . . .	81
Figura 45 – Marcação após Disparo de $t_5$ . . . . .	82
Figura 46 – Cálculo da Possibilidade Associado a $t_6$ . . . . .	82
Figura 47 – Marcação após o Disparo de $t_6$ . . . . .	82
Figura 48 – Cálculo da Possibilidade Associado a $t_{16}$ . . . . .	83
Figura 49 – Marcação após Disparo de $t_{16}$ . . . . .	83
Figura 50 – Modelo para o Processo de Tratamento de Reclamações com Recursos Discretos. . . . .	86
Figura 51 – Modelo da <i>Workflow net</i> - Recursos Discretos. . . . .	87
Figura 52 – Atividades A1, A2, A3 e A4 do Processo de Tratamento de Reclamações - Recursos Discretos. . . . .	89
Figura 53 – Atividades A5, A6, A7 e A8 do Processo de Tratamento de Reclamações - Recursos Discretos. . . . .	90
Figura 54 – Modelo para o Processo de Tratamento de Reclamações com Recursos Discretos e Contínuos. . . . .	91
Figura 55 – Modelo da <i>Workflow net</i> - Recursos Discretos e Contínuo. . . . .	92
Figura 56 – Atividades A1, A2, A3 e A4 do Processo de Tratamento de Reclamações - Recursos Discretos e Contínuos. . . . .	93
Figura 57 – Atividades A5, A6, A7 e A8 do Processo de Tratamento de Reclamações - Recursos Discretos e Contínuos. . . . .	95
Figura 58 – Modelo para o Processo de Tratamento de Reclamações com Recursos Discretos e <i>Fuzzy</i> . . . . .	96

Figura 59 – Modelo da <i>Workflow net</i> - Recurso Discretos e <i>Fuzzy</i> . . . . .	97
Figura 60 – Atividades A1, A2, A3 e A4 do Processo de Tratamento de Reclamações - Recursos Discretos e <i>Fuzzy</i> . . . . .	99
Figura 61 – Atividades A5, A6, A7 e A8 do Processo de Tratamento de Reclamações - Recursos Discretos e <i>Fuzzy</i> . . . . .	100
Figura 62 – Intervalo <i>Fuzzy</i> . . . . .	108
Figura 63 – <i>Time WorkFlow net fuzzy</i> com Intervalos Estáticos para o “Processo de Tratamento de Reclamações”. . . . .	109
Figura 64 – Bordas <i>fuzzy</i> mínimas - Roteiro Sequencial. . . . .	110
Figura 65 – Bordas <i>fuzzy</i> máximas - Roteiro Sequencial. . . . .	110
Figura 66 – Bordas <i>fuzzy</i> mínimas - Roteiro Paralelo. . . . .	110
Figura 67 – Bordas <i>fuzzy</i> máximas - Roteiro Paralelo. . . . .	111
Figura 68 – Bordas <i>fuzzy</i> mínimas - Sincronização. . . . .	111
Figura 69 – Bordas <i>fuzzy</i> máximas - Sincronização. . . . .	112
Figura 70 – Bordas <i>fuzzy</i> mínimas - Roteiro Alternativo. . . . .	112
Figura 71 – Bordas <i>fuzzy</i> máximas - Roteiro Alternativo. . . . .	113
Figura 72 – Intervalos de Visibilidade iniciais - “Processo de Tratamento de Recla- mações”. . . . .	113
Figura 73 – Modelo para “Processo de Tratamento de Reclamações” após Aplicação dos Mecanismos de Propagação de Restrição. . . . .	114
Figura 74 – Modelo para “Processo de Tratamento de Reclamações” com Intervalos de Tempo na forma Trapezoidal. . . . .	115
Figura 75 – <i>Workflow net</i> com Restrições de Tempo na Forma Trapezoidal. . . . .	116
Figura 76 – Distribuição de possibilidade de W6. . . . .	118
Figura 77 – Fragmento da <i>Workflow net</i> - “Processo de Tratamento de Reclamações”. . . . .	119
Figura 78 – Modelo da <i>Workflow net</i> . . . . .	121
Figura 79 – Atividades A1, A2, A3 e A4 do Processo de Tratamento de Reclamações - Possibilidade Conjunta. . . . .	123
Figura 80 – Atividades A5, A6, A7 e A8 do Processo de Tratamento de Reclamações - Possibilidade Conjunta. . . . .	124
Figura 81 – Resultado da Simulação: Mecanismo de Alocação de Recurso Discreto. . . . .	153
Figura 82 – Resultado da Simulação: Mecanismo de Alocação de Recurso Contínuo e Discreto. . . . .	154
Figura 83 – Resultado da Simulação: Mecanismo de Alocação de Recurso <i>Fuzzy</i> e Discreto. . . . .	154
Figura 84 – Resultado da Simulação - Distribuição Normal. . . . .	155
Figura 85 – Resultado da Simulação - Distribuição Exponencial: Cenário 1. . . . .	156
Figura 86 – Resultado da Simulação - Distribuição Exponencial: Cenário 2. . . . .	157
Figura 87 – Resultado da Simulação - Distribuição Exponencial: Cenário 3. . . . .	158





---

## Lista de tabelas

Tabela 1 – Declarações para o Modelo CPN com Recursos Discretos. . . . .	88
Tabela 2 – Variáveis para o Modelo CPN com Recursos Discretos . . . . .	88
Tabela 3 – Constantes para o Modelo CPN com Recursos Discretos. . . . .	88
Tabela 4 – Declarações para o modelo CPN com Recurso Discreto e Contínuo. . .	93
Tabela 5 – Declarações para o modelo CPN com Recurso Discreto e <i>Fuzzy</i> . . . . .	98
Tabela 6 – Tempos de Execução para o Mecanismo de Alocação de Recurso Discreto	102
Tabela 7 – Tempos de Execução para o Mecanismo de Alocação de Recurso Dis- creto e Contínuo . . . . .	103
Tabela 8 – Tempos de Execução para o Mecanismo de Alocação de Recurso Dis- creto e <i>Fuzzy</i> . . . . .	103
Tabela 9 – Valores Comparativos - Mecanismo de Alocação de Recurso Discreto, Contínuo e <i>Fuzzy</i> . . . . .	104
Tabela 10 – Melhora <i>Fuzzy</i> em relação ao Mecanismo de Alocação de Recurso Dis- creto e Contínuo . . . . .	104
Tabela 11 – Declarações para o modelo CPN com Recursos e Restrições de Tempo <i>Fuzzy</i> . . . . .	120
Tabela 12 – Variáveis para o modelo CPN com Recursos e Restrições de Tempo <i>Fuzzy</i>	122
Tabela 13 – Constantes para o modelo CPN com Recursos e Restrições de Tempo <i>Fuzzy</i> . . . . .	122
Tabela 14 – Estatística Sobrecarga dos Recursos <i>Fuzzy</i> . . . . .	127
Tabela 15 – Valores Ideais (definidos) para Duração das Atividades . . . . .	128
Tabela 16 – Estatística para Duração das Atividades - Recursos e Tempo <i>Fuzzy</i> . .	129
Tabela 17 – Estatística - Quantidade de Casos Finalizados . . . . .	129
Tabela 18 – Estatística Sobrecarga dos Recursos <i>Fuzzy</i> . . . . .	132
Tabela 19 – Estatística - Quantidade de Casos Finalizados . . . . .	133
Tabela 20 – Estatística Sobrecarga dos Recursos <i>Fuzzy</i> . . . . .	133
Tabela 21 – Estatística - Quantidade de Casos Finalizados . . . . .	134
Tabela 22 – Estatística Sobrecarga dos Recursos <i>Fuzzy</i> . . . . .	135

Tabela 23 – Estatística - Quantidade de Casos Finalizados . . . . .	135
Tabela 24 – Estatística Sobrecarga dos Recursos <i>Fuzzy</i> . . . . .	136
Tabela 25 – Estatística - Quantidade de Casos Finalizados . . . . .	136

---

# Sumário

<b>1</b>	<b>INTRODUÇÃO . . . . .</b>	<b>25</b>
<b>1.1</b>	<b>Motivação . . . . .</b>	<b>28</b>
<b>1.2</b>	<b>Formulação do Problema . . . . .</b>	<b>28</b>
<b>1.3</b>	<b>Hipóteses . . . . .</b>	<b>28</b>
<b>1.4</b>	<b>Objetivos e Desafios da Pesquisa . . . . .</b>	<b>29</b>
1.4.1	Objetivo Geral . . . . .	29
1.4.2	Objetivos Específicos . . . . .	30
<b>1.5</b>	<b>Contribuições . . . . .</b>	<b>31</b>
<b>1.6</b>	<b>Organização da Tese . . . . .</b>	<b>31</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA . . . . .</b>	<b>33</b>
<b>2.1</b>	<b>Fundamentos Teóricos . . . . .</b>	<b>33</b>
2.1.1	Conjuntos <i>Fuzzy</i> . . . . .	33
2.1.2	Teoria das Possibilidades . . . . .	36
2.1.3	Redes de Petri . . . . .	41
2.1.4	Redes de Petri com Tempo . . . . .	46
2.1.5	Rede de Petri Colorida . . . . .	47
2.1.6	Jogador de Rede de Petri - “ <i>Token Player</i> ” . . . . .	56
2.1.7	<i>Workflow nets</i> . . . . .	58
2.1.8	Distribuições de Probabilidade . . . . .	61
<b>2.2</b>	<b>Trabalhos Relacionados . . . . .</b>	<b>66</b>
<b>3</b>	<b>MODELOS DE ALOCAÇÃO DE RECURSOS <i>WORKFLOW</i></b>	
	<b><i>NET</i> . . . . .</b>	<b>69</b>
<b>3.1</b>	<b>Mecanismos de Alocação de Recursos . . . . .</b>	<b>71</b>
3.1.1	Mecanismo de Alocação de Recurso Discreto . . . . .	71
3.1.2	Mecanismos de Alocação de Recurso Contínuo . . . . .	74
3.1.3	Mecanismo de Alocação de Recurso Contínuo <i>Fuzzy</i> . . . . .	76

<b>3.2</b>	<b>Implementação de <i>Workflow nets</i> com Recursos . . . . .</b>	<b>84</b>
3.2.1	Implementação de uma <i>Workflow net</i> com Mecanismos de Alocação de Recursos Discretos baseada na Linguagem CPN Tools sem Restrições de Tempo . . . . .	85
3.2.2	Implementação de uma <i>Workflow net</i> com Mecanismo de Alocação de Recursos Discretos e Contínuos baseada na Linguagem do CPN Tools sem Restrições de Tempo . . . . .	90
3.2.3	Implementação de uma <i>Workflow net</i> com Mecanismos de Alocação de Recursos Discretos e <i>Fuzzy</i> baseado na Linguagem do CPN Tools sem Restrições de Tempo . . . . .	94
<b>3.3</b>	<b>Jogador de <i>Workflow net</i> com Recursos e Resultados de Simulação . . . . .</b>	<b>98</b>
3.3.1	Resultado da Simulação: Mecanismo de Alocação de Recurso Discreto .	101
3.3.2	Resultado da Simulação: Mecanismo de Alocação de Recurso Discreto e Contínuo . . . . .	102
3.3.3	Resultado da Simulação: Mecanismo de Alocação de Recurso Discreto e <i>Fuzzy</i> . . . . .	103
3.3.4	Resultados Comparativos . . . . .	103
<b>4</b>	<b>MODELOS DE ALOCAÇÃO DE RECURSOS EM <i>WORKFLOW NET</i> COM RESTRIÇÕES DE TEMPO . . . . .</b>	<b>105</b>
4.1	Problema de Escalonamento em SGWs . . . . .	105
4.2	Modelagem de Restrições de Tempo em <i>Workflow net</i> . . . . .	107
4.3	Jogador de <i>Workflow net</i> com Recursos e Restrições de Tempo <i>Fuzzy</i> . . . . .	115
4.3.1	Implementação de uma <i>Workflow net</i> com Recursos e Restrições de Tempo <i>Fuzzy</i> baseada na Linguagem CPN Tools . . . . .	120
4.3.2	Jogador com Regras de Disparo baseadas na Noção de Possibilidade Conjunta - Distribuição Uniforme . . . . .	124
4.3.3	Resultados da Simulação - Distribuição Uniforme . . . . .	127
4.3.4	Jogador com Regras de Disparo baseadas na Noção de Possibilidade Conjunta - Distribuição Exponencial . . . . .	128
4.3.5	Resultados da Simulação - Distribuição Exponencial . . . . .	131
<b>5</b>	<b>CONCLUSÃO . . . . .</b>	<b>137</b>
5.1	Principais Contribuições . . . . .	138
5.2	Trabalhos Futuros . . . . .	139
5.3	Contribuições Representadas pela Produção Bibliográfica . . .	141
	<b>REFERÊNCIAS . . . . .</b>	<b>143</b>

<b>APÊNDICE A</b>	<b>RELATÓRIOS COMPLETOS DA SIMULAÇÃO</b>	
	<b>NO CPN TOOLS . . . . .</b>	<b>153</b>
<b>A.1</b>	<b>Capítulo 3 . . . . .</b>	<b>153</b>
<b>A.2</b>	<b>Capitulo 4 . . . . .</b>	<b>153</b>



## Introdução

Segundo o *Workflow Management Coalition* - WfMC (WfMC, 1999), *workflow*, ou fluxo de trabalho, é a automação de um processo de negócio, por inteiro ou por partes, durante o qual documentos, informações e atividades são passadas de um participante para outro para que estes desenvolvam ações respeitando um conjunto de regras procedimentais. Um processo de negócio pode ser definido como um conjunto de atividades inter-relacionadas iniciadas como resposta a um evento, com o objetivo de obter resultados com valor para um cliente, sendo este interno ou externo à organização (SHARP; MCDERMOTT, 2001).

*Workflow* também pode ser definido como qualquer conjunto de atividades executadas de forma coordenada, em série ou em paralelo, por dois ou mais membros de um grupo de trabalho, visando um objetivo comum (AALST; HEE, 2004). Atividades, por sua vez, pressupõem que existam atores que as executam (interpretando seus papéis). Os documentos que são produzidos e consumidos durante a execução das atividades, as rotas que definem os destinos de cada fluxo de informação e as regras procedurais que devem ser respeitadas durante a execução das atividades (LOMAZOVA, 2010).

*Workflow* é um conceito intimamente relacionado à reengenharia e automação de negócios e de processos de informação em uma organização (GEORGAKOPOULOS; HORNICK; SHETH, 1995). A necessidade de escritórios mais eficientes resultou no conceito de Reengenharia de Processo de Negócio e na tecnologia chamada de Sistemas de Gerenciamento de *Workflow* (SGWs) (BOSZORMENYI; EISNER; GROISS, 1999; GEORGAKOPOULOS; HORNICK; SHETH, 1995).

SGWs são pacotes de *software* genéricos para gerenciamento de processos de negócio, (AALST; HEE, 2004; ESHUIS, 2002; ALONSO et al., 1997; HOLLINGSWORTH, 1995). Estes sistemas, apresentam-se como uma solução capaz de melhorar a eficiência e a gestão dos processos organizacionais, uma vez que provêm uma automação procedimental do gerenciamento de processos de negócios assim como mecanismos de alocação de recursos apropriados associados com as diversas atividades. Em particular, SGWs permitem o gerenciamento das atividades fazendo com que a informação certa chegue até a pessoa

certa no momento certo.

Em SGWs, os processos de negócio são modelados através de modelos de *workflows* representando as sequências de atividades que precisam ser executadas em uma organização para tratar casos específicos e atingir uma meta previamente estabelecida (AALST; HEE, 2004). Para Lomazova (2010), a modelagem dos processos de negócio é a primeira e mais importante fase no ciclo de vida de Sistemas de Gerenciamento de *Workflow*.

Várias linguagens de modelagem podem ser utilizadas no contexto da modelagem de processos de negócio como, por exemplo, BPMN (*Business Process Model and Notation*) (OMG, 2011a), YAWL (*Yet Another Workflow Language*) (HOFSTEDE et al., 2010b), redes de Petri (MURATA, 1989) e, até mesmo, Diagramas de Atividade da UML (*Unified Modeling Language*) (OMG, 2011b). É importante frisar que ambigüidades e confusões não podem ser prevenidas em um modelo de processos de negócio informal. Para resolver este problema, é necessário um modelo de processos de negócio com uma semântica formal (AALST; HEE, 2004). Modelos com rigor matemático não são usuais para profissionais da área de negócio. No entanto, modelos formais como as redes de Petri podem auxiliar na solução desse problema, pois possuem representação gráfica, são de fácil aprendizado, funcionam como linguagem de comunicação entre especialistas de diversas áreas, permitem a descrição dos aspectos estáticos e dinâmicos do sistema a ser representado, e ainda possuem o formalismo matemático que permite a implementação de importantes métodos de análise (MURATA, 1989).

As redes de Petri que modelam processos de negócio foram definidas por Aalst (1997) e são chamadas de *WorkFlow nets*. Vários trabalhos passaram a considerar este modelo formal e específico para a modelagem dos processos de negócio em suas abordagens. Isto pode ser visto, por exemplo, em Liu et al. (2015), Freitas, Julia e Rezende (2015a), Gharib e Giorgini (2015), Passos e Julia (2015), Hofstede et al. (2010a) e outros.

As Redes de Petri *Fuzzy*, uma combinação da teoria dos conjuntos nebulosos (ZADEH, 1965) com a teoria das redes de Petri (MURATA, 1989), são uma ferramenta para representação do conhecimento incerto sobre um estado de um sistema. A partir do trabalho de Looney (1988), diversos autores, tanto da comunidade das redes de Petri quanto de Inteligência Artificial, têm proposto diferentes tipos de redes de Petri *Fuzzy*; ver por exemplo: Rezende e Julia (2015), Rezende, Julia e Cardoso (2014), Cardoso (1999), Murata (1996) e Cardoso, Valette e Dubois (1989). Apesar de possuírem o mesmo nome, esses modelos diferem entre si pelas diferentes ferramentas *fuzzy* utilizadas (lógica fuzzy (ZADEH, 1988) ou teoria da possibilidade (DUBOIS; PRADE, 2009)) e pelos elementos que são *fuzzificados* (por exemplo: marcações, tempo de disparo, disponibilidade de recursos, entre outros).

O gerenciamento do tempo em *workflow* tem sido uma das áreas de pesquisa mais ativas tanto na comunidade acadêmica quanto na indústria, como por exemplo em Li e Cui (2014), Bertolini, Liu e Srba (2013), Tang et al. (2010). Existem muitos tipos de



restrição de tempo em processos de negócio. Pode-se citar o fato de que uma tarefa deve ser realizada em um período de tempo limitado, pois as empresas podem sofrer perdas significativas quando esses limites são violados. Se, por exemplo, em um processo específico, um pedido de um cliente não for tratado a tempo, a empresa poderá receber sanções financeiras por não completar o pedido dentro de um determinado período de tempo. Por isso, é de grande importância gerenciar informações de tempo em um *workflow* de forma eficiente, a fim de evitar problemas de violação do prazo.

Um estudo sobre a gestão do tempo em *workflow* concentra-se principalmente no planejamento do tempo global de execução do processo e na estimativa de duração das atividades, isso a fim de evitar violações de restrição de tempo nas atividades dos processos (DENG et al., 2014; EDER; PANAGOS; RABINOVICH, 2013b). Um modelo de *workflow* deve então descrever o tempo de forma explícita e analisar o comportamento temporal de utilização dos recursos aplicados a certas atividades. Em geral, fenômenos temporais típicos incluem tempo de espera entre atividades, limite de intervalos válidos sobre visibilidade dos recursos, etc.

É importante salientar também que as atividades de um processo de negócio geralmente dependem de recursos para serem executadas. O problema é que esses recursos são geralmente finitos e precisam ser compartilhados entre as diferentes instâncias do processo de negócio que podem estar concorrentemente em execução. Portanto, é necessário quantificar os recursos necessários para a execução de *workflow* a fim de evitar que muitas tarefas sejam atribuídas ao mesmo recurso, e, como resultado, o recurso se tornar um gargalo para o sistema. Além disto, quando se trata de recursos humanos, tem-se a existência de incerteza associada à execução da tarefa, pois cada funcionário tem seu próprio ritmo de trabalho.

Em sistemas onde as noções de tempo e de concorrência são tratadas explicitamente, conceitos e técnicas de escalonamento formam o ponto central na previsibilidade do comportamento de sistemas em tempo real. A principal diferença entre o tradicional problema de escalonamento dos sistemas de produção e o problema de escalonamento dos sistemas de gerenciamento de *workflow* é a natureza dos recursos usados para tratar as atividades. Nos sistemas de produção, os recursos representam equipamentos físicos e podem ser representados por exemplo por simples fichas nos lugares de uma rede de Petri. Eles são recursos do tipo discreto. No caso de sistemas de gerenciamento de *workflow*, recursos podem representar tanto equipamentos físicos quanto funcionários humanos. A maioria dos recursos humanos pode tratar diversas atividades simultaneamente e por isso não podem ser representados por simples fichas nos lugares de uma rede de Petri. Resumidamente, no caso da manufatura, os recursos são 100% discretos, enquanto que em processo administrativos a noção de alocação de recursos é mais nebulosa e incerta, em particular em relação ao esforço de realização do trabalho e do tempo de uso do recurso.

A resolução do problema do escalonamento em SGW deve considerar características

importantes inerentes a estes sistemas. Em particular, se muitos casos são executados simultaneamente, situações de conflito poderão ocorrer e deverão ser resolvidas em tempo real (sem um mecanismo de retrocesso). Deverão ser consideradas também restrições temporais relativas aos prazos de entrega dos casos específicos.

## 1.1 Motivação

O problema fundamental a ser tratado durante este trabalho de tese é o gerenciamento de *Workflow nets* com recursos. Processos de negócio distintos podem compartilhar recursos comuns. Por exemplo, um funcionário pode atuar em diversas atividades que se encontram em diversos processos não diretamente relacionados entre si. Além disso, na execução das atividades de processos de negócio num ambiente real, dependendo da urgência, os recursos podem ser sobrecarregados ou os prazos de entrega podem ser alterados (atrasados). Para obter a melhor utilização dos recursos, as cargas de trabalho devem ser gerenciadas para evitar a sobrecarga em alguns recursos, e/ou taxas de ocupação baixa de certos funcionários.

Sendo assim, deseja-se produzir modelos e mecanismos de inferências especializados aplicados a tais modelos para tratar a gestão dos recursos em tempo de execução, ou, em outras palavras, tratar o problema de escalonamento de atividades existentes em SGW. Para tanto, pretende-se usar técnicas de Inteligência Artificial aplicadas a modelos de processos mais tradicionais para tornar a gestão dos recursos mais consistente com o comportamento humano de funcionários envolvidos na boa execução de processos de negócio.

## 1.2 Formulação do Problema

Dada a motivação do trabalho, as seguintes questões de pesquisa foram levantadas.

- a) Como os modelos *fuzzy* podem aprimorar a representação do comportamento humano em relação a outros modelos existentes?
- b) Como Técnicas de Inteligência Artificial podem introduzir a noção de flexibilidade do trabalho que existe nas estruturas administrativas que dependem de funcionários humanos?
- c) Como avaliar o desempenho global de sistemas administrativos e a disponibilidade dos recursos envolvidos nas atividades com os modelos de simulação existentes?

## 1.3 Hipóteses

A fim de responder as questões de pesquisa quatro hipóteses são assumidas.

- a) É possível expressar de maneira mais realista o comportamento humano usando conjuntos *fuzzy* quando se trata de Sistemas de Gerenciamento de *Workflow*.
- b) Como o tempo real necessário para a execução de uma atividade em um Sistema de Gerenciamento de *Workflow* é não-determinístico e não é facilmente previsível, uma duração *fuzzy* pode ser atribuída a cada atividade a fim de representar a incerteza do tempo gasto para a realização de cada atividade.
- c) O conceito de possibilidade conjunta que considera dois conjuntos nebulosos simultaneamente - tanto a disponibilidade dos recursos quanto restrições de tempo para respeitar prazos preestabelecidos de conclusão dos casos tratados - flexibiliza os modelos de alocação de recursos bem como a duração das atividades de maneira a expressar melhor o comportamento de funcionários humanos que atuam em atividades de processos de negócio.
- d) É factível, usando CPN (Colored Petri Nets) e o ambiente CPN Tools, implementar simular e avaliar diversas propostas de modelos de alocação de recursos de processos de negócio.

## 1.4 Objetivos e Desafios da Pesquisa

Em Jeske (2006), foi proposto um modelo de rede de Petri p-temporal com recursos híbridos *fuzzy* como solução para o problema de alocação de recursos humanos em SGWs de forma a expressar de maneira mais realista o mecanismo de alocação de recursos quando o comportamento humano é considerado. A fim de alcançar o objetivo proposto, os mecanismos de alocação de recursos foram fuzzyficados e uma apresentação informal de jogadores associados a tais modelos foi introduzida mas não implementada. Tais mecanismos de inferência especializados tem como papel principal encontrar sequências de atividades que respeitam os prazos de conclusão dos casos considerados nos processos de acordo com as disponibilidades dos recursos existentes. Isso constitui de certa forma um problema de escalonamento de atividades no caso de processos de negócio, num contexto menos de otimização e mais de satisfação de um conjunto de restrição, como é o caso geralmente da abordagem de Inteligência Artificial aplicadas ao problema de escalonamento de um conjunto de atividades. A partir dos resultados alcançados nessa pesquisa, pode-se delinear novos objetivos.

### 1.4.1 Objetivo Geral

O objetivo deste trabalho é propor um novo modelo de alocação de recurso em processos de negócio que consideram o gerenciamento do tempo (essencialmente os prazos de conclusão dos casos tratados pelos processos existentes) através de restrições de tipo incertas e nebulosas ( intervalos de tempo *fuzzy*). Além disso, deve-se definir mecanismos

de inferências especializados de tipo *token player* para uma gestão dos recursos e do tempo consistente com a forma de trabalhar de funcionários humanos envolvidos na execução das atividades dos processos a fim de garantir a execução das tarefas previstas dentro de prazos bem estabelecidos.

### 1.4.2 Objetivos Específicos

- a) Construir, simular e validar o protótipo do mecanismo de alocação de recursos apresentados em (JESKE, 2006) usando a ferramenta de programação e simulação CPN Tools. Os mecanismos de inferências do tipo *token player* serão implementados para a supervisão dos mecanismos de alocação de recursos discretos, contínuos, híbridos e eventualmente *fuzzy*. A construção, simulação e validação dos jogadores será feita no CPN Tools.
- b) Definir um modelo de *workflow* que considera intervalos de tempo *fuzzy* associados às atividades do processo. Baseados em tais intervalos de tempo, mecanismos de propagação de restrições de tempos serão aplicados para produzir uma estimativa *fuzzy* das datas de início das atividades envolvidas nos processos.
- c) Introduzir a noção de possibilidade conjunta que envolve o caráter incerto e nebuloso tanto dos tempos de execução das atividades dos processos quanto das intensidades de carga dos recursos utilizados para a realização de atividades. Definir novas regras de disparo aplicadas ao novo modelo de alocação de recursos que considera informações incertas e nebulosas nos diversos aspectos das atividades do processo, isso com a finalidade de tornar o comportamento do novo modelo o mais próximo e consistente com o comportamento de funcionários humanos envolvidos na execução de atividades de processos de negócio.
- d) Construir, simular e validar o protótipo do novo modelo de alocação de recursos que considera informações incertas e nebulosas nos diversos aspectos das atividades do processo (tanto dos tempos de execução das atividades dos processos quanto das intensidades de carga dos recursos utilizados para a realização de atividades). Os mecanismos de inferência do tipo *token player* serão implementados para a supervisão dos mecanismos de alocação de recursos *fuzzy* bem como a duração das atividades. A construção, simulação e validação dos jogadores será feita no CPN Tools.
- e) Definir heurísticas para melhorar a taxa de sucesso de entrega dos casos tratados nos diversos processos do ponto de vista dos prazos de entrega. A fim de evitar que casos não cheguem ao final, as heurísticas tratarão casos que chegam antes do prazo estabelecido, dentro do prazo ou atrasados (após o prazo).

## 1.5 Contribuições

Os processos de negócio e seus projetos ganharam importância desde o início dos anos noventa. A habilidade em projetar estes processos de forma eficiente e flexível é o fator de sucesso mais difícil de ser alcançado pelas companhias atualmente. É desta dificuldade que surgiu a necessidade de desenvolver e lidar com técnicas e ferramentas adequadas para especificar, analisar e simular processos de negócio.

Complexos mecanismos de alocação de recursos existem em processos de negócio e raramente são considerados formalmente. Por outro lado, eles geralmente são as causas de criação de gargalos e baixos desempenhos em processos e podem até levar a problemas sérios de gargalos ou bloqueios. É então de fundamental interesse analisar os processos de negócio considerando a existência de restrições sobre os mecanismos de alocação de recurso nas fases iniciais do desenvolvimento de sistemas administrativos.

A principal contribuição que se espera deste trabalho é a de propor um novo modelo de *Workflow net* com mecanismos de alocação de recursos e restrições de tempo *fuzzy* que deverão tornar o modelo de supervisão das atividades executadas nos processos de negócio o mais próximo possível do comportamento de funcionários humanos que atuem diretamente na execução das atividades dos processos. As tarefas de um processo de negócio geralmente dependem de diferentes recursos para serem executadas. Além disso, recursos do tipo humano podem trabalhar em diferentes atividades simultaneamente e ainda gastar mais ou menos tempo em cada tarefa. Em particular, o desempenho esperado para o processo de negócio depende de como os recursos são provisionados e usados. Para isso, pretende-se associar técnicas clássicas de representação de processos a conceitos avançados de Inteligência Artificial que consideram o caráter incerto e nebuloso tanto de recursos de tipo humano quanto do tempo para realização das tarefas.

Mecanismos de inferência especializados do tipo *token player* (jogadores de redes de Petri) baseados nos diversos modelos de alocação de recursos servirão então para encontrar um planejamento das datas de início de execução das atividades do processo (uma solução de escalonamento em tempo de execução que não considera necessariamente políticas de disparo mais cedo do tipo *First-in-First-out*), a fim de respeitar os prazos de entrega da maioria dos casos tratados no sistema.

## 1.6 Organização da Tese

Este documento está dividido em cinco capítulos organizados da forma como se segue.

O Capítulo 2 especifica a Fundamentação Teórica desta pesquisa; em particular na seção 2.1 são apresentados os fundamentos teóricos e na seção 2.2 os trabalhos relacionados. O Capítulo 3 mostra na ferramenta de programação e simulação CPN Tools os mecanismos de inferências especializados aplicados a diversos mecanismos de alocação de recursos

(discretos, contínuos e *fuzzy*) e um estudo comparativo dos resultados de simulação entre os diversos jogadores classificados de acordo com a natureza dos recursos envolvidos. O Capítulo 4 introduz o problema de escalonamento (sequência de atividades que consideram tanto as intensidades dos recursos envolvidos quanto as restrições de tempo relativas aos prazos de entrega dos casos envolvidos nos processos) em tempo de execução (sem mecanismo de retrocesso); em particular, uma abordagem baseada na combinação de três mecanismos é apresentada: um mecanismo de propagação de restrições aplicados aos roteiros dos processos; novas regras de disparo de transições baseadas no conceito de possibilidade conjunta; e, finalmente um jogador de redes de Petri, implementado na ferramenta de programação e simulação CPN Tools, que, combinando as informações relativas à disponibilidade dos recursos compartilhados e às informações de prazo dos casos envolvidos nos processos permite uma gestão dos recursos mais apropriada ao tipo de atividades que existem em sistemas administrativos. No Capítulo 5, são apresentadas as considerações finais, bem como as principais contribuições (em particular as produções bibliográficas associadas ao trabalho de tese) e sugestões de trabalhos futuros.

## Fundamentos Teóricos

Neste capítulo é apresentada a fundamentação teórica necessária para compreensão deste trabalho bem como os trabalhos relacionados. A seção 2.1 apresenta os conceitos teóricos relacionados aos Conjuntos *Fuzzy*, à Teoria das Possibilidades, às Redes de Petri, às Redes de Petri Coloridas, ao conceito do Jogador de Redes de Petri, às *Workflow nets* e os conceitos matemáticos relacionados a modelos probabilísticos. Os Trabalhos Relacionados são apresentados na seção 2.2.

### 2.1 Fundamentos Teóricos

Esta seção apresenta os conceitos teóricos relacionados aos Conjuntos *Fuzzy* (2.1.1), à Teoria das Possibilidades (2.1.2) às Redes de Petri (2.1.3), às Redes de Petri com Tempo (2.1.4), às Redes de Petri Coloridas (CPN) (2.1.5), ao conceito do Jogador de Rede de Petri (2.1.6), às *Workflow nets* (2.1.7) e aos Modelos Probabilísticos (2.1.8).

#### 2.1.1 Conjuntos *Fuzzy*

A noção de conjunto *fuzzy* foi introduzida por Zadeh (1965) a fim de representar a incerteza nas informações. Por exemplo, um homem, com base na sua estatura, pode ser considerado pela maioria da população como alto, normal, baixo. No entanto, não se pode definir uma única unidade de medida (um número) para especificar quando uma pessoa é alta ou baixa. Para tanto, é necessário definir um intervalo para esta representação. Sendo assim, um certo grau de crença pode ser anexado a cada interpretação possível de uma informação simbólica e pode simplesmente ser formalizado por um conjunto *fuzzy*  $F$  de um conjunto de referência  $X$  que pode ser definido pela função  $\mu_F : X \rightarrow [0, 1]$ .

Em particular, dado um elemento  $x \in X$ ,  $\mu_F(x) = 0$  denota que  $x$  não é membro do conjunto  $F$ ,  $\mu_F(x) = 1$  denota que  $x$  é, definitivamente, um membro do conjunto  $F$ , e valores intermediários denotam o fato que  $x$  é um elemento de  $F$  com um certo grau de pertinência. Por exemplo, considerando que a Figura 1 representa o conjunto *fuzzy*

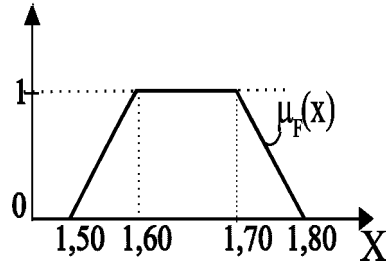


Figura 1 – Representação de um Conjunto *Fuzzy* para Tamanhos Médios.

$F$  dos tamanhos médios,  $X$  é o conjunto dos tamanhos possíveis para pessoas do sexo masculino. Se  $x \leq 1,50m$  ou se  $x \geq 1,80m$ , o tamanho não é médio e  $\mu_F(x) = 0$ . Se  $x \in [1,60m; 1,70m]$  o tamanho é completamente médio e  $\mu_F(x) = 1$ . Entre  $1,50m$  e  $1,60m$ , e entre  $1,70m$  e  $1,80m$  quanto maior o valor de  $\mu_F(x)$ , mais o tamanho da pessoa é compatível com o conceito de tamanho médio.

Um conjunto *fuzzy* pode ser representado por um trapézio  $A = [a1, a2, a3, a4]$  como mostrado na Figura 2, onde o subconjunto menor, chamado núcleo, corresponde aos membros com valor igual a 1. O subconjunto maior, o qual corresponde aos membros com valor maior que zero, é chamado suporte.

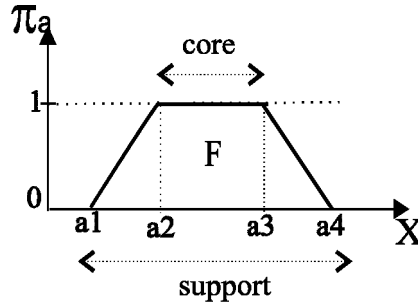


Figura 2 – Representação de um Conjunto *Fuzzy*.

Podem existir três casos particulares de conjuntos *fuzzy* que, geralmente, são:

- a) forma triangular,  $a_2 = a_3$ ,
- b) caso impreciso,  $a_1 = a_2$  e  $a_3 = a_4$ ,
- c) caso preciso,  $a_1 = a_2 = a_3 = a_4$ .

Quando dois conjuntos *fuzzy* distintos  $A$  e  $B$  são considerados, as operações aritméticas existentes são (KLIR; YUAN, 1995):

- a) a soma *fuzzy*  $A \oplus B$  definida como:

$$[a1, a2, a3, a4] \oplus [b1, b2, b3, b4] = [a1 + b1, a2 + b2, a3 + b3, a4 + b4] \quad (1)$$

- b) a subtração *fuzzy*  $A \ominus B$  definida como:

$$[a1, a2, a3, a4] \ominus [b1, b2, b3, b4] = [a1 - b4, a2 - b3, a3 - b2, a4 - b1] \quad (2)$$



c) o produto *fuzzy*  $A \otimes B$  definido como:

$$[a1, a2, a3, a4] \otimes [b1, b2, b3, b4] = [a1.b1, a2.b2, a3.b3, a4.b4] \quad (3)$$

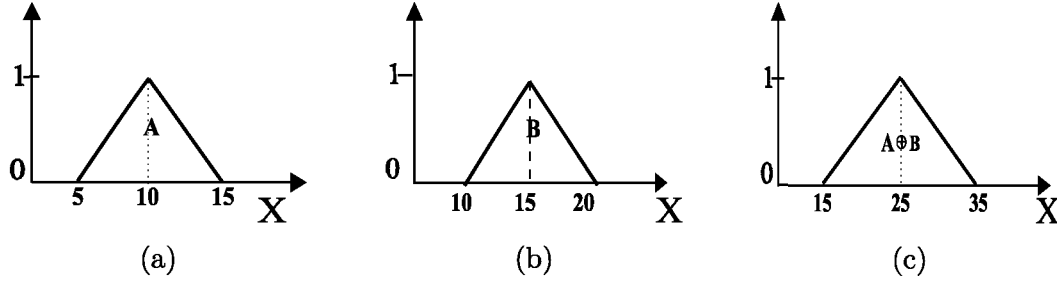


Figura 3 – Exemplo de Soma *Fuzzy*.

Para exemplificar as operações com intervalos *fuzzy*, suponha que um produto  $A$  possui um preço em torno de R\$10,00. Então, ele pode ser representado por um intervalo *fuzzy*  $A = [5, 10, 10, 15]$  (Figura 3a), que é um caso particular de conjunto *fuzzy* (distribuição triangular); o produto  $B$  possui um preço em torno de R\$15,00 e pode ser representado por  $B = [10, 15, 15, 20]$  (Figura 3b). É totalmente aceitável que o valor gasto para comprar estes dois produtos será algo em torno de R\$25,00, que é comprovado pela operação de soma *fuzzy*, resultando em um intervalo  $I = [15, 25, 25, 35]$ . Pode-se, então, concluir que, para comprar os produtos  $A$  e  $B$  seriam gastos no mínimo R\$15,00 e no máximo R\$35,00. Esta operação pode ser visualizada na Figura 3c.

Supondo que se deseja comprar os dois produtos e o dinheiro disponível para esta compra seja, somente, algo em torno de R\$20,00, representado pelo intervalo  $D = [15, 20, 20, 25]$  (Figura 4a). Neste caso, deve-se utilizar a equação 2, onde o resultado será  $R = [-20, -5, -5, 10]$  (Figura 4b). Pela subtração *fuzzy* pode-se verificar que será possível comprar os dois produtos somente em alguns casos por exemplo, quando o produto  $A$  e o produto  $B$  forem mais baratos (pertencerem a borda inferior do intervalo).

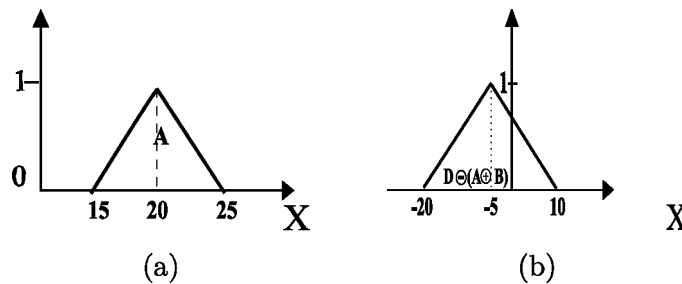


Figura 4 – Exemplo de Subtração *Fuzzy*.

No caso em que se deseja comprar quatro (4) produtos  $A$ , pode-se fazer este cálculo usando a equação 3, onde a quantidade é um conjunto preciso  $Q = [4, 4, 4, 4]$ ; Assim o resultado será  $R = [20, 40, 40, 60]$  (Figura 5). No caso, para comprar quatro produtos  $A$ , pode-se dizer que serão gastos em torno de R\$40,00.

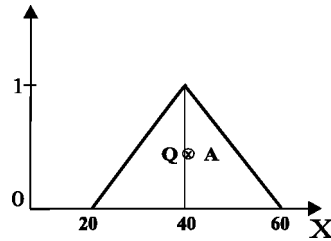


Figura 5 – Exemplo de Multiplicação *Fuzzy*.

### 2.1.2 Teoria das Possibilidades

A teoria das possibilidades, introduzida por Zadeh (1965) e desenvolvida por Dubois e Prade (2009), apresentaram uma abordagem para formalizar as incertezas subjetivas sobre a ocorrência de eventos, isto é, um meio de dizer em qual medida a realização de um evento é possível sem ter a disposição a avaliação da possibilidade de ocorrência do evento porque, por exemplo, não se conhece um evento semelhante que poderia ser usado como referência, ou porque a incerteza é a consequência da falta de confiabilidade dos instrumentos de observação ou das próprias dúvidas sobre os resultados obtidos a partir da observação. Por exemplo, considerando o enunciado “Pedro participará da festa de Maria em São Paulo terça-feira”, é difícil encontrar uma referência já conhecida para estabelecer um valor de probabilidade associado a tal evento. Poderiam ser considerados diversos fatos conhecidos do tipo:

Pedro dificilmente participa de festas durante a semana.

Pedro viaja muito durante a semana a trabalho e não se encontra em São Paulo.

Pedro briga muito com Maria.

Por outro lado, pode-se estabelecer subjetivamente até que ponto é possível e também até que ponto é certo que Pedro participará da festa baseando-se globalmente sobre o que se conhece de Pedro.

Tais avaliações subjetivas podem ser formalizadas através dos conceitos de medidas de possibilidade e de necessidade de ocorrência de um evento.

Considerando um conjunto de referência  $X$ , atribua-se a cada evento definido sobre  $X$  um valor que pertence ao conjunto  $[0, 1]$  e que avalia até que ponto tal evento é possível. Em particular, é possível definir uma distribuição de possibilidade associada a um dado  $f$  a partir de um conjunto *fuzzy*  $F$  (DUBOIS, 1997; CARDOSO; VALETTE; DUBOIS, 1989) de tal forma que:

$$\forall x \in X, \quad \Pi_f(x) = \mu_F(x) \quad (4)$$

O cálculo da possibilidade e da necessidade de ocorrência de um evento sobre o conjunto de referência  $X$  é definido da seguinte forma:

**Definição 1 (Cálculos de Possibilidade e de Necessidade)** *Os cálculos de possibilidade  $\Pi(S)$  e de necessidade  $N(S)$  de ocorrência de um evento  $S$  sobre o conjunto referência  $X$ , quando se considera a distribuição de possibilidade definida por  $\Pi_f : X \rightarrow [0, 1]$ , são dados por:*

$$\Pi(S) = \sup_{x \in S} \Pi_f(x) \quad (5)$$

e

$$N(S) = \inf_{x \notin S} (1 - \Pi_f(x)) = 1 - \Pi(\bar{S}) \quad (6)$$

Se  $\Pi(S) = 0$ , é impossível que  $S$  aconteça; se  $\Pi(S) = 1$ , é possível que  $S$  aconteça no entanto, depende do valor de  $N(S)$ . Se  $N(S) = 1$ , é certeza que  $S$  acontecerá. Em particular, existe a relação de dualidade entre as modalidades de possibilidade e necessidade, a qual postula que um evento é necessário quando o contrário é impossível.

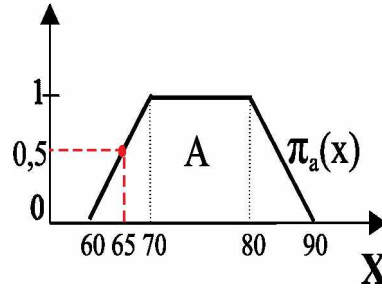


Figura 6 – Distribuição de Possibilidade de Passageiros em um Vagão de Metrô

Considerando, por exemplo, o conjunto de referência  $X$  dos inteiros menores ou iguais a 100 e a possibilidade que um certo número  $a$  de pessoas se encontram em um vagão de metrô. A configuração permite, por exemplo, a distribuição de possibilidade dada pela Figura 6. Considerando que o número  $a$  de passageiros corresponde ao que se espera no horário de saída do trabalho. Se  $S = [0, 65]$  então  $\Pi(S) = 0,5$  representa a possibilidade para que o número de pessoas no horário de saída do trabalho seja menor que 65. Pode-se dizer também que a certeza sobre o fato de que o número de passageiros no vagão seja menor que 65 é dado por  $N(S) = 1 - \Pi[66, 100] = 1 - 1 = 0$ .

No caso de dois dados  $a$  e  $b$  caracterizados por dois conjuntos fuzzy  $A$  e  $B$ , o cálculo da possibilidade para que  $a \leq b$  é definido como (Figura 7):

$$\begin{aligned} \Pi(a \leq b) = \sup_{x \leq y} (\min(\Pi_a(x)), \min(\Pi_b(y))) = \\ \max([A, +\infty[ \cap ] - \infty, B]) \end{aligned} \quad (7)$$

.

A necessidade é então definida por (Figura 8):

$$N(a \leq b) = 1 - \sup_{x \leq y} (\min(\Pi_a(x)), \min(\Pi_b(y))). \quad (8)$$

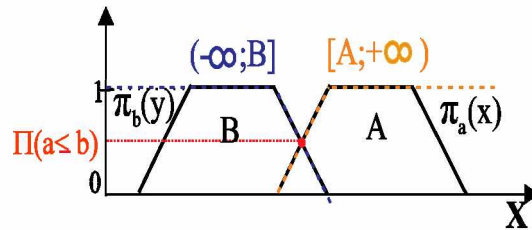


Figura 7 – Cálculo da Possibilidade para que  $a \leq b$ .

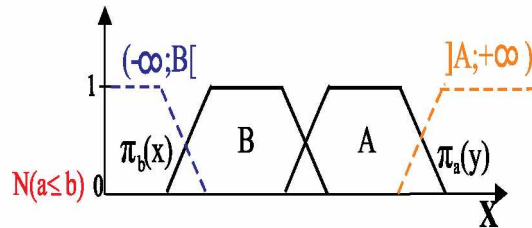


Figura 8 – Cálculo da Necessidade para que  $a \leq b$ .

Por exemplo, seja  $X$  o conjunto de datas para a realização de um evento. As Figuras 9a e 9b mostram a possibilidade dos eventos  $A$  e  $B$ , respectivamente, ocorrerem em uma determinada data. Deseja-se saber qual a possibilidade do evento  $A$  ocorrer antes do evento  $B$ . Neste caso, a possibilidade do evento  $A$  ocorrer antes do evento  $B$  é igual a 1, como mostra a Figura 9c.

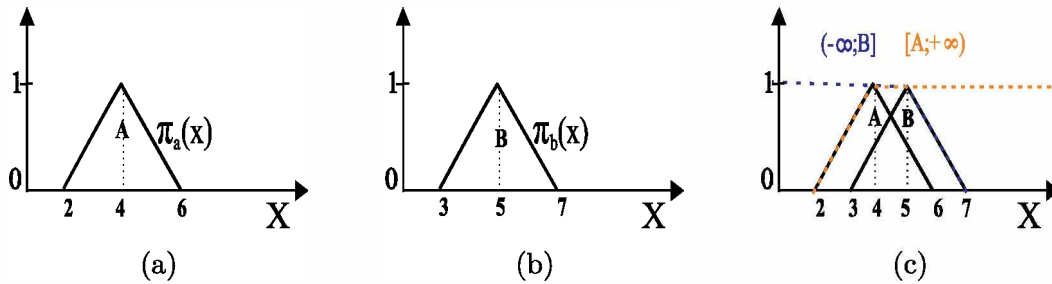


Figura 9 – Cálculo de Possibilidade entre dois Eventos.

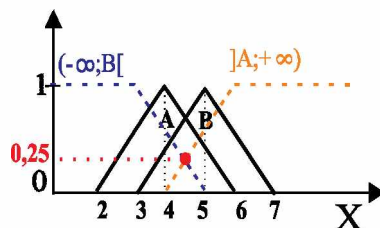


Figura 10 – Cálculo de Necessidade entre dois Eventos.

A certeza sobre o fato de que o evento  $A$  ocorrerá antes do evento  $B$  é dado pelo cálculo da necessidade, que neste caso é 0,25, como pode ser visualizado na Figura 10.

Dada uma distribuição de possibilidade  $\pi_a$ , (DUBOIS; PRADE, 1989) definem os seguintes conjuntos *fuzzy* para *time point*, que são:

- a) possivelmente depois de  $a$ :  $\mu_{[A,+\infty)}(x) = \sup_{x \in X} \pi_a(s)$  (Figura 11a);
- b) necessariamente depois de  $a$ :  $\mu_{]A,+\infty)}(x) = \inf_{x \in X} (1 - \pi_a(s))$  (Figura 11b);

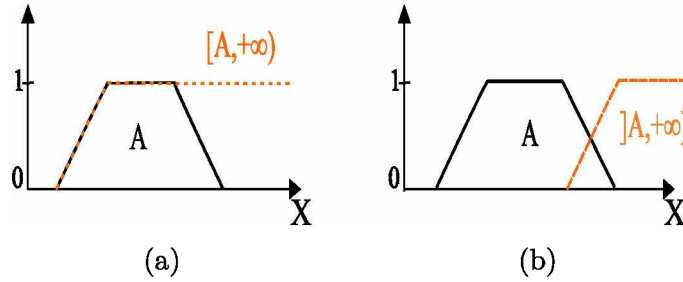


Figura 11 – Possivelmente/Necessariamente depois de  $a$ .

- c) possivelmente antes de  $a$ :  $\mu_{(-\infty, A]}(x) = \sup_{x \in X} \pi_a(s)$  (Figura 12a);
- d) necessariamente antes de  $a$ :  $\mu_{(-\infty, A[)}(x) = \inf_{x \in X} (1 - \pi_a(s))$  (Figura 12b).

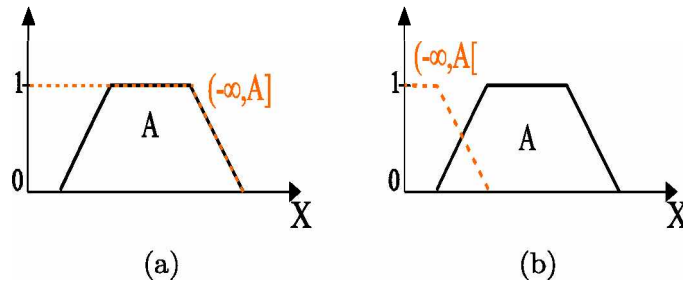


Figura 12 – Possivelmente/Necessariamente antes de  $a$ .

O intervalo de visibilidade de tempo fuzzy é definido em (CARDOSO, 1999).

**Definição 2 (Intervalo de visibilidade de tempo.)** Um intervalo de visibilidade de tempo  $[a, b]$  é um período de tempo entre duas datas  $a$  e  $b$ . No caso onde  $a$  e  $b$  são datas fuzzy  $A$  e  $B$  (delimitadas por  $\pi_a$  e  $\pi_b$ ), respectivamente, o intervalo  $[a, b]$  é representado pelo seguinte par de conjuntos fuzzy:

- a)  $[A, B]$ , conjunção de instantes de tempo que representam o conjunto de datas possivelmente depois de  $A$  e possivelmente antes de  $B$  (Figura 13).

- a)  $]A, B[$ , conjunção de instantes de tempo que representam o conjunto de datas necessariamente depois de  $A$  e necessariamente antes de  $B$  (Figura 14).

Dada a proposição “ $X$  é  $F$ ” onde  $X$  é uma variável e  $F$  uma restrição fuzzy sobre  $X$ ,  $F$  define a distribuição de possibilidade de uma variável  $X$ , denotada por  $\Pi_X$ , atuando sobre os valores de  $X$ . Formalmente:

$$\Pi_X = R(X) = F \quad (9)$$

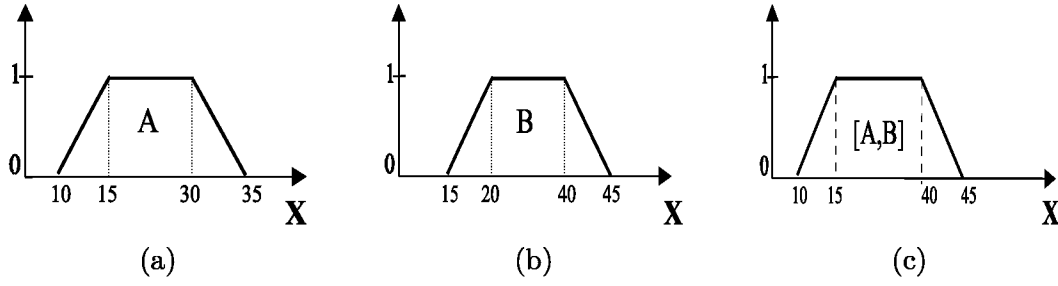


Figura 13 – Possivelmente depois de  $A$  e Possivelmente antes de  $B$ .

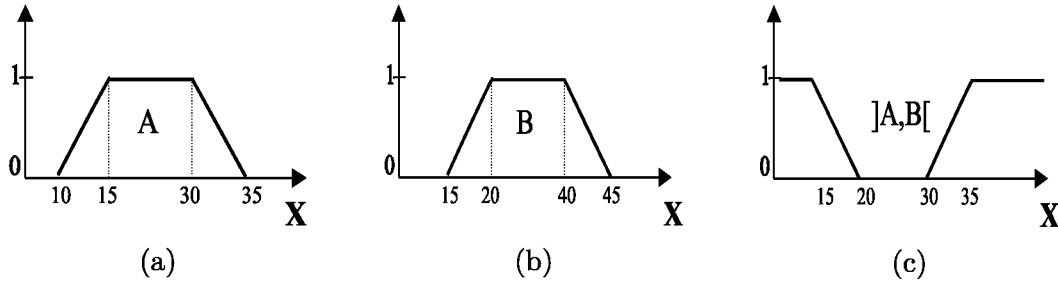


Figura 14 – Necessariamente depois de  $A$  e Necessariamente antes de  $B$ .

A função de distribuição de possibilidade  $X$ , denotada por  $\pi_X$ , é numericamente igual à função  $F$ .

$$\pi_X(u) = \mu_F(u) \quad (10)$$

onde  $\pi_X(u)$  é a possibilidade de  $X = u$ .

Na equação 9 assume-se que  $X$  depende de um único atributo. De modo mais geral, se a proposição “ $X$  é  $F$ ” contém mais que um atributo relacionado a variável  $X$  ( $A_i(X)$ ), então temos a definição de possibilidade conjunta  $\Pi_{(A_i(X).A_n(X))} = F$ . O correspondente à distribuição de possibilidade conjunta é dada por:

$$\pi_{(A_i(X).A_n(X))}(u_1..u_n) = \mu_F(u_1..u_n) \quad (11)$$

onde  $F$  é a relação *fuzzy* do produto cartesiano  $U = U_1X..XU_n$ .

A distribuição de possibilidade conjunta é dada por:

$$\pi_{(A_i(X).A_n(X))}(u_1..u_n) = \pi_{A_1(X)}(u_1) \wedge .. \wedge \pi_{A_n(X)}(u_n) \quad (12)$$

onde  $\pi_{A_i(X)}(u_i)$  denota a distribuição do atributo  $A_i(X)$  dado pela equação 10.

Suponha, por exemplo, a proposição “a pressão da caldeira é alta”.  $U$  é o universo dos valores de pressão,  $F$  é o conjunto *fuzzy* “alto” e  $X$  é a variável “pressão da caldeira”. O conjunto *fuzzy* “alto” atua como uma restrição sobre os valores da pressão da caldeira, isto é, os valores associados a esta variável que devem ser considerados são somente “alto” em algum grau. No entanto, se a estabilidade da caldeira depende da pressão ( $\Delta P$ ) e temperatura ( $\Delta T$ ), a proposição “a caldeira está estável” pode ser traduzida pela seguinte possibilidade conjunta.

$$\begin{aligned}
\text{caldeira} = \text{estável} &\Rightarrow \Pi_{\text{ESTABILIDADE}(\text{caldeira})} \\
&= \Pi_{(\Delta P(\text{caldeira}), \Delta T(\text{caldeira}))} \\
&= (\Pi_{\Delta P(\text{caldeira})} = \text{BAIXO} \wedge \Pi_{\Delta T(\text{caldeira})} = \text{BAIXO})
\end{aligned}$$

Dados dois conjuntos de referência  $X$  e  $Y$ , DuBois (1997) define a possibilidade conjunta como limite superior, tal que:

$$\forall x \in X \quad \forall y \in Y \quad \pi(x, y) = \min(\pi_X(x), \pi_Y(y)) \quad (13)$$

quando os conjuntos de referência são não-interativos (o valor de  $x \in X$  não tem influência sobre o valor de  $y \in Y$ , e vice versa).

### 2.1.3 Redes de Petri

As redes de Petri têm origem na tese de Carl Adam Petri intitulada *Comunicação com Autômatos* apresentada em 1962 na Universidade de Darmstadt. Apesar de ser considerada uma teoria relativamente jovem, a teoria se adapta bem a um grande número de aplicações em que as noções de eventos e de evoluções simultâneas são importantes (CARDOSO; VALETTE, 1997).

Redes de Petri são consideradas como uma ferramenta gráfica e matemática de representação formal que permite modelagem, análise e controle de sistemas a eventos discretos que comportam atividades paralelas, concorrentes e assíncronas (MURATA, 1989). Uma rede de Petri pode ser vista em particular como um grafo bipartido e direcionado com dois tipos de nós chamados de lugares e transições. Os nós são conectados por meio de arcos direcionados. Conexões entre dois nós do mesmo tipo não são permitidas. Dessa forma, os elementos básicos que permitem a definição das redes de Petri são:

- a) **Lugar:** em uma rede de Petri um lugar é representado por um círculo. Um lugar pode ser interpretado como uma condição, um estado parcial, um procedimento, a existência de um recurso, etc..
- b) **Transição:** uma transição é representada graficamente por uma barra ou retângulo. As transições são associadas a eventos que ocorrem em um sistema.
- c) **Ficha:** a ficha é representada por um ponto em um lugar. É um indicador significando que a condição associada ao lugar é verificada, como, por exemplo, um recurso disponível em um certo processo.

A Figura 15 ilustra um exemplo de uma rede de Petri e seus elementos básicos.

Formalmente, as redes de Petri são definidas da seguinte forma (CARDOSO; VALETTE, 1997):

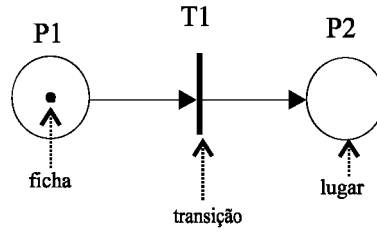


Figura 15 – Rede de Petri.

**Definição 3 (Rede de Petri)** Uma rede de Petri é uma quádrupla  $R = \langle P, T, \text{Pré}, \text{Pós} \rangle$  onde:

- a)  $P$  é um conjunto finito de lugares de dimensão  $n$ ,
- b)  $T$  é um conjunto finito de transições de dimensão  $m$ ,
- c)  $\text{Pré}: P \times T \rightarrow \mathbf{N}$  é a aplicação de entrada (lugares precedentes ou incidência anterior), com  $\mathbf{N}$  sendo o conjunto dos números naturais,
- d)  $\text{Pós}: T \times P \rightarrow \mathbf{N}$  é a aplicação de saída (lugares seguintes ou incidência posterior).

Um lugar  $p$  é chamado lugar de entrada de uma transição  $t$  se existe um arco direcionado de  $p$  para  $t$ . Por exemplo, na Figura 15, o lugar  $P1$  é lugar de entrada da transição  $T1$ . Já um lugar é chamado lugar de saída de uma transição se existe um arco direcionado de  $t$  para  $p$ . Ainda na Figura 15, o lugar  $P2$  é um lugar de saída de  $T1$ .

No caso das redes de Petri marcadas, a definição formal é a seguinte:

**Definição 4 (Rede de Petri Marcada)** Uma rede marcada  $N$  é uma dupla  $N = \langle R, M \rangle$  onde:

- a)  $R$  é uma rede de Petri,
- b)  $M$  é a marcação inicial dada pela aplicação  $M: P \rightarrow \mathbf{N}$ .

$M(p)$  é então o conjunto de fichas contidas no lugar  $p$ . A marcação  $M$  é a distribuição das fichas nos diversos lugares, sendo representada por um vetor coluna cuja dimensão é o número de lugares e os elementos são  $M(p)$ .

Um exemplo de rede de Petri marcada é dado na Figura 16 onde a marcação é dada por  $M^T = [1 \ 0 \ 3 \ 0 \ 1]$  ( $M^T$  é o vetor transposto).

Formalmente, a evolução dinâmica de uma rede de Petri é dada pelas definições seguintes:

**Definição 5** Uma transição  $t$  está sensibilizada ou habilitada se e somente se:

$$\forall p \in P, \quad M(p) \geq \text{Pré}(p, t) \quad (14)$$

Isto é, uma transição está sensibilizada se o número de fichas em cada um dos seus lugares de entrada for maior (ou igual) ao peso do arco que liga este lugar à transição.



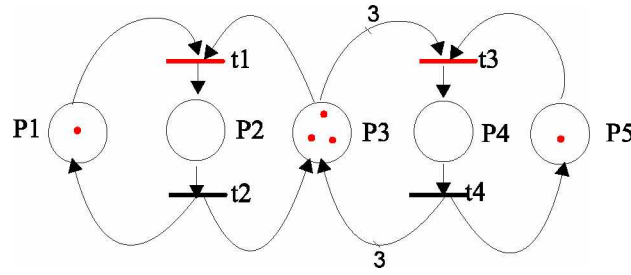


Figura 16 – Exemplo de uma Rede de Petri Marcada.

**Definição 6** Se  $t$  está sensibilizada por uma marcação  $M$ , uma nova marcação  $M'$  é obtida através do disparo de  $t$  de maneira que:

$$\forall p \in P, \quad M'(p) = M(p) - \text{Pre}(p, t) + \text{Post}(p, t) \quad (15)$$

O disparo de uma transição  $t$  consiste então em retirar as fichas dos lugares de entrada ( $\text{Pre}(p, t)$ ), e em depositar fichas em cada lugar de saída ( $\text{Post}(p, t)$ ). Um exemplo de disparo de transição é dado na Figura 17.

O disparo em seqüência de transições é chamado de *seqüência de disparo*. O vetor  $s$  é chamado vetor característico da seqüência  $s$ , onde cada componente  $s(t)$  representa o número de ocorrências da transição  $t$ . A dimensão do vetor característico  $s$  é igual ao número de transições da rede de Petri.

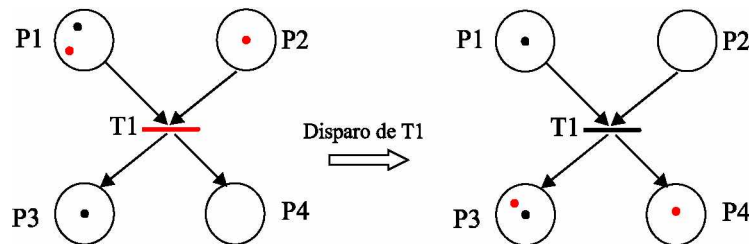


Figura 17 – Exemplo de Disparo de uma Transição.

Uma observação importante a ser feita é que as interpretações dos lugares e fichas podem ser bastante variadas. Estas, podem ser usadas para descrever entidades abstratas como condições ou estados, mas também entidades físicas como peças ou depósito. Uma transição pode ser associada a um evento que ocorre no sistema, ou até mesmo a execução de uma operação com duração estritamente positiva (não nulas). Uma ficha pode representar um objeto (máquina, peça) ou um indicador. Neste sentido, as redes de Petri permitem uma visão sintética do sistema a ser modelado e permite procedimentos de análise.

### 2.1.3.1 Propriedades das redes de Petri

Redes de Petri não se resumem apenas a uma ferramenta que permite a modelagem de problemas que tenham atividades concorrentes. Elas são utilizadas também para descrever e analisar um sistema (DAVID; ALLA, 2010). Para tal, foram desenvolvidos métodos que permitem a análise de várias propriedades de sistemas a eventos discretos (MURATA, 1989; PETERSON, 1981).

A dinâmica de um sistema descrito por uma rede de Petri é dada pela evolução das marcações. De acordo com o conjunto de marcações acessíveis a partir da marcação inicial, são definidas algumas propriedades. Essas propriedades são definidas por Murata (1989) como: alcançabilidade, limitabilidade, vivacidade e reiniciabilidade. Esse conjunto de propriedades é conhecido na literatura como boas propriedades.

- a) Alcançabilidade: uma marcação  $M_n$  é dita alcançável se existe uma seqüência de disparo de transições que a partir da marcação inicial  $M_0$  possibilita a chegada a  $M_n$ . Essa propriedade garante que certos estados sempre serão atingidos.
- b) Limitabilidade: uma rede é limitada ou k-limitada se o número de fichas em cada lugar não excede um número finito k para qualquer marcação alcançável a partir da marcação inicial.
- c) Vivacidade: uma rede é considerada viva se toda transição  $t$  pode ser sensibilizada a partir de qualquer marcação  $M'$  do grafo de marcações alcançáveis. Esse conceito, na prática, garante que o sistema será livre de bloqueios (*deadlock free*).
- d) Reiniciabilidade: uma rede de Petri é reiniciável se a partir de qualquer marcação acessível, existe uma seqüência de disparo que leva à marcação inicial  $M_0$ .

Existem também *propriedades estruturais* (CARDOSO; VALETTE, 1997) que são dependentes da estrutura da rede e não da marcação inicial. Estas propriedades são definidas através dos componentes conservativos de lugar e dos componentes repetitivos estacionários. Baseado nestes elementos estruturais, é possível definir invariantes de lugar e de transição que fornecem algumas informações sobre a dinâmica da rede.

### 2.1.3.2 Componentes Conservativos e Invariantes de Lugares

Considerando a Figura 18, é possível observar um circuito formado pelos lugares  $p1$  e  $p2$  e pelas transições  $t1$  e  $t2$ . A soma  $M(p1) + M(p2)$  vale 1 para a marcação inicial  $M_0 = [1 \ 0 \ 3 \ 0 \ 1]^T$ . O disparo de  $t1$  não modifica em nada esta soma, da mesma forma que o disparo de  $t2$  embora a marcação de cada lugar seja modificada a cada disparo de transição. O disparo de  $t3$  e  $t4$  também não modifica esta soma. Para este exemplo, pode-se verificar que, para todas as marcações acessíveis a partir da marcação inicial, tem-se  $M(p1) + M(p2) = 1$ , ou de modo mais geral  $\forall M \in A(R, M_0), M(p1) + M(p2) = M_0(p1) + M_0(p2)$ .<sup>1</sup>

<sup>1</sup> Todos os estados que podem ser atingidos em uma rede R com estado inicial  $M_0$  são denotados como  $A(R, M_0)$ .

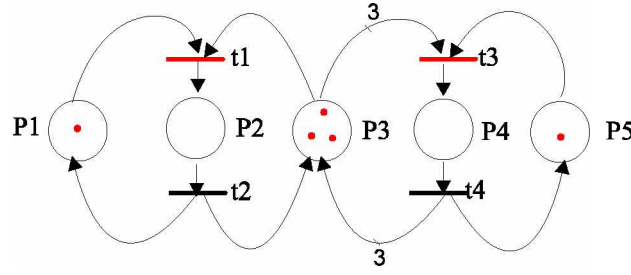


Figura 18 – Exemplo de uma Rede de Petri Marcada.

A forma linear  $M(p1) + M(p2) = M_0(p1) + M_0(p2)$  é chamada invariante de lugar, pois a soma das fichas se conserva para estes lugares. O conjunto de lugares  $p1$  e  $p2$  forma um componente conservativo da rede.

**Definição 7 (Invariante de Lugar)** *Um invariante de lugar é uma função linear da marcação dos lugares cujo valor é uma constante que depende apenas da marcação inicial da rede. Ele corresponde a uma restrição sobre os estados e as atividades do sistema que será sempre verificada, quaisquer que sejam suas evoluções.*

A equação que permite determinar a evolução da rede é a equação fundamental (equação 15). Para se obter uma soma ponderada das marcações, pré-multiplica-se a equação por um vetor  $f^T$ :

$$f^T M'(p) = f^T M(p) + f^T C s \quad (16)$$

onde  $C$  é a matriz de incidência definida como  $C = Pos(p, t) - Pre(p, t)$  e  $s$  é o vetor característico da seqüência de disparo  $s$ .

**Definição 8 (Componente conservativo)** *um componente conservativo de uma rede de Petri é o conjunto de lugares  $p_i \in P$  correspondentes aos elementos não nulos  $f_i$  do vetor coluna  $f$ , solução da equação:*

$$f^T C = 0 \quad (17)$$

Do ponto de vista gráfico, um componente conservativo define uma sub-rede de Petri. Uma rede de Petri é dita conservativa se todos os lugares da rede pertencem a um componente conservativo.

### 2.1.3.3 Componentes Repetitivos e Invariantes de Transição

Considerando novamente a Figura 18 e a sub-rede formada pelas transições  $t3$  e  $t4$  juntamente com seus lugares de entrada ou saída, é possível observar que o disparo da seqüência  $s = t3t4$  a partir da marcação inicial leva de volta à mesma marcação.

A seqüência  $s = t3t4$  é um invariante de transição, pois o disparo de tal seqüência não modifica a marcação da rede. O invariante de transição corresponde a uma seqüência cíclica de eventos que pode ser repetida indefinidamente. O conjunto das transições  $t3$  e  $t4$  do invariante forma um componente repetitivo estacionário.

Para encontrar o conjunto de transições que, uma vez disparadas a partir de uma marcação da rede, faz retornar a esta mesma marcação, utiliza-se a equação fundamental (equação 15). É fácil verificar que para obter-se  $M' = M$ , a seqüência  $s$  deve ser tal que o vetor  $s$  verifique:

$$Cs = 0 \quad (18)$$

Toda solução  $s$  desta equação é chamada componente repetitivo estacionário, e a seqüência  $s$  é dita invariante de transição. Uma rede de Petri é dita repetitiva se todas as transições  $t \in T$  pertencem a um componente repetitivo estacionário.

Com base nas boas propriedades e nas propriedades estruturais, é possível verificar através de métodos analíticos se o sistema modelado pela rede possui certos comportamentos. Há três formas de análise do comportamento de uma rede de Petri: a análise por enumeração das marcações, estrutural e por redução (MURATA, 1989).

### 2.1.4 Redes de Petri com Tempo

As redes de Petri clássicas não permitem a modelagem do tempo de forma explícita (durações de atividades e datas de ocorrência de início e fim de operações), mas estas podem ser estendidas em relação ao tempo, já que informações temporais são úteis para sequenciar eventos, comparar suas durações, bem como determinar o intervalo existente entre elas. Desta forma, é possível representar e analisar problemas ligados a diversas atividades onde a avaliação do tempo é um critério de fundamental importância. Vários trabalhos foram realizados no sentido de utilizar explicitamente o tempo, como parâmetro contínuo e quantificável em redes de Petri.

Com a adição de tempo, o indeterminismo com relação ao disparo de uma transição é, de certa forma, reduzido já que a informação temporal acrescenta uma nova relação de ordem entre os disparos das transições.

Um aspecto importante, seja qual for a forma de especificar o tempo, é como se interpreta o tempo. Este aspecto é denominado de semântica do tempo e, para as redes de Petri, existem basicamente, duas semânticas distintas:

- a) Tempo de Sensibilização, determina o tempo que deve transcorrer desde o momento da habilitação de uma transição até o momento do seu disparo. O disparo ocorre de forma atômica, ou seja, disparo instantâneo, indivisível.
- b) Tempo de Disparo, o disparo de uma transição ocorre em três fases: retirada da marca, disparo atômico e colocação da marca.

Outro aspecto importante, em relação ao tempo, é quanto a efetivação do disparo de uma transição sensibilizada. Para alguns sistemas, o disparo de uma transição sensibilizada deve ocorrer em determinada data. Nestes casos, a semântica do tempo é denominada semântica forte, em oposição à semântica fraca, na qual o disparo de uma transição sensibilizada não é obrigatório, porém, se ocorrer, deve ser em uma determinada data também.

As primeiras redes de Petri com interferência do tempo foram apresentadas por Ramchandani (1974), Merlin (1974), em suas teses de doutorado. Desde então, um grande número de diferentes modelos de redes de Petri com tempo tem sido proposto na literatura. Geralmente, os modelos se diferenciam em aspectos, tais como: tipo de temporização, localização da restrição temporal e propriedade da restrição (CERONE; MAGGIOLO-SCHETTINI, 1999). Independente desses aspectos, podem-se agrupar as extensões temporais das redes de Petri em quatro grandes categorias:

- a) Redes de Petri Temporais (*Time Petri Nets*). A restrição temporal é um intervalo de tempo associado a cada transição. Foram inicialmente usadas na descrição de protocolos de comunicação, entretanto, seu campo de aplicação tem se ampliado para áreas tais como: manufatura, sistemas em tempo real, validação e verificação de sistemas. (BERTHOMIEU; MENASCHE, 1983; CERONE; MAGGIOLO-SCHETTINI, 1999; BOWDEN, 2000; BERARD et al., 2013).
- b) Redes de Petri Temporizadas (*Timed Petri Nets*). Permitem que transições ou lugares retenham marcas durante um intervalo de tempo. São comumente usadas na análise de sistemas de manufatura (LAHAYE; KOMENDA; BOIMOND, 2014; DOTOLI et al., 2014; ZUBEREK, 1991).
- c) Marcas Temporizadas. A restrição temporal é associada à ficha ao se propagar na rede. São largamente usadas em problemas de produção (CELKO, 1982; TSAI; YANG; CHANG, 1995).
- d) Modelos Probabilísticos. São as chamadas redes de Petri Estocásticas e são usadas para estimar o desempenho de grandes sistemas. Uma probabilidade é então associada com uma das opções, transição, lugar ou marca (MURATA, 1989).

### 2.1.5 Rede de Petri Colorida

Geralmente as fichas em redes de Petri representam objetos ou recursos em um sistema modelado (AALST, 1992). Esses objetos podem ter atributos, os quais não são facilmente representados por uma simples ficha de uma rede de Petri. A fim de resolver este problema, Jensen (1981) definiu o que hoje se conhece por redes de Petri Coloridas, ou *Colored Petri Nets* (CPN).

CPN combina rede de Petri com uma linguagem de programação para obter uma linguagem de modelagem para sistemas concorrentes. Redes de Petri fornecem a base

formal para modelar concorrência e sincronização; uma linguagem de programação fornece as primitivas para modelar manipulação de dados e criação de modelos compactos e parametrizáveis. Sendo assim, em CPN as fichas possuem valores que são conhecidos como cores. Essas cores não significam apenas cores ou padrões, elas podem representar tipos de dados complexos (JENSEN; KRISTENSEN, 2009). Dessa forma, as CPNs foram projetadas para tornar mais compactos modelos de sistemas complexos, permitindo a individualização de fichas (usando as cores que lhe são atribuídas). Assim, diferentes processos ou recursos podem ser representados em uma mesma estrutura gráfica.

A aplicação prática da modelagem e análise das CPNs depende fortemente da existência de ferramentas computacionais que ofereçam suporte à criação e manipulação de tais modelos. CPN Tools é uma ferramenta adequada para edição, simulação e para análise do espaço de estado (*state space*) dos modelos de redes de Petri Coloridas (JENSEN; KRISTENSEN, 2009).

Há várias razões para o uso de Redes de Petri Coloridas e CPN Tools:

- a) O uso de redes de Petri fornece uma análise para correção do modelo por meio de, por exemplo, uma análise do *state space*. É importante salientar que a análise de estado é realizada a partir do modelo de alto nível através da replicação de cenário e não do estudo do modelo de rede de Petri ordinário subjacente. Isso é então um tipo de análise de propriedade de um modelo de alto nível que é o equivalente a um programa produzido numa linguagem de programação de alto nível.
- b) É muito fácil fazer pequenas adaptações dos modelos e depois comparar os resultados, dadas várias propostas de modelagem.
- c) A integração de processos e de dados complexos é essencial para modelagem de processos complexos. CPN Tools oferece a possibilidade de integrá-los em um único modelo.
- d) CPN Tools contém um ambiente de simulação, em que se pode seguir passo-a-passo uma sequência de disparo. Simulação de tipo Monte Carlo (com replicação de um mesmo cenário aleatório e o cálculo de estimativas de valores médios que pertencem a intervalos de confianças calculados automaticamente pelo *software*)

Por estes motivos, diversos trabalhos têm sido desenvolvidos quando se trata da modelagem e validação de modelos usando CPN Tools. Por exemplo, (AALST, 2015; LU et al., 2014; CHAN et al., 2014; VENERO; SILVA, 2014; XIAO; MING, 2011; VANDER-FEESTEN; AALST; REIJERS, 2005) e outros. Sendo assim, a ferramenta CPN Tools foi a escolhida para modelar, simular e validar o modelo proposto neste trabalho.

#### 2.1.5.1 Introdução a CPN Tools

As redes de Petri coloridas resultam em uma representação mais compacta em relação as redes lugar-transição devido as fichas carregarem valores de dados, que podem ser

complexos (ex: uma lista onde o primeiro elemento é um inteiro, o segundo é uma *string*, o terceiro é um par de inteiros), denominados cores das fichas. Cada lugar da rede deve possuir fichas cujas cores pertençam a um tipo específico, sendo este tipo denominado conjunto de cores do lugar. É necessário, portanto, (similarmente ao caso das linguagens de programação), declarar todos os conjuntos de cores (tipos) que existam na rede. É permitido definir cores a partir de operações entre cores já existentes (ex: união, produto, etc.) (CPN..., 2016).

Aos arcos desta rede, ao invés de números inteiros, são associadas inscrições que correspondem a expressões. Essas expressões determinam, na ocorrência de uma transição, quantas e de que cores são as fichas que devem ser removidas dos lugares de entrada e adicionadas aos lugares de saída. Inscrições também podem ser associadas às transições, sendo então denominadas guardas. Uma guarda é uma expressão que tem como resultado um valor booleano, cuja finalidade é restringir a ocorrência de uma dada transição.

As declarações de uma CPN (também denominadas nó de declaração) servem para especificar o tipo dos elementos que são suportados pelos lugares da rede, bem como as variáveis e funções utilizadas nas inscrições de arcos e guardas.

As inscrições dos arcos são construídas em função de variáveis de transição definidas nas declarações, cuja cor coincide com a cor do lugar de onde as fichas devem ser retiradas/adicionadas. O mesmo é válido para as guardas das transições. Uma ligação (*binding*) é a associação das variáveis de transição de um dado arco a um valor possível de cores. Um elemento de ligação é um par  $(t, b)$  onde  $t$  é uma transição e  $b$  é uma ligação para  $t$ .

Uma transição em uma CPN é dita estar sensibilizada se possuir em seus lugares um número de fichas suficiente para satisfazer as inscrições dos arcos (existir um elemento de ligação) e se sua guarda for satisfeita. Com isso são retiradas fichas dos lugares de entrada e são adicionadas fichas aos lugares de saída de acordo com as expressões (avaliação das inscrições) dos arcos que ligam a transição aos seus lugares de entrada e de saída. Uma CPN pode então ser definida como sendo composta de uma estrutura (um grafo bipartido e direcionado), de um conjunto de inscrições e de um conjunto de declarações.

No CPN Tools, cada elemento da rede de Petri possui atributos descritos na linguagem Standard ML. Os atributos dos respectivos elementos de um modelo serão apresentados a seguir.

### Inscrições nos Lugares

Existem três tipos de inscrições que podem ser associadas a um lugar como mostra a Figura 19, sendo que duas são opcionais e somente uma é obrigatória.

- a) PLACE TYPE (obrigatória) – inscrição do conjunto de cores associado ao lugar;
- b) INIT MARK (opcional) – inscrição da marcação inicial de um lugar;

- c) NAME (opcional) – nome do lugar, que não tem significado semântico, mas é importante para a compreensão do modelo.

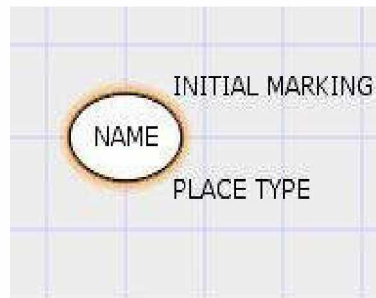


Figura 19 – Atributos de um Lugar de uma Rede de Petri Colorida.

### Inscrições nos Arcos

A expressão do arco *expr* tem que coincidir com o conjunto de cores associado ao lugar ligado ao arco, caso contrário, uma mensagem de erro aparecerá próxima ao arco, durante a verificação de sintaxe.

Existe uma diferença básica entre as inscrições dos arcos de entrada e as dos arcos de saída de uma transição. A expressão do arco de entrada da transição é constituída pela escolha de fichas dos lugares de entrada da transição. A expressão do arco de saída da transição é um construtor para a criação de novas fichas. Este construtor, frequentemente, usa variáveis das inscrições dos arcos de entrada da transição, como pode ser visto na Figura 20.

Na rede apresentada na Figura 20, a marcação inicial do lugar p1 consiste em cinco fichas de valor 3 (5 '3). A marcação inicial em p2 é composta por uma ficha de valor 4 (1 '4) e três fichas de valor 7 (3 '7). Então, p2, inicialmente, possui um total de quatro fichas. As variáveis x e y escolhem aleatoriamente uma ficha de seus respectivos lugares de entrada, retirando uma ficha de cada lugar p1 e p2. A expressão do arco de saída da transição é o construtor de uma nova ficha criada a partir da soma dos valores que as variáveis x e y assumem no momento do disparo da transição. No exemplo,  $x=3$  e  $y=7$ .

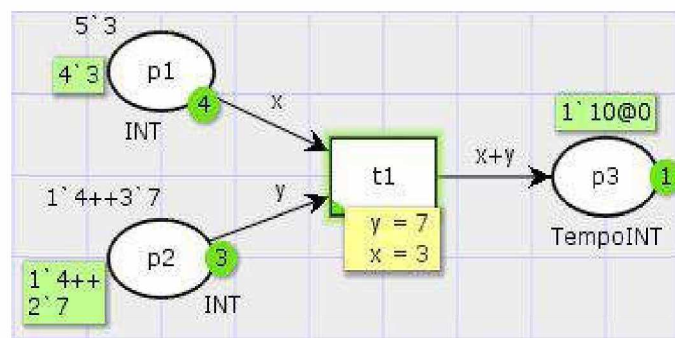


Figura 20 – Expressões nos Arcos de Entrada e de Saída de uma Transição.



As expressões dos arcos de saída das transições podem ser temporizadas. No exemplo apresentado na Figura 21, a cada 50 unidades de tempo, uma cópia da ficha que se encontra em p1 é colocada no lugar p2 e uma nova ficha de mesmo valor (4) é colocada em p1 com sua temporização acrescentada de +50 unidades de tempo.

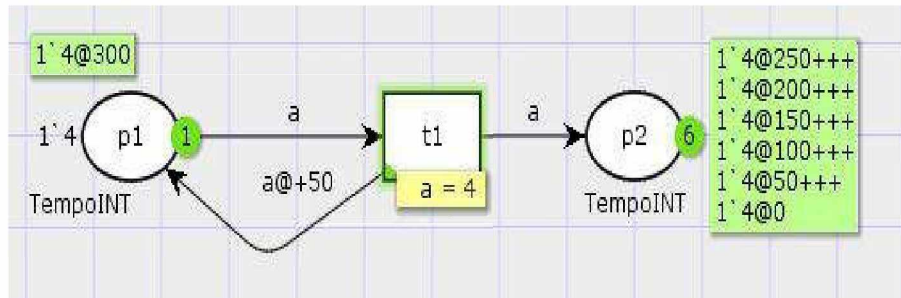


Figura 21 – Expressão Temporizada em um Arco de Saída de uma Transição.

### Inscrições nas transições

São quatro as inscrições que podem ser associadas às transições, como mostra a Figura 22. Todas são opcionais.

- a) Nome (NAME);
- b) Guarda (GUARD);
- c) Retardo de tempo (TIME DELAY);
- d) Segmento de Código (ACTION).

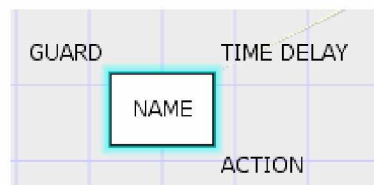


Figura 22 – Inscrições que Podem ser Associadas a uma Transição.

A inscrição *TIME DELAY* tem que ser uma expressão positiva. Ela é precedida pelos símbolos @+. Assim a inscrição possui a seguinte forma @+ delay-expr. O retardo de tempo é sempre relativo ao tempo atual, ou seja, se, por exemplo, o tempo atual é 20 e o retardo de tempo associado à transição é @ + 5, então, o tempo associado às fichas colocadas nos lugares de saída da transição, quando do seu disparo, será igual @ + 25.

Se a transição não possui tempo associado, isto indica que o atraso de tempo associado ao disparo da transição é zero. Diferente do atraso de tempo associado a um arco, que associa este atraso somente às fichas relativas àquele arco, o retardo de tempo associado a uma transição associa esse retardo a todas as fichas colocadas nos lugares de saída. A Figura 23 mostra a diferença da temporização associada ao arco de saída da transição em relação à temporização associada à própria transição.

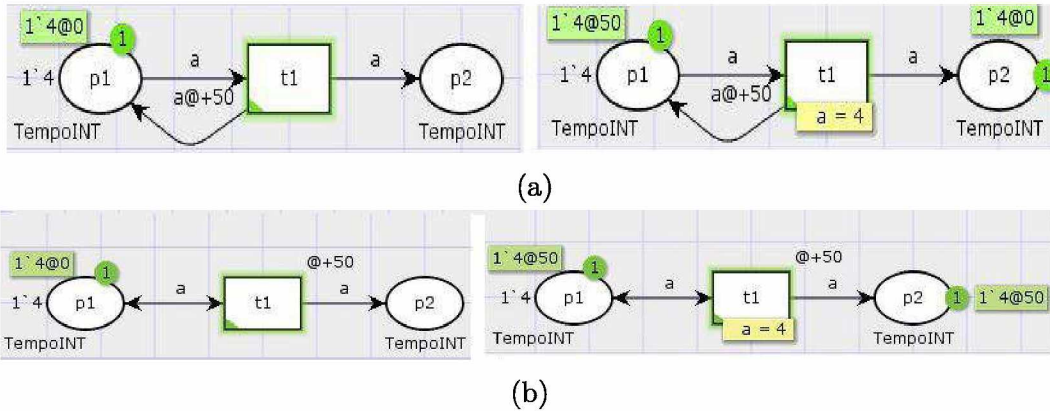


Figura 23 – Diferença da Temporização de uma Transição para a Temporização de um Arco de Saída da Transição.

Na Figura 23a, é apresentada uma rede com temporização somente no arco que liga t1 a p1. Na marcação inicial, a rede possui uma ficha (1'4@ + 0) no lugar p1. Após o disparo de t1, uma ficha 1'4@ + 0 foi colocada em p2 e uma ficha 1'4@ + 50 foi colocada em p1. Como não existe temporização nem na transição, nem no arco que liga t1 a p2, então a ficha nesse lugar continua com a mesma temporização da ficha em p1 da marcação anterior.

Na Figura 23b a temporização é associada à transição (@ + 50). Assim, na marcação inicial a rede possui uma ficha (1'4@ + 0) no lugar p1. Após o disparo de t1, as fichas nos lugares de saída da transição possuem uma temporização de @ + 50, pois este é o tempo acrescentado no disparo da transição.

A inscrição *GUARD* é uma expressão booleana da linguagem Standard ML, que é avaliada como verdadeira ou falsa. O valor inicial da guarda é sempre verdadeiro, antes que uma expressão seja adicionada. Uma guarda pode ser uma única expressão ou uma lista de expressões booleanas, [b-expr1, b-expr2, ..., b-exprn]. Quando uma expressão de guarda é associada a uma transição, então esta só poderá disparar se a expressão for verdadeira.

Considerando a Figura 24, por exemplo, a marcação inicial do lugar p1 consiste num total de 17 fichas definidas da seguinte forma: 4 fichas de valor 2 (4'2), 1 ficha de valor 4 (1'4), 9 fichas de valor 5 (9'5) e 3 fichas de valor 7 (3'7). A expressão de guarda da transição t1 é  $[a = 5 \text{ or else } a = 2]$ . Neste caso, t1 está sensibilizada, pois existem quatro fichas de valor 2 e nove fichas de valor 5 em p1. Após t1 ser disparada treze vezes, a nova marcação da rede é uma ficha de valor 4 (1'4) e três fichas de valor 7 (3'7) em p1, e quatro fichas de valor 2 (4'2) e nove fichas de valor 5 (9'5) em p2. Nesta nova marcação, a transição não está mais sensibilizada pois sua expressão *booleana* é avaliada como falsa, já que não existe mais nenhuma ficha em p1 de valor 2 ou 5.

Usando a expressão de guarda, é possível comparar fichas (ou parte) de diferentes lugares de entrada da transição. No exemplo da Figura 25, o valor do segundo campo

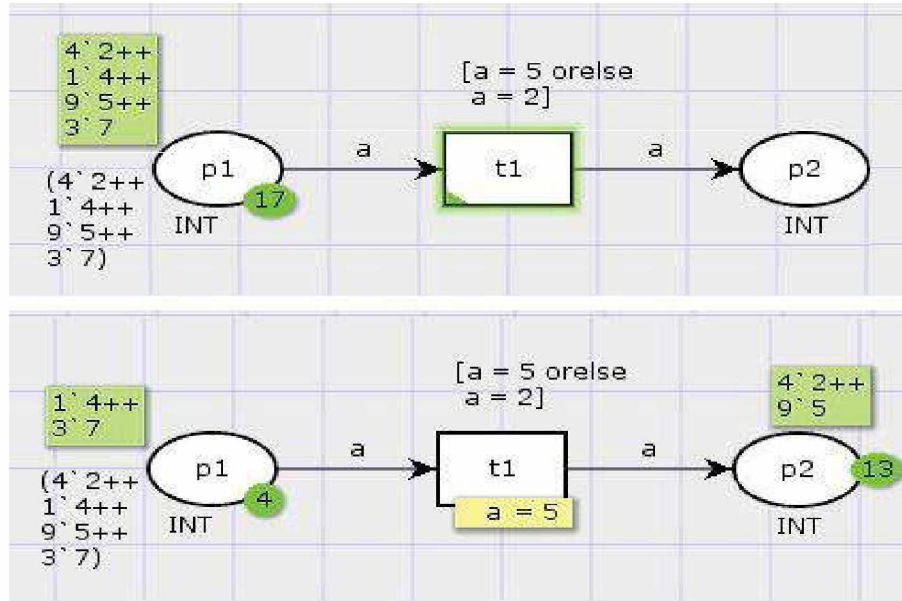


Figura 24 – Expressão de Guarda em uma Transição Restringe o Disparo para quando a Expressão for Verdadeira.

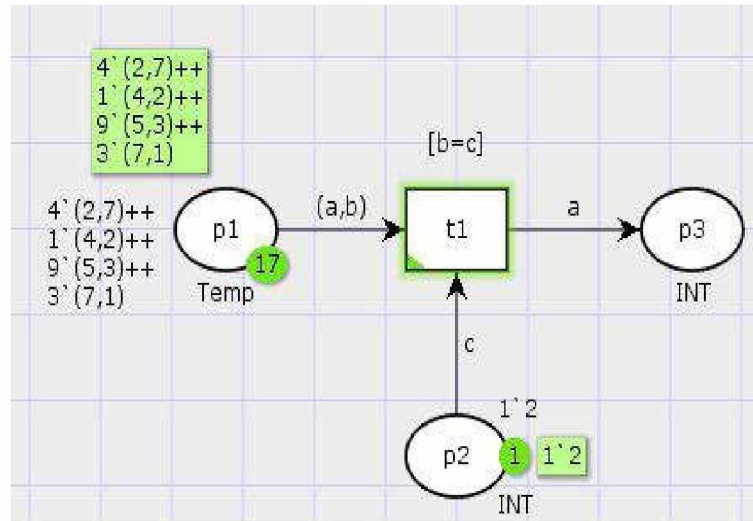


Figura 25 – Exemplo de aplicação da expressão de guarda de uma transição.

da ficha em  $p1$  é comparado com o valor da ficha em  $p2$ . Considerando a marcação da Figura 25 então a transição está sensibilizada, porque  $p1$  possui uma ficha (4,2) e  $p2$  uma ficha de valor 2. Na hora do disparo, o segundo campo da ficha (4,2) em  $p1$  é associado a variável  $b$  e a ficha 2 do lugar  $p2$  é associada a variável  $c$ ; assim, a expressão  $[b = c]$  se torna verdadeira.

A inscrição ACTION, também opcional, contém um código ML. Um segmento de código é executado quando a transição é disparada. Cada segmento de código pode utilizar variáveis CPN e pode associar estas variáveis aos arcos de saída de uma transição, mesmo que elas não tenham sido associadas a nenhuma ficha dos lugares de entrada da transição. Um segmento de código pode conter:



- a) Padrão de entrada (*Input pattern*) – opcional;
- b) Padrão de saída (*Output pattern*) – opcional;
- c) Código (*Code action*) – obrigatório;

Um *Input pattern* é um conjunto de variáveis CPN, precedidas pela palavra reservada *input*. O *Input Pattern* lista as variáveis CPN que podem ser usadas no *Code action*, sem que os valores destas variáveis possam ser modificados. Se o *Input pattern* não é declarado, então, nenhuma variável CPN pode ser usada no *Code action*.

Um *Output pattern* é um conjunto de variáveis CPN, precedidas pela palavra reservada *output*. O *Output pattern* lista as variáveis CPN a serem modificadas como resultado da execução do *Code action*. Se o *Output pattern* não é declarado, então, nenhuma variável CPN pode ser calculada.

Um *Code action* é uma expressão ML precedida pela palavra reservada *action*. O *Code action* não pode conter nenhuma declaração de conjunto de cores, variáveis CPN, ou variáveis de referência. No entanto, ele pode utilizar constantes pré-declaradas ou declaradas pelo usuário, operações e funções. Em adição, novas funções e constantes podem ser definidas localmente por meio da cláusula *let-in-end*. O *Code action* é executado como uma declaração local em um ambiente contendo variáveis CPN especificadas no *Input pattern*. Isso garante que o *Code action* não pode modificar diretamente quaisquer variáveis CPN, somente cópias locais.

Quando o *Code action* é executado, seu resultado é aplicado às variáveis CPN declaradas no *Output pattern*. Quando o *Code action* é avaliado em um ambiente contendo um *Input pattern*, as variáveis, obrigatoriamente, produzem um resultado do mesmo tipo das variáveis do *Output pattern*.

Normalmente, o segmento de código é usado para um processamento mais complexo das fichas de entrada. A soma de fichas apresentada no exemplo da Figura 20 pode ser representada usando segmento de código, como mostra a Figura 26.

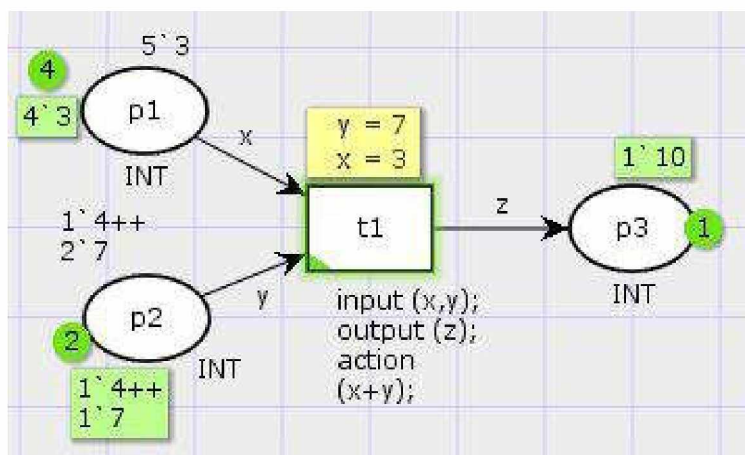


Figura 26 – Exemplo de Segmento de Código Associado a uma Transição,

### 2.1.5.2 CPN Hierárquica

O CPN Tools também permite estruturar os modelos CPN em módulos. O conceito de módulos em CPN é baseado em um mecanismo de estruturação hierárquica. A ideia básica da hierarquia por trás das CPNs é permitir a construção de um amplo modelo combinando um número de pequenas redes de Petri Coloridas em um único modelo (JENSEN; KRISTENSEN, 2009). Desta forma, as CPN Hierárquicas permitem relacionar um número de CPN individuais uma com a outra de um modo formal, ou seja, de um modo que se tenha uma semântica bem definida e permita, desta forma, uma análise formal. CPNs Hierárquicas podem ser transformadas em CPNs com modos comportamentais equivalentes a um modelo global único e vice-versa. Sendo assim, os conceitos básicos e os métodos de análise de CPN podem ser generalizados para as CPN Hierárquicas.

Dois dos construtores de linguagens de CPN Hierárquicas são transições de substituição e lugares de fusão. Estes construtores permitem a construção de uma CPN Hierárquica mais complexa, através da composição de várias CPNs menos complexas. O objetivo das transições de substituição é permitir a relação de uma transição e seus respectivos arcos com uma CPN mais complexa, que, usualmente, fornece uma descrição mais detalhada e precisa da funcionalidade representada pela transição de substituição. Um conjunto de CPNs é chamado de página. Subpágina é a página que contém a descrição detalhada da funcionalidade modelada pela correspondente transição de substituição. Já o objetivo dos lugares de fusão é permitir a especificação de um conjunto de lugares como sendo idênticos, ou seja, todos estes lugares representam um único lugar, conceitualmente, embora sejam desenhados como lugares individuais. Isto significa que, quando uma ficha é adicionada/removida de um destes lugares, uma ficha idêntica é adicionada/removida de todos os outros lugares.

A Figura 27 apresenta um exemplo de lugares de fusão. Cada lugar pertencente a um conjunto de lugares de fusão possui associado uma etiqueta com um único nome que identifica o conjunto. Todos os lugares de um conjunto de fusão são considerados pelo CPN Tools como um único lugar. Se uma marcação muda em um dos lugares de fusão, ela mudará também em todos os outros lugares do conjunto. Desta forma, o mesmo conjunto de cores é associado a todos os lugares de um mesmo conjunto de fusão, isto pode ser visto na Figura 27 observando-se os lugares A, B e C antes e depois do disparo.

A transição de substituição é a forma usada pelo CPN Tools para se substituir uma transição de uma rede de mais alto nível por uma rede de mais baixo nível. Seja o exemplo apresentado na Figura 28. Nele, a página de mais alto nível *Principal* possui a transição de substituição *t1*, que é substituída pela subpágina *Secundaria*. A substituição é assinalada pela etiqueta Secundaria que aparece próxima à transição *t1* na página *Principal*. É importante notar também que a subpágina associada a *t1* possui o mesmo nome da etiqueta.

Deve-se observar que os lugares *P1* e *P2*, que são respectivamente entrada e saída de

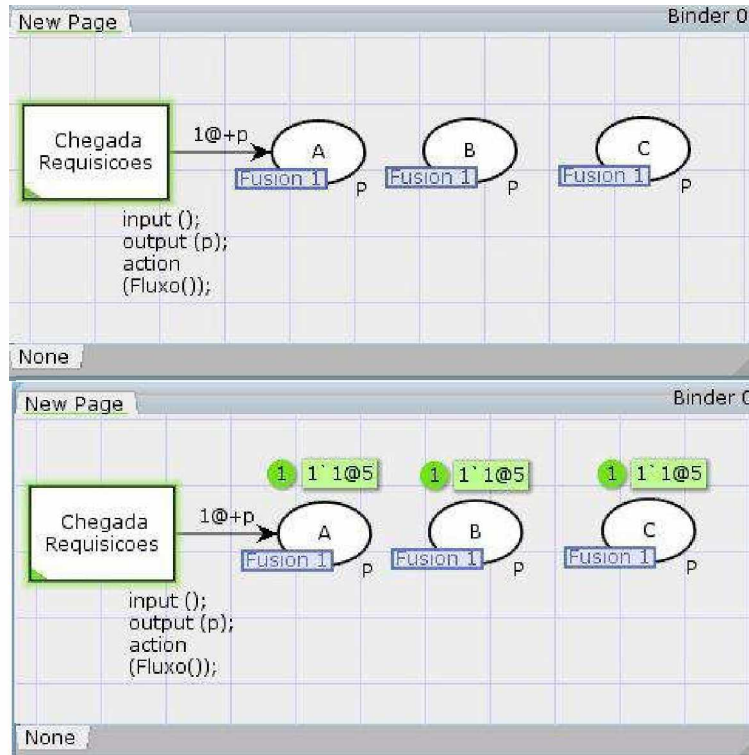


Figura 27 – Um conjunto de lugares de fusão representa um único lugar.

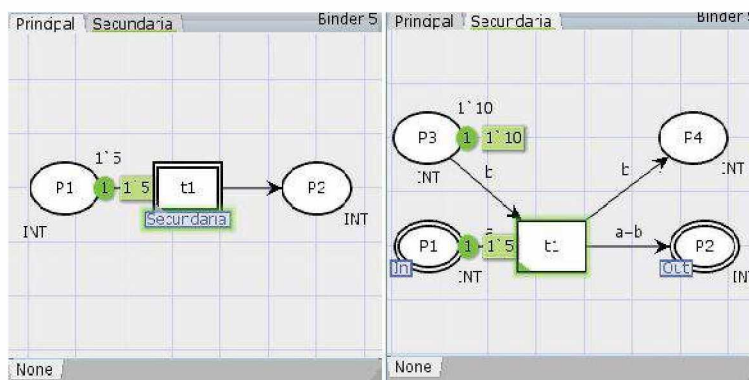


Figura 28 – Transição de substituição  $t_2$  e sua subrede associada secundaria.

$t_1$ , aparecem tanto na página *Principal*, como na página *Secundaria*. Na página *Principal* estes lugares são chamados de *sockets* e na *Secundaria* são chamados de *ports* e são etiquetados, respectivamente, com as etiquetas In e Out.

### 2.1.6 Jogador de Rede de Petri - “*Token Player*”

Segundo (CARDOSO; VALETTE, 1997), a rede de Petri pode ser considerada como um sistema de regras de produção. Cada transição é então considerada como uma regra de transformação do estado; a partir de um estado dado (marcação), deve-se procurar uma regra aplicável para passar ao estado seguinte.

Uma maneira de implementar o comportamento dinâmico de redes de Petri em uma

grande variedade de domínios é o conceito de Jogador de Rede de Petri, também conhecido como “*Token Player*” (VALETTE; ATABAKHCHE, 1987).

Para (CARDOSO; VALETTE, 1997), um motor de inferência especializado, frequentemente chamado jogador de rede de Petri, é capaz de jogar a rede de Petri, isto é, de deslocar as fichas de modo a respeitar as regras de disparo das transições. O funcionamento do jogador é descrito pela Figura 29. O disparo das transições externas deve ser sincronizado com a ocorrência do evento associado. Isto corresponde à recepção de mensagens de atualização do conhecimento ou, então, a requisições provenientes do sistema.

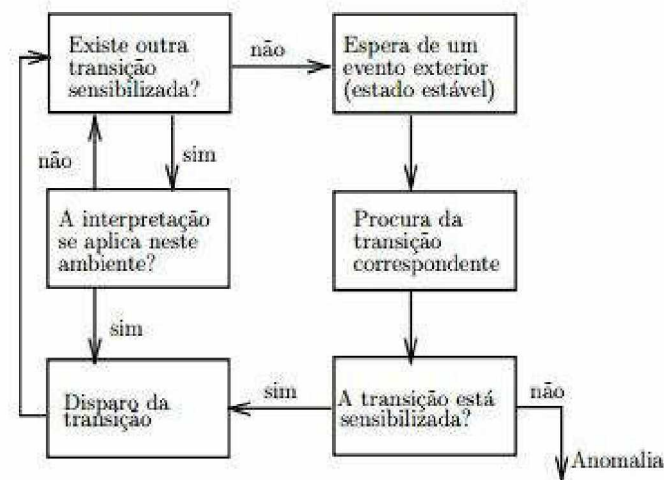


Figura 29 – Estrutura de um Jogador de Rede de Petri.

Por outro lado, a partir de uma marcação dada, todas as transições internas sensibilizadas devem ser disparadas antes de considerar o evento externo seguinte. Elas correspondem a conclusões (decisões) internas e se traduzem, eventualmente, por emissões de mensagens (comandos ou respostas) para o exterior.

O estado estável de uma rede é uma marcação para a qual somente as transições associadas a eventos externos estão sensibilizadas. O jogador deve então se colocar num estado de “escuta” do ambiente externo.

Quando uma mensagem é recebida, é necessário procurar a transição que lhe é associada e dispará-la. Em seguida, a partir da nova marcação, é necessário disparar todas as transições internas sensibilizadas a fim de alcançar um novo estado estável. Disparando todas as transições internas, as ações associadas são executadas, o que provoca operações sobre os dados (parte operativa) e envios de mensagens para o ambiente externo do sistema.



### 2.1.7 Workflow nets

Uma rede de Petri que modela um processo de negócio é chamada de *Workflow net* (AALST, 1998; AALST; HEE, 2004). Uma *Workflow net* satisfaz os seguintes requisitos (AALST, 1998):

- a) uma *Workflow net* tem apenas um lugar de início ( $i$ ) e apenas um lugar de fim ( $o$ ). Esses dois lugares são tratados como lugares especiais. O lugar  $i$  tem apenas arcos de saída e o lugar  $o$  tem apenas arcos de entrada. Uma ficha em  $i$  representa um caso que precisa ser tratado. Uma ficha em  $o$  representa um caso que já foi tratado.
- b) em uma *Workflow net* cada tarefa (transição) e condição (lugar) deve estar em um caminho que se encontra entre o lugar de início ( $i$ ) e o lugar de término ( $o$ ).

A definição formal de uma *Workflow net* é apresentada a seguir (AALST; HEE, 2004).

**Definição 9 (Workflow Net)** A rede de Petri  $R = (P, T, F)$  é uma *Workflow net* se e somente se:

1. existe um único lugar de início  $i \in P$  tal que  $\bullet i = \phi$ ;
2. existe um único lugar de saída  $o \in P$  tal que  $o \bullet = \phi$ ;
3. cada nó  $x \in P \cup T$  está em um caminho entre os lugares  $i$  e  $o$ .

Um processo especifica quais tarefas precisam ser executadas e em qual ordem executá-las. Modelar um processo de negócio em termos de uma *Workflow net* é bem direto: as tarefas são modeladas por transições, condições são modeladas por lugares e os casos são modelados pelas fichas (AALST, 1998).

Para ilustrar o mapeamento de um processo por uma *Workflow net*, considera-se o processo de tratamento de reclamações (“Handle Complaint Process”) apresentado em (AALST; HEE, 2004):

An incoming complaint first is recorded. Then the client who has complained and the department affected by the complaint are contacted. The client is approached for more information. The department is informed of the complaint and may be asked for its initial reaction. These two tasks may be performed in parallel – that is, simultaneously or in any order. After this, the data are gathered and a decision is taken. Depending upon the decision, either a compensation payment is made or a letter is sent. Finally, the complaint is filed (AALST; HEE, 2004).

A Figura 30 mostra a *Workflow net* que representa este processo. De acordo com Aalst e Hee (2004), as seguintes construções básicas para o roteamento de tarefas devem ser consideradas:

- a) sequencial: a forma mais simples de execução de tarefas, onde uma tarefa é executada após a outra, havendo, claramente, dependência entre elas;



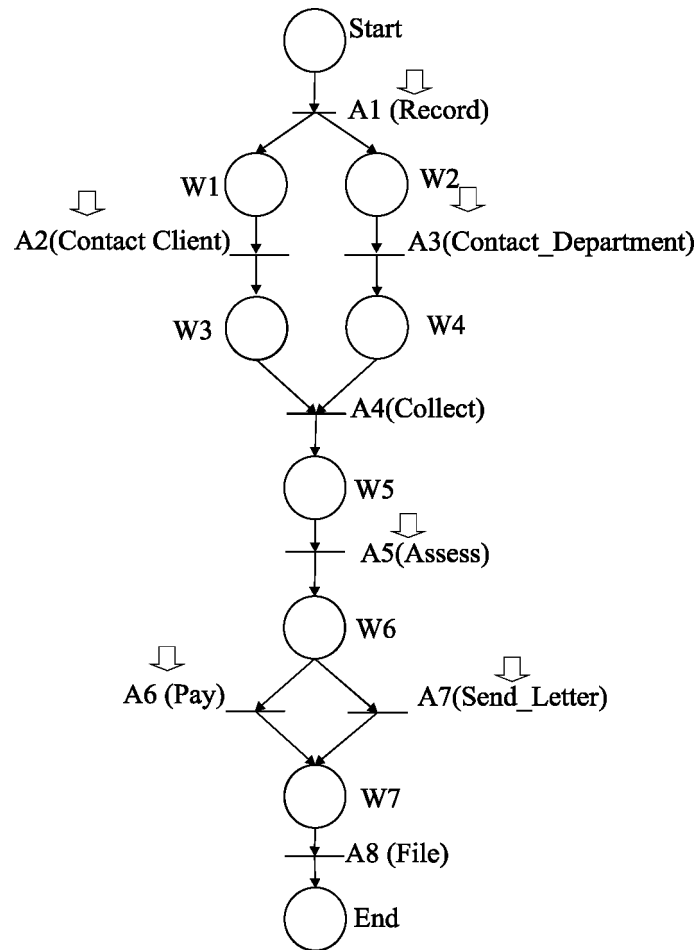


Figura 30 – *Workflow net* para o “Processo de Tratamento de Reclamações”.

- b) paralela: mais de uma tarefa pode ser executada simultaneamente, ou em qualquer ordem;
- c) condicional (ou rota seletiva): quando há uma escolha entre duas ou mais tarefas;
- d) iterativa: quando é necessário executar uma mesma tarefa (ou conjunto de tarefas) múltiplas vezes.

Considerando o processo de tratamento de reclamações mostrado na Figura 30, as tarefas *Contact\_Client* e *Contact\_Department* são um exemplo de roteamento paralelo. As tarefas *Collect* e *Assess* são um exemplo de roteamento sequencial. Já as tarefas *Pay* e *Send\_Letter* são um exemplo de roteamento condicional.

Aalst e Hee (2004) também definem a noção de acionamento, sendo um acionamento uma condição externa que guia a execução de uma tarefa sensibilizada. De acordo com Aalst e Hee (2004), há quatro tipos distintos de acionamento: usuário, mensagem, tempo e automático. No entanto, somente nos acionamentos do tipo usuário (este tipo de acionamento é identificado através do símbolo  $\Downarrow$  associado às transições), onde uma tarefa é acionada por um recurso (humano, uma máquina, uma equipe etc.), há a necessidade de alocação de recurso para o tratamento da tarefa. Nos demais tipos de acionamento,

não é obrigatória a alocação de recursos associada. Na Figura 30, as tarefas *Record*, *Contact\_Client*, *Contact\_Department*, *Assess*, *Send\_Letter* e *Pay* são tarefas do tipo usuário e as três demais tarefas (*Record*, *Collect* e *File*) são tratadas automaticamente.

*Soundness* é o principal critério de correção definido para as *Workflow nets*. Uma *Workflow net* é *Sound* se, e somente se, os três requisitos a seguir são satisfeitos (AALST; HEE, 2004):

- a) para cada ficha colocada no lugar de início  $i$ , uma (e apenas uma) aparece no lugar de término;
- b) quando uma ficha aparece no lugar  $o$ , todos os outros lugares estão vazios, considerando o caso em questão;
- c) considerando uma tarefa associada a uma transição, é possível evoluir da marcação inicial até uma marcação que sensibiliza tal transição, ou seja, não deve haver nenhuma transição morta na *Workflow net*.

A definição formal do critério de correção *Soundness* no contexto das *Workflow nets*, proposto por Aalst (1998), é apresentada na sequência.

**Definição 10 (*Soundness*)** Um processo de workflow modelado por uma *Workflow net*  $PN = (P, T, F)$  é *Sound* se, e somente se:

- a) para cada marcação  $M$  alcançável a partir da marcação  $i$ , existe uma sequência de disparo que leva da marcação  $M$  para a marcação  $o$ . Formalmente:

$$\forall_M (i \xrightarrow{*} M) \Rightarrow (M \xrightarrow{*} o);$$

- b) a marcação  $o$  é o único estado alcançável a partir da marcação  $i$  com pelo menos uma ficha no lugar  $o$ . Formalmente:

$$\forall_M (i \xrightarrow{*} M \wedge M \geq o) \Rightarrow (M = o);$$

- c) não há nenhuma transição morta em  $(PN, i)$ . Formalmente:

$$\forall_{t \in T} \exists_{M, M'} (i \xrightarrow{*} M \xrightarrow{t} M');$$

A *Workflow net* mostrada na Figura 30 é *Sound*, pois quando uma ficha atinge o lugar de término, não existe nenhuma ficha remanescente, considerando qualquer sequência de disparo.

## 2.1.8 Distribuições de Probabilidade

Existem algumas distribuições especiais de probabilidade contínuas que serão utilizadas neste trabalho e por isto merecem ser apresentadas. São elas: distribuição uniforme, exponencial e normal. Estas e outras definições podem ser encontradas em (JAMES, 1981).

### 2.1.8.1 Distribuição Uniforme

A distribuição uniforme é a mais simples distribuição contínua, entretanto uma das mais importantes e utilizadas dentro da teoria de probabilidade. A distribuição uniforme tem uma importante característica a qual a probabilidade de acontecer um fenômeno de mesmo comprimento é a mesma.

**Definição 11 (*Distribuição Uniforme*)** Uma variável aleatória  $X$  tem distribuição uniforme no intervalo  $[a, b]$  se sua função densidade de probabilidade for dada por:

$$f(x) = \begin{cases} \frac{1}{b-a}, & \text{se } a \leq x \leq b; \\ 0, & \text{caso contrário.} \end{cases}$$

Como exemplo de distribuição uniforme, considere que a ocorrência de panes em qualquer ponto de uma rede telefônica de 7 km foi modelada por uma distribuição uniforme no intervalo  $[0, 7]$ . Qual é a probabilidade de que uma pane venha a ocorrer nos primeiros 800 metros? E qual a probabilidade de que ocorra nos 3 km centrais da rede?

A função densidade da distribuição uniforme é dada por  $f(x) = \frac{1}{7}$  se  $0 \leq x \leq 7$  e zero, caso contrário. Assim, a probabilidade de ocorrer pane nos primeiros 800 metros é:

$$\mathbb{P}(X \leq 0,8) = \int_0^{0,8} f(x)dx = \frac{0,8 - 0}{7} = 0,1142.$$

e a probabilidade de ocorrer pane nos 3 km centrais da rede é:

$$\mathbb{P}(2 \leq X \leq 5) = \int_2^5 f(x)dx = \mathbb{P}(X \leq 5) - \mathbb{P}(X \leq 2) = 5/7 - 2/7 \approx 0,4285.$$

### Função Geradora de Momentos, Valor Esperado e Variância

Seja  $X$  uma variável contínua com distribuição uniforme, então sua função geradora de momentos é dada por:

$$M_X(t) = \mathbb{E}[e^{tX}] = \int_a^b \frac{e^{tx}}{b-a} dx = \frac{e^{tb} - e^{ta}}{t(b-a)} \Big|_a^b = \frac{e^{tb} - e^{ta}}{t(b-a)}$$

O valor esperado de uma variável aleatória  $X$  com distribuição uniforme é dado por:

$$\mathbb{E}(X) = \int_a^b x \frac{1}{b-a} dx = \frac{a+b}{2}$$

Outra forma de calcular é utilizando a função geradora de momentos.

$$M'_X(t) = \frac{e^{at}(at-1) + e^{bt}(1-bt)}{t^2(a-b)}$$

de onde segue que:

$$\mathbb{E}(X) = M'_X(0) = \lim_{t \rightarrow 0} \frac{e^{at}(at-1) + e^{bt}(1-bt)}{t^2(a-b)}.$$

Aplicando a regra de L'Hospital (uma vez que, no limite, tanto o numerador quanto o denominador vão para zero, tem-se que:

$$M'_X(0) = \lim_{t \rightarrow 0} \frac{t(e^{at}a^2 - e^{bt}b^2)}{2t(a-b)} = \frac{(a-b)(a+b)}{2(a-b)} = \frac{a+b}{2}$$

e, portanto,

$$\mathbb{E}(X) = \frac{a+b}{2}.$$

Calcula-se agora  $E[X^2]$  a partir da função geradora de momentos.

$$M''_X(t) = \frac{e^{at}(a^2t^2 - 2at + 2) - e^{bt}(b^2t^2 - 2bt + 2)}{t^3(a-b)}.$$

E então,

$$\mathbb{E}(X^2) = M''_X(0) = \lim_{t \rightarrow 0} \frac{e^{at}(a^2t^2 - 2at + 2) - e^{bt}(b^2t^2 - 2bt + 2)}{t^3(a-b)}$$

e aplicando a regra de L'Hospital, segue que

$$M''_X(0) = \lim_{t \rightarrow 0} \frac{t^2(a^3e^{at} - b^3e^{bt})}{3(a-b)} = \frac{a^3 - b^3}{3(a-b)} = \frac{(a-b)(a^2 + ab + b^2)}{3(a-b)}$$

Desta forma, tem-se que a variância  $\text{Var}(X)$  é dada por:

$$\text{Var}(X) = \mathbb{E}(X^2) - \mathbb{E}^2(X) = M''_X(0) - (M'_X(0))^2 = \frac{(a^2 + ab + b^2)}{3} - \left(\frac{(a+b)}{2}\right)^2 = \frac{b^2 - 2ab + a^2}{12}.$$

Assim,

$$\text{Var}(X) = \mathbb{E}(X^2) - \mathbb{E}^2(X) = \frac{(b-a)^2}{12}.$$

### 2.1.8.2 Distribuição Exponencial

Esta é uma distribuição que se caracteriza por ter uma função de taxa de falha constante. A distribuição exponencial é a única com esta propriedade. Ela é considerada uma das mais simples em termos matemáticos. Esta distribuição tem sido usada extensivamente como um modelo para o tempo de vida de certos produtos e materiais.

**Definição 12 (Distribuição Exponencial)** *A variável aleatória  $X$  tem distribuição Exponencial com parâmetro  $\lambda$ ,  $\lambda > 0$ , se tiver função densidade de probabilidade dada por:*

$$f(x) = \begin{cases} \lambda e^{-\lambda x} & \text{se } x \geq 0 \\ 0 & \text{se } x < 0 \end{cases}$$

em que  $\lambda$  é o parâmetro de taxa da distribuição e deve satisfazer  $\lambda > 0$ . Neste caso,  $\lambda$  é o tempo médio de vida e  $x$  é um tempo de falha. O parâmetro deve ter a mesma unidade do tempo da falha  $x$ . Isto é, se  $x$  é medido em horas,  $\lambda$  também será medido em horas.

A função de distribuição acumulada  $F(x)$  é dada por

$$F(x) = \int_0^x f(s)ds = \begin{cases} 1 - e^{-\lambda x} & \text{se } x \geq 0 \\ 0 & \text{se } x < 0 \end{cases}$$

Utiliza-se a notação  $X \sim \text{Exp}(\lambda)$ .

A distribuição Exponencial pode ser parametrizada de uma forma alternativa segundo a função densidade de probabilidade dada por

$$f(x) = \begin{cases} \frac{1}{\beta} e^{-\frac{x}{\beta}} & \text{se } x \geq 0 \\ 0 & \text{se } x < 0 \end{cases}.$$

Neste caso, pode-se dizer que  $\beta > 0$  é o parâmetro de escala da distribuição e é o inverso do parâmetro taxa na definição acima. Nesta definição alternativa, a variável aleatória  $X$  pode ser interpretada como a duração de tempo em que um sistema mecânico ou biológico sobrevive. Para este caso, denota-se  $X \sim \text{Exp}(\beta)$  e, infelizmente, esta definição alternativa torna-se ambígua. Neste caso, deve-se verificar qual das duas especificações está sendo utilizada quando se escreve  $X \sim \text{Exp}(\lambda)$ . Ou seja, deve-se sempre verificar se  $\lambda$  está se referindo ao parâmetro taxa ou ao parâmetro escala da distribuição. Especificamente nestas definições, sempre que se escreve  $X \sim \text{Exp}(\lambda)$  está se referindo à parametrização em que  $\lambda$  é o parâmetro taxa. É importante ressaltar que a função exponencial, na verdade, é um caso particular da função Gama, pois se  $X \sim \text{Exp}(\lambda)$ , então  $X \sim \text{Gama}(1, \lambda)$ .

A Figura 31 mostra a distribuição exponencial com parâmetros  $\lambda = 1/2, 1$  e  $3/2$ .

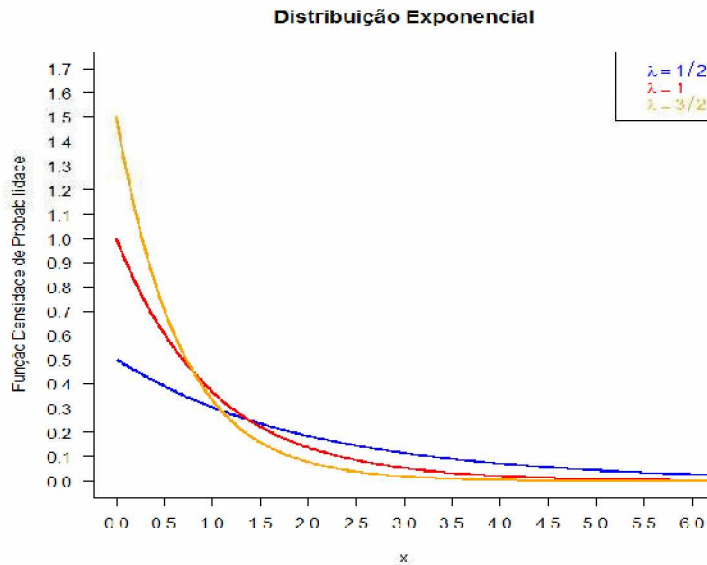


Figura 31 – Distribuição Exponencial com Parâmetros  $\lambda = 1/2, 1$  e  $3/2$ .

Como exemplo, suponha que o tempo de vida de uma determinada espécie de inseto tenha uma distribuição exponencial de parâmetro  $\lambda = 1/12$  dia. Suponha também que estes insetos atinjam a maturidade sexual após 3 dias de seu nascimento. Qual a função densidade de probabilidade, em dias, dos insetos que conseguem se reproduzir? E qual a probabilidade de que um inseto reprodutor viva mais de 24 dias?

Seja  $X$  a distribuição do tempo de vida dos insetos, e  $Y$  a distribuição do tempo de vida dos insetos que chegam a reprodução. Observem que  $Y = X + 3$ , assim

$$F_Y(y) = \mathbb{P}(Y \leq y) = \mathbb{P}(X + 3 \leq y) = \mathbb{P}(X \leq y - 3) = F_X(y - 3).$$

Portanto, a função densidade de probabilidade de  $Y$  é dada por:

$$f_Y(y) = \begin{cases} \frac{1}{12} e^{-\frac{(y-3)}{12}}, & \text{se } y \in (3, \infty) \\ 0, & \text{caso contrário.} \end{cases},$$

Para encontrar a probabilidade de que o inseto reprodutor dure mais de 24 dias, usa-se a densidade e tem-se que:

$$\mathbb{P}(Y > 24) = 1 - \mathbb{P}(Y \leq 24) = 1 - F_Y(24) = 1 - \int_{-\infty}^{24} f_Y(y) dy = 1 - \int_3^{24} \frac{1}{12} e^{-\frac{(y-3)}{12}} \approx 0,1738.$$

### Função Geradora de Momentos, Valor Esperado e Variância

Seja  $X$  um variável aleatória com distribuição exponencial com parâmetro  $\lambda$ . Então sua função geradora de momentos é dada por:

$$M_X(t) = \mathbb{E}(e^{tX}) = \int_0^\infty \lambda e^{-\lambda x} dx = \int_0^\infty \lambda e^{(t-\lambda)x} dx = \left|_0^\infty \frac{\lambda}{t-\lambda} = \frac{\lambda}{\lambda-t}.\right.$$

Tem-se que o valor esperado e a variância de uma variável aleatória  $X$  com distribuição exponencial com parâmetro  $\lambda$  são dados, respectivamente, por:

$$\mathbb{E}(X) = \int_{-\infty}^\infty x f(x) dx = \int_0^\infty x \lambda e^{-\lambda x} dx = \lambda \int_0^\infty x e^{-\frac{1}{\lambda}x} dx$$

e, resolvendo esta integral conclui-se que:

$$\mathbb{E}(X) = \frac{1}{\lambda}$$

Portanto, o valor esperado de  $X$  é  $\frac{1}{\lambda}$ . Para encontrar a variância de  $X$ , deve-se primeiramente calcular o valor esperado de  $X^2$ .

$$\mathbb{E}(X^2) = \int_{-\infty}^\infty g(x) f(x) dx = \int_0^\infty x^2 \lambda e^{-\lambda x} dx$$

e, resolvendo a integral, obtém-se que:

$$\mathbb{E}(X^2) = \frac{2}{\lambda^2}.$$

Portanto a variância de  $X$  é dada por:

$$\text{Var}(X) = \mathbb{E}(X^2) - \mathbb{E}^2(X) = \frac{2}{\lambda^2} - \frac{1}{\lambda^2} = \frac{1}{\lambda^2}.$$

Assim, o valor esperado e a variância de  $X$  são dados, respectivamente por:

$$\mathbb{E}(X) = \frac{1}{\lambda} \quad \text{Var}(X) = \frac{1}{\lambda^2}.$$

Pode-se calcular também o valor esperado e a variância utilizando a função geradora de momentos:

$$M'_X(t) = \frac{\lambda}{(\lambda - t)^2}$$

e

$$M''_X(t) = \frac{2\lambda}{(\lambda - t)^3}.$$

Portanto, o valor esperado e a variância podem ser calculados por:

$$\mathbb{E}(X) = M'_X(0) = \frac{1}{\lambda}$$

e

$$\text{Var}(X) = \mathbb{E}(X^2) - \mathbb{E}^2(X) = M''_X(0) - (M'_X(0))^2 = \frac{2}{\lambda^2} - \frac{1}{\lambda^2} = \frac{1}{\lambda^2}.$$

## 2.2 Trabalhos Relacionados

*Workflow nets* têm sido usadas como um sólido modelo para processos de *workflow*, por exemplo em (WANG; LI, 2013; MARTOS-SALGADO; ROSA-VELARDO, 2011; HEE; SIDOROVA; VOORHOEVE, 2006; KOTB; BADREDDIN, 2005; AALST, 1997). A maioria dos modelos existentes não contempla importantes aspectos do sistema de gerenciamento de *workflow*, como por exemplo a alocação de recursos e a gestão do tempo.

Em (HEE; SIDOROVA; VOORHOEVE, 2006; AALST, 1997) o mecanismo de alocação de recursos é apresentado através de uma representação informal. Em (WANG; LI, 2013; KOTB; BADREDDIN, 2005) o mecanismo de alocação de recursos é representado por simples fichas como é geralmente o caso de sistemas de produção. Mas uma simples ficha não pode representar de forma realística empregados que podem tratar simultaneamente diversos casos em um único dia, como é, geralmente, o caso nas administrações de empresas.

Sistemas modernos de gerenciamento de *workflow* têm que suportar tarefas com restrições complexas de dependência e lidar com o problema de alocação de recursos. Muito se tem pesquisado sobre alocação de recursos modelados por redes de Petri, por exemplo em (LIU, 2015; PLA et al., 2014; HUANG et al., 2011; OLIVEIRA; JULIA; PASSOS, 2011; MIKA; WALIGORA; WEGLARZ, 2011; JESKE; JULIA; VALETTE, 2009; JULIA; OLIVEIRA; VALETTE, 2008).

Em (JESKE; JULIA; VALETTE, 2006; JESKE; JULIA; VALETTE, 2009) um mecanismo de alocação de recursos *fuzzy* foi proposto a fim de tratar os recursos do tipo humano de forma mais próxima à realidade. No entanto, o jogador do modelo (mecanismo de inferência especializado para execução dinâmica do modelo) não foi implementado e a proposta não foi validada experimentalmente.

O tempo em processos de negócio é de suma importância, já que no caso de prazos não cumpridos, as empresas podem sofrer sanções financeiras. Um estudo sobre a gestão do tempo em *workflow* concentra-se principalmente no planejamento de tempo de execução do processo como um todo, em estimativa de duração da atividade, evitando violações de restrição de tempo em atividades ou processos como pode ser visto em (DENG et al., 2014; EDER; PANAGOS; RABINOVICH, 2013a; EDER; PANAGOS; RABINOVICH, 2013b; LARCO; WIERS; FRANSOO, 2013; BERTOLINI; LIU; SRBA, 2013). No entanto, no mundo real, devido às características dinâmicas de utilização dos recursos e das atividades de processos de negócios, a maioria das informações relativas ao tempo é incerta e não pode ser descrita com precisão.

Um modelo que surgiu, baseado na ideia inicial da *Workflow net* foi a *Time Workflow Net*, criada por Ling e Schmidt (2000). Seu objetivo é permitir a associação de tempos a atividades, possibilitando análises de restrições de tempo do processo. Os tempos na *Time Workflow Net* são expressos na forma de intervalos no formato  $[min, max]$ , que são associados às transições. Tal modelo permite a representação de recursos e a exis-



tência de instâncias múltiplas na representação do processo, mas de maneira *ad-hoc*, não apresentando uma descrição formal para tal.

Quanto à função de escalonamento em tempo real dos SGWs, o trabalho apresentado por (AALST; HEE, 2004) aponta a habilidade limitada dos SGWs no que se refere à alocação dos recursos e à decisão com relação à ordem em que as tarefas que usam os mesmos recursos devem ser executadas. Segundo Baggio, Wainer e Ellis (2004) não há resultados na literatura sobre escalonamento que cubram uma porção significativa das características dos SGWs.

A fim de solucionar o problema do escalonamento em SGW, em (JULIA; OLIVEIRA; VALETTE, 2008) foi proposto um modelo baseado em redes de Petri que permite representar tanto recursos discretos quanto recursos contínuos. Também foi utilizada uma técnica de propagação de restrições que torna possível calcular os tempos que um caso pode esperar entre uma atividade e outra para que o seu tempo de execução esteja dentro do intervalo permitido para o processo em consideração. A proposta se mostrou eficiente e o resultado da simulação foi dado por um cenário admissível correspondente a uma sequência específica de atividades que respeita as restrições temporais. A limitação desta técnica é que podem ocorrer situações em que a borda máxima de um intervalo de visibilidade seja alcançada, não havendo mais garantia de que todos os prazos sejam respeitados. Além disso, a representação de recursos através de recursos contínuos não mostra da melhor maneira o real comportamento humano ao realizar tarefas.

Em (MEDEIROS, 2009) foi apresentada uma proposta para a solução do problema de escalonamento tempo real dos SGWs. Uma técnica de Raciocínio Energético foi aplicada sobre os recursos e atividades dos processos de *workflow* a fim de buscar uma atualização dos intervalos de datas de início das atividades cujo objetivo é de melhorar as restrições de tempo que um algoritmo de escalonamento deverá explorar para produzir uma solução aceitável (que satisfaça o conjunto de restrições). Malaquias et al. (2013) propõe um SGW que se baseia em redes de Petri para a modelagem formal de processos e em Algoritmos Genéticos (AG) como técnica de otimização para o escalonamento de tarefas em tempo real. A limitação destes modelos continua sendo o fato de que os recursos humanos são representados como contínuos, não representando o aspecto nebuloso do comportamento humano.

Existe um interesse crescente no uso de sistemas que manipulam conjuntos nebulosos (*fuzzy*) para lidar com imprecisão, incerteza e indefinição. Desde o início dos anos 2000 diversas pesquisas tem sido desenvolvidas, por exemplo em (BUCKLEY, 2005; ZIMMERMANN, 2001) e outros. Nos últimos anos, sistemas *fuzzy* tornaram-se uma área extensa de pesquisa, principalmente devido ao fato de que os modelos determinísticos têm grandes limitações quando se trata de representação do comportamento humano. Algumas pesquisas relacionadas ao uso de conjuntos *fuzzy* podem ser encontrados em (ANAND; RAMAN; SUBRAMANIAN, 2014; LIU et al., 2013; BHARATHI; PRAMOD; RAMAN,

2013; SU et al., 2012; ZHOU et al., 2012; TÜYSÜZ; KAHRAMAN, 2010).

Em (ZHOU; MURATA, 1999; CARDOSO; VALETTE; DUBOIS, 1999; MURATA, 1996; CARDOSO; VALETTE; DUBOIS, 1989; VALETTE; CARDOSO; DUBOIS, 1989), os autores utilizaram as redes de Petri juntamente com a lógica possibilística e conjuntos nebulosos para incluir uma noção de incerteza nos processos da manufatura permitindo modelos mais compactos.

Pode-se perceber que os trabalhos publicados tratam formalmente somente dos mecanismos de alocação de recursos e do gerenciamento do tempo no processo (problema de planejamento mas não de escalonamento). É importante salientar que a correta descrição de mecanismos de alocação de recursos e um bom gerenciamento de tempo é imprescindível para garantir que os objetivos das empresas sejam cumpridos.

Dentre as várias definições para a simulação em computação, destaca-se a afirmação de (PEGDEN; SADOWSKI; SHANNON, 1995): a simulação é um processo de projetar um modelo computacional de um sistema real e conduzir experimentos com este modelo com o propósito de entender seu comportamento e/ou avaliar estratégias para sua operação. Uma característica importante para a simulação é a escolha da ferramenta a ser utilizada. Dentre as diversas existentes no mercado, com suas vantagens e desvantagens, pode-se citar a ferramenta CPN Tools que apresenta uma característica peculiar de permitir que estruturas complexas sejam programadas. Muitos trabalhos tem sido desenvolvidos utilizando a ferramenta CPN Tools, como por exemplo: (XU; ZENG; LU, 2013; WESTERGAARD; SLAATS, 2013; VASCONCELOS; DIAS, 2016; GEHLOT; KULKARNI; BRZOZOWSKI, 2016) entre outros.

---

## Modelos de Alocação de Recursos em *Workflow net*

Este capítulo apresenta modelos de alocação de recursos que podem ser usados em SGWs. Inicialmente, em (JESKE, 2006), foram apresentadas as definições estáticas dos principais mecanismos de alocação de recursos, assim como novas regras de disparo de transição no caso de mecanismo de alocação não convencionais (recursos *fuzzy*). A proposta deste capítulo é de implementar mecanismos de inferências especializados chamados de jogadores de redes de Petri que serão aplicados a diversos mecanismos de alocação, a fim de produzir sequências de disparo de transições que poderão ser vistas como um tipo de solução do problema de escalonamento em tempo de execução das atividades de processos de negócios quando são consideradas nos diversos roteiros dos processos de *workflow* restrições de recursos compartilhados entre as diversas atividades dos processos modelados. Neste capítulo, um intervalo de tempo será atribuído para a execução de cada atividade; no entanto, somente a alocação de recursos será monitorada, isto é, não haverá gerenciamento explícito do tempo (controle de prazos). Sendo assim, a execução das atividades dos casos de cada processo acontecerá independentemente de existir ou não violação nos prazos de entrega previstos.

Para ilustrar o método de modelagem, simulação e avaliação de desempenho dos diversos mecanismos de alocação de recursos, será usado um exemplo de *Workflow net* para representar o “Processo de Tratamento de Reclamações” apresentado na seção 2.1.7.

É importante lembrar que tal processo é ativado pela presença de uma ficha (representando o cliente) no lugar “Start” que, juntamente com a ficha presente em R1 (recurso disponível), habilita a transição A1 que corresponde à atividade para registrar a reclamação. Depois, o cliente que fez a reclamação (transição A2) e o departamento afetado pela reclamação (transição A3) devem ser contactados. Nesta etapa do processo, é o recurso R2 que fica responsável pela realização destas duas tarefas. Em particular, dependendo da disponibilidade do recurso R2, tais atividades poderão ser realizadas em qualquer ordem ou até em paralelo. Quando estas duas atividades são finalizadas, a transição que

representa a atividade A4 é habilitada e em seguida uma decisão é tomada (transição A5), caso o recurso R4 esteja disponível. Dependendo da decisão tomada, ou um pagamento é feito (transição A6 - recurso A5) ou uma carta é enviada ao cliente (transição A7 - recurso R2). Ao final do processo, a reclamação é arquivada (transição A8). A Figura 32 mostra a *Workflow net* que representa este processo com os recursos humanos definidos de acordo com a noção de acionamento apresentada em Aalst e Hee (2004).

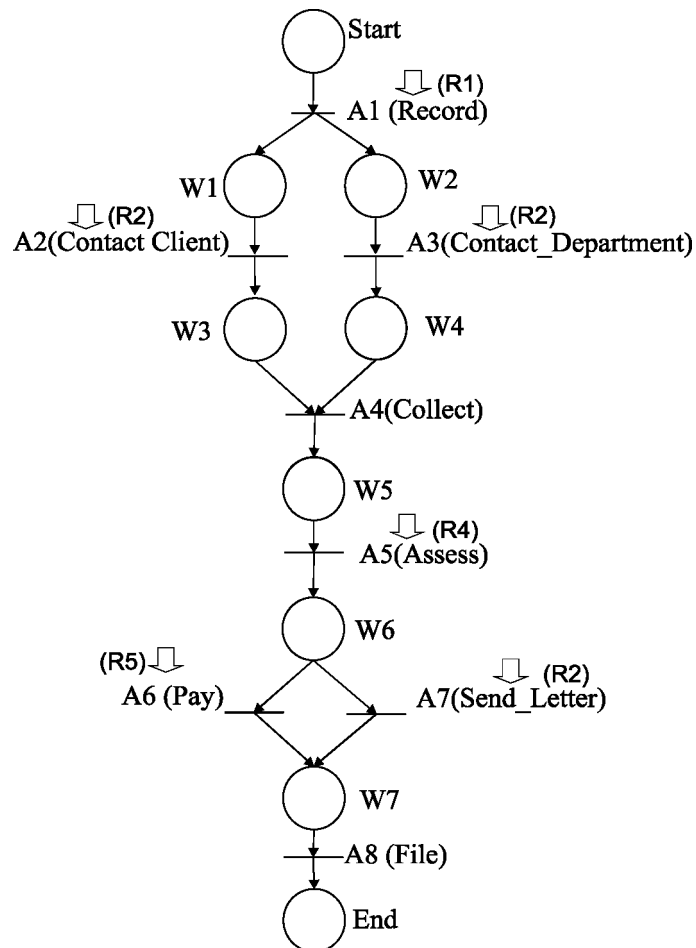


Figura 32 – “*WorkFlow net* para o Processo de Tratamento de Reclamações”.

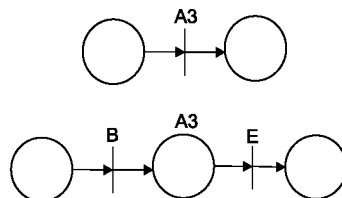


Figura 33 – Representação de uma Atividade.

Como mencionado previamente, uma atividade está associada a uma transição em uma *Workflow net*. No entanto, a fim de mostrar explicitamente a utilização dos recursos alocados para realizar uma atividade, uma transição atividade é trocada por uma sequência transição-lugar-transição (WANG; TEPFENHART; ROSCA, 2009). Como pode ser

observado na Figura 33, as transições  $B$  e  $E$  representam, respectivamente, o início e o fim de uma atividade em execução. O lugar  $A_3$  representa a atividade em execução. Do ponto de vista da análise de alcançabilidade (verificação de boas propriedades), a Figura 33 pode ser reduzida a uma única transição, a qual representa toda a atividade em execução como uma única unidade lógica.

A Figura 34 apresenta a *Workflow net* definida na Figura 32 baseada no conceito de representação de atividades apresentado em Wang, Tepfenhart e Rosca (2009).

## 3.1 Mecanismos de Alocação de Recursos

Existem diferentes tipos de recursos usados para realizar atividades em SGWs. Esses recursos podem ser agrupados de acordo com as unidades organizacionais a que pertencem. Os recursos também podem ser agrupados de acordo com papéis. Para recursos do tipo humano, um papel pode se referir a habilidades ou responsabilidades, por exemplo. Para equipamentos, um papel pode se referir à capacidade computacional. Às vezes, o conceito de *pool* também é utilizado. Um *pool* é um conjunto de recursos que podem desempenhar um mesmo papel.

Alguns dos recursos usados em um SGW podem ser considerados do tipo discreto, podendo ser representados por simples fichas. Por exemplo, uma impressora usada para tratar uma classe específica de documentos será representada como um recurso disjuntivo e pode ser alocada para um único documento num mesmo momento. Mas existem alguns recursos que não podem ser representados por uma simples ficha. Isso é o caso da maioria dos recursos de tipo humano. De fato, não é incomum um funcionário que trabalha em um escritório tratar muitos casos simultaneamente. Por exemplo, em uma companhia de seguros, um funcionário pode normalmente tratar vários casos durante um dia de trabalho e não necessariamente de uma maneira puramente sequencial. Um outro exemplo seria o caso de uma enfermeira que pode ser alocada para cuidar de vários pacientes ao mesmo tempo durante o seu dia de trabalho. A seguir, os diversos mecanismos de alocação de recursos considerados neste trabalhos serão formalmente definidos.

### 3.1.1 Mecanismo de Alocação de Recurso Discreto

A fim de definir os mecanismos de alocação de recursos, os termos utilizados seguem as definições apresentadas em David e Alla (2010). A seguir é apresentada a definição formal do mecanismo de alocação de recurso discreto.

**Definição 13** *O mecanismo de alocação de recurso discreto pode ser definido formalmente por um modelo de rede de Petri ordinária marcada  $C_{DR} = \langle A_{DR}, T_{DR}, Pre_{DR}, Pos_{DR}, M_{DR} \rangle$  com :*

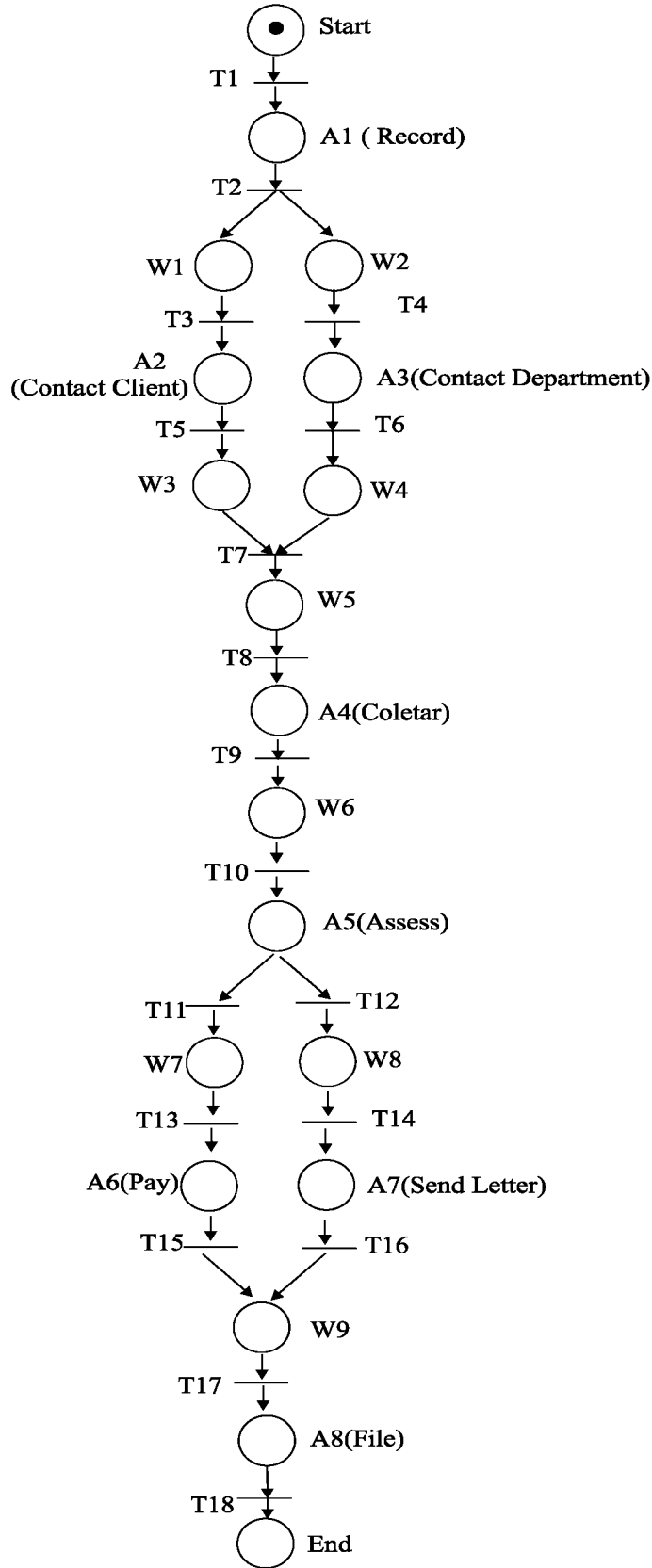


Figura 34 – *Workflow net* com Representação Explícita das Atividades.

- a)  $A_{DR} = \bigcup_{\alpha=1}^{N_{DR}} A_{(DR)_\alpha} \cup \{R_D\}$  onde  $R_D$  representa o lugar do recurso discreto,  $A_{(DR)_\alpha}$  o lugar da atividade  $\alpha$  e  $N_{DR}$  o número de atividades que estão conectadas com o lugar

do recurso discreto  $R_D$ .

b)  $T_{DR} = \bigcup_{\alpha=1}^{N_{DR}} T_{in_\alpha} \cup \bigcup_{\alpha=1}^{N_{DR}} T_{out_\alpha}$  onde  $T_{in_\alpha}$  representa a transição de entrada da atividade  $A_{(DR)_\alpha}$  e  $T_{out_\alpha}$  representa a transição de saída da atividade  $A_{(DR)_\alpha}$ .

c)  $Pre_{DR} : A_{DR} \times T_{DR} \rightarrow \{0, 1\}$  é a aplicação de incidência anterior tal que:

$$Pre_{DR}(R_D, T_{in_\alpha}) = 1 \text{ e } Pre_{DR}(A_{(DR)_\alpha}, T_{out_\alpha}) = 1$$

(outras combinações de lugar/transição são iguais a zero).

d)  $Pos_{DR} : A_{DR} \times T_{DR} \rightarrow \{0, 1\}$  é a aplicação da incidência posterior tal que:

$$Pos_{DR}(R_D, T_{out_\alpha}) = 1 \text{ e } Pos_{DR}(A_{(DR)_\alpha}, T_{in_\alpha}) = 1$$

(outras combinações de lugar/transição são iguais a zero).

e)  $M_{DR} : A_{DR} \rightarrow N$  é a marcação inicial tal que:

$$M_{DR}(R_D) = m_D$$

e

$$M_{DR}(A_{(DR)_\alpha}) = 0 \text{ para } \alpha = 1 \text{ até } N_{DR}$$

onde  $M_{DR}$  representa o número de recursos discretos do mesmo tipo.

Por se tratar de recursos renováveis, as alocações de recursos do tipo discreto deverão verificar a propriedade de invariante de lugar. Pode-se verificar esta propriedade pela seguinte equação:

$$M_{DR}(R_D) + (M_{DR}(A1)) + (M_{DR}(A2)) + \dots + (M_{DR}(A_{n_{DR}})) = m_D \quad (19)$$

Supondo-se que no “Processo de Tratamento de Reclamações” apresentado anteriormente, um único funcionário trata as atividades “Contactar-Cliente”, “Contactar-Departamento” e “Enviar Carta”; então a modelagem do mecanismo de alocação de recurso discreto é dado pela Figura 35.

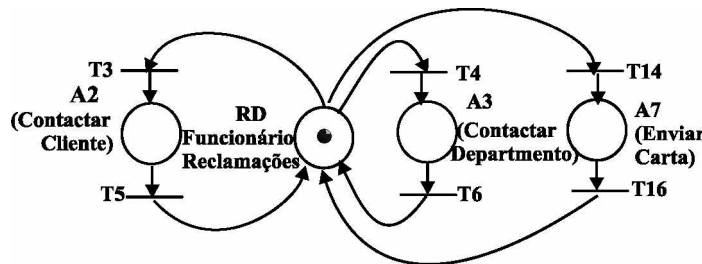


Figura 35 – Alocação de Recurso Discreto.

Nesta Figura, é claro que se a ficha em  $R_D$  é usada para realizar a atividade  $A_2$  nenhuma outra atividade poderá ser realizada. Isto significa que o recurso  $R_D$  pode ser usado somente de modo disjuntivo. Em particular, se a atividade  $A_2$  é iniciada e o funcionário não pode entrar em contato imediatamente com o cliente, ele não pode usar seu tempo disponível (o tempo de espera da resposta do cliente) para iniciar outra atividade, como enviar uma carta por exemplo (atividade  $A_7$ ). Como já foi dito, é evidente que na prática, esta situação não ocorre. Se o cliente não está disponível em um dado instante, o funcionário usará sua disponibilidade para executar outra tarefa.

### 3.1.2 Mecanismos de Alocação de Recurso Contínuo

A fim de definir os mecanismos de alocação de recursos, os termos utilizados seguem as definições apresentadas no livro de (DAVID; ALLA, 2010). A seguir é apresentada a definição formal do mecanismo de alocação de recurso contínuo.

**Definição 14** *O mecanismo de alocação de recurso contínuo pode ser definido formalmente por um modelo de rede de Petri híbrida marcada com transições discretas  $C_{CR} = \langle A_{CR}, T_{CR}, Pre_{CR}, Pos_{CR}, M_{CR} \rangle$  com:*

- a)  $A_{CR} = \bigcup_{\alpha=1}^{N_{CR}} A_{(CR)_\alpha} \cup \{R_C\}$  onde  $R_C$  representa o lugar de recurso contínuo,  $A_{(CR)_\alpha}$  o lugar da atividade  $\alpha$  e  $N_{CR}$  o número de atividades que estão conectados com o lugar do recurso contínuo  $R_C$ .
- b)  $T_{CR} = \bigcup_{\alpha=1}^{N_{CR}} T_{in_\alpha} \cup \bigcup_{\alpha=1}^{N_{CR}} T_{out_\alpha}$  onde  $T_{in_\alpha}$  representa a transição de entrada discreta da atividade  $A_{(CR)_\alpha}$  e  $T_{out_\alpha}$  representa a transição de saída discreta da atividade  $A_{(CR)_\alpha}$ .
- c)  $Pre_{CR} : A_{CR} \times T_{CR} \rightarrow R^+$  é a aplicação da incidência anterior tal que:

$$Pre_{CR}(R_C, T_{in_\alpha}) = X_\alpha \text{ com } X_\alpha \in R^+ \text{ e } Pre_{CR}(A_{(CR)_\alpha}, T_{out_\alpha}) = 1$$

(outras combinações de lugar/transição são iguais a zero).

- d)  $Pos_{CR} : A_{CR} \times T_{CR} \rightarrow R^+$  é a aplicação da incidência posterior tal que:

$$Pos_{CR}(R_C, T_{out_\alpha}) = X_\alpha \text{ e } Pos_{CR}(A_{(CR)_\alpha}, T_{in_\alpha}) = 1$$

(outras combinações de lugar/transição são iguais a zero).

- e)  $M_{CR} : A_{(CR)_\alpha} \rightarrow R^+$  é a marcação inicial tal que

$$M_{CR}(R_C) = m_C$$

e

$$M_{CR}(A_{(CR)_\alpha}) = 0 \text{ para } \alpha = 1 \text{ até } N_{CR}$$



onde  $M_{CR}$  representa a disponibilidade (em porcentagem) do recurso contínuo.

Por se tratar de recursos renováveis, as alocações de recursos do tipo contínuo deverão verificar a propriedade de invariante de lugar que se escreverá na seguinte forma:

$$M_{CR}(R_C) + (X_1 \times M_{CR}(A_1)) + (X_2 \times M_{CR}(A_2)) + \dots + (X_{N_{CR}} \times M_{CR}(A_{N_{CR}})) = m_C \quad (20)$$

Supondo-se que no “Processo de Tratamento de Reclamações” apresentado anteriormente, um único funcionário trata as atividades “Contactar-Cliente”, “Contactar-Departamento” e “Enviar Carta”; então o mecanismo de alocação de recurso contínuo é dado pela Figura 36.

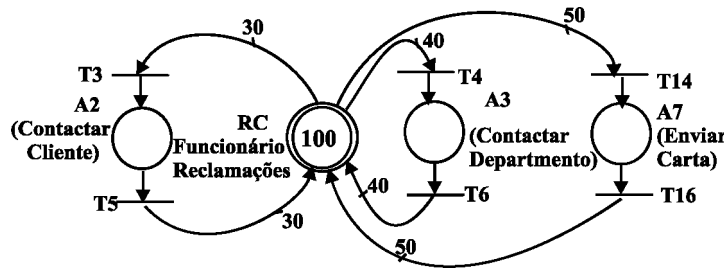


Figura 36 – Alocação de Recurso Contínuo.

Nesta Figura, fica especificado que somente 30% da disponibilidade do funcionário é necessária para realizar a atividade “Contactar-Cliente”, 40% para realizar a atividade “Contactar-Departamento” e 50% para realizar a atividade “Enviar-Carta”. Desta forma, então, é possível que o funcionário trate simultaneamente mais de uma atividade. Por exemplo, depois do início da atividade “Contactar-Cliente”, se o cliente não estiver disponível para responder as questões do funcionário, este último poderá usar sua disponibilidade de tempo (esperando a resposta do cliente) para iniciar outra atividade, como, por exemplo, enviar uma carta (atividade  $A_7$ ). De fato, 50% da disponibilidade do funcionário é necessária para realizar a atividade “Enviar Carta”, e, depois de iniciada a atividade “Contactar Cliente”, o funcionário tem ainda 70% de disponibilidade.

Como o “Funcionário-Reclamações” ( $R_C$ ) é um recurso renovável, a propriedade do invariante de lugar deverá ser respeitada. Usando a equação 20, o invariante de lugar terá, neste caso, a forma:

$$M_{CR}(R_C) + (X_2 \times M_{CR}(A_2)) + (X_3 \times M_{CR}(A_3)) + (X_7 \times M_{CR}(A_7)) = 100$$

A limitação deste modelo está relacionada ao fato que a representação do comportamento humano em termos de disponibilidade, na prática, é conhecida somente como um valor incerto. Não é possível prever com certeza o quanto cada funcionário gasta em termos de disponibilidade para realizar uma atividade, além do fato de que um mesmo funcionário pode não ter a mesma eficiência durante seu dia de trabalho. Por exemplo,

se no início do dia somente 30% da disponibilidade de um funcionário é requerida para realizar a atividade  $A_2$ , é provável que mais de 30% seja requerido ao final do dia por causa do cansaço. Sendo assim, torna-se necessário uma representação mais próxima da realidade. A representação da disponibilidade em termos de valores nebulosos (fuzzy) corresponde mais ao que acontece na prática.

### 3.1.3 Mecanismo de Alocação de Recurso Contínuo *Fuzzy*

**Definição 15** *O mecanismo de alocação de recurso contínuo fuzzy pode ser definido formalmente por um modelo de rede de Petri híbrida fuzzy marcada com transições discretas (JESKE, 2006)  $C_{FCR} = \langle A_{FCR}, T_{FCR}, Pre_{FCR}, Pos_{FCR}, M_{FCR} \rangle$  com:*

- a)  $A_{FCR} = \bigcup_{\alpha=1}^{N_{FCR}} A_{FCR_\alpha} \cup \{R_{FC}\}$  onde  $R_{FC}$  representa o lugar de recurso contínuo fuzzy,  $A_{(FCR)_\alpha}$  o lugar da atividade  $\alpha$  e  $N_{FCR}$  o número de atividades que estão conectadas com o lugar do recurso contínuo fuzzy  $R_{FC}$ .
- b)  $T_{FCR} = \bigcup_{\alpha=1}^{N_{FCR}} T_{in_\alpha} \cup \bigcup_{\alpha=1}^{N_{FCR}} T_{out_\alpha}$  onde  $T_{in_\alpha}$  representa a transição de entrada discreta da atividade  $A_{(FCR)_\alpha}$  e  $T_{out_\alpha}$  representa a transição de saída discreta da atividade  $A_{(FCR)_\alpha}$ .
- c)  $Pre_{FCR} : A_{FCR} \times T_{FCR} \rightarrow F$  é a aplicação da incidência anterior tal que:

$$\begin{aligned} Pre_{FCR}(R_{FC}, T_{in_\alpha}) &= [w1, w2, w3, w4] \text{ com } w2 = w3 \text{ e} \\ Pre_{FCR}(A_{(FCR)_\alpha}, T_{out_\alpha}) &= [1, 1, 1, 1] \\ \text{onde } (w1, w2, w3) &\in \mathbb{R} \text{ e } w4 \in \mathbb{R}^+ \end{aligned}$$

(outras combinações de lugar/transição são iguais a zero) com  $F$  sendo o conjunto dos números da forma triangular.

- d)  $Pos_{FCR} : A_{FCR} \times T_{FCR} \rightarrow F$  é a aplicação da incidência posterior tal que:

$$\begin{aligned} Pos_{FCR}(R_{FC}, T_{out_\alpha}) &= [w1, w2, w3, w4] \text{ com } w2 = w3 \text{ e} \\ Pos_{FCR}(A_{(FCR)_\alpha}, T_{in_\alpha}) &= [1, 1, 1, 1] \\ \text{onde } (w1, w2, w3) &\in \mathbb{R} \text{ e } w4 \in \mathbb{R}^+ \end{aligned}$$

(outras combinações de lugar/transição são iguais a zero)

- e)  $M_{FCR} : A_{FCR} \rightarrow F$  é a marcação inicial tal que

$$\begin{aligned} M_{FCR}(R_{FC}) &= [m1, m2, m3, m4] \text{ com } (m1, m2, m3) \in \mathbb{R} \text{ e } m4 \in \mathbb{R}^+ \\ &e \\ M_{FCR}(A_{(FCR)_\alpha}) &= 0 \end{aligned}$$

onde  $M_{FCR}$  representa a disponibilidade fuzzy (em porcentagem) do recurso contínuo fuzzy.

Supondo-se que no “Processo de Tratamento de Reclamações” apresentado anteriormente, um único funcionário trata as atividades “Contactar-Cliente”, “Contactar-Departamento” e “Enviar Carta”; então o mecanismo de alocação de recurso contínuo *fuzzy* é dado pela Figura 37.

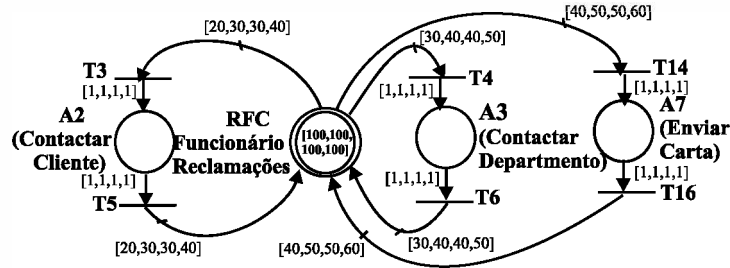


Figura 37 – Alocação de Recurso Contínuo *Fuzzy*.

Esta Figura mostra que  $30\% \pm 10\%$  da disponibilidade do recurso  $R_{FC}$  é necessário para realizar a atividade A2 (Contactar-Cliente),  $40\% \pm 10\%$  para realizar a atividade A3 (Contactar-Departamento) e  $50\% \pm 10\%$  para realizar a atividade A7 (Enviar-Carta).

O comportamento de um modelo de alocação de recurso contínuo *fuzzy* pode ser definido através dos conceitos de “transição habilitada” e “equação fundamental”.

Como dito anteriormente, em uma rede de Petri ordinária, uma transição  $t$  é habilitada se e somente se para todas as entradas dos lugares  $p$  da transição  $t$ ,  $M(p) \geq Pre(p, t)$ . Isto é, se o número de fichas de cada lugar de entrada é maior ou igual ao peso associado aos arcos que conectam os lugares de entrada da transição  $t$ . No caso de uma alocação híbrida *fuzzy*, a definição da sensibilização de uma transição é a seguinte:

**Definição 16** Em uma alocação de recurso contínuo *fuzzy*, uma transição  $t$  está habilitada se e somente se:

$$\Pi_t = \Pi(Pre_{FCR}(p, t) \leq M_{FCR}(p)) > 0 \quad (21)$$

Por exemplo, a transição  $t3$  na Figura 37 está habilitada porque:

$$\Pi_{t3} = \Pi(Pre_{FCR}(R_{FC}, t3) \leq M_{FCR}(R_{FC})) = 1 > 0$$

como mostra a Figura 38 com  $A = Pre_{FCR}(R_{FC}, t3)$  e  $B = M_{FCR}(R_{FC})$ .

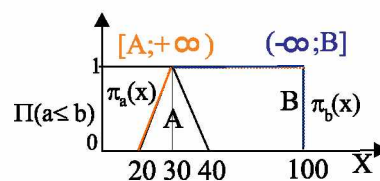


Figura 38 – Cálculo de Possibilidade Associado a  $t3$ .

Para uma rede de Petri ordinária, a evolução da marcação de um lugar  $p$  após o disparo de uma transição  $t$  é obtida de acordo com a equação fundamental:

$$M'(p) = M(p) - Pre(p, t) + Pos(p, t) \quad (22)$$

Com o modelo de alocação de recurso contínuo *fuzzy*, a evolução da marcação é definida através da equação fundamental:

$$M'_{FCR}(p) = M_{FCR}(p) \ominus Pre_{FCR}(p, t) \boxplus Pos_{FCR}(p, t) \quad (23)$$

A operação “ $\ominus$ ” corresponde à subtração *fuzzy*. A operação “ $\boxplus$ ”, quando considerada como soma de dois conjuntos *fuzzy*, é diferente da soma tradicional “ $\oplus$ ” dada nos conjuntos *fuzzy* e é definida como:

$$[a1, a2, a3, a4] \boxplus [b1, b2, b3, b4] = [a1 + b4, a2 + b3, a3 + b2, a4 + b1] \quad (24)$$

O uso da operação  $\boxplus$  vem do fato de que a operação *fuzzy* “ $\oplus$ ” não mantém o invariante de lugar associado à alocação de recursos. De fato, já que se trata de um recurso renovável, depois da realização de diferentes atividades, a disponibilidade do recurso deve voltar a 100%, mesmo no caso *fuzzy*, o que não seria o caso com o uso da operação “ $\oplus$ ”. Do ponto de vista do mecanismo de alocação de recursos híbridos *fuzzy*, a operação  $\boxplus$  pode ser vista como um tipo de operação de *defuzzificação*. Em particular, usando esta operação será possível encontrar a expressão linear da marcação *fuzzy* que sempre será constante e que corresponde à seguinte expressão:

$$M_{FCR}(R_{FC}) \boxplus (w_1 \otimes M_{FCR}(A1)) \boxplus (w_2 \otimes M_{FCR}(A2)) \boxplus \dots \boxplus (w_{N_{FCR}} \otimes M_{FCR}(A_{N_{FCR}})) = [m1, m2, m3, m4]$$

com  $w_\alpha = Pre_{FCR}(R_{FC}, t_{in_\alpha}) = Pos_{FCR}(R_{FC}, t_{out_\alpha})$  para  $\alpha = 1$  até  $N_{FCR}$ .

Para ilustrar os conceitos *fuzzy* de “transição habilitada”, “disparo de transição” e “invariante de lugar”, a seqüência de disparo  $t_3 \rightarrow t_4 \rightarrow t_{14} \rightarrow t_5 \rightarrow t_6 \rightarrow t_{16}$  será considerada usando o mecanismo de alocação de recurso contínuo *fuzzy* apresentado na Figura 37.

a) Disparo de  $t_3$ :

o cálculo da possibilidade associado a  $t_3$  é :

$$\Pi_{t3} = \Pi(Pre_{FCR}(R_{FC}, t3) \leq M_{FCR}(R_{FC})) = 1 > 0$$

como mostra a Figura 38 com  $A = Pre_{FCR}(R_{FC}, t3)$  e  $B = M_{FCR}(R_{FC})$ .

Depois do disparo de  $t_3$ , as novas marcações de  $R_{FC}$  e A2 são (Figura 39):

$$\begin{aligned} M'_{FCR}(R_{FC}) &= M_{FCR}(R_{FC}) \ominus Pre_{FCR}(R_{FC}, t3) = \\ &[100, 100, 100, 100] \ominus [20, 30, 30, 40] = [60, 70, 70, 80] \end{aligned}$$

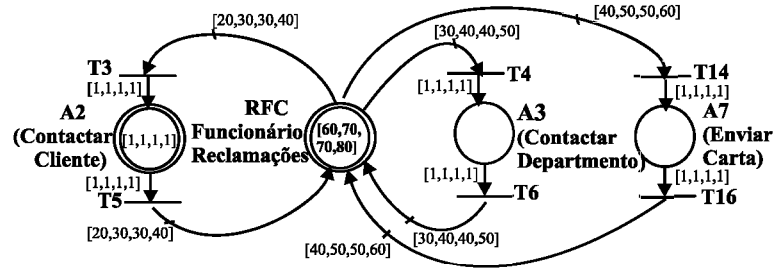


Figura 39 – Marcação após Disparo de t3

$$M'_{FCR}(A2) = M_{FCR}(A2) \boxplus Pos_{FCR}(A2, t3) = [0, 0, 0, 0] \boxplus [1, 1, 1, 1] = [1, 1, 1, 1]$$

O invariante de lugar associado ao modelo de recurso contínuo *fuzzy* é:

$$\begin{aligned} &M_{FCR}(R_{FC}) \boxplus (Pre_{FCR}(R_{FC}, t3) \otimes M_{FCR}(A2)) \boxplus (Pre_{FCR}(R_{FC}, t4) \otimes \\ &M_{FCR}(A3)) \boxplus (Pre_{FCR}(R_{FC}, t14) \otimes M_{FCR}(A7)) = \\ &[60, 70, 70, 80] \boxplus ([20, 30, 30, 40] \otimes [1, 1, 1, 1]) \boxplus ([30, 40, 40, 50] \otimes [0, 0, 0, 0]) \boxplus \\ &([40, 50, 50, 60] \otimes [0, 0, 0, 0]) = [60, 70, 70, 80] \boxplus [20, 30, 30, 40] = [100, 100, 100, 100] \end{aligned}$$

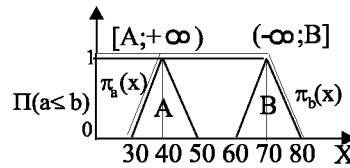


Figura 40 – Cálculo de Possibilidade Associado a t4

b) Disparo de  $t_4$ :

o cálculo de possibilidade associado a  $t_4$  é:

$$\Pi_{t4} = \Pi(Pre_{FCR}(R_{FC}, t4) \leq M_{FCR}(R_{FC})) = 1 > 0$$

como mostra a Figura 40 com  $a = Pre_{FCR}(R_{FC}, t4)$  e  $b = M_{FCR}(R_{FC})$ .

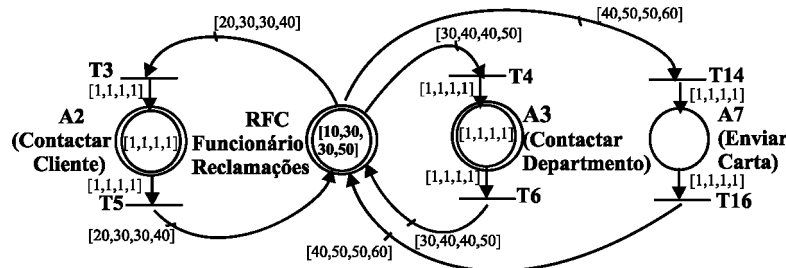


Figura 41 – Marcação após o Disparo de t4

Depois do disparo de  $t_4$ , as novas marcações de  $R_{FC}$  e A3 são (Figura 41):

$$\begin{aligned}
M'_{FCR}(R_{FC}) &= M_{FCR}(R_{FC}) \ominus Pre_{FCR}(R_{FC}, t4) = \\
&[60, 70, 70, 80] \ominus [30, 40, 40, 50] = [10, 30, 30, 50] \\
&\text{e} \\
M'_{FCR}(A3) &= M_{FCR}(A3) \boxplus Pos_{FCR}(A3, t4) = \\
&[0, 0, 0, 0] \boxplus [1, 1, 1, 1] = [1, 1, 1, 1]
\end{aligned}$$

O invariante de lugar associado ao modelo de recurso contínuo *fuzzy* é:

$$\begin{aligned}
&M_{FCR}(R_{FC}) \boxplus (Pre_{FCR}(R_{FC}, t3) \otimes M_{FCR}(A2)) \boxplus (Pre_{FCR}(R_{FC}, t4) \otimes \\
&M_{FCR}(A3)) \boxplus (Pre_{FCR}(R_{FC}, t14) \otimes M_{FCR}(A7)) = \\
&[10, 30, 30, 50] \boxplus ([20, 30, 30, 40] \otimes [1, 1, 1, 1]) \boxplus ([30, 40, 40, 50] \otimes [1, 1, 1, 1]) \boxplus \\
&([40, 50, 50, 60] \otimes [0, 0, 0, 0]) = \\
&[10, 30, 30, 50] \boxplus [20, 30, 30, 40] \boxplus [30, 40, 40, 50] = \\
&[50, 60, 60, 70] \boxplus [30, 40, 40, 50] = [100, 100, 100, 100]
\end{aligned}$$

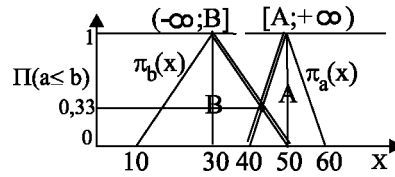


Figura 42 – Cálculo de Possibilidade Associado a  $t_{14}$

c) Disparo de  $t_{14}$ :

o cálculo de possibilidade associado a  $t_{14}$  é:

$$\Pi_{t4} = \Pi(Pre_{FCR}(R_{FC}, t14) \leq M_{FCR}(R_{FC})) = 0,33 > 0$$

como mostra a Figura 42 com  $A = Pre_{FCR}(R_{FC}, t14)$  e  $B = M_{FCR}(R_{FC})$ .

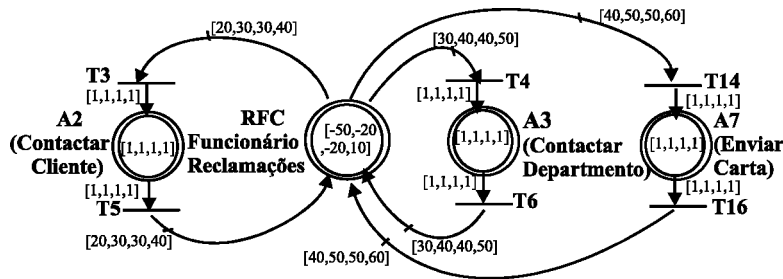


Figura 43 – Marcação após o Disparo de  $t_{14}$

Depois do disparo de  $t_{14}$ , as novas marcações de  $R_{FC}$  e A3 são (Figura 43):

$$\begin{aligned}
M'_{FCR}(R_{FC}) &= M_{FCR}(R_{FC}) \ominus Pre_{FCR}(R_{FC}, t14) = \\
&[10, 30, 30, 50] \ominus [40, 50, 50, 60] = [-50, -20, -20, 10]
\end{aligned}$$

e

$$M'_{FCR}(A7) = M_{FCR}(A7) \boxplus Pos_{FCR}(A7, t14) = \\ [0, 0, 0, 0] \boxplus [1, 1, 1, 1] = [1, 1, 1, 1]$$

A parte negativa da marcação *fuzzy* de  $R_{FC}$  que aparece depois do disparo de  $t14$  simplesmente mostra a possibilidade de sobrecarga do recurso (o funcionário trabalha além da sua capacidade normal). É importante saber que a parte negativa da marcação não é inconsistente com a teoria de rede de Petri. De fato, somente a parte positiva da marcação *fuzzy* pode ser usada para habilitar uma transição no modelo de recurso contínuo *fuzzy*.

O invariante de lugar associado ao modelo de recurso contínuo *fuzzy* é:

$$M_{FCR}(R_{FC}) \boxplus (Pre_{FCR}(R_{FC}, t3) \otimes M_{FCR}(A2)) \boxplus (Pre_{FCR}(R_{FC}, t4) \otimes \\ M_{FCR}(A3)) \boxplus (Pre_{FCR}(R_{FC}, t14) \otimes M_{FCR}(A7)) = \\ [-50, -20, -20, 10] \boxplus ([20, 30, 30, 40] \otimes [1, 1, 1, 1]) \boxplus ([30, 40, 40, 50] \otimes [1, 1, 1, 1]) \boxplus \\ ([40, 50, 50, 60] \otimes [1, 1, 1, 1]) = \\ [-50, -20, -20, 10] \boxplus [20, 30, 30, 40] \boxplus [30, 40, 40, 50] \boxplus [40, 50, 50, 60] = \\ [-10, 10, 10, 30] \boxplus [30, 40, 40, 50] \boxplus [40, 50, 50, 60] = \\ [40, 50, 50, 60] \boxplus [40, 50, 50, 60] = [100, 100, 100, 100]$$

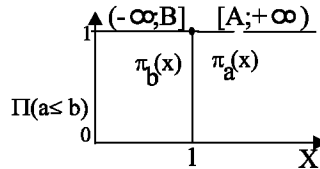


Figura 44 – Cálculo de Possibilidade Associado a  $t_5$

d) Disparo  $t_5$ :

o cálculo de possibilidade associado a  $t_5$  é :

$$\Pi_{t_5} = \Pi(Pre_{FCR}(A2, t5) \leq M_{FCR}(A2)) = 1 > 0$$

como mostra a Figura 44 com  $A = Pre_{FCR}(A2, t5)$  e  $B = M_{FCR}(A2)$ .

Depois do disparo de  $t_5$ , as novas marcações de  $R_{FC}$  e A2 são (Figura 45):

$$M'_{FCR}(R_{FC}) = M_{FCR}(R_{FC}) \boxplus Pos_{FCR}(R_{FC}, t5) = \\ [-50, -20, -20, 10] \boxplus [20, 30, 30, 40] = [-10, 10, 10, 30]$$

e

$$M'_{FCR}(A2) = M_{FCR}(A2) \ominus Pre_{FCR}(A2, t5) = \\ [1, 1, 1, 1] \ominus [1, 1, 1, 1] = [0, 0, 0, 0]$$

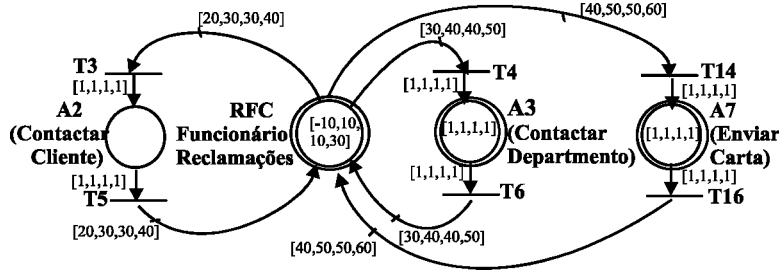


Figura 45 – Marcação após Disparo de t5

O invariante de lugar associado ao modelo de recurso contínuo *fuzzy* é:

$$\begin{aligned}
 &M_{FCR}(R_{FC}) \boxplus (Pre_{FCR}(R_{FC}, t3) \otimes M_{FCR}(A2)) \boxplus (Pre_{FCR}(R_{FC}, t4) \otimes \\
 &\quad M_{FCR}(A3)) \boxplus (Pre_{FCR}(R_{FC}, t14) \otimes M_{FCR}(A7)) = \\
 &[-10, 10, 10, 30] \boxplus ([20, 30, 30, 40] \otimes [0, 0, 0, 0]) \boxplus ([30, 40, 40, 50] \otimes [1, 1, 1, 1]) \boxplus \\
 &\quad ([40, 50, 50, 60] \otimes [1, 1, 1, 1]) = \\
 &[-10, 10, 10, 30] \boxplus [30, 40, 40, 50] \boxplus ([40, 50, 50, 60] = \\
 &[40, 50, 50, 60] \boxplus [40, 50, 50, 60] = [100, 100, 100, 100]
 \end{aligned}$$

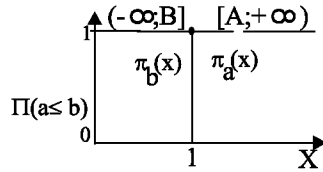


Figura 46 – Cálculo da Possibilidade Associado a t6

e) Disparo de  $t_6$ :

o cálculo da possibilidade associado a  $t_6$  é :

$$\Pi_{t6} = \Pi(Pre_{FCR}(A3, t6) \leq M_{FCR}(A3)) = 1 > 0$$

como mostra a Figura 46 com  $A = Pre_{FCR}(A3, t6)$  e  $B = M_{FCR}(A3)$ .

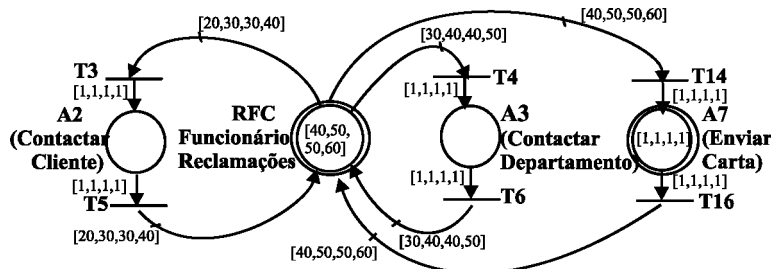


Figura 47 – Marcação após o Disparo de t6

Depois do disparo de  $t_6$ , as novas marcações de  $R_{FC}$  e A3 são (Figura 47:



$$M'_{FCR}(R_{FC}) = M_{FCR}(R_{FC}) \boxplus Pos_{FCR}(R_{FC}, t6) =$$

$$[-10, 10, 10, 30] \boxplus [30, 40, 40, 50] = [40, 50, 50, 60]$$

e

$$M'_{FCR}(A3) = M_{FCR}(A3) \ominus Pre_{FCR}(A3, t6) =$$

$$[1, 1, 1, 1] \ominus [1, 1, 1, 1] = [0, 0, 0, 0]$$

O invariante de lugar associado ao modelo de recurso contínuo *fuzzy* é:

$$M_{FCR}(R_{FC}) \boxplus (Pre_{FCR}(R_{FC}, t3) \otimes M_{FCR}(A2)) \boxplus (Pre_{FCR}(R_{FC}, t4) \otimes$$

$$M_{FCR}(A3)) \boxplus (Pre_{FCR}(R_{FC}, t14) \otimes M_{FCR}(A7)) =$$

$$[40, 50, 50, 60] \boxplus ([20, 30, 30, 40] \otimes [0, 0, 0, 0]) \boxplus ([30, 40, 40, 50] \otimes [0, 0, 0, 0]) \boxplus$$

$$([40, 50, 50, 60] \otimes [1, 1, 1, 1]) =$$

$$[40, 50, 50, 60] \boxplus ([40, 50, 50, 60]) = [100, 100, 100, 100]$$

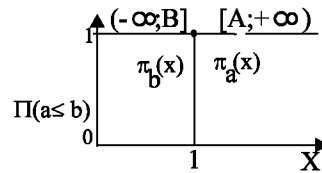


Figura 48 – Cálculo da Possibilidade Associado a  $t_{16}$

f) Disparo de  $t_{16}$ :

o cálculo da possibilidade associado a  $t_{16}$  é :

$$\Pi_{t_{16}} = \Pi(Pre_{FCR}(A7, t_{16}) \leq M_{FCR}(A7)) = 1 > 0$$

como mostra a Figura 48 com  $A = Pre_{FCR}(A7, t_{16})$  e  $B = M_{FCR}(A7)$ .

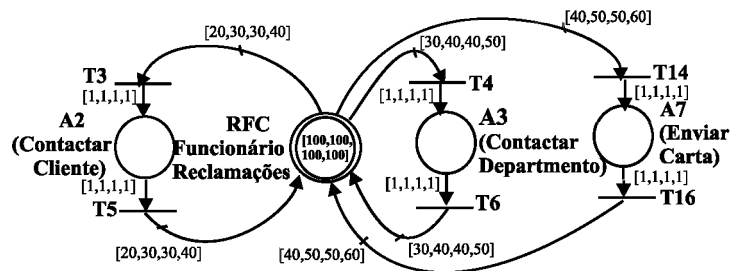


Figura 49 – Marcação após Disparo de  $t_{16}$

Depois do disparo de  $t_{16}$ , as novas marcações de  $R_{FC}$  e  $A7$  são (Figura 49):

$$M'_{FCR}(R_{FC}) = M_{FCR}(R_{FC}) \boxplus Pos_{FCR}(R_{FC}, t_{16}) =$$

$$[40, 50, 50, 60] \boxplus [40, 50, 50, 60] = [100, 100, 100, 100]$$

e

$$M'_{FCR}(A7) = M_{FCR}(A7) \ominus Pre_{FCR}(A7, t16) = \\ [1, 1, 1, 1] \ominus [1, 1, 1, 1] = [0, 0, 0, 0]$$

O invariante de lugar associado ao modelo de recurso contínuo *fuzzy* é:

$$M_{FCR}(R_{FC}) \boxplus (Pre_{FCR}(R_{FC}, t3) \otimes M_{FCR}(A2)) \boxplus (Pre_{FCR}(R_{FC}, t4) \otimes \\ M_{FCR}(A3)) \boxplus (Pre_{FCR}(R_{FC}, t14) \otimes M_{FCR}(A7)) = \\ [100, 100, 100, 100] \boxplus ([20, 30, 30, 40] \otimes [0, 0, 0, 0]) \boxplus ([30, 40, 40, 50] \otimes [0, 0, 0, 0]) \boxplus \\ ([40, 50, 50, 60] \otimes [0, 0, 0, 0]) = [100, 100, 100, 100]$$

Supondo-se que a operação “ $\oplus$ ” *fuzzy* fosse usada para calcular as novas marcações após os disparos. Usando o mesmo mecanismo de alocação de recurso contínuo *fuzzy* apresentado na Figura 37 após o disparo de  $t_3$ , as novas marcações de  $R_{FC}$  e  $A2$  seriam as mesmas apresentadas anteriormente. No entanto, após o disparo de  $t_5$ , a nova marcação de  $R_{FC}$  seria:

$$M'_{FCR}(R_{FC}) = M_{FCR}(R_{FC}) \oplus Pos_{FCR}(R_{FC}, t5) = \\ [60, 70, 70, 80] \oplus [20, 30, 30, 40] = [80, 100, 100, 120]$$

O recurso  $R_{FC}$  (funcionário-reclamações) é um recurso renovável, por isso, após a realização de alguma atividade sua capacidade deve voltar ao valor inicial, neste caso  $[100, 100, 100, 100]$ . Como se pode perceber, quando se usa a operação “ $\oplus$ ” para o cálculo de novas marcações não é possível manter esta propriedade. No exemplo apresentado anteriormente, o resultado obtido foi um intervalo  $[80, 100, 100, 120]$ , o que definitivamente não é o valor atribuído ao recurso  $R_{FC}$  antes da realização de qualquer atividade. Isso ocorre porque em operações *fuzzy* a soma e a subtração não são operações inversas. Neste caso, justifica-se a criação da operação “ $\boxplus$ ” que tem como único objetivo ser uma operação inversa à operação “ $\ominus$ ”, garantindo assim que um recurso renovável, após a sua utilização, volte à sua capacidade inicial.

## 3.2 Implementação da *Workflow net* com Recursos

O CPN Tools é uma ferramenta de força industrial para a construção e análise de modelos CPN (*Coloured Petri Nets*). Utilizando o CPN Tools é possível investigar o comportamento de um sistema modelado utilizando algoritmos para verificar propriedades e simulações de tipo Monte Carlo para conduzir a análise de desempenho. O CPN Tools é, então, um *software* que oferece as funcionalidades de edição, simulação, análise de espaço dos estados e análise de performance de um modelo CPN (JENSEN; KRISTENSEN, 2009).

Com o objetivo de mostrar a modelagem e implementação dos diversos mecanismos de alocação de recursos, será utilizada a ferramenta computacional CPN Tools para o exemplo “Processo de Tratamento de Reclamações” definido em (AALST; HEE, 2004), apresentado na seção 2.1.7 e no início deste capítulo. Somente o recurso *R2* (Funcionário-Reclamações) será usado para demonstrar os diferentes tipos de mecanismo de alocação de recursos.

A ferramenta CPN Tools permite desenvolver uma rede de Petri colorida de forma hierárquica, ou seja, é possível subdividir o modelo em submodelos, sendo que cada um destes módulos aparece em uma subpágina da área de trabalho do *software*. Por sua vez, cada uma dessas subpáginas está associada a uma transição do modelo principal. É importante salientar que com o uso do modelo hierárquico será possível representar os recursos somente nas subpáginas. Então a representação do processo ficará completamente separada dos mecanismos de alocação de recurso. Assim, percebe-se que existe uma sistemática de modelagem no CPN Tools muito próximo dos modelos formais.

Para definir a duração de cada atividade um intervalo estático foi usado e, entre o valor mínimo e máximo, um tempo aleatório baseado na distribuição uniforme será utilizado. A duração total esperada para um caso deverá ser 105 minutos. É importante salientar que neste modelo não existe restrição de tempo para fixar a data de início de cada atividade.

### 3.2.1 Implementação de uma *Workflow net* com Mecanismos de Alocação de Recursos Discretos baseada na Linguagem CPN Tools sem Restrições de Tempo

A Figura 50 apresenta o modelo para o “Processo de Tratamento de Reclamações” definido em (AALST; HEE, 2004) com inclusão explícita de mecanismos de recursos discretos e intervalos de durações de tempo (modelo p-temporal apresentado em (JULIA; OLIVEIRA; VALETTE, 2008)) associados às atividades do processo. Com base neste modelo, a implementação de uma *Workflow net* com mecanismos de alocação de recursos discretos baseada na linguagem CPN Tools será apresentada.

A implementação do modelo mostrado na Figura 50 foi construída usando CPN Hierárquica. A Figura 51 representa o modelo de processo e as Figuras 52 e 53 mostram o mecanismo de alocação de recurso discreto. É importante salientar que com o uso do modelo hierárquico a representação do processo fica completamente separada dos mecanismos de alocação de recurso.

Cada atividade, aqui representada em uma transição de substituição (“*abstract transition*”), só é habilitada caso o recurso esteja disponível, representado explicitamente na subpágina correspondente. Também deve-se ressaltar que o modelo do “Processo de Tratamento de Reclamações” possui um ponto de sincronização visto que o processo só pode avançar depois que o cliente e o departamento afetado forem contactados. Na implemen-

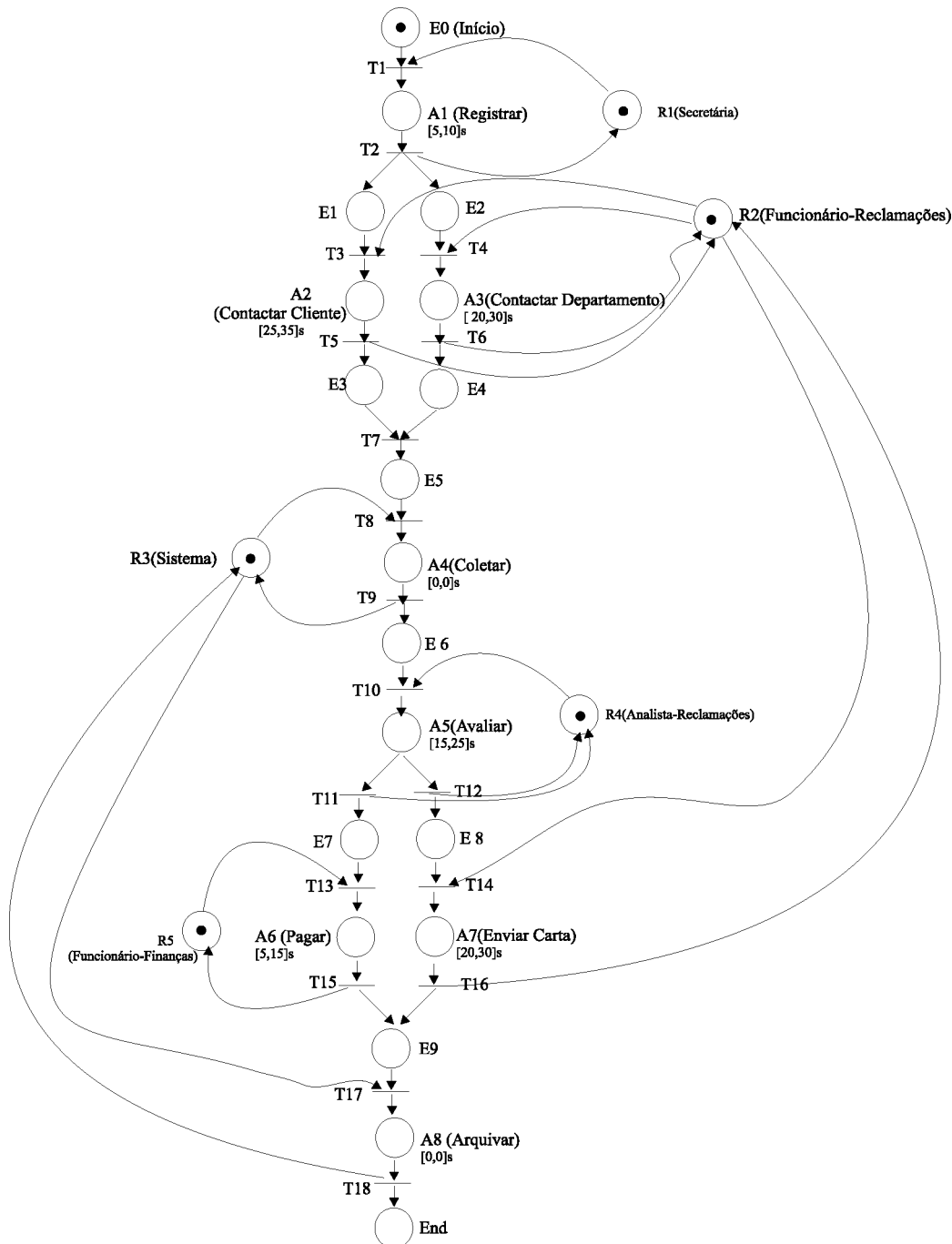


Figura 50 – Modelo para o Processo de Tratamento de Reclamações com Recursos Discretos.

tação isto é explicitado pela transição *transition* e a condição  $\text{if } (\#1(\text{tp1})) > (\#1(\text{tp2}))$  **then**  $\text{tp1}$  **else**  $\text{tp2}$  que verifica o tempo em cada ficha e prossegue com o maior tempo do caso tratado no momento do disparo.

A Tabela 1 apresenta as declarações utilizadas no modelo.

Conforme observado na Tabela 1, o modelo define o seguinte conjunto de Cores (*colset*).

a) TI: representa o tempo como inteiro, podendo ser o tempo inicial, final ou duração

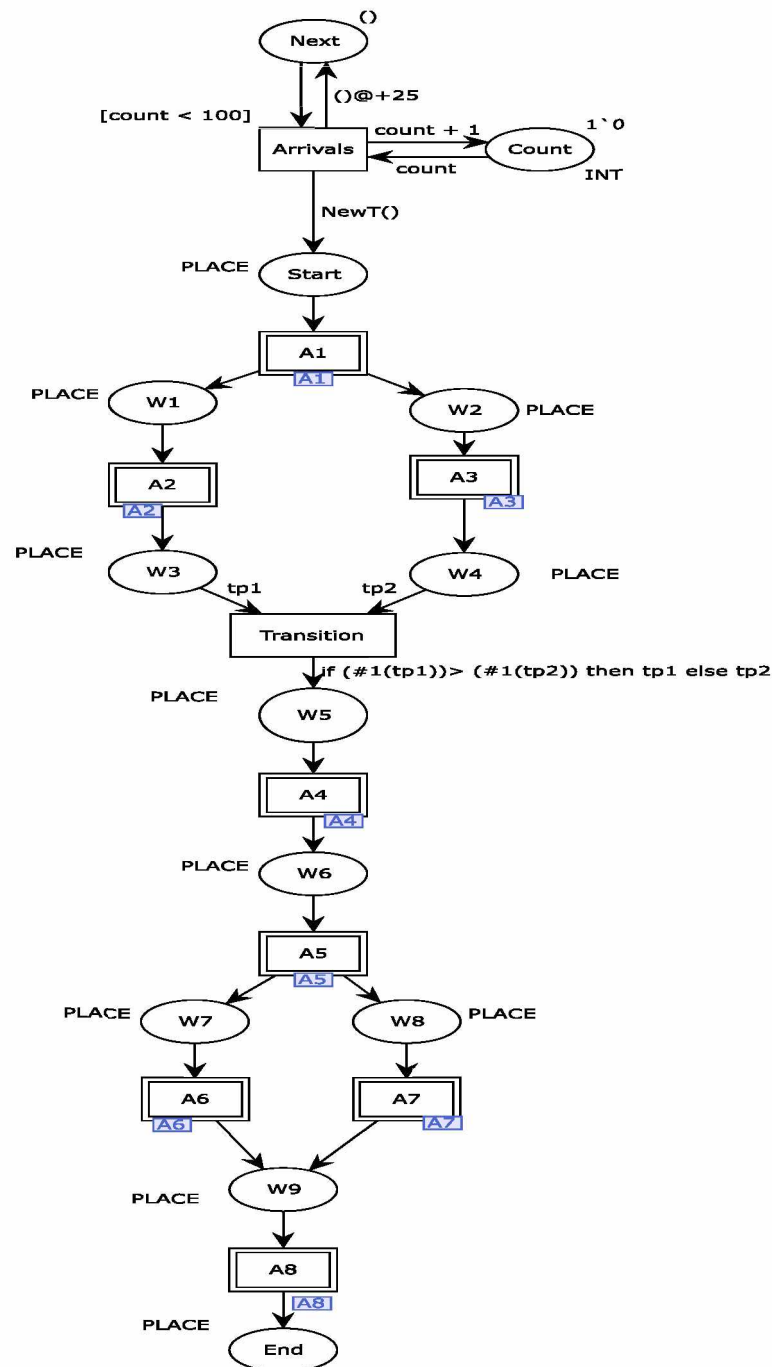


Figura 51 – Modelo da *Workflow net* - Recursos Discretos.

de um caso.

- b) t1: representa o tempo inicial do intervalo de uma atividade.
- c) t2: representa o tempo final do intervalo de uma atividade.
- d) W : representa o intervalo de tempo de uma atividade.
- e) RESOURCED: define o recurso discreto.
- f) PLACE: representa os lugares do Processo de Tratamento de Reclamações, mostra

colset TI = INT;
colset t1 = REAL;
colset t2 = REAL;
colset W = product t1*t2;
colset RESOURCED = UNIT;
colset PLACE = product TI*TI timed;
colset CHOICE = with pay  letter;

Tabela 1 – Declarações para o Modelo CPN com Recursos Discretos.

o tempo de duração do caso, assim como a data de início do caso tratado.

- g) CHOICE: define a escolha a ser feita - se um pagamento será efetuado ou uma carta será enviada.

A tabela 2 apresenta as variáveis utilizadas no modelo. Variáveis CPN tem a mesma característica de variáveis em qualquer linguagem de programação, porém os tipos de dados são as Cores (*colset*).

var t, upd_t, tp1, tp2: PLACE;
var r1, r2, r3, r4, r5: RESOURCED;
var c: CHOICE;
var count: INT;
var proctime, tt, ti: Tp;

Tabela 2 – Variáveis para o Modelo CPN com Recursos Discretos

A tabela 3 apresenta as constantes que foram definidas para o modelo. Neste caso, estas constantes representam o intervalo de duração de cada atividade. É importante salientar que as atividades A4 e A8 tem execução imediata, sendo assim, seu tempo de execução é igual a zero, não sendo necessária a sua definição.

val A1 = (5.0,10.0):W;
val A2 = (25.0, 35.0):W;
val A3 = (20.0,30.0):W;
val A5 = (15.0,25.0):W;
val A6 = (5.0,15.0):W;
val A7 = (20.0,30.0):W;

Tabela 3 – Constantes para o Modelo CPN com Recursos Discretos.

As subpáginas para as transições A1, A2, A3 e A4 são mostradas na Figura 52. Nestas subpáginas, cada atividade é definida como uma transição de entrada para verificar se existe recurso disponível. Se a transição é disparada então o tempo de execução é atualizado somando o valor da duração da atividade ao tempo que está na ficha que representa o cliente ( $(upd\_t)@ + proctime$ ). Quando a atividade termina, o recurso é devolvido.

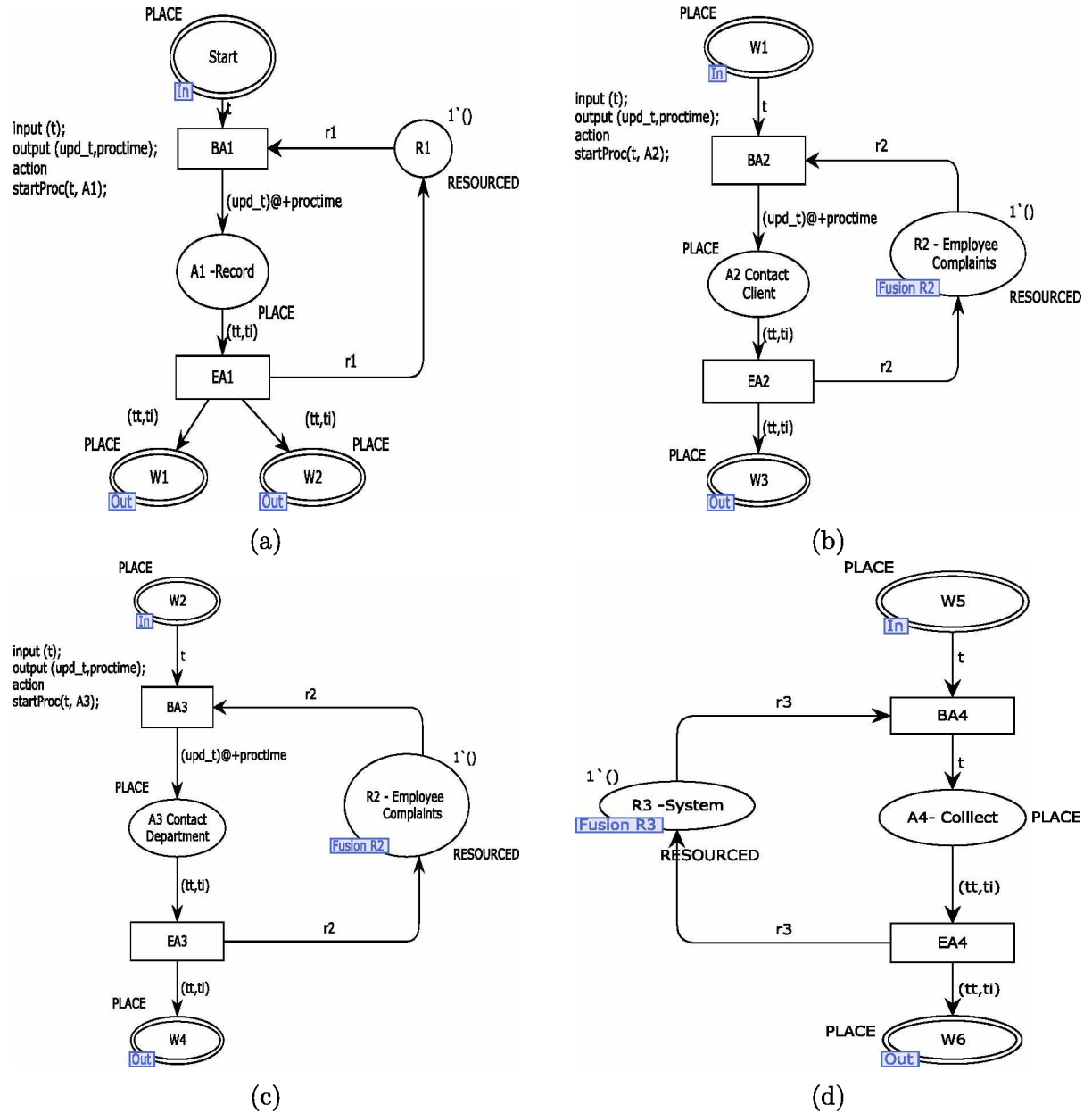


Figura 52 – Atividades A1, A2, A3 e A4 do Processo de Tratamento de Reclamações - Recursos Discretos.

As subpáginas para as transições A5, A6, A7 e A8 são mostradas na Figura 53. Basicamente, o comportamento destas atividades é o mesmo que o apresentado nas atividades da Figura 52, com exceção da atividade A5. Como o roteiro é alternativo, é necessário que uma decisão seja tomada - ou um pagamento é feito, ou uma carta é enviada. Torna-se indispensável então definir uma condição de guarda que verifica qual foi a decisão tomada, ou seja, para habilitar a transição *EA5pay*, a decisão deve ter sido “efetuar o pagamento” satisfazendo a condição de guarda  $[c = \text{pay}]$ . Caso a decisão tenha sido “enviar uma carta”, a transição *EA5letter* será habilitada pois a condição de guarda  $[c = \text{letter}]$  foi satisfeita. O conceito de *fusion place* foi utilizado nos recursos R2 e R3 a fim de manter a integridade do modelo e a característica de compartilhamento dos recursos. Desta forma,

pode-se garantir que uma alteração no recurso *R2* ou *R3* em qualquer subpágina é atualizado instantaneamente nas outras subpáginas. A *fusiontag* definida para o recurso *R2* é *fusion R2* e para o recurso *R3* é *fusion R3*.

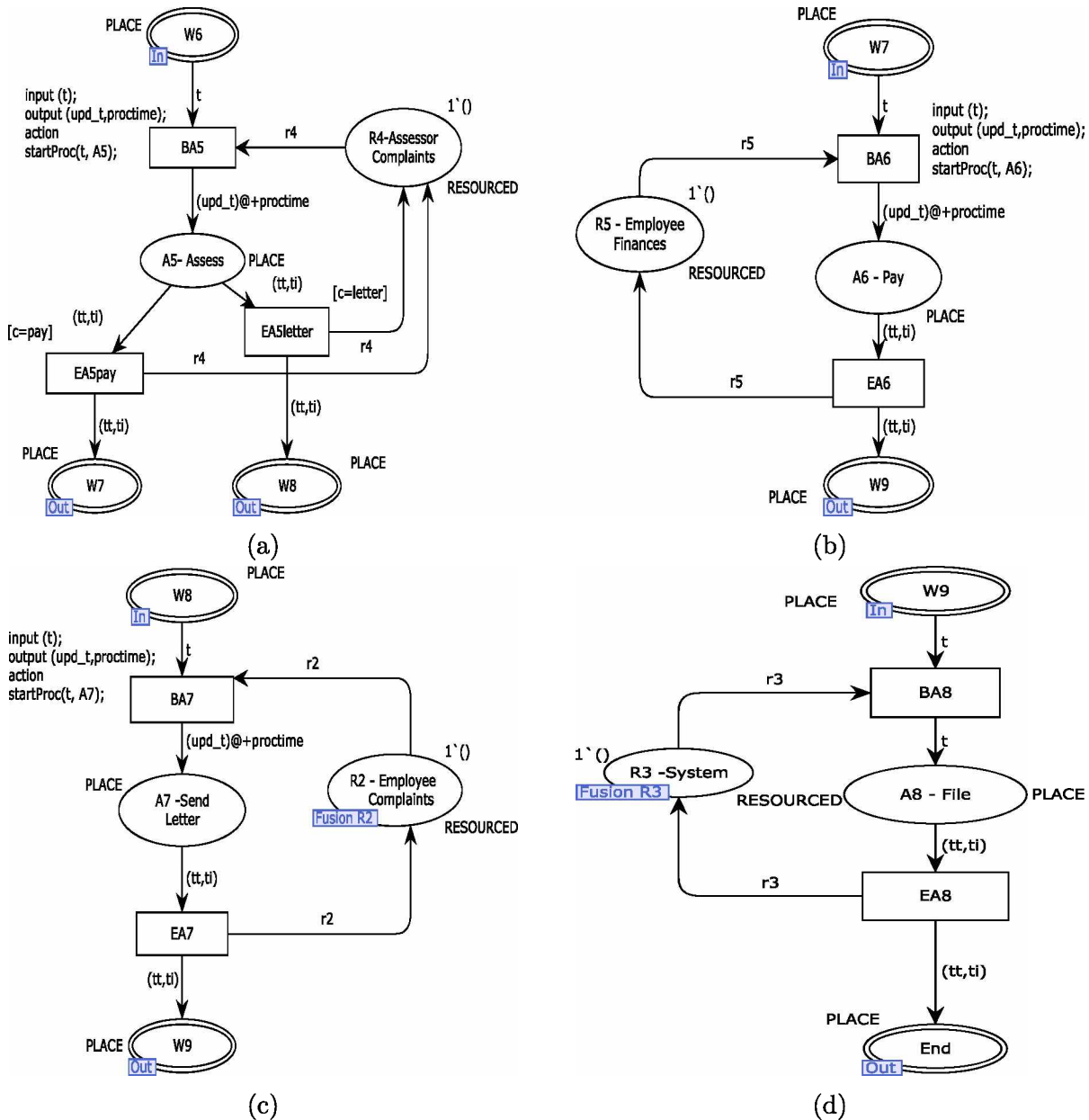


Figura 53 – Atividades A5, A6, A7 e A8 do Processo de Tratamento de Reclamações - Recursos Discretos.

### 3.2.2 Implementação de uma *Workflow net* com Mecanismo de Alocação de Recursos Discretos e Contínuos baseada na Linguagem do CPN Tools sem Restrições de Tempo

A Figura 54 apresenta o modelo para o “Processo de Tratamento de Reclamações” definido em (AALST; HEE, 2004) com inclusão explícita de mecanismos de recursos





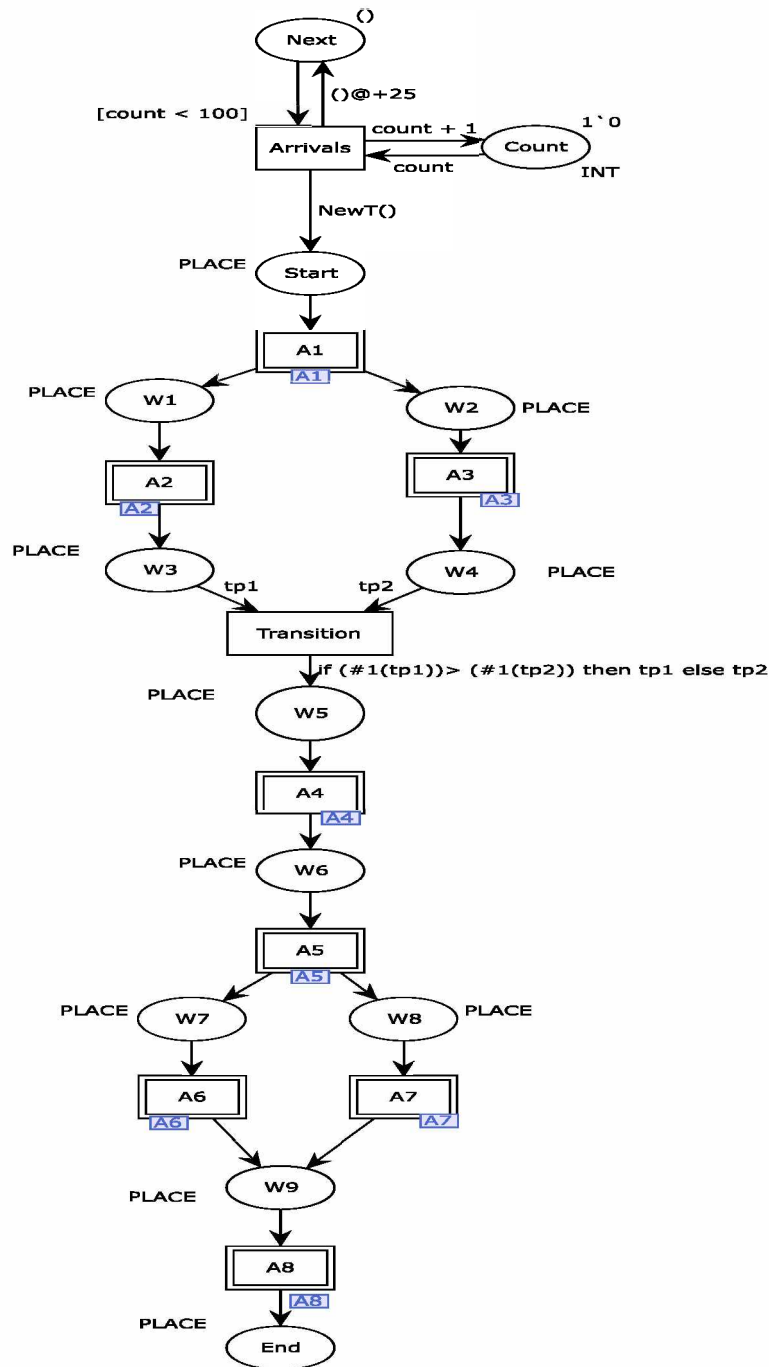


Figura 55 – Modelo da *Workflow net* - Recursos Discretos e Contínuo.

a serem acrescentadas ao modelo.

O funcionamento deste modelo é similar ao apresentado na subseção 3.2.1, exceto quando se trata do mecanismo de alocação de recurso contínuo *R2* que se dá nas atividades *A2*, *A3* e *A7*. O recurso contínuo é definido pelo *colset RESOURCEC*. A porcentagem para a conclusão de cada atividade é definida pelos pesos associados aos arcos de entrada/saída do lugar recurso e especificada no modelo através dos valores constantes definidos em *RBA2*, *RBA3* e *RBA7*.

As subpáginas para as transições *A1*, *A2*, *A3*, *A4*, *A5*, *A6*, *A7* e *A8* são mostradas nas

colset RESOURCEC = INT;
var r2: RESOURCEC;
val RBA2 = 30:INT;
val RBA3 = 40 : INT;
val RBA7 = 50 : INT;

Tabela 4 – Declarações para o modelo CPN com Recurso Discreto e Contínuo.

Figuras 56 e 57. As Figuras 56b, 56c e 57c mostram as atividades A2, A3 e A7 onde o recurso alocado é o recurso contínuo  $R2$ .

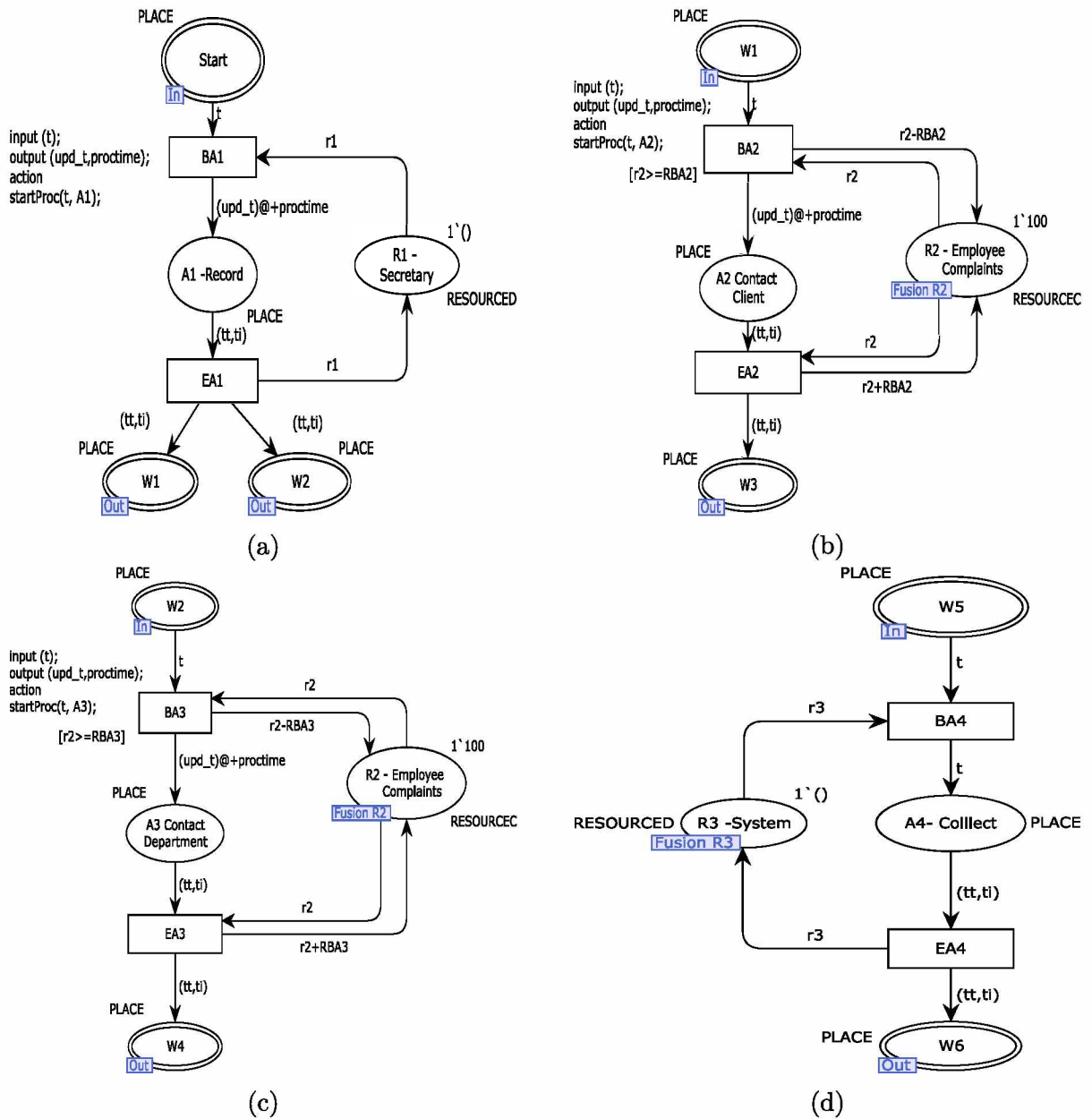


Figura 56 – Atividades A1, A2, A3 e A4 do Processo de Tratamento de Reclamações - Recursos Discretos e Contínuos.

Cada vez que uma transição é habilitada e o recurso  $R2$  precisa ser alocado, deve-

se garantir que a disponibilidade do recurso esteja suficiente (em porcentagem) para a realização da atividade correspondente. Isto acontece com a utilização de uma *Condição de Guarda* associada com a transição que corresponde à atividade a ser executada, onde a disponibilidade do recurso deve ser maior ou igual ao peso de arco ( $[r2 \geq RBA * ^1]$ ). Além disso, quando a transição é disparada, o valor de disponibilidade do recurso deve ser imediatamente atualizada. Se, por exemplo, a disponibilidade do recurso  $R2$  (em porcentagem) estiver suficiente para o disparo de uma transição, o valor  $RBA*$  associado ao arco de entrada da transição a ser disparada deve ser subtraído ao valor de disponibilidade do recurso antes do disparo ( $r2 - RBA*$ ). No final da execução da atividade que reservou parte da disponibilidade do recurso, o valor  $RBA*$  associado com o arco de entrada do recurso deve ser somado ao valor de disponibilidade do recurso que fica associado com a ficha que se encontra no lugar RECURSOC ( $r2 + RBA*$ ). No momento do disparo um arco suplementar de retorno precisa ser conectado ao recurso para realizar a atualização da disponibilidade do recurso.

### 3.2.3 Implementação de uma *Workflow net* com Mecanismos de Alocação de Recursos Discretos e *Fuzzy* baseada na Linguagem do CPN Tools sem Restrições de Tempo

A Figura 58 apresenta o modelo para o “Processo de Tratamento de Reclamações” definido em (AALST; HEE, 2004) com inclusão explícita de mecanismos de recursos discretos e *fuzzy* e intervalos de durações de tempo (modelo p-temporal apresentado em (JULIA; OLIVEIRA; VALETTE, 2008) associados às atividades do processo. Com base neste modelo, a implementação de uma *Workflow net* com mecanismos de alocação de recursos discretos e *fuzzy* baseada na linguagem CPN Tools será apresentada.

A implementação no CPN Tools do modelo mostrado na Figura 58 é mostrado nas Figuras 55, 60 e 61. A Figura 59 representa o modelo de processo e as Figuras 60 e 61 mostram o mecanismo de alocação de recurso discreto e *fuzzy*.

Somente o recurso  $R2$  será modificado a fim de de ilustrar o mecanismo de alocação *fuzzy*. Para tanto, além das declarações de *colset*, variáveis e constantes da seção 3.2.1, devem ser feitas novas declarações para o recurso *fuzzy*. A Tabela 5 apresenta tais declarações a serem acrescentadas ao modelo.

O tipo RESOURCEF que é um conjunto de 4 elementos  $[a1, a2, a3, a4]$  foi criado para definir o recurso  $R2$  como *fuzzy*. Assim, um valor de atributo valendo inicialmente  $[100, 100, 100, 100]$  (intervalo *fuzzy* que representa 100 % de disponibilidade) pode ser definido. A porcentagem para a realização de cada atividade é definida pelos pesos associados aos arcos de entrada e saída do lugar recurso, especificados no modelo do CPN Tools pelas constantes  $RBA2 = [20, 30, 30, 40]$ ,  $RBA3 = [30, 40, 40, 50]$  e  $RBA7 = [40, 50, 50, 60]$ .

<sup>1</sup> \* é o número atividade

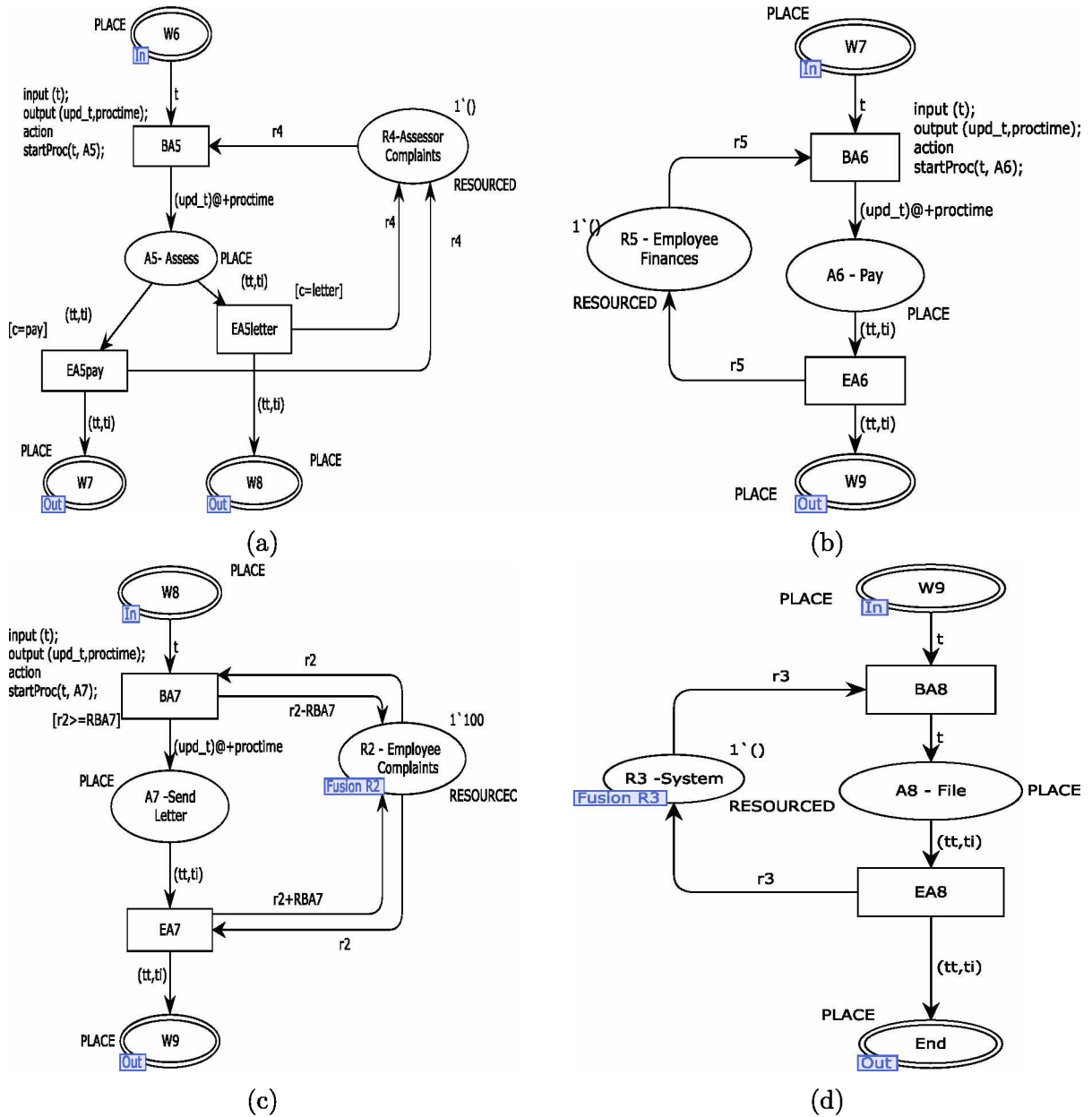


Figura 57 – Atividades A5, A6, A7 e A8 do Processo de Tratamento de Reclamações - Recursos Discretos e Contínuos.

Cada vez que uma transição é habilitada e o recurso  $R2$  precisa ser alocado, deve-se garantir que a disponibilidade do recurso esteja suficiente para a realização da atividade correspondente. Isto acontece com a utilização de uma *Condição de Guarda* associada com a transição que corresponde a atividade a ser executada, onde a disponibilidade do recurso deve ser maior ou igual ao peso de arco. Como se trata de intervalo *fuzzy*, deve-se comparar o elemento  $a4$  do intervalo *fuzzy* que representa a disponibilidade do recurso com o elemento  $a1$  que representa o peso do arco. Assim, a condição de guarda é definida como  $[(\#4(r2)) \geq (\#1(RBA *^2))]$ . Neste caso, é possível garantir que sempre haverá pelo menos um número positivo no intervalo *fuzzy* que define a disponibilidade do recurso,

<sup>2</sup> \* é o número atividade

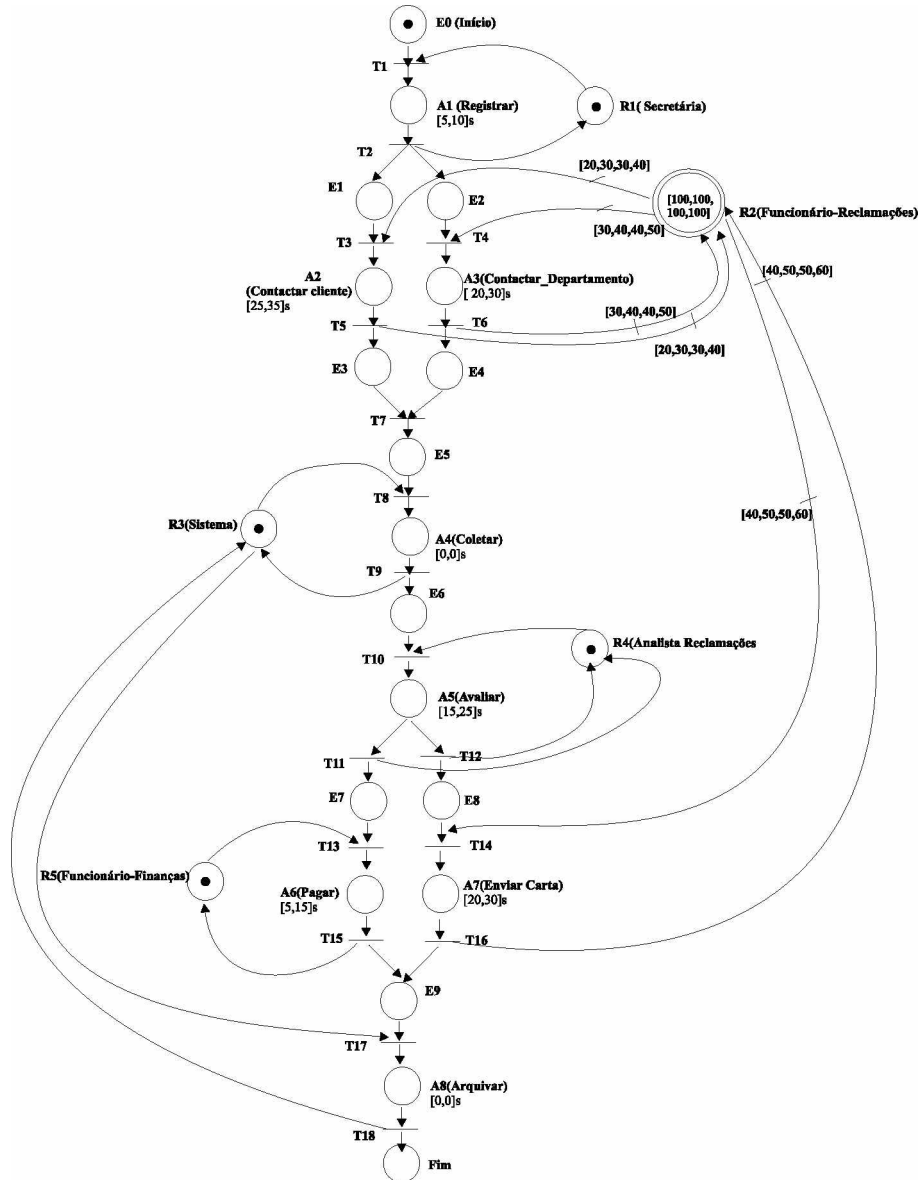


Figura 58 – Modelo para o Processo de Tratamento de Reclamações com Recursos Discretos e *Fuzzy*.

sabendo que, apesar do recurso estar sobrecarregado (quando existe algum elemento negativo), ainda existe possibilidade de alocação do recurso. Além disso, quando a transição é disparada, o valor de disponibilidade do recurso deve ser imediatamente atualizada. Se, por exemplo, o recurso *R2* fica disponibilizado para o disparo de uma transição, o valor  $RBA^*$  associado ao arco de entrada da transição a ser disparada deve ser subtraído ao valor de disponibilidade do recurso antes do disparo ( $r2 \ominus RBA^*$ ). No final da execução da atividade que reservou parte da disponibilidade do recurso, o valor  $RBA^*$  associado com o arco de entrada do recurso deve ser somado ao valor de disponibilidade do recurso que fica associado com a ficha que se encontra no lugar  $RECURSOF$  ( $r2 \boxplus RBA^*$ ). As operações de soma e subtração para o mecanismo de alocação de recursos *fuzzy* foram definidas na seção 3.1.3. No modelo CPN estas operações são implementadas pelas funções

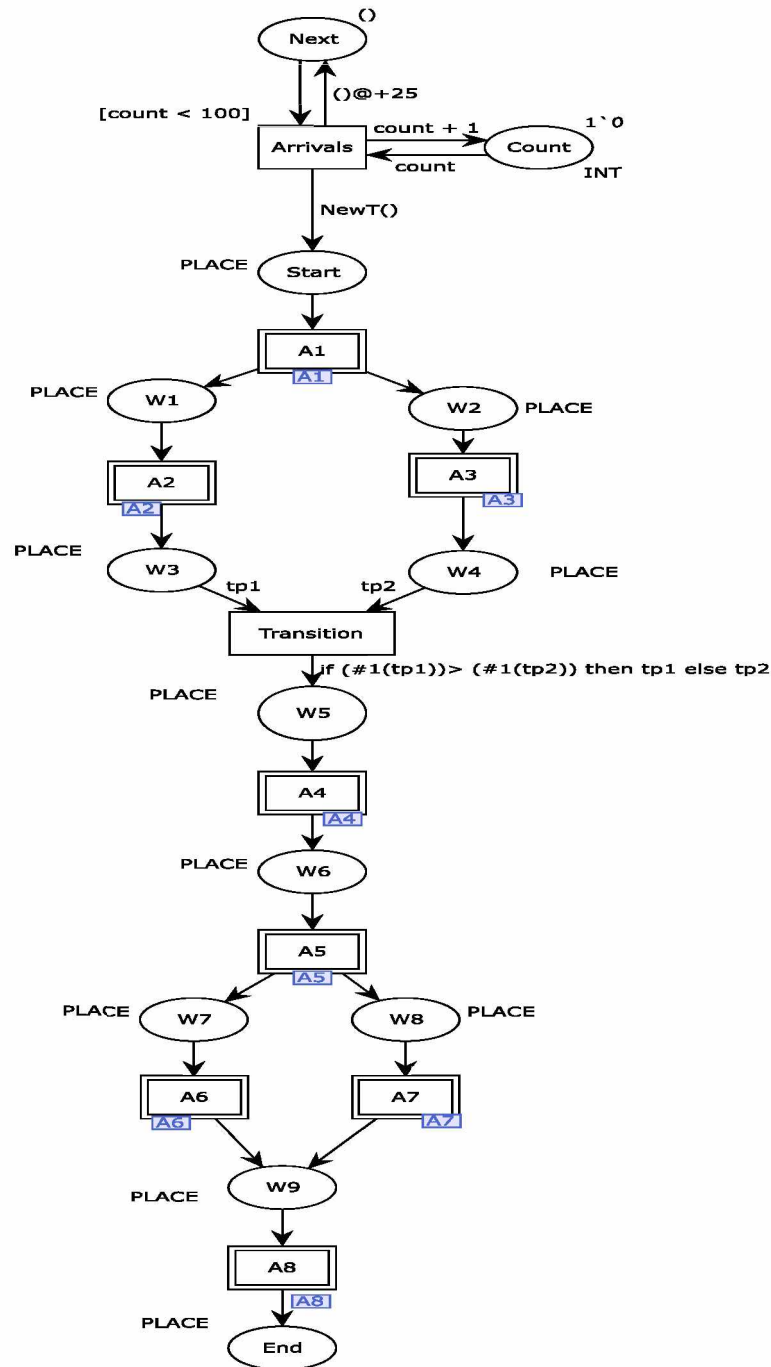


Figura 59 – Modelo da *Workflow net* - Recurso Discretos e *Fuzzy*

**fun soma** e **fun sub** apresentadas na Tabela 5. É imprescindível acrescentar um arco de retorno adicional no momento do disparo da transição a fim de manter o valor associado a disponibilidade do recurso atualizado.

As subpáginas para as transições A1, A2, A3, A4, A5, A6, A7 e A8 são mostradas nas Figuras 60 e 61. As Figuras 60b, 60c e 61c mostram as atividades A2, A3 e A7 onde o recurso alocado é o recurso *fuzzy* R2.

Aqui, cabe salientar que não seria possível construir o mecanismo de alocação *fuzzy* usando outros simuladores que não tenham a capacidade de programação do CPN Tools.

colset a1 = INT;
colset a2 = INT;
colset a3 = INT;
colset a4 = INT;
colset RESOURCEF = product a1 * a2 * a3* a4 ;
var r2: RESOURCEF;
val RBA2 = (20,30,30,40 ): RESOURCEF;;
val RBA3 = (30,40,40,50): RESOURCEF;
val RBA7 = (40,50,50,60) : RESOURCEF;
fun sub (r : RESOURCEF, RBA : RESOURCEF) = ((#1(r))-(#4(RBA)), (#2(r))-(#3(RBA)), (#3(r))-(#2(RBA)), (#4(r))-(#1(RBA)))
fun soma (r : RESOURCEF, RBA : RESOURCEF) = ((#1(r))+(#4(RBA)), (#2(r))+(#3(RBA)), (#3(r))+(#2(RBA)), (#4(r))+(#1(RBA)))

Tabela 5 – Declarações para o modelo CPN com Recurso Discreto e *Fuzzy*.

Somente mecanismos de alocação convencionais (discreto e eventualmente contínuos) poderiam ser representados então, neste caso, os mecanismos de alocação *fuzzy* não poderiam ser simulados.

### 3.3 Jogador de *Workflow net* com Recursos e Resultados de Simulação

Uma das técnicas que permite a execução dinâmica das redes de Petri é o algoritmo do jogador de rede de Petri (JULIA; VALETTE, 2000), que pode ser usado para a resolução do problema do escalonamento de sistemas em tempo real. No contexto deste capítulo, o jogador de rede de Petri permite a simulação de uma rede de Petri com funcionamento mais cedo, ou seja, as transições são disparadas no momento em que estão habilitadas.

Para demonstrar a abordagem proposta, um processo de simulação, utilizando o simulador CPN Tools (RATZER et al., 2003) foi executado. Os jogadores foram simulados em CPN Tools usando os diferentes mecanismos de alocação de recursos (discreto, contínuo e *fuzzy*).

Para adicionar expressividade ao modelo visando alcançar a correta simulação do modelo proposto, foram definidas algumas funções:

- a) **mda**: a cada atividade fica associada uma duração aleatória que pertence a um intervalo de tempo uniformemente distribuído. Para isto, uma função denominada *mda* foi definida para que durante a simulação a distribuição estatística seja uniforme e a probabilidade de cada instante seja a mesma.

```
1. fun mda(Z:W) = round(uniform((#1(Z)), (#2(Z))));
```



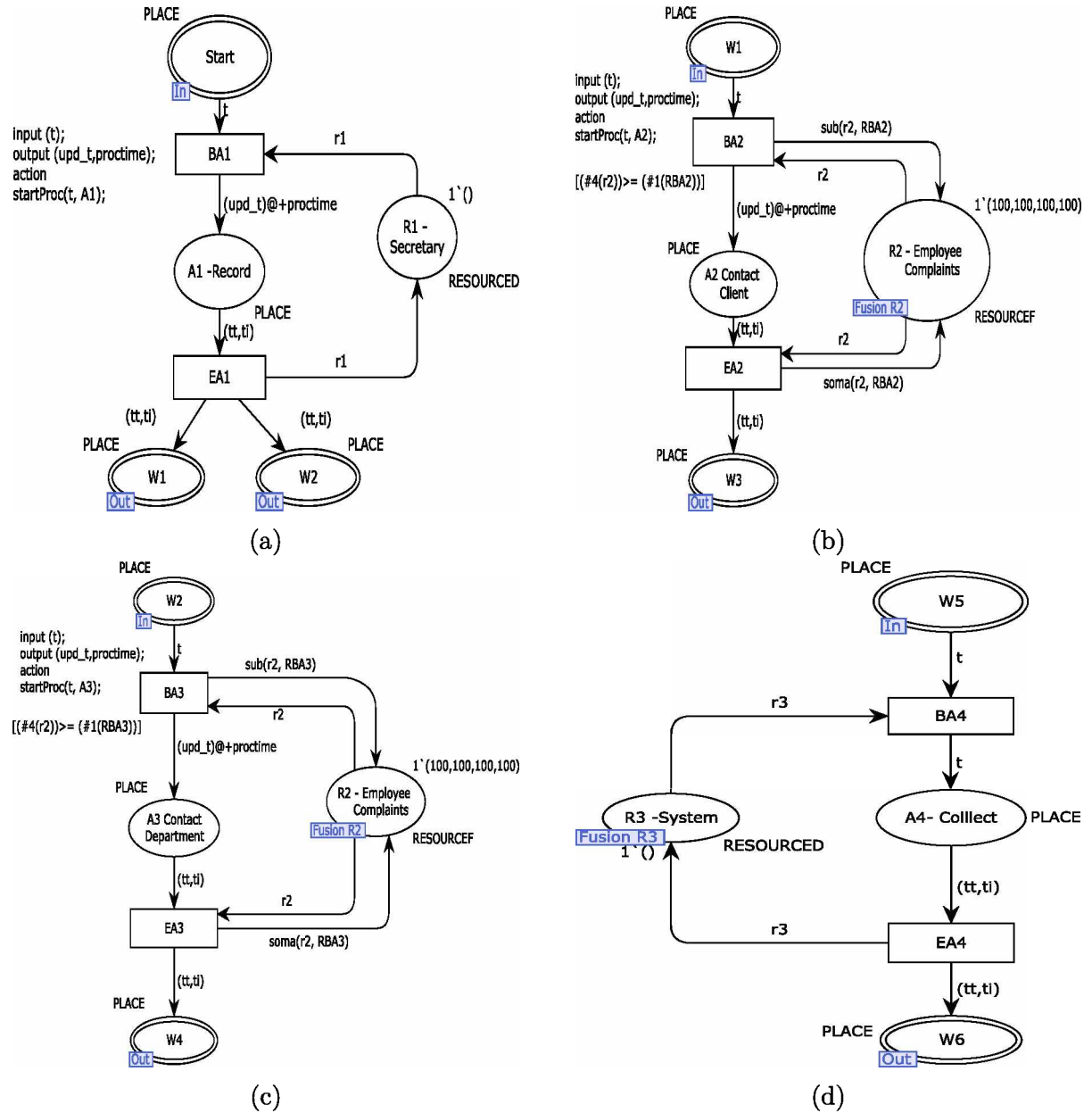


Figura 60 – Atividades A1, A2, A3 e A4 do Processo de Tratamento de Reclamações - Recursos Discretos e *Fuzzy*.

a) **intTime**: a função `intTime` é declarada para retornar a data de simulação corrente na forma de um valor inteiro. Esta função pertence a biblioteca de funções predefinidas da ferramenta CPN.

```
1. fun intTime() = IntInf.toInt(time()): int;
```

a) **startProc**: a função `startProc` é responsável por atualizar o tempo de duração de cada caso. Inicialmente, a duração da atividade é calculada pela chamada à função `mda`. O tempo atual é acrescido a duração da atividade usando a função da biblioteca `Base SML ModelTime.add` e, então, a variável `tt` (que armazena a duração total do caso) é atualizada.

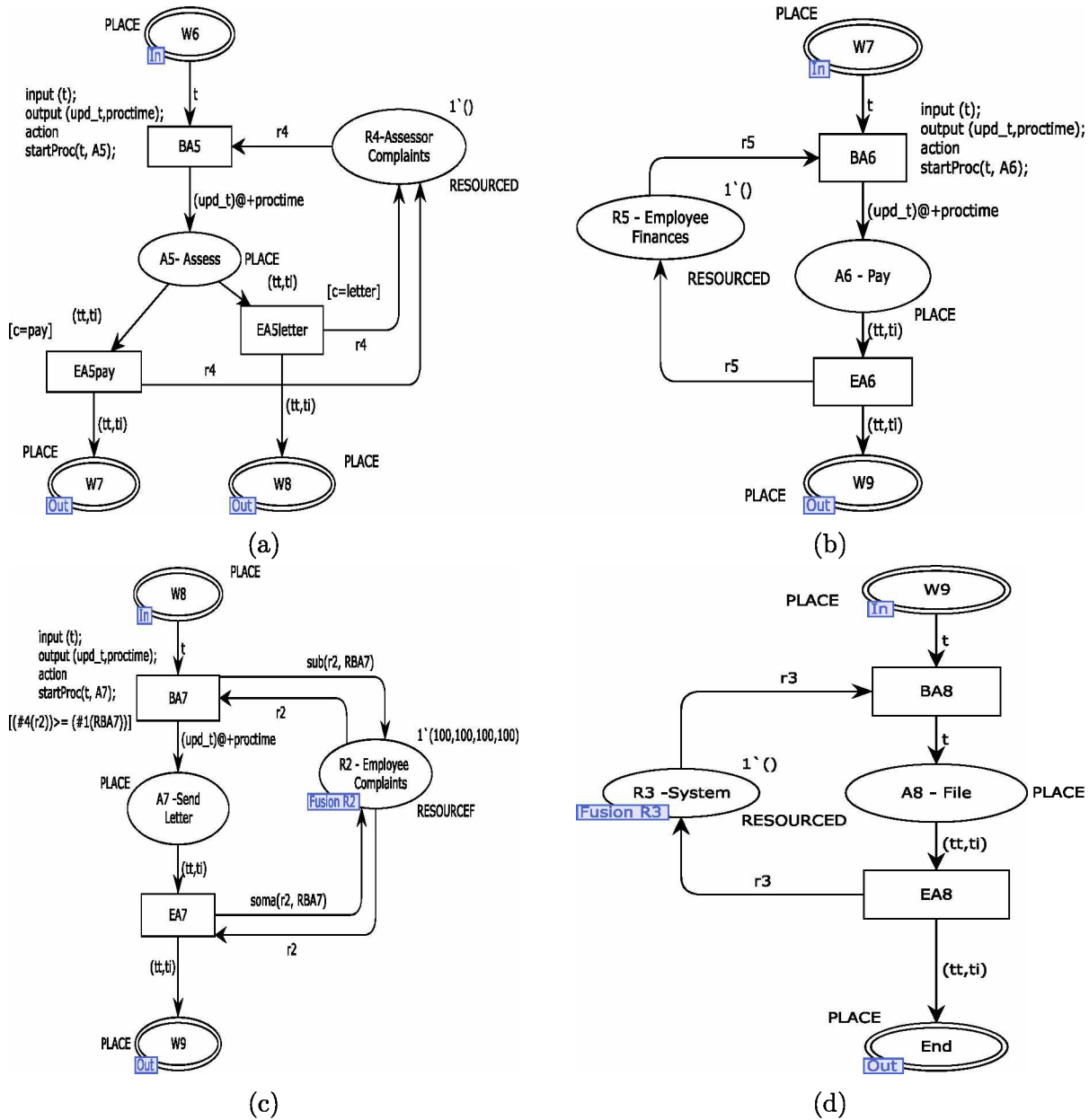


Figura 61 – Atividades A5, A6, A7 e A8 do Processo de Tratamento de Reclamações - Recursos Discretos e *Fuzzy*.

```

1. fun startProc ((tt,ti):PLACE, Z:W) =
2.   let
3.     val proc_time = mda(Z)
4.     val time_stamp =
5.       ModelTime.add(time(), ModelTime.fromInt(proc_time))
6.   in
7.     ((new_tt, ti), proc_time)
8.   end

```

- a) **NewT**: a função NewT é responsável por gerar novos casos. Cada caso tem o tempo de duração e o tempo inicial.

```
1. fun NewT() = (intTime(),intTime());
```

A fim de padronizar os resultados, 100 casos serão gerados em intervalos regulares de 25 minutos. A duração dos intervalos foi definida baseada em experimentos de forma a mostrar, da melhor maneira, o funcionamento dos modelos. Rozinat et al. (2008) mostram que quanto maior o número de replicações, maior a precisão do resultado. No entanto, é necessário decidir um limite aceitável de replicações. Sendo assim, 20 experimentos de simulação (replicações) foram considerados para cada mecanismo de alocação de recursos com um horizonte de simulação de 1600 eventos <sup>3</sup>. Com esta quantidade de repetições a dispersão obtida foi pequena; menor que 10% em relação à ordem de grandeza das métricas, considerando o intervalo de confiança de 95%.

O simulador CPN Tool permite a criação de monitores para coleta de dados de simulação utilizando-se *Data Collector Monitoring Functions*, as quais apresentam um alto grau de flexibilidade, uma vez que as funções são escritas em código CPN-ML para atender necessidades específicas. Um monitor inclui uma função de predicado (*predicate function*), que determina quando o dado deve ser coletado, e uma função de observação (*observation function*) que determina qual dado deve ser coletado. Sabendo que a distribuição aleatória escolhida, associada às atividades do modelo, foi a distribuição uniforme com limites mínimo e máximo pode-se garantir que a probabilidade de cada tempo ocorrer é uniforme. Sendo assim, a duração das atividades não poderá gerar um atraso maior que o valor da borda máxima associada ao intervalo de tempo de atividade; somente a espera por um recurso poderá atrasar além do esperado (quando são considerados os valores máximos de todas as atividades envolvidas na execução do processo) Como mostrado na seção 3.1, somente o recurso *R2* (Funcionário-Reclamações) poderá influenciar no resultado da avaliação do mecanismo de alocação de recursos. A fim de saber a duração de cada caso o monitor **Duracao Total** foi definido:

- a) **Duracao Total** - a função deste monitor é calcular o tempo total de duração de cada caso. Por definição, o prazo máximo para o término do processo é de 105 minutos. Com a análise deste monitor, será possível verificar se os prazos foram cumpridos.

### 3.3.1 Resultado da Simulação: Mecanismo de Alocação de Recurso Discreto

A Tabela 6 apresenta os valores estatísticos para a simulação do mecanismo de alocação de recurso discreto. A duração máxima permitida para a conclusão de cada caso é de

<sup>3</sup> valor suficiente para simular os 100 casos gerados

105 minutos. O valor (1,A) representa o valor médio dos menores tempos de duração encontrados, sendo o mínimo menor valor 60 mostrado em (1,C) e o mínimo do maior valor 115 mostrado em (1,D). O valor (2,A) representa o valor médio dos maiores tempos de duração encontrados, sendo o maior valor entre os mínimos 5805 mostrado em (2,C) e o máximo maior valor 6714 mostrado em (2,D). O valor (3,A) representa o valor médio do tempo de execução calculado com base na média dos valores mínimos (1,C) e (2,C) e na média dos valores máximos (1,D) e (2,D). Percebe-se que a duração máxima permitida (105 minutos) foi respeitada em alguns casos, visto que os valores encontrados na linha *Min\_id* representam os valores no *melhor caso*. No entanto, em outros casos, o valor é muito alto em relação ao máximo permitido como mostra a linha *Max\_id* que representa os valores no *pior caso*. A média do tempo de execução dos 100 casos utilizando somente recursos discretos é de 2405,43 ( $\pm 56,21$ ), muito acima do esperado.

		A	B	C	D
	Nome	Média	Intervalo de Confiança (95%)	Mínimo	Máximo
1	Min_id	87,30	7,84	60	115
2	Max_id	6255,55	139,23	5805	6714
3	Média_id	2405,43	56,21	2237,83	2634,18

Tabela 6 – Tempos de Execução para o Mecanismo de Alocação de Recurso Discreto

O relatório completo pode ser encontrado no Apêndice A.1.

### 3.3.2 Resultado da Simulação: Mecanismo de Alocação de Recurso Discreto e Contínuo

A Tabela 7 apresenta os valores estatísticos para a simulação do mecanismo de alocação de recurso discreto e contínuo. Percebe-se que a duração máxima permitida (105 minutos) não foi respeitada em alguns casos (como mostra a linha *Max\_id*). No entanto, quando este resultado é comparado com resultado do mecanismo de alocação de recurso discreto, pode-se notar que o tempo médio gasto para terminar cada caso melhorou em cerca de 80%. Como foi apresentado na seção 3.1.2, o fato que um recurso contínuo melhora a performance é justamente porque o caso do modelo puramente discreto não corresponde ao tipo de recursos a ser considerado em processos de negócios; afinal, o esforço de um funcionário pode ser dividido para realizar tarefas distintas em um mesmo período de tempo.

O relatório completo pode ser encontrado no Apêndice A.1.

Nome	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Min_id	62,65	2,46	56	72
Max_id	1751,00	168,14	1235	2479
Média_id	457,53	35,47	337,90	594,25

Tabela 7 – Tempos de Execução para o Mecanismo de Alocação de Recurso Discreto e Contínuo

### 3.3.3 Resultado da Simulação: Mecanismo de Alocação de Recurso Discreto e *Fuzzy*

A Tabela 8 apresenta os valores estatísticos para a simulação do mecanismo de alocação de recurso discreto e *fuzzy*. Percebe-se que a duração máxima permitida (105 minutos) foi respeitada na maioria dos casos, visto que o tempo médio gasto para concluir um caso foi de 80,66 ( $\pm 0,95$ ) como mostra a linha *Média\_id*. No entanto, neste experimento os recursos foram sobrecarregados de forma sistemática. Em situações reais, tal melhoria poderá ser explorada, em particular, quando certos casos críticos não previstos aparecerão nos fluxos dos processos de negócios. É claro que se os recursos começam a trabalhar em permanência num regime forçado, a qualidade de processamento dos casos vai baixar; por isso que o uso da margem adicional que os recursos *fuzzy* produzem deverá ser usada somente em casos excepcionais.

Nome	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Min_id	59,60	0,78	56	62
Max_id	108,55	5,23	98	138
Média_id	80,66	0,95	77,30	86,13

Tabela 8 – Tempos de Execução para o Mecanismo de Alocação de Recurso Discreto e *Fuzzy*

O relatório completo pode ser encontrado no Apêndice A.1.

### 3.3.4 Resultados Comparativos

Considerando um intervalo de confiança de 95% a tabela 9 apresenta os valores médios para o melhor caso, pior caso e caso médio dos diferentes mecanismos de alocação de recursos.

A tabela 10 mostra a porcentagem de melhora no tempo de conclusão de cada caso considerando a realização de tarefas que usam o mecanismo de alocação de recurso *fuzzy* em comparação com o mecanismo de alocação de recurso discreto e contínuo. Há uma melhora significativa do mecanismo de alocação de recurso *fuzzy* quando comparado com os outros mecanismos de alocação de recurso. Isto pode ser justificado pelo fato do recurso compartilhado *R2* no mecanismo de alocação de recurso *fuzzy* ter sido sobrecarregado sem

Recurso	Melhor Caso	Pior Caso	Caso Médio
Discreto	87,30 $\pm$ 7, 84	6255,55 $\pm$ 139, 23	2405,43 $\pm$ 56, 21
Contínuo	62,65 $\pm$ 2, 46	1751,00 $\pm$ 168, 14	457,53 $\pm$ 35, 47
<i>Fuzzy</i>	59,60 $\pm$ 0, 78	108,55 $\pm$ 5, 23	80,66 $\pm$ 0, 95

Tabela 9 – Valores Comparativos - Mecanismo de Alocação de Recurso Discreto, Contínuo e *Fuzzy*.

Recurso	Melhor Caso(%)	Pior Caso (%)	Caso Médio(%)
Discreto	46,47	5662,82	2882,18
Contínuo	empate	1513,08	467,23

Tabela 10 – Melhora *Fuzzy* em relação ao Mecanismo de Alocação de Recurso Discreto e Contínuo

nenhuma restrição. Então, este recurso trabalhou com toda a possibilidade disponível possível. No entanto, apesar do prazo ter sido cumprido na maioria dos casos, a sobrecarga de trabalho pode acarretar diminuição da qualidade do serviço. Sendo assim, deve-se ampliar o monitoramento dos casos considerando, na medida do possível, informações adicionais relacionadas com os prazos de conclusão previstos de cada caso para tentar implementar uma política de escalonamento que não seja simplesmente do tipo FIFO. De fato, em certas situações, quando um caso não precisa ser processado imediatamente (caso cuja parte das atividades foram adiantadas em relação ao prazo de entrega), poderá ser interessante usar a disponibilidade dos recursos para outros casos mais urgentes e deixar o caso adiantado em espera, para não sobrecarregar o trabalho dos recursos.

É importante salientar que além dos resultados de simulação apresentados neste capítulo, a implementação dos mecanismos de alocação de recursos no CPN Tools comprova que tais mecanismos poderão ser então implementados em SGW reais. Em particular, as simulações executadas no CPN Tools podem ser vistas como um tipo de escalonamento em tempo de execução (sem mecanismos de retrocessos) das atividade dos processos de negócios, no caso de um disparo mais cedo, mas com um certo grau de flexibilidade na resolução dos conflitos de recursos devido ao uso de uma política de alocação de recursos *fuzzy*.

---

## Modelos de Alocação de Recursos em *Workflow net* com Restrições de Tempo

Este capítulo apresenta modelos de alocação de recursos em *Workflow net* com restrições de tempo que podem ser usado para a simulação de estratégias de escalonamento em tempo de execução (sem mecanismos de retrocesso) de SGWs. A ideia central deste capítulo se baseia no fato de que, para cumprir os prazos de entrega dos casos processados, torna-se imprescindível um tipo de monitoramento da disponibilidade dos recursos usados nas atividades dos processos em função das restrições de prazos de conclusão dos casos processados pelo *Workflow*. A seção 4.1 apresenta o problema de escalonamento em SGWs. A seção 4.2 apresenta uma proposta de modelagem de restrições de tempo em *Workflow net*. A seção 4.3 mostra as novas regras de disparo de transições baseadas em um conceito de possibilidade conjunta de habilitação de atividades que dependem tanto da disponibilidade de recursos quanto de informações relativas ao tempo decorrido e aos prazos de conclusão dos casos tratados nos processos; também a implementação do modelo CPN Tools, bem como os resultados da simulação de estratégias de escalonamento são apresentadas.

### 4.1 Problema de Escalonamento em SGWs

No contexto dos SGWs, um escalonamento é definido pelo planejamento de execução das atividades, isto é, a ordem de execução das atividades bem como as datas de início e de fim das atividades, além da alocação dos diversos recursos envolvidos. Uma solução para o problema de escalonamento de um SGW deve então considerar características importantes inerentes a estes sistemas. Em particular, se muitos casos são executados simultaneamente, situações de conflito poderão ocorrer pela utilização de um mesmo recurso e deverão ser resolvidas em tempo real, ou seja, sem um mecanismo de retrocesso. Tal solução deverá em particular considerar as restrições temporais relativas aos prazos de entrega dos casos específicos, além dos diferentes roteiros que cada caso pode seguir.

Uma situação de conflito poderá ocorrer sempre que um mesmo recurso não preemptivo (que não pode ser interrompido) for chamado dentro de um intervalo de tempo comum para a execução de atividades pertencentes a diferentes cenários (diferentes casos).

Esquirol, Huguet e Lopez (1995) apresentaram uma abordagem para solucionar o problema de escalonamento de sistemas representados por grafos com restrições não conjuntivas (mecanismos de alocação de recursos em caso não preemptivo) que utiliza um mecanismo de propagação de restrição que permite o cálculo de intervalos de datas admissíveis para o cálculo de datas de início de atividades mais cedo e mais tarde. Um algoritmo do tipo *branch and bound* cuja meta é encontrar uma sequência ótima de atividades que respeita as restrições temporais (intervalos de datas) complementa então a abordagem proposta. Geralmente, tais abordagens foram desenvolvidas no contexto de sistemas de produção cujos recursos envolvidos não são do tipo humano (máquinas, sistemas de transportes automatizados, robôs etc.) e seguem um comportamento bastante previsível (JUNG; KIM; LEE, 2015), (HAN et al., 2014) e (ZHOU, 2012). Contudo, para os SGWs, tal abordagem apresenta certas limitações, uma vez que a solução gerada por um algoritmo de otimização geralmente é dada por uma sequência de atividades que dificilmente poderá ser seguida em tempo de execução, em particular para sistema que associam recursos humanos a grande parte das atividades a serem executadas nos processos.

Silva e Valette (1990) apresentam uma abordagem baseada em um jogador de rede de Petri onde foi mostrado que disparar as transições assim que elas se tornam disponíveis nem sempre é a melhor estratégia. Dessa forma, um jogador de redes de Petri tradicional (que dispara as transições tão logo elas ficam habilitadas), dificilmente respeitará as restrições referentes aos prazos de entrega dos casos de um processo de negócio porque o problema de escalonamento no caso de SGWs, em vez de se resumir a um problema de otimização, se trata mais de um problema de satisfação de um conjunto de restrições temporais, o objetivo sendo de entregar os casos nos prazos estabelecidos e não necessariamente o mais rápido possível.

Sendo assim, a proposta apresentada neste capítulo mostra uma abordagem baseada na combinação de dois mecanismos: um mecanismo de propagação de restrições de tempo *fuzzy* aplicados aos roteiros dos processos para o cálculo de uma estimativa sobre as datas de início das atividades dos processos mais cedo e mais tarde; e um jogador de redes de Petri que tem como objetivo calcular uma sequência de atividades que respeite as restrições disjuntivas (mecanismo de alocação de recursos) e as restrições temporais (intervalos de datas) sem mecanismo de retrocesso (solução de escalonamento produzida em tempo de execução); em particular, a característica de tal jogador será de não disparar necessariamente as transições habilitadas mais cedo (tão logo elas ficam disponíveis) considerando novas regras de disparo baseadas num conceito de possibilidade conjunta que envolve dados *fuzzy* tanto sobre a disponibilidade dos recursos quanto sobre as datas previstas de



realização das atividades.

## 4.2 Modelagem de Restrições de Tempo em *Workflow net*

No caso de *Workflow nets*, dificilmente pode-se fazer uma previsão correta dos tempos de realização das atividades. Por exemplo, o tempo de atendimento a um cliente pode variar de acordo com o problema a ser tratado ou da destreza do funcionário em resolver o problema. De fato, um modelo simplesmente temporizado não poderá servir para um escalonamento previsional e, menos ainda, para um escalonamento em tempo real. A introdução de durações incertas no modelo formal torna-se, então, necessária. A escolha de restrições de tempo representadas por intervalos *fuzzy* fica mais adequada à representação do comportamento humano e das atividades desenvolvidas em processos de negócios.

No contexto das redes de Petri temporais *fuzzy*, uma *Time Workflow net fuzzy* será então uma *Workflow net* estendida com intervalos de tempo *fuzzy* associados às transições, representando a execução das tarefas, e intervalos de tempo *fuzzy* associados aos lugares representando os intervalos de visibilidade dos casos tratados (intervalos de datas).

**Definição 17 (Time Workflow net fuzzy)** Uma *Time Workflow net fuzzy* é uma *quín-tupla*  $\langle P, T, F, I, IE \rangle$  tal que:

a)  $\langle P, T, F \rangle$  é uma *Workflow net*, onde:

- $P$  é um conjunto finito de lugares,
- $T$  é um conjunto finito de transições e
- $F \subseteq (P \times T) \cup (T \times P)$  é um conjunto de arcos (relação de fluxo).

b)  $I$  é uma aplicação que associa a cada transição  $t \in T$  um intervalo de duração de atividade  $I(t) = [[a_i], [b_i]]$ , onde  $[a_i]$  representa a duração *fuzzy* mínima (na forma triangular) associado à transição  $t$  e  $[b_i]$  representa a duração *fuzzy* máxima (na forma triangular) associada à transição  $t$ .

c)  $IE$ : é uma aplicação que associa a cada lugar  $p \in P$  um intervalo de visibilidade de um caso  $I(t) = [[\alpha_i], [\beta_i]]$ , onde  $[\alpha_i]$  representa a data *fuzzy* mínima de visibilidade de um caso no lugar  $p$  (na forma triangular) e  $[\beta_i]$  representa a data *fuzzy* máxima de visibilidade de um caso no lugar  $p$  (na forma triangular).

Conforme apresentado na definição (2), sabendo-se que um intervalo de tempo  $[x_i, y_i]$  é um período de tempo entre duas durações  $x_i$  e  $y_i$ , com  $x_i$  e  $y_i$  sendo durações *fuzzy*  $X_i$  e  $Y_i$  (delimitadas por  $\pi_{x_i}$ ,  $\pi_{y_i}$  respectivamente), o intervalo  $[x_i, y_i]$  é representado pela conjunção de instâncias de tempo, que representam o conjunto de durações possivelmente após  $X_i$  e possivelmente antes de  $Y_i$ . Sendo assim, é possível definir um conjunto de intervalos

de tempo *fuzzy* em único intervalo *fuzzy*. Por exemplo, considerando o intervalo *fuzzy* representado por  $[[20, 35, 50], [55, 70, 85]]$ ; então, pode-se produzir um intervalo *fuzzy* único  $[20, 35, 70, 85]$  (tanto para datas quanto para as durações apresentadas na definição 17). A Figura 62 mostra os intervalos de forma esquemática.

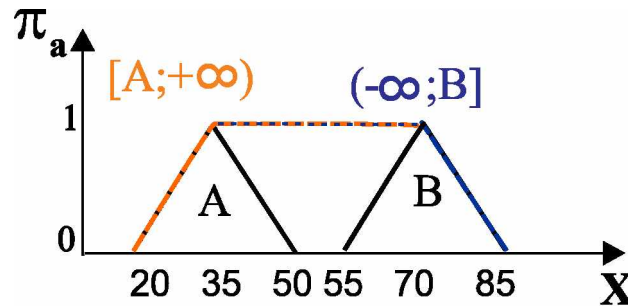


Figura 62 – Intervalo *Fuzzy*.

O modelo temporal para o “Processo de Tratamento de Reclamações” é mostrado na Figura 63. Os intervalos associados às atividades A4 - *Collect* e A8 - *File* são iguais a  $[[0,0,0],[0,0,0]]$  porque a duração dessas atividades é desprezível se comparada com as outras atividades do “Processo de Tratamento de Reclamações”. Os intervalos associados às atividades foram estipulados apenas a título de exemplificação, tomando como base as médias de duração das atividades. apresentadas em Aalst e Hee (2004).

Então, dadas as definições, é necessário entender que as durações das operações serão representadas por intervalos estáticos. Tais intervalos ficarão associados somente com as tarefas da *Workflow net*. Por exemplo, sabendo que para executar uma tarefa A deve demorar entre 10 e 30 minutos, com uma tolerância de mais ou menos 5 minutos, então seus intervalos *fuzzy* mínimos e máximos seriam definidos como  $[[5,10,15],[25,30,35]]$ .

O comportamento dinâmico de uma *Workflow net* depende da marcação da rede e da situação temporal das fichas que é dada por intervalo de visibilidade, que representa um intervalo de datas (e não de duração como é o caso dos intervalos estáticos); tais datas devem fixar as disponibilidades previstas das fichas (casos) que aparecerão nos lugares (condições) da *Workflow net*. De acordo com as datas de chegadas dos casos na *Workflow net*, com as durações das operações e com os prazos de entrega dos casos no final do processamento previstos, tais datas (bordas mínimas e máximas dos intervalos de visibilidade associados com os lugares) podem ser produzidas através de um mecanismo de propagação de restrições que depende em particular das estruturas dos roteiros da *Workflow net*. Assim, conhecendo a data de início e a máxima duração de um caso, é possível calcular os intervalos de visibilidade associados às fichas nos lugares condições da *Workflow net* (lugares de espera entre as diversas atividades do processo) usando técnicas de propagação de restrição classicamente utilizadas em problemas de escalonamento baseados em grafos sem circuitos (GONDRAN; MINOUX; VAJDA, 1984).

A fim de ilustrar a técnica de propagação de restrição proposta neste trabalho, serão

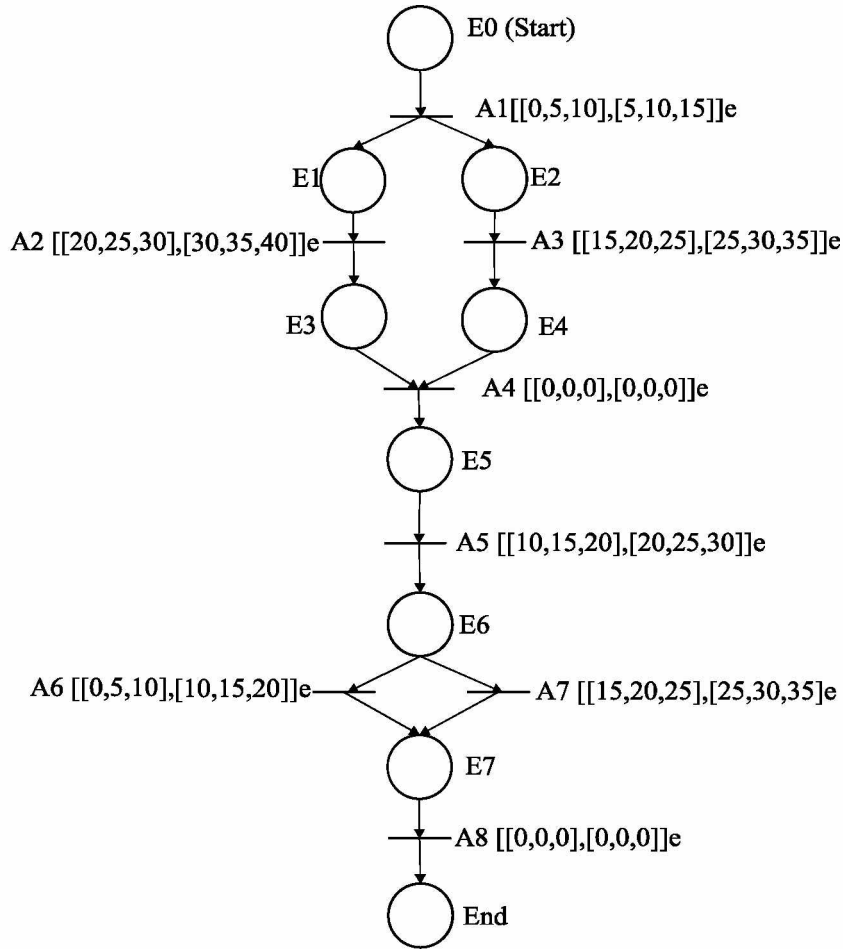
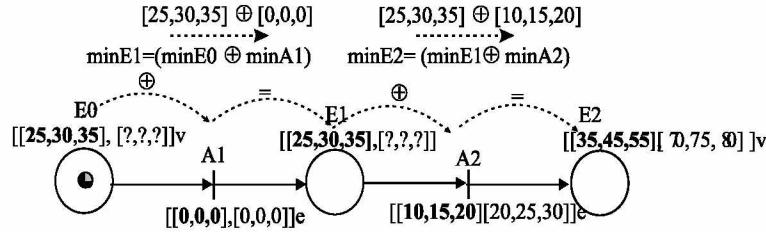
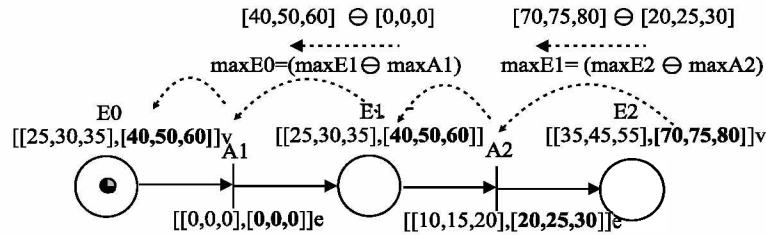


Figura 63 – *Time WorkFlow net fuzzy* com Intervalos Estáticos para o “Processo de Tratamento de Reclamações”.

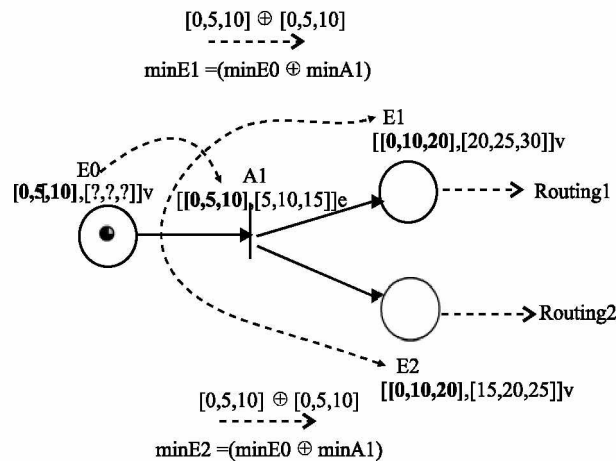
utilizados alguns exemplos que mostram como calcular as bordas *fuzzy* mínimas e máximas dos intervalos de visibilidade. As seguintes notações serão utilizadas nos exemplos:

- a)  $\min EY$  = borda *fuzzy* mínima do intervalo de visibilidade associado a um caso no lugar de espera EY.
- b)  $\max EY$  = borda *fuzzy* máxima do intervalo de visibilidade associado a um caso no lugar de espera EY.
- c)  $\min AY$  = borda *fuzzy* mínima do intervalo estático associado à transição que representa a atividade AY.
- d)  $\max AY$  = borda *fuzzy* máxima do intervalo estático associado à transição que representa a atividade AY.

Por exemplo, a expressão “ $\min E1 = \min E0 \oplus \min A1$ ” significa que a borda *fuzzy* mínima do intervalo de visibilidade associado a um caso específico no lugar *E1* é igual à borda *fuzzy* mínima do intervalo de visibilidade associado com o mesmo caso no lugar *E0* mais a borda *fuzzy* mínima do intervalo estático associado à transição que representa a atividade *A1*.

Figura 64 – Bordas *fuzzy* mínimas - Roteiro Sequencial.Figura 65 – Bordas *fuzzy* máximas - Roteiro Sequencial.

As Figuras 64 e 65 mostram o cálculo das bordas *fuzzy* mínimas e máximas, respectivamente, em um roteiro sequencial. O caso considerado, representado por uma ficha no lugar  $E0$ , pode ser iniciado na data *fuzzy*  $[25,30,35]$  (borda *fuzzy* mínima de intervalo de visibilidade associado ao caso em  $E0$ ). Se a máxima duração permitida do caso (o tempo máximo para completar o processo) é de 45 minutos então, a borda *fuzzy* máxima do intervalo de visibilidade associado ao caso em  $E2$  será  $[25, 30, 35] \oplus [45, 45, 45]$  (máxima duração do caso) =  $[70, 75, 80]$ . A fim de se calcular as outras bordas dos intervalos de visibilidade associados aos lugares de espera, foram consideradas as bordas *fuzzy* mínimas e máximas associadas às transições que representam as atividades como mostram as fórmulas associadas às Figuras 64 e 65.

Figura 66 – Bordas *fuzzy* mínimas - Roteiro Paralelo.

As Figuras 66 e 67 mostram o cálculo das bordas *fuzzy* mínimas e máximas, respectivamente, em um roteiro paralelo. O caso considerado, representado por uma ficha no lugar  $E0$ , pode ser iniciado na data *fuzzy*  $[0, 5, 10]$  (borda *fuzzy* mínima do intervalo de

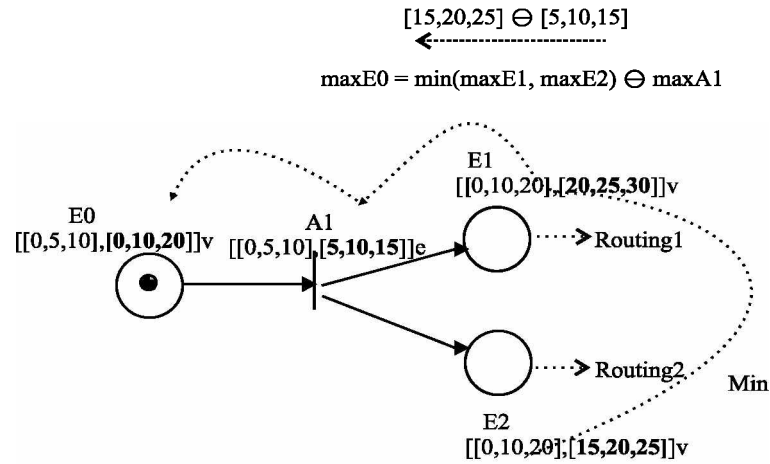


Figura 67 – Bordas fuzzy máximas - Roteiro Paralelo.

visibilidade associado ao caso em  $E0$ ). A máxima duração deste caso é 20 e 15 respectivamente. A borda fuzzy máxima nos lugares  $E1$  e  $E2$  é  $[20, 25, 30]$  e  $[15, 20, 25]$  ((borda fuzzy mínima associada com o caso em  $E0$ )  $\oplus$  (máxima duração do caso)) respectivamente. A borda fuzzy mínima nos lugares  $E1$  e  $E2$  será calculada considerando a borda fuzzy mínima do intervalo de visibilidade no lugar  $E0$  e a borda fuzzy mínima do intervalo estático em  $A1$  como mostra a Figura 66. A borda fuzzy máxima do intervalo de visibilidade associada ao no lugar  $E0$  será calculada considerando o valor mínimo da borda fuzzy máxima nos lugares  $E1$  e  $E2$  ( $\min(\max E0, \max E2)$ ) e a borda fuzzy máxima do intervalo estático associado com a atividade  $A1$ , como mostra a Figura 67.

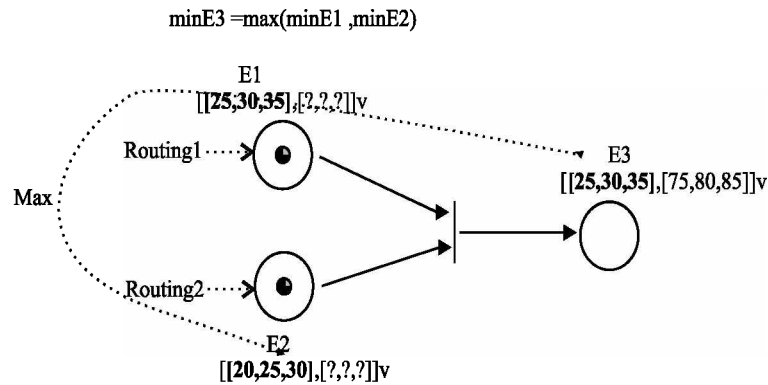
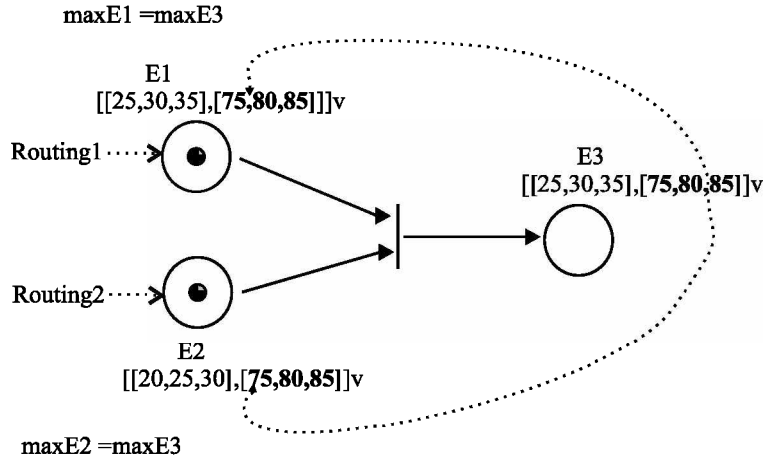
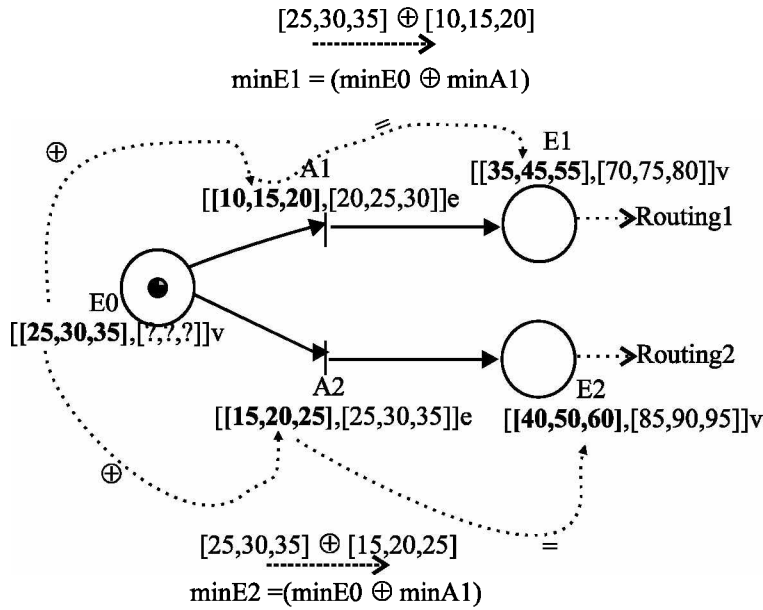


Figura 68 – Bordas fuzzy mínimas - Sincronização.

As Figuras 68 e 69 mostram o cálculo das bordas fuzzy mínimas e máximas, respectivamente, em um ponto de sincronização de dois roteiros paralelos. A data mínima para o início do caso é tal que a borda fuzzy mínima dos intervalos de visibilidade nos lugares  $E1$  e  $E2$  (estes lugares pertencem a diferentes rotas sequenciais e suas bordas fuzzy mínimas não são necessariamente as mesmas) são  $[20, 25, 30]$  e  $[25, 30, 35]$ , respectivamente. A máxima duração para o caso considerado é 50 e, então, a borda fuzzy máxima do intervalo de visibilidade no lugar  $E3$  é  $[75, 80, 85]$  (máximo valor da borda fuzzy mínima do intervalo de visibilidade nos lugares  $E1$  e  $E2$ )  $\oplus$  máxima duração do caso. A borda fuzzy mínima

Figura 69 – Bordas *fuzzy* máximas - Sincronização.

do intervalo de visibilidade no lugar  $E3$  será calculada considerando o valor máximo da borda *fuzzy* mínima dos intervalos de visibilidade nos lugares  $E1$  e  $E2$  ( $\max(\min E1; \min E2)$ ) como mostra a Figura 68. A borda *fuzzy* máxima do intervalo de visibilidade associada com a ficha nos lugares  $E1$  e  $E2$  será igual à borda *fuzzy* máxima do intervalo de visibilidade no lugar  $E3$  (veja Figura 69).

Figura 70 – Bordas *fuzzy* mínimas - Roteiro Alternativo.

As Figuras 70 e 71 mostram o cálculo das bordas *fuzzy* mínimas e máximas, respectivamente, em um roteiro alternativo. O caso considerado, representado por uma ficha no lugar  $E0$ , pode ser iniciado na data *fuzzy*  $[25, 30, 35]$ . A máxima duração deste caso é tal que as bordas *fuzzy* máximas calculadas para os lugares  $E1$  e  $E2$  são  $[70, 75, 80]$  e  $[85, 90, 95]$ , respectivamente. Estes lugares pertencem a diferentes roteiros e suas bordas não são necessariamente iguais. As bordas *fuzzy* mínimas associadas ao caso nos lugares  $E1$  e  $E2$  serão calculadas. Considerando a borda *fuzzy* mínima do intervalo de visibili-

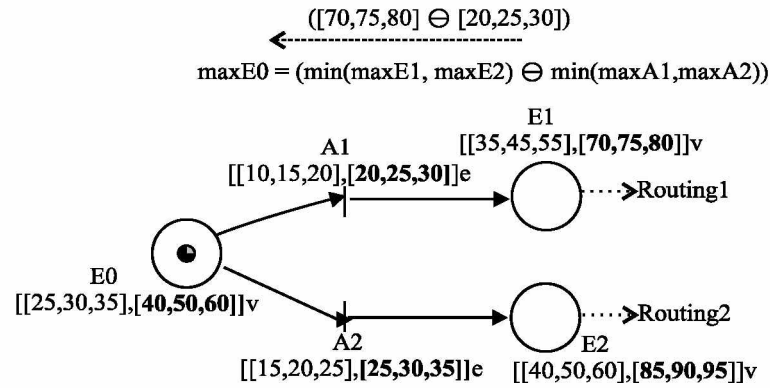


Figura 71 – Bordas fuzzy máximas - Roteiro Alternativo.

dade no lugar  $E0$  e a borda fuzzy mínima do intervalo estático em  $A1$ , como mostra a Figura 70. A borda fuzzy máxima do intervalo de visibilidade associado à ficha no lugar  $E0$  será calculada considerando o valor mínimo da borda fuzzy máxima nos lugares  $E1$  e  $E2$  ( $\min(\max E1; \max E2)$ )  $\ominus$  o valor mínimo da borda fuzzy máxima do intervalo estático associado com as atividades  $A1$  e  $A2$  ( $\min(\max A1; \max A2)$ ), como mostra a Figura 71.

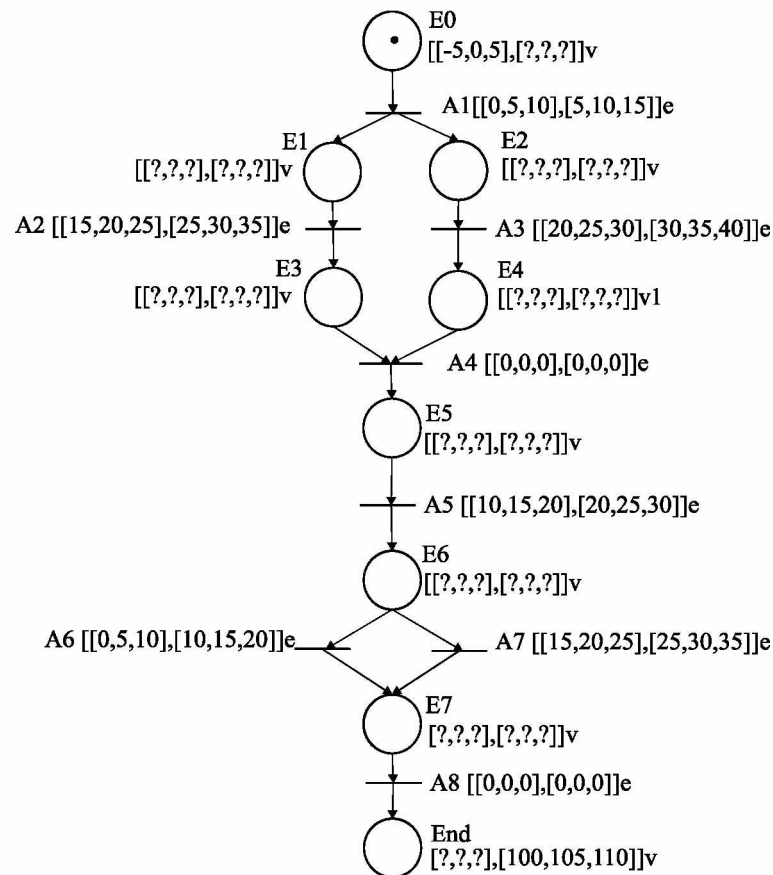


Figura 72 – Intervalos de Visibilidade iniciais - “Processo de Tratamento de Reclamações”.

Na *Workflow net* da Figura 72, um caso representado por uma ficha no lugar  $E0$  é considerado. É possível perceber que apesar de não existir tempo negativo, existe a necessidade de se mostrar os valores negativos para que todos os passos para construir

intervalos de tempo *fuzzy* sejam visualizados. Tem-se que a possibilidade máxima ( $\mu = 1$ ) de início de um caso é zero; então, a variação de chegada será  $-5$  e  $5$ . Logo, a data *fuzzy* para o primeiro caso será  $[-5, 0, 5]$ , que é representada como a borda *fuzzy* mínima do intervalo de visibilidade  $v$  em  $E0$ . Como a duração máxima permitida para o caso é de 105 minutos, a borda *fuzzy* máxima do intervalo de visibilidade associado ao último lugar “End” do processo será igual a  $[-5, 0, 5] \oplus [105, 105, 105] = [100, 105, 110]$ .

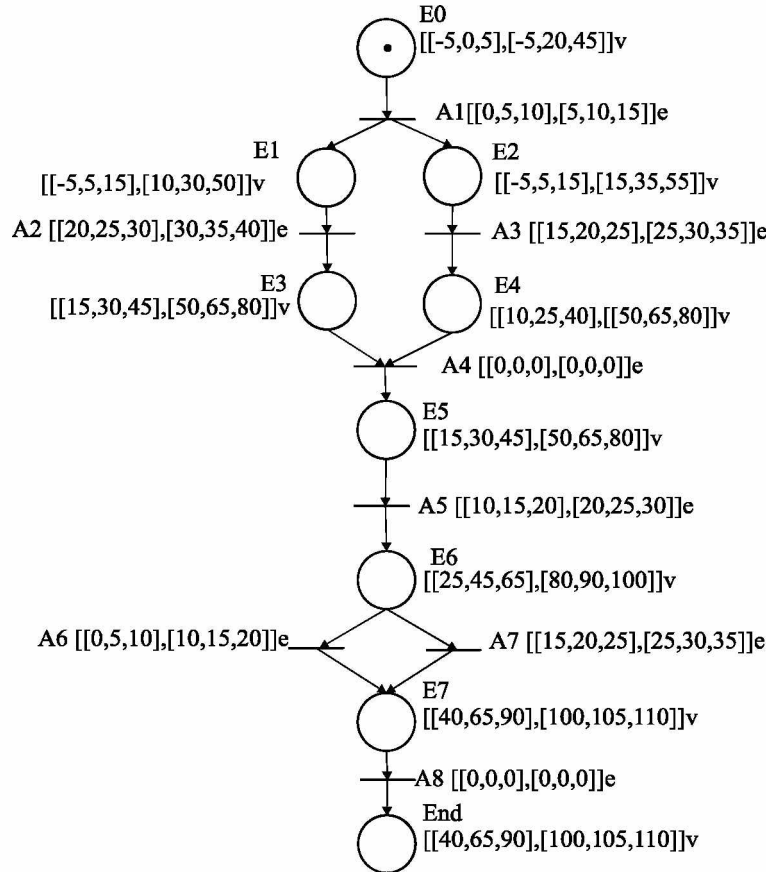


Figura 73 – Modelo para “Processo de Tratamento de Reclamações” após Aplicação dos Mecanismos de Propagação de Restrição.

Na Figura 73, as bordas *fuzzy* mínimas e máximas dos intervalos de visibilidade estimados para os lugares de espera são calculados de acordo com os mecanismos de propagação apresentados neste capítulo.

Através da aplicação da definição 2, todos os intervalos de visibilidade serão transformados em intervalos *fuzzy* únicos. Em particular, com base em tais informações, os cálculos de possibilidade de disparo da *Workflow net fuzzy* poderão ser realizados. É importante salientar que tempos negativos não são permitidos; consequentemente, estes valores devem ser substituídos por 0. A Figura 74 mostra os intervalos de visibilidade *fuzzy* na forma trapezoidal correspondente para o “Processo de Tratamento de Reclamações”. Tal modelo representa então a *Workflow net* do processo com restrições de tempo *fuzzy* derivadas das datas *fuzzy* de início dos casos, das durações *fuzzy* das diversas tare-



fas, e dos prazos máximos aceitáveis de entrega dos casos aos clientes (durações máximas para completar os casos).

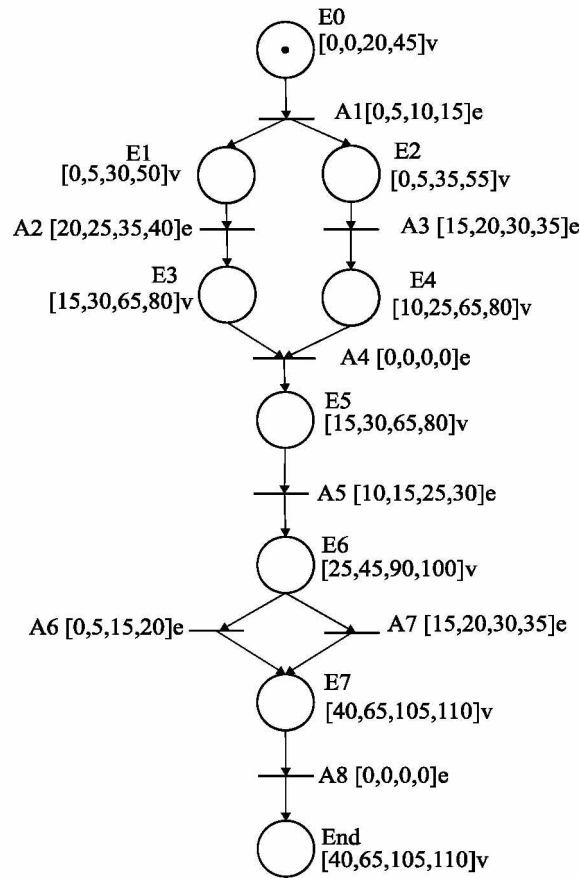


Figura 74 – Modelo para “Processo de Tratamento de Reclamações” com Intervalos de Tempo na forma Trapezoidal.

### 4.3 Jogador de *Workflow net* com Recursos e Restrições de Tempo *Fuzzy*

Como mencionado previamente, uma tarefa está associada a uma transição em uma *Workflow net*, mas a fim de mostrar explicitamente os recursos alocados para realizar uma tarefa, duas transições sequenciais e um lugar para modelar a atividade são usados (WANG; TEPFENHART; ROSCA, 2009).

A Figura 75 mostra a alocação de recursos *fuzzy* e os intervalos estáticos *fuzzy* (intervalos de durações *fuzzy*) associados aos locais de atividade do processo e os intervalos de visibilidade *fuzzy* (intervalos de datas imprecisas) associados com os lugares de espera (condições do processo). É importante notar que não há nenhuma restrição de tempo sobre os recursos. Os limites *fuzzy* foram calculados usando a técnica de propagação apresentada na seção 4.2.

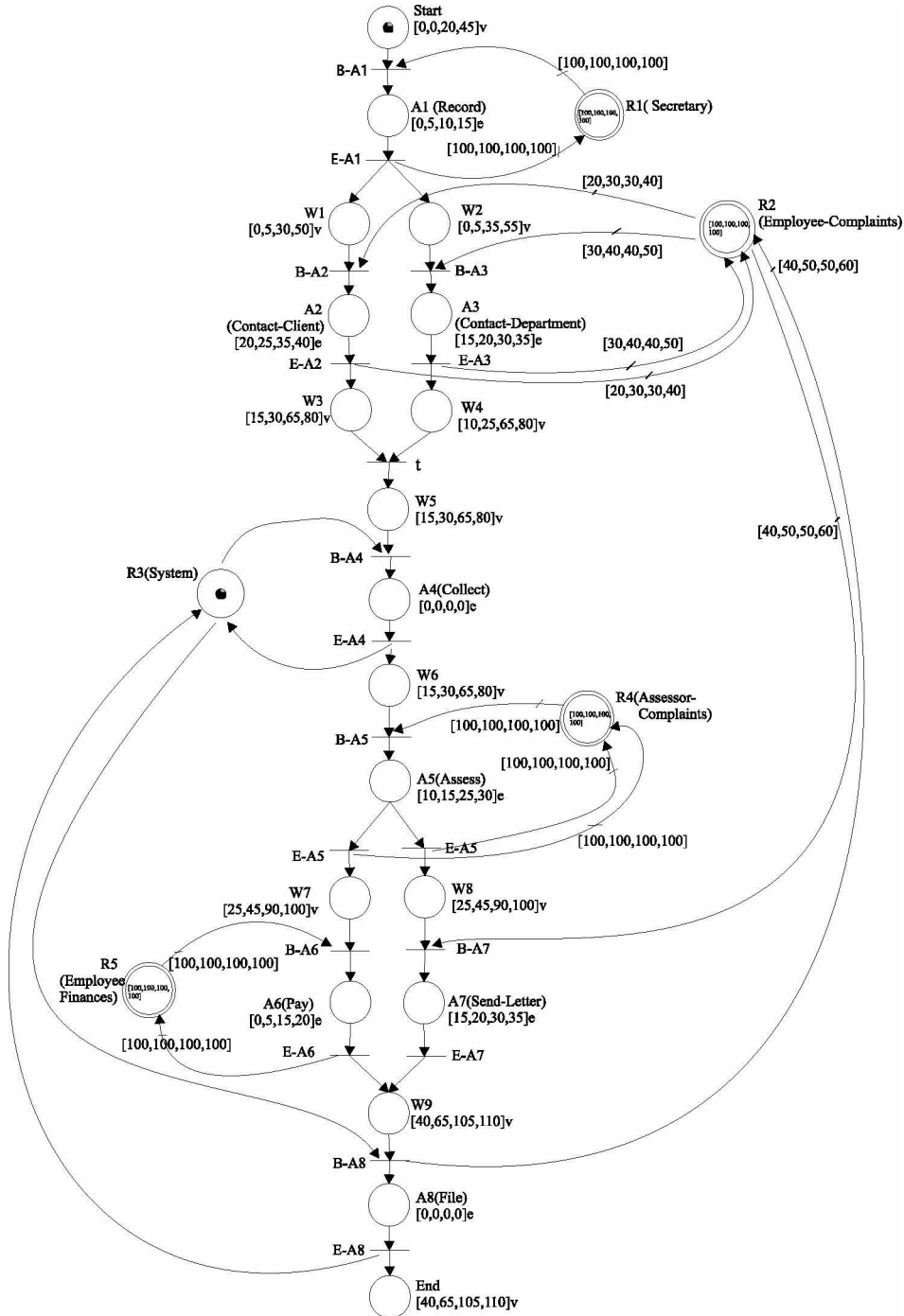


Figura 75 – *Workflow net* com Restrições de Tempo na Forma Trapezoidal.

Em uma rede de Petri ordinária, uma transição  $t$  é habilitada se e somente se para todas as entradas dos lugares  $p$  à transição  $t$ ,  $M(p) \geq Pre(p, t)$ . Isto é, se o número de fichas de cada lugar de entrada é maior ou igual ao peso associado aos arcos que conectam os lugares de entrada da transição  $t$ . No caso de uma alocação híbrida *fuzzy* em que o tempo também deve ser levado em consideração, a definição da sensibilização de uma transição é baseada no conceito de possibilidade conjunta definida pela equação (13) na página 41 apresentada novamente.

Dados dois conjuntos de referência  $X$  e  $Y$ , DuBois (1997) define a possibilidade con-

junta como limite superior, tal que:

$$\forall x \in X \quad \forall y \in Y \quad \pi(x, y) = \min(\pi_X(x), \pi_Y(y))$$

quando os conjuntos referência são não-interativos (o valor de  $x \in X$  não tem influência sobre o valor de  $y \in Y$ , e vice versa).

Quando se trata de sensibilização de transição relacionada à data de chegada da ficha, tem-se que se a ficha aparece em um local  $p$  na data  $\delta$  e se o intervalo de visibilidade é dado por  $[a1, a2, a3, a4]$ , então esta ficha somente poderá ser utilizada para o disparo de uma transição mais cedo na data  $a1$  e mais tarde na data  $a4$ , caso o recurso envolvido na atividade correspondente fique disponível dentro do intervalo *fuzzy* de visibilidade.

Portanto, para disparar uma transição, é necessário que a data de chegada da ficha no local de entrada da transição pertença ao intervalo de visibilidade *fuzzy* associado com o local de entrada da transição e exista a disponibilidade de recursos necessária para realizar a atividade. Para avaliar a disponibilidade de recurso e a visibilidade dos casos envolvidos simultaneamente, a possibilidade conjunta deve ser calculada.

Formalmente, tem-se que:

**Definição 18** *Em uma Workflow net com recursos definidos pelo conjunto referência  $X$  e restrições de tempo fuzzy definidas pelo conjunto referência  $Y$ , uma transição  $t$  está habilitada se e somente se:*

$$\forall x \in X \quad \forall y \in Y \quad \pi(x, y) = \min(\pi_X(x), \pi_Y(y)) > 0 \quad (25)$$

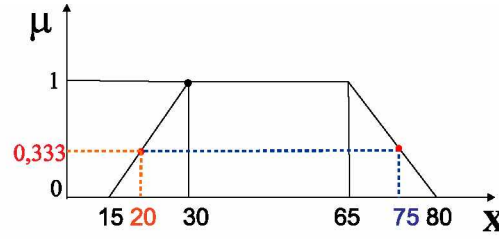
onde:

$\pi_X(x)$  representa a possibilidade associada à alocação de recurso na transição  $t_i$  e deve ser calculado conforme equação (7),  $\Pi(Pre_{FCR}(p, t) \leq M_{FCR}(p)) > 0$

$\pi_Y(y)$  representa a possibilidade associada à data de chegada de um caso ( $\delta$ ) no lugar de entrada da transição  $t_i$  e deve ser calculada pela equação (5),  $\Pi(\delta \in Y) > 0$

Se uma transição tem  $n$  lugares de entrada e que têm, cada um, uma ou várias fichas, então o intervalo de sensibilização desta transição  $[a1, a2, a3, a4]$  é obtido escolhendo-se para cada um dos  $n$  lugares uma ficha e o seu intervalo de visibilidade. Não existe nenhuma restrição de tempo para os recursos e, como consequência, o intervalo de tempo que habilita uma transição será simplesmente igual ao intervalo de visibilidade associado com o caso a ser tratado pela transição correspondente. Por exemplo, sabendo que o intervalo de visibilidade ligado ao caso representado por uma ficha no lugar  $W1$  é igual a  $[0, 5, 30, 50]v$ , o intervalo de tempo que habilita a transição  $B - A2$  será de  $[0, 5, 30, 50]$  também.

Por exemplo, se uma ficha é produzida no lugar  $W6$  da Figura 75 na data  $\delta = 30$ , considerando a distribuição possibilística mostrada na Figura 76, o cálculo da possibilidade de disparo da transição  $B - A5$  será igual a  $\Pi_{B-A5} = 1$  (de acordo com a equação (5)).

Figura 76 – Distribuição de possibilidade de  $W6$ .

Sabendo que o intervalo estático associado a  $A5$  é igual a  $[10, 15, 25, 30]$  e a data de início é  $[30, 30, 30, 30]$  então, o intervalo de sensibilização desta transição será  $[30, 30, 30, 30] \oplus [10, 15, 25, 30]$  (intervalo estático associado a  $A5$ ) =  $[40, 45, 55, 60]$ . Se a ficha no lugar  $W6$  é produzida mais cedo na data  $\delta = 20$ , por exemplo, a possibilidade de disparo de transição  $B - A5$  será  $\Pi_{B-A5} = 0,333$  e o intervalo de sensibilização será  $[20, 20, 20, 20] \oplus [10, 15, 25, 30]$  (intervalo estático associado a  $A5$ ) =  $[30, 35, 45, 50]$ . Neste caso, o disparo poderia, eventualmente, ser adiado até chegar a uma data que corresponde a uma possibilidade igual a  $\Pi_{B-A5} = 1$ . Finalmente, se a ficha no lugar  $W6$  é produzida na data  $\delta = 75$ , a possibilidade de disparo de transição  $B - A5$  também será igual a  $\Pi_{B-A5} = 0,333$ , mas com um significado diferente. Esta situação corresponderá a um caso em que algumas das atividades anteriores no processo foram atrasadas e seu intervalo de sensibilização será  $[75, 75, 75, 75] \oplus [10, 15, 25, 30]$  (intervalo estático associado a  $A5$ ) =  $[85, 90, 100, 105]$ . Será importante, em seguida, disparar imediatamente a transição  $B - A5$  correspondente ao início da próxima atividade e informar o recurso responsável por executar esta atividade em atraso.

Com o objetivo de entender o mecanismo de disparo de uma transição em uma *Workflow net* com recursos *fuzzy* e tempo *fuzzy*, será considerado um fragmento do Processo de Tratamento de Reclamações, como mostra a Figura 77.

a) No intervalo *fuzzy*  $[0, 0, 20, 45]$ :

- Em algum momento o caso no lugar *start* pode se tornar disponível para ser tratado pelo recurso  $R1$ . Foi escolhido o disparo da transição  $B-A1$  na data 0 para alcançar o maior valor possível (situação normal para tratar o caso) quando considerado a possibilidade conjunta pois  $\pi_Y(0) = 1$  (possibilidade do tempo igual a 1 quando *data* = 0) e  $(\pi_X(x) = \Pi(a \leq b) = 1$  (possibilidade de recurso disponível). Depois do disparo da transição, uma ficha é produzida no lugar  $A1$  com um intervalo de sensibilização igual a  $[0, 0, 0, 0] \oplus [0, 5, 10, 15]_s = [0, 5, 10, 15]_v$ .

b) No intervalo *fuzzy*  $[0, 5, 10, 15]$ :

- se a atividade associada a  $A1$  é finalizada na data 5, a ficha torna-se disponível em  $A1$ , a transição  $E-A1$  é disparada porque a possibilidade conjunta é igual

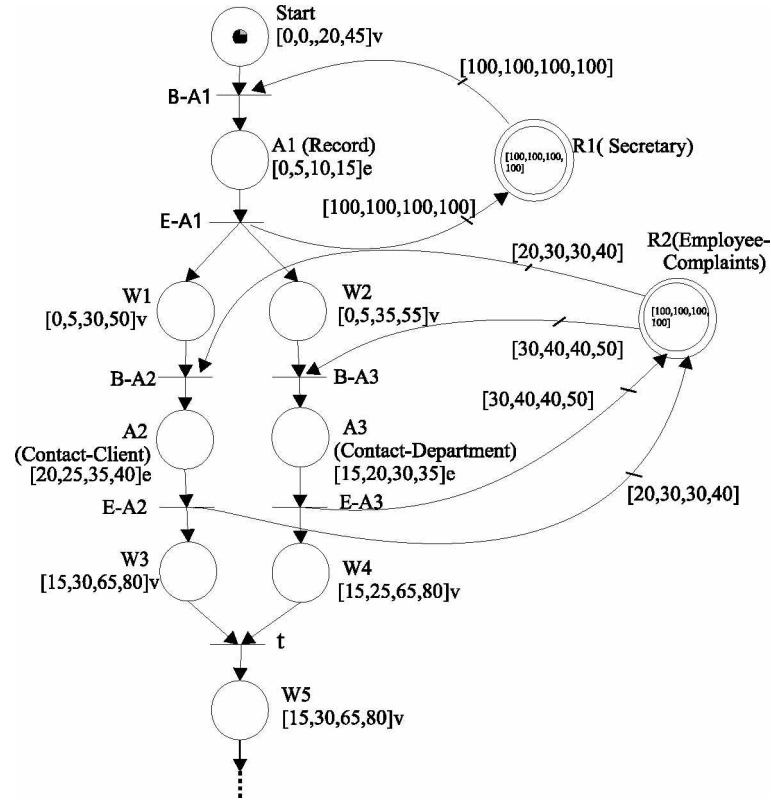


Figura 77 – Fragmento da *Workflow net* - “Processo de Tratamento de Reclamações”.

a 1, pois a possibilidade da chegada da ficha é igual a 1 quando a data é igual a 5) e a possibilidade da disponibilidade do recurso também é igual a 1, então o recurso é devolvido para  $R1$ . Ao mesmo tempo, fichas são produzidas em  $W1$  e  $W2$ . Para disparar  $B-A2$  e  $B-A3$  é necessário avaliar o cálculo da possibilidade da chegada da ficha e a disponibilidade do recurso através da equação (18). Para  $B-A2$ , a possibilidade conjunta é igual a 1 já que a possibilidade do tempo é igual a 1 quando  $data = 5$ ) e a possibilidade de recurso disponível também é igual a 1.

Da mesma maneira, para  $B-A3$ , a possibilidade conjunta é igual a 1, pois para a data igual a 5 a possibilidade é igual a 1 e a possibilidade de recurso disponível também é 1. Então, as transições  $B-A2$  e  $B-A3$  são disparadas e uma ficha é produzida em  $A2$  com intervalo de sensibilização  $[25, 30, 40, 45]_v$  e outra em  $A3$  com intervalo de sensibilização  $[20, 25, 35, 40]_v$ . Neste momento, a disponibilidade do recurso é  $R2 = [10, 30, 30, 50]$ .

c) No intervalo *fuzzy*  $[20, 25, 35, 40]$

- se a atividade  $A3$  é finalizada na data 30, a ficha torna-se disponível em  $A3$ , então a transição  $E-A3$  é disparada, pois possibilidade da chegada da ficha é igual a 1 quando  $data = 30$ ) e a possibilidade do recurso disponível também é 1, assim, o recurso é devolvido a  $R2$ . Neste momento,  $R2 =$

[60, 70, 70, 80]. Uma ficha é produzida em  $W4$  e o intervalo de sensibilização será [50, 55, 60, 70];

d) No intervalo *fuzzy* [30, 35, 45, 50]

- se a atividade  $A2$  é finalizada na data 35, a ficha torna-se disponível em  $A2$ , então a transição  $E-A2$  é disparada e o recurso é devolvido a  $R2$ . Neste momento,  $R2 = [100, 100, 100, 100]$ . Uma ficha é produzida em  $W3$  com intervalo de sensibilização igual a [60, 65, 75, 80]. A transição  $t$  é disparada e uma ficha é produzida em  $W5$ .

### 4.3.1 Implementação de uma *Workflow net* com Recursos e Restrições de Tempo *Fuzzy* baseada na Linguagem CPN Tools

A implementação do modelo mostrado na Figura 75 foi construída usando CPN Hierárquica e é apresentada nas Figuras 78, 79 e 80.

Cada atividade, aqui representada em uma “*abstract transition*”, só é habilitada caso o recurso esteja disponível, representado explicitamente na subpágina correspondente. Também deve-se ressaltar que o modelo do “Processo de Tratamento de Reclamações” possui um ponto de sincronização visto que o processo só pode avançar depois que o cliente e o departamento afetado forem contactados. Na implementação isto é explicitado pela transição *transition* e a condição `if (#1(tp1)) > (#1(tp2)) then tp1 else tp2` que verifica o tempo em cada ficha e prossegue com o maior tempo do caso tratado no momento do disparo.

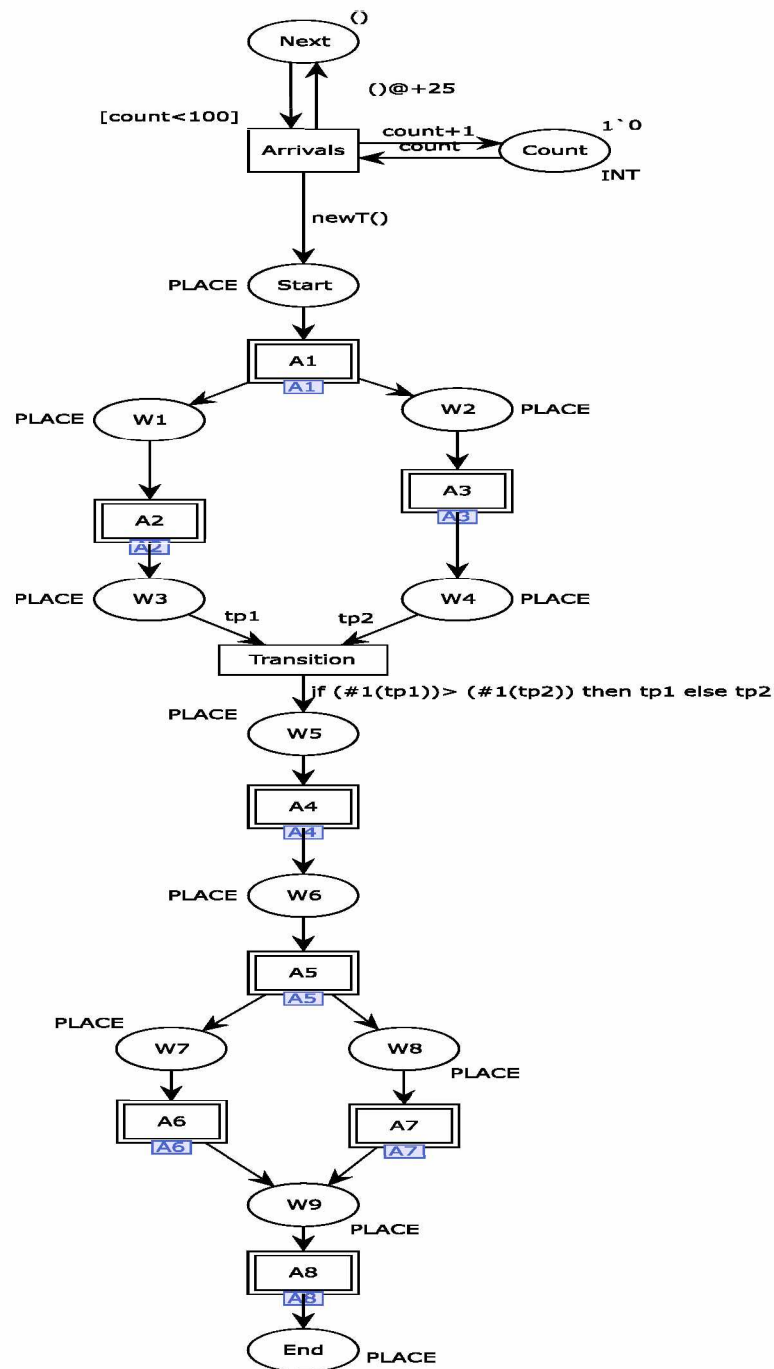
A Tabela 11 apresenta as declarações utilizadas no modelo.

colset t = REAL;
colset a = INT;
colset TI = INT;
colset RESOURCE = UNIT;
colset RESOURCEF = product a * a * a * a ;
colset W = product t * t * t * t;
colset PLACE = product TI*TI timed;
colset CHOICE = with pay  letter;

Tabela 11 – Declarações para o modelo CPN com Recursos e Restrições de Tempo *Fuzzy*

Conforme observado na Tabela 11, o modelo define o seguinte conjunto de Cores (*colset*).

- t: define um valor real que será associado à duração da atividade
- a: define um valor inteiro que será associado à disponibilidade do recurso.

Figura 78 – Modelo da *Workflow net*.

- c) TI: define um valor inteiro que será associado ao *lugar*.
- d) RESOURCE: define um recurso discreto.
- e) RESOURCEF: define o conjunto *fuzzy* que representa a disponibilidade de um recurso.
- f) W : define o intervalo de tempo de uma atividade como um conjunto *fuzzy*.
- g) PLACE: representa os lugares do Processo de Tratamento de Reclamações, mostra o tempo de duração do caso e armazena a data de entrada do caso no processo.

- h) CHOICE: define a escolha a ser feita - se um pagamento será efetuado ou se uma carta será enviada.

A Tabela 12 apresenta as variáveis utilizadas no modelo. Variáveis CPN tem a mesma característica de variáveis em qualquer linguagem de programação, porém os tipos de dados são as Cores (*colset*).

var t, upd_t, tp1,tp2: PLACE;
var proctime, nt, tt, ti: TI;
var r3: RESOURCE;
var r, r1, r2, r4, r5, RBA:RESOURCEF;
var proctime, tt, ti: TI;
var c: CHOICE;
var c2,c3,c7,count: INT;

Tabela 12 – Variáveis para o modelo CPN com Recursos e Restrições de Tempo *Fuzzy*

A Tabela 13 apresenta as constantes que foram definidas para o modelo. As constantes A1, A2, A3, A5, A6, A7 definem o intervalo de duração de cada atividade. As constantes RBA1, RBA2, RBA3, RBA5, RBA6, RBA7 definem os pesos associados com os arcos para o cálculo da disponibilidade do recurso. É importante salientar que as atividades A4 e A8 tem alocação de recurso discreto e execução imediata; sendo assim, seu tempo de execução é igual a zero, não sendo necessária a definição da duração da atividade e nem do peso no arco.

val RBA1 = (100,100,100,100 ): RESOURCEF;
val RBA2 = (20,30,30,40):RESOURCEF;
val RBA3 = (30,40,40,50):RESOURCEF;
val RBA5 = (100,100,100,100 ): RESOURCEF;
val RBA6 = (100,100,100,100):RESOURCEF;
val RBA7 = (40,50,50,60):RESOURCEF;
val A1 = (0.0,5.0,10.0,15.0):W;
val A2 = (20.0,25.0,35.0,40.0):W;
val A3 = (15.0,20.0, 30.0,35.0):W;
val A5 = (10.0,15.0,25.0,30.0):W;
val A6 = (0.0,5.0,15.0,20.0):W;
val A7 = (15.0,20.0,30.0,35.0):W;

Tabela 13 – Constantes para o modelo CPN com Recursos e Restrições de Tempo *Fuzzy*

As subpáginas para as transições A1, A2, A3 e A4 são mostradas na Figura 79. Nessas subpáginas, cada atividade é definida como uma transição de entrada para verificar se existe recurso disponível e se o tempo da ficha pertence ao intervalo de visibilidade (possibilidade conjunta). Se a transição é disparada, então o tempo de execução é atualizado, somando o valor da duração da atividade ao tempo que está na ficha que representa o caso em tratamento. Quando a atividade termina, o recurso é devolvido.



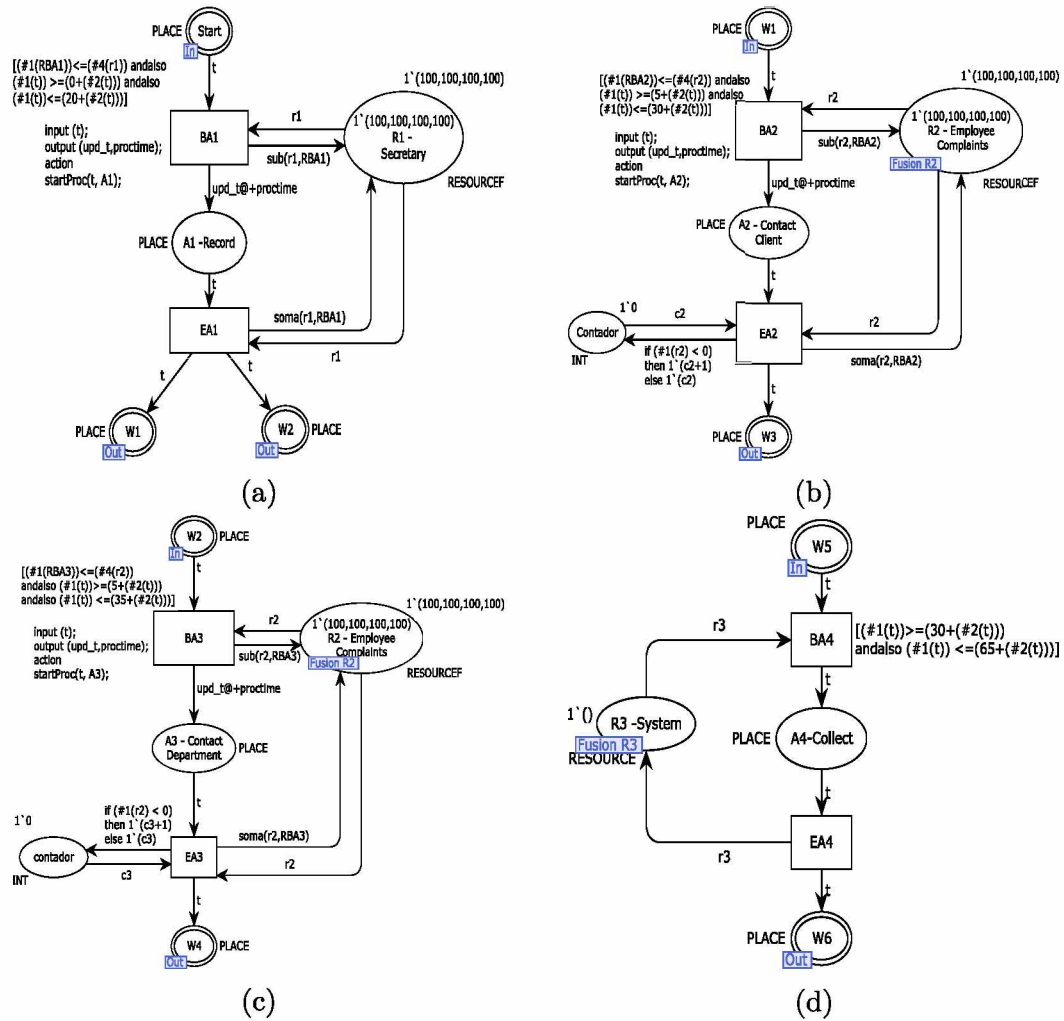


Figura 79 – Atividades A1, A2, A3 e A4 do Processo de Tratamento de Reclamações - Possibilidade Conjunta.

As subpáginas para as transições A5, A6, A7 e A8 são mostradas na Figura 80. Basicamente, o comportamento destas atividades é o mesmo que o apresentado nas atividades anteriores, isto é, uma transição só é disparada quando existe disponibilidade de recurso e se o tempo da ficha pertence ao intervalo de visibilidade associado ao lugar. A única variação está na definição da atividade A5. Como o roteiro é alternativo, é necessário que uma decisão seja tomada - ou um pagamento é feito, ou uma carta é enviada. Torna-se indispensável definir uma condição de guarda que verifica qual foi a decisão tomada, ou seja, para habilitar a transição *EA5pay*, a decisão deve corresponder à realização do pagamento, satisfazendo a condição de guarda  $[c = \text{pay}]$ . Caso a decisão tenha sido enviar uma carta, a transição *EA5letter* será habilitada, pois a condição de guarda  $[c = \text{letter}]$  foi satisfeita.

O conceito de *fusion place* foi utilizado nos recursos *R2* e *R3* a fim de manter a integridade do modelo e a característica de compartilhamento dos recursos. Desta forma, pode-se garantir que uma alteração no recurso *R2* ou *R3* em qualquer subpágina é atualizado instantaneamente nas outras subpáginas. A *fusion tag* definida para o recurso *R2*

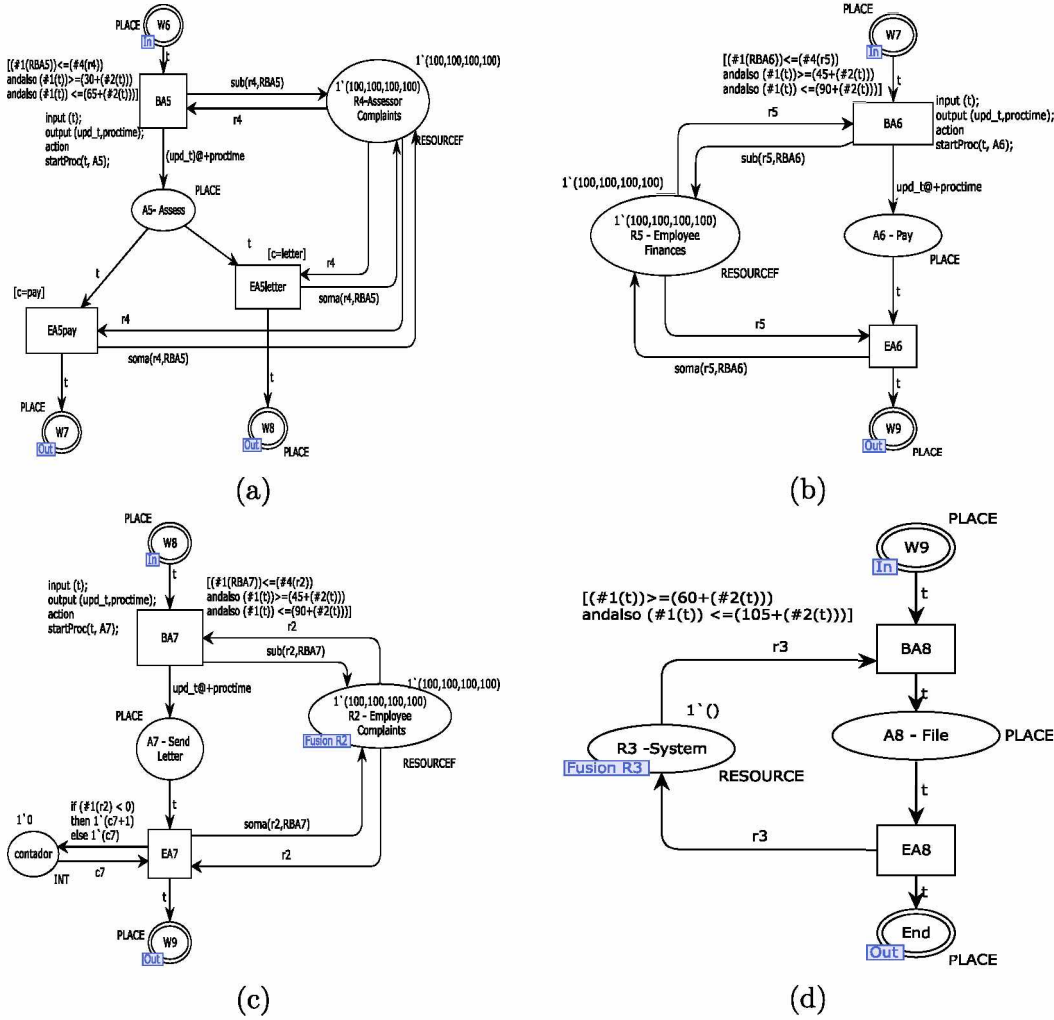


Figura 80 – Atividades A5, A6, A7 e A8 do Processo de Tratamento de Reclamações - Possibilidade Conjunta.

é *fusion R2* e para o recurso *R3* é *fusion R3*.

### 4.3.2 Jogador com Regras de Disparo baseadas na Noção de Possibilidade Conjunta - Distribuição Uniforme

No contexto deste capítulo, o jogador de rede de Petri permite a simulação de uma rede de Petri sendo as transições habilitadas por uma possibilidade conjunta que envolve tanto a disponibilidade de um recurso quanto a visibilidade das datas de realização esperadas para o caso em tratamento.

A característica de tal jogador, em relação ao jogador apresentado no capítulo 3, é que as transições atividades não são necessariamente disparadas imediatamente no momento em que o recurso necessário à realização da atividade se torna disponível. Dependendo das datas estimadas de início da atividade relacionada com o caso em tratamento, um recurso poderá ser usado para uma atividade associada a um outro caso mais prioritário. Em particular, na prática, em atividades de escritórios é assim que são realizadas as

atividades. Quando um trabalho é repassado para um funcionário, o funcionário olha o prazo. Dependendo, ele vai iniciar o trabalho com urgência ou deixar para depois, aproveitando o tempo de folga para tratar o mais urgente. Um funcionário com pouca capacidade de planejamento vai executar as tarefas repassadas a ele por ordem de chegada, sem pensar muito, mas isso não vai necessariamente produzir um efeito bom.

Para demonstrar a abordagem proposta, um processo de simulação, utilizando o simulador CPN Tools (RATZER et al., 2003) foi executado. O jogador foi simulado em CPN Tools usando a possibilidade conjunta para habilitar as transições. A possibilidade conjunta de tempo e disponibilidade de recursos foi definida como uma condição de guarda nas transições de entrada das atividades.

Para adicionar expressividade ao modelo, visando alcançar a correta simulação do jogador proposto, foram definidas algumas funções.

- a) **mda**: A cada atividade fica associada uma duração aleatória que pertence a um intervalo de tempo uniformemente distribuído. Para isto, uma função denominada *mda* foi definida para que durante a simulação, a distribuição estatística seja uniforme e a probabilidade de cada instante seja a mesma dentro de um intervalo definido. Deve-se notar que, neste caso, os valores utilizados compõem o núcleo do intervalo (quando a duração da atividade possui possibilidade igual a 1).

```
1. fun mda(Z:W) = round(uniform((#2(Z)),(#3(Z))));
```

- a) **intTime**: a função intTime é declarada para retornar o tempo atual como um inteiro.

```
1. fun intTime() = IntInf.toInt(time()): int;
```

- a) **startProc**: a função startProc é responsável por atualizar o tempo de duração de cada caso. Inicialmente a duração da atividade é calculada pela chamada a função mda. O tempo corrente é acrescido à duração da atividade usando a função da biblioteca Base SML ModelTime.add, e então a variável tt (que armazena a duração total do caso) é atualizada.

```
1. fun startProc ((tt,ti):PLACE, Z:W) =
2.   let
3.     val proc_time = mda(Z)
4.     val time_stamp =
5.       ModelTime.add(time(),ModelTime.fromInt(proc_time))
6.   in
7.     ((new_tt, ti), proc_time)
```

8. end

- a) **conta**: a função conta é responsável por contabilizar quantas vezes o recurso *R2* foi sobrecarregado em cada atividade (A2, A3 e A7). Assim, será possível monitorar a quantidade de vezes que o recurso foi sobrecarregado.

```
1. fun conta((s1,s2,s3,s4):RESOURCEF, x: INT) =
2.   if s1< 0
3.   then x+1
4.   else x
```

- a) **NewT**: a função NewT é responsável por gerar novos casos. Cada caso tem o tempo de duração e o tempo inicial.

```
1. fun NewT() = (intTime(),intTime());
```

A fim de padronizar os resultados, 100 casos serão gerados em intervalos regulares de 25 minutos. A duração dos intervalos foi definida baseada em experimentos de forma a mostrar, da melhor maneira, o funcionamento dos modelos. Rozinat et al. (2008) mostram que quanto maior o número de replicações, maior a precisão do resultado. No entanto, é necessário decidir um limite aceitável de replicações. Sendo assim, 20 experimentos de simulação (replicações) foram considerados para cada mecanismo de alocação de recursos com um horizonte de simulação de 1600 eventos <sup>1</sup>. Com esta quantidade de repetições a dispersão obtida foi pequena, menor que 5% em relação à ordem de grandeza das métricas, considerando o intervalo de confiança de 95%.

O simulador CPN Tool permite a criação de monitores para coleta de dados de simulação. Utilizando-se o *Data Collector Monitoring Functions*, foram escritas 9 funções definidas em código CPN-ML para monitorar a duração das atividades e também a sobrecarga do recurso *R2*. Utilizando o *Count transition occurrence data collector* foi possível contar quantos casos respeitaram os prazos e foram concluídos corretamente.

- a) **Duracao A1** - a função deste monitor é de monitorar o tempo de execução da atividade A1.
- b) **Duracao A2** - a função deste monitor é de monitorar o tempo de execução da atividade A2.
- c) **Duracao A3** - a função deste monitor é de monitorar o tempo de execução da atividade A3.

<sup>1</sup> valor suficiente para simular os 100 casos gerados

- d) **Duracao A5** - a função deste monitor é de monitorar o tempo de execução da atividade A5.
- e) **Duracao A6** - a função deste monitor é de monitorar o tempo de execução da atividade A6.
- f) **Duracao A7** - a função deste monitor é de monitorar o tempo de execução da atividade A7.
- a) **Sobrecarga de R2 em A2** - a função deste monitor é monitorar quantas vezes o recurso foi sobrecarregado quando a atividade A2 foi executada.
- b) **Sobrecarga de R2 em A3** - a função deste monitor é monitorar quantas vezes o recurso foi sobrecarregado quando a atividade A3 foi executada.
- c) **Sobrecarga de R2 em A7** - a função deste monitor é monitorar quantas vezes o recurso foi sobrecarregado quando a atividade A7 foi executada.

### 4.3.3 Resultados da Simulação - Distribuição Uniforme

A Tabela 14 apresenta o número de vezes que o recurso compartilhado *R2* foi sobrecarregado. Considerando 100 casos, o recurso *R2* foi sobrecarregado em média:

- a) 46 vezes para a atividade A2,
- b) 39 vezes para a atividade A3,
- c) 23 vezes para a atividade A7.

Sobrecarga de R2 em A2				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Min_id	0,00	0,00	0	0
Max_id	93,85	0,70	91	97
Média_id	46,87	0,51	45,08	48,98
Sobrecarga de R2 em A3				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Min_id	0,00	0,00	0	0
Max_id	79,40	2,10	71	85
Média_id	39,58	1,23	34,09	43,51
Sobrecarga de R2 em A7				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Min_id	0,00	0,00	0	0
Max_id	47,25	2,07	38	54
Média_id	23,63	1,08	18,55	27,26

Tabela 14 – Estatística Sobrecarga dos Recursos *Fuzzy*

A Tabela 15 apresenta os valores definidos como sendo ideais para a realização de cada atividade.

	Média	Mínimo	Máximo
Duração A1	7,5	5	10
Duração A2	30	25	35
Duração A3	25	20	30
Duração A5	20	15	25
Duração A6	10	5	15
Duração A7	25	20	30

Tabela 15 – Valores Ideais (definidos) para Duração das Atividades

A Tabela 16 apresenta os valores estatísticos da simulação do jogador com disparo das transições baseado na possibilidade conjunta. Percebe-se que a duração média das atividades se manteve muito próximo dos valores considerados ideais para a realização das atividades; por exemplo, a média da duração ideal para a atividade A6 é 10. O valor médio obtido na simulação foi  $10,23 \pm 0,35$ .

A Tabela 17 apresenta o número de casos que foram finalizados. Isto significa que os casos respeitaram o prazo para realização de cada atividade e também a disponibilidade de recursos. É importante notar que apesar do grande número de casos finalizados (quase 100% -  $99,55 \pm 0,41$ ), a quantidade de vezes que o recurso R2 foi sobrecarregado é muito alto, como se vê na Tabela 14. Por exemplo, na atividade A2, o recurso foi sobrecarregado em média 46 vezes (quase a metade do tempo).<sup>2</sup>

O relatório completo pode ser encontrado no Apêndice A.2.

#### 4.3.4 Jogador com Regras de Disparo baseadas na Noção de Possibilidade Conjunta - Distribuição Exponencial

Em todos os experimentos de simulação, até o presente momento desta tese, os casos foram gerados por um intervalo regular de 25 minutos (1 novo caso a cada 25 minutos) e foi utilizada a distribuição uniforme para definir a duração das atividades. A distribuição uniforme garante que, considerando um intervalo de tempo, cada instante dentro do intervalo terá a mesma probabilidade de acontecer; sendo assim, como era necessário comparar os diferentes modelos, foi imprescindível garantir que as durações das atividades também pertencessem ao intervalo de tempo estático associado a cada atividade do processo. No entanto, na prática, é impossível garantir que uma determinada tarefa seja realizada com duração pertencente a um intervalo de tempo fixado; em particular, quando funcionários humanos ficam envolvidos no processo, os tempos de operação previstos nem sempre podem ser garantidos. Por exemplo, a experiência do funcionário, o seu estado emocional, o acúmulo de tarefas pode ter um efeito significativo sobre o tempo de execução de uma atividade. É de fato uma situação muito comum ter funcionários que atrasam além de prazos estabelecidos as tarefas que recebem. Mesmo assim, os fluxos de atividades dos

<sup>2</sup>  $46,87 \pm 0,51$  vezes

<b>Duracao A1</b>				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Min_id	5,00	0,00	5	5
Max_id	10,00	0,00	10	10
Média_id	7,50	0,06	7,31	7,76
<b>Duracao A2</b>				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Min_id	25,00	0,00	25	25
Max_id	35,00	0,00	35	35
Média_id	30,01	0,15	29,34	30,52
<b>Duracao A3</b>				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Min_id	20,00	0,00	20	20
Max_id	30,00	0,00	30	30
Média_id	25,11	0,12	24,79	25,75
<b>Duracao A5</b>				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Min_id	15,00	0,00	15	15
Max_id	25,00	0,00	25	25
Média_id	19,92	0,11	19,50	20,31
<b>Duracao A6</b>				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Min_id	5,00	0,00	5	5
Max_id	14,95	0,10	14	15
Média_id	9,98	0,14	9,09	10,41
<b>Duracao A7</b>				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Min_id	20,15	0,17	20	21
Max_id	30,00	0,00	30	30
Média_id	24,82	0,44	23,34	26,64

Tabela 16 – Estatística para Duração das Atividades - Recursos e Tempo *Fuzzy*

<b>Casos Finalizados</b>				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Quantidade	99,55	0,41	97	100

Tabela 17 – Estatística - Quantidade de Casos Finalizados

processos não são interrompidos. O que se espera é que o atraso causado por alguns será compensado por outros até o processo ser finalizado.

Sabendo da possibilidade de variação do tempo de execução de cada atividade, decidiu-se então, depois de comprovar que o modelo desenvolvido que utiliza a possibilidade conjunta pode gerenciar o tempo e a alocação de recursos, usar a distribuição exponencial para descrever a duração de cada atividade e para produzir também o fluxo de entrada

dos casos tratados no processo.

Segundo Anderson e Sweeney (2011), uma distribuição exponencial de probabilidade é muito útil para descrever o tempo que se leva para completar uma tarefa; por exemplo, descrever o tempo entre a chegada de um motoboy à casa de um cliente, o tempo exigido para alguma tarefa dentro de uma fábrica, podendo assim ser aplicado em qualquer área aonde exista a necessidade de identificar tempos percorridos aleatórios.

Algumas funções usadas no jogador apresentado na Seção 4.3 foram modificadas a fim de garantir que as durações das atividades sejam calculadas seguindo a distribuição exponencial.

As funções alteradas são:

- a) **mda**: a função denominada *mda* foi definida para calcular a média da duração da atividade - considerando somente o intervalo onde a possibilidade é igual a 1, isto é, sabendo que  $[a1, a2, a3, a4]$  representa o intervalo *fuzzy*, a média será calculada somente com os valores dentro do intervalo  $[a2, a3]$ .

```
1. fun mda(Z:W) = (((#2(Z))+(#3(Z))) div 2);
```

- a) **expTime**: a função *expTime* é declarada para calcular a duração exponencial da atividade e para gerar as fichas iniciais.

```
1. fun expTime (mean: int) =
2.   let
3.     val realMean = Real.fromInt mean
4.     val rv = exponential((1.0/realMean))
5.   in
6.     floor (rv+0.5)
7. end;
```

- a) **startProc**: a função *startProc* é responsável por atualizar o tempo de duração de cada caso. Inicialmente a duração da atividade é calculada pela chamada a função *expTime*. O tempo corrente é acrescido da duração da atividade usando a função da biblioteca *Base SML ModelTime.add* e então a variável que armazena a duração total do caso é atualizada.

```
1. fun startProc ((tsi,at):PLACE, Z:W) =
2.   let
3.     val proc_time = expTime(mda(Z))
4.     val time_stamp =
5.       ModelTime.add(time(),ModelTime.fromInt(proc_time))
6.     val new_tsi= IntInf.toInt(time_stamp)
```



```
6.      in
7.      ((new_tsi, at), proc_time)
8.      end;
```

A fim de padronizar os resultados, a quantidade de experimentos e monitores são os mesmos que os apresentados na seção 4.3.2 deste Capítulo. A quantidade de casos gerados usará a função exponencial considerando o intervalo médio de chegada de 25 minutos ( $\text{expTime}(25)$ ).

### 4.3.5 Resultados da Simulação - Distribuição Exponencial

O intervalo de visibilidade *fuzzy* é uma característica apresentada para o Processo de Tratamento de Reclamações modelado neste capítulo. Com o objetivo de mostrar as vantagens, ou mesmo desvantagens, do uso de intervalos *fuzzy* para representar o tempo, diferentes cenários para simulação serão definidos.

a) Cenário 1:

- Duração das atividades - distribuição exponencial.
- Intervalo de visibilidade *fuzzy* - somente será utilizado o núcleo do intervalo, ou seja, quando a possibilidade é igual a 1. Considerando o intervalo  $[a1, a2, a3, a4]$ , serão disparadas as transições quando o tempo estiver dentro do intervalo  $[a2, a3]$ .

b) Cenário 2:

- Duração das atividades - distribuição exponencial.
- Intervalo de visibilidade *fuzzy* - neste cenário, fichas que chegarem antes do prazo (núcleo), poderão ser disparadas desde que pertençam ao intervalo. Sabendo que  $[a1, a2, a3, a4]$  representa o intervalo *fuzzy*, fichas que chegarem depois de  $a1$  poderão ser disparadas.

c) Cenário 3:

- Duração das atividades - distribuição exponencial.
- Intervalo de visibilidade *fuzzy* - neste cenário, fichas que chegarem depois do prazo (núcleo), poderão ser disparadas desde que pertençam ao intervalo. Sabendo que  $[a1, a2, a3, a4]$  representa o intervalo *fuzzy*, fichas que chegarem antes de  $a4$  poderão ser disparadas.

d) Cenário 4:

- Duração das atividades - distribuição exponencial.
- Intervalo de visibilidade *fuzzy* - neste cenário, fichas que chegarem dentro do intervalo *fuzzy* (depois de  $a1$  e antes de  $a4$ ), poderão ser disparadas.

#### 4.3.5.1 Cenário 1 - Resultado de Simulação

O cenário 1 foi definido como tendo as durações das atividades definidas pela distribuição exponencial (sendo o parâmetro da exponencial a duração média quando é considerado o intervalo estático) e o intervalo de visibilidade *fuzzy* caracterizado somente pelo uso dos elementos do núcleo.

A Tabela 18 apresenta o número de vezes que o recurso compartilhado *R2* foi sobrecarregado. Para a realização da atividade *A2*, o recurso foi sobrecarregado em média 16 vezes; para a atividade *A3*, o recurso foi sobrecarregado em média 18 vezes e para a atividade *A7*, o recurso foi sobrecarregado em média 2 vezes. Se somente estes valores forem levados em consideração, aparentemente o recurso *R2* fica pouco sobrecarregado neste cenário (quando se utiliza somente o núcleo do intervalo). No entanto, deve-se analisar também a quantidade de casos que foram executados em cada atividade. Na Tabela 18, a linha *Count\_id* mostra a quantidade de vezes que a atividade foi executada. Assim, sabendo que deveriam ser 100 casos, os valores apresentados estão bem abaixo do que eram esperados.

Sobrecarga de R2 em A2				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Count_id	58,65	2,59	48	68
Min_id	0	0	0	0
Max_id	32,95	3,61	19	53
Média_id	16,60	1,84	9,04	25,16
Sobrecarga de R2 em A3				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Count_id	60,65	2,79	48	71
Min_id	0,00	0,00	0	0
Max_id	35,95	4,19	18	55
Média_id	17,85	2,27	7,21	27,90
Sobrecarga de R2 em A7				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Count_id	8,20	1,75	3	17
Min_id	0,00	0,00	0	0
Max_id	3,95	1,27	1	11
Média_id	1,98	0,60	0,33	5,47

Tabela 18 – Estatística Sobrecarga dos Recursos *Fuzzy*

A Tabela 19 apresenta o número de casos que foram finalizados. Como era de se esperar, considerando que a duração das atividades foi definida usando a distribuição exponencial, algumas atividades terminaram bem antes do esperado e outras bem depois (atraso). Isso porque quando a duração de atividade gerada é baseada num parâmetro de uma distribuição exponencial, não se pode garantir que as durações produzidas vão pertencer ao intervalo estático inicialmente especificado junto à atividade considerada.

Em ambos os casos, a ficha não pode ser disparada, então, o número de casos finalizados é muito pequeno; em média, 12 casos ( $11,90 \pm 1,97$ ).

Casos Finalizados				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Quantidade	11,90	1,97	5	23

Tabela 19 – Estatística - Quantidade de Casos Finalizados

O relatório completo pode ser encontrado no Apêndice A.2.

#### 4.3.5.2 Cenário 2 - Resultado de Simulação

Neste cenário, as durações das atividades são calculadas pela distribuição exponencial e as fichas que chegarem antes do prazo (núcleo), poderão ser disparadas desde que pertençam ao intervalo de visibilidade *fuzzy*. Sabendo que  $[a1, a2, a3, a4]$  representa o intervalo *fuzzy*, fichas que chegarem depois de  $a1$  poderão ser disparadas.

A Tabela 20 apresenta o número de vezes que o recurso compartilhado *R2* foi sobrecarregado. Para a realização da atividade *A2*, o recurso foi sobrecarregado em média 35 vezes; para a atividade *A3*, o recurso foi sobrecarregado em média 37 vezes; e para a atividade *A7*, o recurso foi sobrecarregado em média 8 vezes. Neste cenário, quase todas as fichas passaram pelas atividades *A2* e *A3*, fato este comprovado pela linha *Count\_id* da tabela.

Sobrecarga de R2 em A2				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Count_id	92,60	1,95	84	98
Min_id	0,00	0,00	0	0
Max_id	70,35	4,09	54	83
Média_id	34,89	2,10	25,17	41,81
Sobrecarga de R2 em A3				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Count_id	94,25	1,98	85	100
Min_id	0,00	0,00	0	0
Max_id	73,25	3,59	61	86
Média_id	36,87	1,71	28,63	42,51
Sobrecarga de R2 em A7				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Count_id	21,00	2,48	11	28
Min_id	0,00	0,00	0	0
Max_id	15,50	2,23	8	23
Média_id	7,79	1,11	3,90	12,39

Tabela 20 – Estatística Sobrecarga dos Recursos *Fuzzy*

A Tabela 21 apresenta o número de casos que foram finalizados. O número de casos finalizados ainda é muito pequeno, em média 32 casos ( $32,20 \pm 3,09$ ). No entanto, é possível perceber uma grande melhora na finalização dos casos, já que houve um aumento, em média, de 20 casos, o que corresponde a mais de 150% quando comparado ao cenário 1.

Casos Finalizados				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Quantidade	32,20	3,09	22	47

Tabela 21 – Estatística - Quantidade de Casos Finalizados

O relatório completo pode ser encontrado no Apêndice A.2.

#### 4.3.5.3 Cenário 3 - Resultado de Simulação

Neste cenário, as durações das atividades são calculadas pela distribuição exponencial e as fichas que chegarem após o prazo (núcleo), poderão ser disparadas desde que pertençam ao intervalo de visibilidade *fuzzy*. Sabendo que  $[a1,a2,a3,a4]$  representa o intervalo *fuzzy*, fichas que chegarem antes de  $a4$  poderão ser disparadas.

A Tabela 22 apresenta o número de vezes que o recurso compartilhado *R2* foi sobrecarregado. Para a realização da atividade *A2*, o recurso foi sobrecarregado em média 17 vezes; para a atividade *A3*, o recurso foi sobrecarregado em média 18 vezes; e para a atividade *A7*, o recurso foi sobrecarregado em média 2 vezes. Neste cenário, percebe-se que a quantidade de casos que passaram pelas atividades *A2*, *A3* e *A7* foi pequeno se comparado ao número total de casos. Fato este comprovado pelos dados da linha Count\_id da tabela.

A Tabela 23 apresenta o número de casos que foram finalizados. O número de casos finalizados é muito pequeno, em média 16 casos ( $16,20 \pm 1,74$ ). Quando os resultados obtidos neste cenário são comparados com o cenário anterior (cenário 2) é possível perceber que o disparo antecipado gera resultados melhores do que disparar fichas após o prazo.

O relatório completo pode ser encontrado no Apêndice A.2.

#### 4.3.5.4 Cenário 4 - Resultado de Simulação

Neste cenário, as durações das atividades são calculadas pela distribuição exponencial e as fichas poderão ser disparadas desde que pertençam ao intervalo de visibilidade *fuzzy*.

A Tabela 24 apresenta o número de vezes que o recurso compartilhado *R2* foi sobrecarregado. Para a realização da atividade *A2*, o recurso foi sobrecarregado em média 35 vezes; para a atividade *A3*, o recurso foi sobrecarregado em média 37 vezes; e para a atividade *A7*, o recurso foi sobrecarregado em média 11 vezes. Neste cenário, quase

Sobrecarga de R2 em A2				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Count_id	61,90	2,53	51	74
Min_id	0,00	0,00	0	0
Max_id	34,65	3,76	16	50
Média_id	17,33	1,85	7,96	25,08
Sobrecarga de R2 em A3				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Count_id	62,10	2,69	51	76
Min_id	0,00	0,00	0	0
Max_id	36,65	3,28	28	55
Média_id	17,96	1,75	12,43	26,31
Sobrecarga de R2 em A7				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Count_id	10,10	1,12	5	13
Min_id	0,00	0,00	0	0
Max_id	5,10	0,77	2	7
Média_id	2,65	0,45	0,80	4,07

Tabela 22 – Estatística Sobrecarga dos Recursos *Fuzzy*

Casos Finalizados				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Quantidade	16,20	1,74	9	24

Tabela 23 – Estatística - Quantidade de Casos Finalizados

todas as fichas passaram pelas atividades A2 e A3, fato este comprovado tanto pela linha Count\_id da tabela.

A Tabela 25 apresenta o número de casos que foram finalizados. O número de casos finalizados aumentou significativamente ( $43,10 \pm 2,77$ ). Este resultado mostra que o uso de intervalos de visibilidade *fuzzy* permitem o melhor gerenciamento do tempo e das atividades.

O relatório completo pode ser encontrado no Apêndice A.2.

Neste capítulo, foi possível mostrar uma ferramenta de simulação onde aparecem mecanismos de alocação de recursos complexos que permitem simular Sistemas de Gerenciamento de *Workflow* em condições de funcionamento variados. Também, foi possível avaliar o comportamento do sistema em funcionamento esperado ou em situação mais crítica. O importante é de poder visualizar através dos resultados de simulação como se comportará o sistema, isso com o objetivo de avaliar diversos cenários possíveis de distribuição de recursos para poder, no futuro, realizar o dimensionamento do sistema se baseando numa ferramenta de modelagem e simulação próxima da forma de trabalhar dos funcionários em ambiente de escritório. Outros modelos e/ou ferramentas de análise

Sobrecarga de R2 em A2				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Count_id	99,50	0,53	96	100
Min_id	0,00	0,00	0	0
Max_id	70,65	2,44	62	81
Média_id	35,54	1,31	32,29	42,55
Sobrecarga de R2 em A3				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Count_id	99,70	0,43	96	100
Min_id	0,00	0,00	0	0
Max_id	75,50	2,15	69	85
Média_id	37,45	1,12	32,25	41,25
Sobrecarga de R2 em A7				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Count_id	27,25	2,00	20	34
Min_id	0,00	0,00	0	0
Max_id	21,65	2,24	14	30
Média_id	10,85	1,07	7,60	15,18

Tabela 24 – Estatística Sobrecarga dos Recursos *Fuzzy*

Casos Finalizados				
	Média	Intervalo de confiança (95%)	Mínimo	Máximo
Quantidade	43,10	2,77	31	52

Tabela 25 – Estatística - Quantidade de Casos Finalizados

e simulação, dificilmente vão contemplar a forma flexível de se trabalhar em ambientes de trabalhos reais quando se trata de processos de negócios e de sistemas administrativos, e consequentemente, existirá um desvio muito importante entre os resultados de análise e o comportamento relendo sistema. Aqui, as estratégias de simulação propostas representam um tipo de solução de escalonamento flexível em tempo de execução que poderá em particular ser implementado após validar o modelo por simulação.

---

## Conclusão

Em Sistema de Gerenciamento de *Workflow*, recursos podem representar tanto equipamentos físicos quanto pessoas. A maioria dos recursos do tipo humano podem tratar várias atividades simultaneamente. Para expressar de forma realista o mecanismo de alocação de recursos onde o comportamento humano é considerado, conjuntos *fuzzy* delimitados pela distribuição de possibilidade na forma triangular foram associados com as marcações dos lugares para representar a disponibilidade humana.

Além disso, a utilização de intervalos *fuzzy* para representar a incerteza dos tempos de atividade de processos de negócio e aplicar mecanismos de propagação de tempo *fuzzy* para encontrar a melhor flexibilização dos prazos de cada atividade foi apresentada. Também, a associação da noção de possibilidade conjunta a sensibilização das transições permitiu uma representação bastante fiel com a forma de trabalhar de funcionários humanos em ambientes de escritório.

A aplicação deste mecanismo de alocação de recurso contínuo *fuzzy* associado a intervalos temporais *fuzzy* com as regras de disparo definidas com a possibilidade conjunta permite o monitoramento de prazos em gestão de projetos. Ainda, a aplicação de algumas heurísticas mostrou que se pode melhorar o gerenciamento tanto dos prazos quanto dos recursos.

Em suma, foram modelados vários mecanismos de alocação de recursos bem como diversas formas de representação do tempo nos processos de negócio. Destes modelos, foram gerados diversos jogadores de redes de Petri (mecanismos de inferência especializados) aplicados aos diversos modelos propostos para tentar encontrar uma forma de gerenciamento dos mecanismos de alocação de recursos em processos de negócio. De certa forma, cada jogador apresenta um tipo de solução ao problema de escalonamento de Sistema de Gerenciamento de *Workflow* em tempo de execução.

Para validar tais mecanismos de alocação de recurso junto com os jogadores usados na execução dinâmica do modelo, usou-se a linguagem CPN-ML para implementar e testar cada proposta apresentada. Através do uso da ferramenta CPN Tools cada proposta foi efetivamente implementada e testada num ambiente de simulação que permite a produção

de eventos aleatórios e a replicação dos experimentos. Então, o aspecto de simulação serviu essencialmente para gerar aleatoriamente os eventos que aconteceriam num ambiente de escritório.

Finalmente, é importante destacar que com uma ferramenta qualquer de simulação, não teria sido possível implementar cada proposta devido ao comportamento complexo, incerto e bastante flexível dos mecanismos apresentados. É por isso que geralmente, as ferramentas de edição e de gerenciamento dos processos de *Workflow* inclusive apresentam os mecanismos de alocação de modo informal ou semi-formal, deixando a cargo dos recursos envolvidos o respeito dos planejamentos de atividades impostos.

Assim, com o CPN Tools é possível mostrar uma ferramenta de simulação onde aparecem mecanismos de alocação de recursos complexos que permitem simular Sistema de Gerenciamento de *Workflow* em condições de funcionamento variados. Também, é possível avaliar o comportamento do sistema em funcionamento esperado ou em situação mais crítica. O importante é poder visualizar através dos resultados de simulação como se comportará o sistema, a fim de avaliar diversos cenários possíveis de distribuição de recursos para poder, no futuro, realizar o dimensionamento do sistema se baseando numa ferramenta de modelagem e simulação próxima da forma de trabalhar dos funcionários em ambiente de escritório. Outros modelos e/ou ferramentas de análise e simulação, dificilmente vão contemplar a forma flexível de se trabalhar em ambientes de trabalhos reais quando se trata de processos de negócio e de sistemas administrativos, e consequentemente, existirá um desvio muito importante entre os resultados de análise e o comportamento real do sistema. Aqui, as estratégias de simulação propostas representam um tipo de solução de escalonamento flexível em tempo de execução que poderá, em particular, ser implementado após validar o modelo por simulação.

## 5.1 Principais Contribuições

Considerando as Hipóteses desta pesquisa, apresentadas na seção 1.3 e as propostas apresentadas no Capítulo 3 e 4 bem como os resultados dos experimentos, tem-se que:

- a) A hipótese “É possível expressar de maneira mais realista o comportamento humano usando conjuntos *fuzzy* quando se trata de SGW” foi validada quando diversos mecanismos foram propostos, formalizados, implementados, simulados e comparados, e disso tudo verificou que mecanismos básicos de alocação baseados em representação de recursos puramente discretos se desviam muito do comportamento de funcionários humanos em ambiente de escritórios. Somente tornando os mecanismos de alocação de recursos contínuos e incertos que os comportamento dos modelos implementados no CPN Tools se tornaram mais parecidos com o que deve acontecer num ambiente real de trabalho (seções 3.1 e 4.1).
- b) A hipótese “Como o tempo real necessário para a execução de uma atividade



em um SGW é não-determinístico e não é facilmente previsível, um tempo *fuzzy* pode ser atribuído a cada atividade” foi validada ao serem definidos intervalos de tempo *fuzzy* e um mecanismo de propagação de restrição de tempo para a *Workflow net* que serve para planejar as datas de início de cada atividade através de uma representação mais ou menos certa, como deve acontecer num ambiente de trabalho real. De fato, uma política simples de início mais cedo das atividades que aparecem de acordo com os casos tratados num processo não correspondem à forma de trabalhar de um sistema administrativo. Caso contrário, como foi validado pelos modelos simulados, grande parte dos casos não seriam tratados dentro dos prazos estabelecidos (seção 4.2).

- c) A hipótese “O conceito de possibilidade conjunta que considera dois conjuntos nebulosos simultaneamente - tanto a disponibilidade dos recursos quanto restrições de tempo para respeitar prazos preestabelecidos de conclusão dos casos tratados - flexibiliza os modelos de alocação de recursos bem como a duração das atividades de maneira a expressar melhor o comportamento de funcionários humanos que atuam em atividades de processos de negócio” foi validada quando novas regras de disparo usando a noção de possibilidade conjunta foram definidas para *Workflow nets* e os *logs* de eventos produzidos pela simulação deste novo modelo mostraram uma melhoria na quantidade de casos tratados da forma correta comparando como uma estratégia mais simples de início mais cedo das atividades em particular em relação ao cumprimento dos prazos estabelecidos, tentando não sobrecarregar sempre que possível os recursos envolvidos (seção 4.3).
- d) A hipótese “É factível, usando CPN (Colored Petri Nets) e o ambiente CPN Tools, implementar e avaliar o desempenho de jogadores que consideram os possíveis modelos de alocação de recursos de processos de negócio” foi validada com a construção dos jogadores que tratam a alocação de recursos, possibilidade conjunta e resolução de conflito (seções 3.3 e 4.3). É importante destacar que o jogadores implementados e simulados no ambiente CPN Tools poderão servir para avaliar a política de gerenciamento de recursos diversos (sobrecarregando ou não os recursos, adiantando ou não o início das atividades etc.) e assim testar e validar diversas propostas de escalonamento e dimensionamento que poderão eventualmente ser implementadas em um SGW utilizado num ambiente real.

## 5.2 Trabalhos Futuros

Esta pesquisa tem se mostrado bastante promissora já que o campo de aplicação é grande. Uma possibilidade de investigação, que não foi realizada, seria a análise do *event log* que mostra as possibilidades de disparo de cada atividade e pode lidar com o tipo de análise qualitativa: se a atividade, na maior parte do tempo, está trabalhando com

possibilidade igual a 1, então, o resultado do trabalho do processo será de boa qualidade. Por outro lado, se um grande número de atividades são associadas com possibilidades perto de 0, então, a qualidade dos processos será de qualidade menor. Além disso, durante o processo de execução das atividades, o gerenciamento das atividades poderia sofrer uma certa influência de acordo com a semântica associada com a possibilidade de disparo baixa. Finalmente, no caso da transição estar em conflito, a informação concernente a possibilidade de disparo poderia ser usada para tomar uma decisão: por exemplo, se a possibilidade é baixa por causa da demora das atividades, seria dada prioridade para a transição em relação a outros processos com possibilidade de disparo alta.

Também, o trabalho de Lima (2007) apresenta a formalização da análise de redes de Petri temporais usando álgebra intervalar. O autor apresenta a junção da álgebra intervalar para tratar os problemas relacionados com imprecisão e os limites mínimos e máximos impostos pelas redes de Petri temporais. Desta maneira, é possível modelar uma equação linear intervalar que possibilita o cálculo de intervalos de tempo de ocorrências de transições sem que seja necessário explorar, completa ou parcialmente, o espaço de estados. Dada esta nova ferramenta matemática, pode ser interessante fazer um estudo comparativo para verificar qual o melhor método para a representação do tempo em *Workflow net* que utiliza recursos *fuzzy*.

No capítulo 4, foi inserido o cálculo de duração das atividades usando a função exponencial de maneira a expressar melhor a realidade dentro de uma empresa. Percebe-se que nesta situação, a quantidade de casos finalizados (que respeitam os prazos estabelecidos pelo planejamento inicial) diminui em comparação com o uso da função uniforme. Entende-se que uma solução possível para o melhor gerenciamento dos recursos e tempo seria aplicar o conceito de redes de Petri Possibilísticas na *Workflow net Fuzzy*. Uma rede de Petri possibilística combina a teoria da possibilidade ((DUBOIS; PRADE, 2009)) com a teoria das redes de Petri a Objeto ((SIBERTIN-BLANC, 2001)) e tem como objetivo levar em conta as informações incompletas sobre um sistema. Neste modelo, um lugar, uma transição e uma sequência de disparo denotam, respectivamente, um possível estado parcial, uma possível troca de estado e um possível comportamento (CARDOSO, 1999). A principal vantagem é a capacidade de atualizar o estado do sistema com informações incompletas sem originar estados inconsistentes. Esta rede permite em particular o uso de pseudo disparo que não corresponde a uma evolução normal do sistema. Considera-se apenas o início do disparo uma vez que não existe informação alguma que permita ter certeza sobre a ocorrência ou não do evento associado com a transição. Neste tipo de disparo, a incerteza sobre a localização dos objetos irá aumentar até que a ocorrência de um evento permitirá deduzir a correta localização dos mesmos, possibilitando assim retornar ao conhecimento certo do sistema. Então, acredita-se que um trabalho futuro interessante é o uso de pseudo-disparos no mecanismo de alocação de recursos *fuzzy* com as regras de disparo definidas como possibilidade conjunta para lidar com situações de

conflito quando recursos são compartilhados e que as resoluções de conflitos de recursos para a melhor execução das diversas atividades envolvidas no processo continuam incertas em relação à melhor estratégia a ser implementada para respeitar os prazos de conclusão de cada caso em execução.

## 5.3 Contribuições Representadas pela Produção Bibliográfica

Até o presente momento as seguintes produções bibliográficas foram publicadas e apresentadas - no caso de conferências - (BARRETO et al., 2014), (FREITAS; JULIA, 2015), (FREITAS; JULIA; REZENDE, 2015a), (FREITAS; JULIA; REZENDE, 2015b), (FREITAS; JULIA; REZENDE, 2016) e (FREITAS; JULIA, 2016).

O artigo (BARRETO et al., 2014) trata de uma introdução à modelagem de Redes de Petri usando CPN Tools apresentada na seção 2.1.5. Este artigo foi publicado e apresentado no ICEIS (*International Conference on Enterprise Information System*) no ano de 2014.

O artigo (FREITAS; JULIA, 2015) apresenta o mecanismo de propagação de restrição mostrado na seção 4.2 publicado e apresentado no ITNG(*International Conference on Information Technology: New Generations*) no ano de 2015.

O artigo (FREITAS; JULIA; REZENDE, 2015a) mostra o uso da possibilidade conjunta em *Workflow net* apresentada na seção 4.3 e foi publicado e apresentado no ICEIS no ano de 2015. Este artigo foi indicado a *best student paper* da conferência. Uma versão estendida com o título “Resource Allocation Mechanisms and Time Constraint Propagation Technics in Fuzzy Workflow Nets” foi publicada em *Lecture Notes in Business Information Processing* em (FREITAS; JULIA; REZENDE, 2015b).

O artigo (FREITAS; JULIA, 2016) apresenta a definição da implementação do Jogador com regras de disparo baseadas na noção de Possibilidade Conjunta apresentados no capítulo 4 foi publicado e apresentado no ano de 2016 no ITNG.

O artigo (FREITAS; JULIA; REZENDE, 2016) mostra os resultados da simulação do Jogador com regras de disparo baseadas na noção de Possibilidade Conjunta apresentados no capítulo 4 e foi publicado e apresentado no ano de 2016 no ICEIS.

Além disso, um outro artigo com o título “Fuzzy Hybrid Resource Allocation Mechanisms in Workflow Management Systems” que trata da modelagem e simulação do mecanismo de alocação de recursos usando CPN Tools (Capítulo 3) foi submetido ao periódico “Fuzzy Sets and Systems” e ainda está sob revisão.



---

## Referências

AALST, W. van der. **Timed coloured Petri nets and their application to logistics**. Tese (Doutorado) — Technische Universiteit Eindhoven, 1992.

AALST, W. van der. Verification of workflow nets. In: **Proceedings of the 18th International Conference on Application and Theory of Petri Nets**. London, UK, UK: Springer-Verlag, 1997. (ICATPN '97), p. 407–426. ISBN 3-540-63139-9.

\_\_\_\_\_. The application of petri nets to workflow management. **Journal of Circuits, Systems, and Computers**, v. 8, n. 1, p. 21–66, 1998.

\_\_\_\_\_. Business process simulation survival guide. In: **Handbook on Business Process Management 1**. Berlin: Springer, 2015. p. 337–370.

AALST, W. van der; HEE, K. van. **Workflow Management: Models, Methods, and Systems**. Cambridge, MA, USA: MIT Press, 2004. ISBN 0262720469.

ALONSO, G. et al. Functionality and limitations of current workflow management systems. **IEEE Expert**, v. 12, 1997.

ANAND, S. K.; RAMAN, T. S.; SUBRAMANIAN, S. Implementing a neuro fuzzy expert system for optimising the performance of chemical recovery boiler. **International Journal of Artificial Intelligence and Soft Computing**, Inderscience, v. 4, n. 2, p. 249–263, 2014.

ANDERSON, D. R.; SWEENEY, D. J. **Estatística aplicada à administração e economia**. 2.ed.. ed. São Paulo: Cengage, 2011.

BAGGIO, G.; WAINER, J.; ELLIS, C. Applying scheduling techniques to minimize the number of late jobs in workflow systems. In: **ACM. Proceedings of the 2004 ACM symposium on Applied computing**. New York, NY: Springer, 2004. p. 1396–1403.

BARRETO, F. M. et al. A straightforward introduction to formal methods using coloured petri nets. In: **ICEIS 2014 - Proceedings of the 16th International Conference on Enterprise Information Systems, Volume 2, Lisbon, Portugal, 27-30 April, 2014**. Lisbon, Portugal: ScitePress, 2014. p. 145–152.

BERARD, B. et al. The expressive power of time petri nets. **Theoretical Computer Science**, Elsevier, v. 474, p. 1–20, 2013.

- BERTHOMIEU, B.; MENASCHE, M. An enumerative approach for analyzing time petri nets. In: **Proceedings IFIP**. London, UK: Elsevier Science Publishers, 1983. p. 41–46.
- BERTOLINI, C.; LIU, Z.; SRBA, J. Verification of timed healthcare workflows using component timed-arc petri nets. In: **Foundations of Health Information Engineering and Systems**. Paris, France: Springer, 2013. p. 19–36.
- BHARATHI, V.; PRAMOD, D.; RAMAN, R. A fuzzy petri-net based conceptual model for risk prediction in enterprise resource planning (erp) acquisition decisions for small and medium enterprises. **Journal of Computer Science**, v. 9, n. 1, p. 139–146, 2013.
- BOSZORMENYI, L.; EISNER, R.; GROISS, H. Adding distribution to a workflow management system. In: **DEXA Workshop**. Florence, Italy: IEEE, 1999. p. 17–21.
- BOWDEN, F. A brief survey and synthesis of the roles of time in petri nets. **Mathematical and Computer Modelling**, v. 31, n. 10, p. 55 – 68, 2000. ISSN 0895-7177.
- BUCKLEY, J. **Simulating Fuzzy Systems**. Germany: Springer Science and Business Media, 2005. (Studies in Fuzziness and Soft Computing). ISBN 9783540241164.
- CARDOSO, J. **Time fuzzy Petri nets**. 1999. Cardoso, Janette (ed.) et al., Fuzziness in Petri nets. Heidelberg: Physica-Verlag. Stud. Fuzziness Soft Comput. 22, 115-145 (1999).
- CARDOSO, J.; VALETTE, R. **Redes de Petri**. Florianópolis, SC: Editora da UFSC, 1997. (Série Didática). ISBN 9788532800954.
- CARDOSO, J.; VALETTE, R.; DUBOIS, D. Petri nets with uncertain markings. In: **Applications and Theory of Petri Nets**. Bonn, Germany: Springer, 1989. p. 64–78.
- \_\_\_\_\_. Possibilistic petri nets. **IEEE Transactions on Systems, Man, and Cybernetics, Part B**, v. 29, n. 5, p. 573–582, 1999.
- CELKO, J. Time token design methodology. **Software: Practice and Experience**, Wiley Online Library, v. 12, n. 10, p. 889–895, 1982.
- CERONE, A.; MAGGIOLO-SCHETTINI, A. Time-based expressivity of time petri nets for system specification. **Theoretical Computer Science**, v. 216, n. 1, p. 1 – 53, 1999. ISSN 0304-3975.
- CHAN, N. N. et al. Mining event logs to assist the development of executable process variants. In: SPRINGER. **Advanced Information Systems Engineering**. Thessaloniki, Greece, 2014. p. 548–563.
- CPN tools. 2016. <<http://www.cpn.org>>. Accessed: 2016-10-20.
- DAVID, R.; ALLA, H. **Discrete, Continuous, and Hybrid Petri Nets**. 2nd. ed. Heidelberg, Germany: Springer Publishing Company, Incorporated, 2010. ISBN 3642106684, 9783642106682.
- DENG, N. et al. Time management model of workflow based on time axis. **Applied Mechanics and Materials**, Trans Tech Publ, v. 442, p. 458–465, 2014.

DOTOLI, M. et al. 12th ifac international workshop on discrete event systems (2014) a timed petri nets model for intermodal freight transport terminals. **IFAC Proceedings Volumes**, v. 47, n. 2, p. 176 – 181, 2014. ISSN 1474-6670.

DUBOIS, D. **Fuzzy Sets and Systems: Theory and Applications**. Orlando, FL, USA: Academic Press, Inc., 1997. ISBN 0122227506.

DUBOIS, D.; PRADE, H. Processing fuzzy temporal knowledge. In: IEEE. **Systems, Man and Cybernetics**. New York, NY, 1989. p. 729–744.

\_\_\_\_\_. **Possibility theory**. New York, NY: Plenum Press, New-York, 2009.

EDER, J.; PANAGOS, E.; RABINOVICH, M. **Time Constraints in Workflow Systems**. Germany: Springer, 2013.

\_\_\_\_\_. Workflow time management revisited. In: **Seminal Contributions to Information Systems Engineering**. Germany: Springer, 2013. p. 207–213.

ESHUIS, H. **Semantics and Verification of UML Activity Diagrams for Workflow Modelling**. Tese (Doutorado) — University of Twente, Enschede, The Netherlands, 2002.

ESQUIROL, P.; HUGUET, M.-J.; LOPEZ, P. Modelling and managing disjunctions in scheduling problems. **Journal of Intelligent Manufacturing**, Kluwer Academic Publishers, v. 6, n. 2, p. 133–144, 1995. ISSN 0956-5515.

FREITAS, J. C. J. de; JULIA, S. Fuzzy time constraint propagation mechanism for workflow nets. In: **ITNG 2015 - Proceedings of the 12th International Conference on Information Technology - New Generations (ITNG)**, Las Vegas, NV, USA April 13, 2015 to April 15, 2015. Las Vegas , NV: Springer, 2015. p. 367 – 372.

\_\_\_\_\_. Fuzzy resource-constrained time workflow nets. In: **Information Technology: New Generations**. Las Vegas , NV: Springer, 2016. p. 543–553.

FREITAS, J. C. J. de; JULIA, S.; REZENDE, L. P. de. Fuzzy resource allocation mechanisms in workflow nets. In: **ICEIS 2015 - Proceedings of the 17th International Conference on Enterprise Information Systems, Volume 1, 27-30 April, 2015**. Barcelona, Spain: SciTPress, 2015. p. 471–478.

\_\_\_\_\_. Resource allocation mechanisms and time constraint propagation techniques in fuzzy workflow nets. In: HAMMOUDI, S. et al. (Ed.). **ICEIS (Revised Selected Papers)**. [S.l.]: Springer, 2015. (Lecture Notes in Business Information Processing, v. 241), p. 214–235. ISBN 978-3-319-29132-1.

\_\_\_\_\_. Modeling a fuzzy resource allocation mechanism based on workflow nets. In: **ICEIS 2016 - Proceedings of the 18th International Conference on Enterprise Information Systems, Volume 2, Rome, Italy, April 25-28, 2016**. Rome, Italy: ScitePress, 2016. p. 559–566.

GEHLOT, V.; KULKARNI, S.; BRZOZOWSKI, J. Modeling and performance simulation of a software architecture for large-scale measurement of broadband networks using colored petri nets. In: SOCIETY FOR COMPUTER SIMULATION

INTERNATIONAL. **Proceedings of the 19th Communications & Networking Symposium**. [S.l.], 2016. p. 4.

GEORGAKOPOULOS, D.; HORNICK, M.; SHETH, A. An overview of workflow management: From process modeling to workflow automation infrastructure. **Distrib. Parallel Databases**, Kluwer Academic Publishers, Hingham, MA, USA, v. 3, n. 2, p. 119–153, abr. 1995. ISSN 0926-8782.

GHARIB, M.; GIORGINI, P. Modeling and reasoning about information quality requirements in business processes. In: **Enterprise, Business-Process and Information Systems Modeling**. Germany: Springer, 2015. p. 231–245.

GONDRAN, M.; MINOUX, M.; VAJDA, S. **Graphs and Algorithms**. New York, NY, USA: John Wiley & Sons, Inc., 1984. ISBN 0471-10374-8.

HAN, L. et al. Deadlock-free genetic scheduling for flexible manufacturing systems using petri nets and deadlock controllers. **International Journal of Production Research**, Taylor & Francis, v. 52, n. 5, p. 1557–1572, 2014.

HEE, K. van; SIDOROVA, N.; VOORHOEVE, M. Resource-constrained workflow nets. **Fundam. Inf.**, IOS Press, Amsterdam, The Netherlands, The Netherlands, v. 71, n. 2,3, p. 243–257, fev. 2006. ISSN 0169-2968.

HOFSTEDE, A. H. M. ter et al. (Ed.). **Modern Business Process Automation - YAWL and its Support Environment**. Germany: Springer, 2010. ISBN 978-3-642-03120-5.

HOFSTEDE, A. t. et al. **Modern Business Process Automation: YAWL and its Support Environment**. Germany: Springer-Verlag, Berlin, 2010.

HOLLINGSWORTH, D. **Workflow Management Coalition - The Workflow Reference Model**. USA, 1995.

HUANG, Z. et al. Reinforcement learning based resource allocation in business process management. **Data & Knowledge Engineering**, v. 70, n. 1, p. 127 – 145, 2011. ISSN 0169-023X.

JAMES, B. **Probabilidade: um curso em nível intermediário**. [S.l.]: Instituto de Matemática Pura e Aplicada, CNPq, 1981. (Projeto Euclides).

JENSEN, K. Coloured petri nets and the invariant-method. **Theoretical Computer Science**, Elsevier, USA, v. 14, n. 3, p. 317 – 336, 1981. ISSN 0304-3975.

JENSEN, K.; KRISTENSEN, L. M. **Coloured Petri Nets**. Germany: Springer, 2009. ISBN 978-3-642-00283-0.

JESKE, J. C. **Mecanismo de Alocação de Recurso Fuzzy para Sistemas de Gerenciamento de Workflow**. Dissertação (Mestrado) — Universidade Federal de Uberlândia, Faculdade de Computação, Uberlândia, 2006.

JESKE, J. C.; JULIA, S.; VALETTE, R. Fuzzy continuous resource allocation mechanisms in workflow management systems. In: **IRI**. Hawai: IEEE Systems, Man, and Cybernetics Society, 2006. p. 472–477.



- \_\_\_\_\_. Fuzzy continuous resource allocation mechanisms in workflow management systems. In: **Proceedings of the 2009 XXIII Brazilian Symposium on Software Engineering**. Washington, DC, USA: IEEE Computer Society, 2009. (SBES '09), p. 236–251. ISBN 978-0-7695-3844-0.
- JULIA, S.; OLIVEIRA, F. F. de; VALETTE, R. Real time scheduling of workflow management systems based on a p-time petri net model with hybrid resources. **Simulation Modelling Practice and Theory**, v. 16, n. 4, p. 462 – 482, 2008. ISSN 1569-190X.
- JULIA, S.; VALETTE, R. Real time scheduling of batch systems. **Simulation Practice and Theory**, v. 8, n. 5, p. 307–319, 2000.
- JUNG, C.; KIM, H.-J.; LEE, T.-E. A branch and bound algorithm for cyclic scheduling of timed petri nets. **Automation Science and Engineering, IEEE Transactions on**, IEEE, v. 12, n. 1, p. 309–323, 2015.
- KLIR, G. J.; YUAN, B. **Fuzzy Sets and Fuzzy Logic: Theory and Applications**. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1995. ISBN 0-13-101171-5.
- KOTB, Y. T.; BADREDDIN, E. Synchronization among activities in a workflow using extended workflow petri nets. In: **CEC**. San Diego, CA: IEEE Computer Society, 2005. p. 548–551. ISBN 0-7695-2277-7.
- LAHAYE, S.; KOMENDA, J.; BOIMOND, J.-L. 12th ifac international workshop on discrete event systems (2014) modeling of timed petri nets using deterministic (max,+) automata. **IFAC Proceedings Volumes**, v. 47, n. 2, p. 471 – 476, 2014. ISSN 1474-6670.
- LARCO, J.; WIERS, V.; FRANSOO, J. Scheduling the scheduling task: A time management perspective on scheduling. **Beta Working Paper series 408**, 2013.
- LI, H.; CUI, X. Modeling and performance analysis of workflow based on advanced fuzzy timing petri nets. In: **Foundations and Applications of Intelligent Systems**. Berlin: Springer, 2014. p. 199–211.
- LIMA, E. A. **ANALISE E APLICACOES EM REDES DE PETRI TEMPORAIS: Uma Abordagem via Algebra Intervalar**. Tese (Doutorado) — Universidade Tecnológica Federal do Paraná, 2007.
- LING, S.; SCHMIDT, H. Time petri nets for workflow modelling and analysis. In: **IEEE. Systems, Man, and Cybernetics, 2000 IEEE International Conference on**. USA, 2000. v. 4, p. 3039–3044.
- LIU, C. et al. Petri net based behavior description of cross-organization workflow with synchronous interaction pattern. In: **Process-Aware Systems**. Germany: Springer, 2015. p. 1–10.
- LIU, G. Complexity of the deadlock problem for petri nets modeling resource allocation systems. **Information Sciences**, Elsevier, USA, 2015.
- LIU, H.-C. et al. Knowledge acquisition and representation using fuzzy evidential reasoning and dynamic adaptive fuzzy petri nets. **Cybernetics, IEEE Transactions on**, IEEE, v. 43, n. 3, p. 1059–1072, 2013.

- LOMAZOVA, I. A. Interacting workflow nets for workflow process re-engineering. **Fundam. Inf.**, IOS Press, Amsterdam, The Netherlands, The Netherlands, v. 101, n. 1-2, p. 59–70, jan. 2010. ISSN 0169-2968.
- LOONEY, C. G. Fuzzy petri nets for rule-based decisionmaking. **IEEE Trans. Syst. Man Cybern.**, IEEE Press, Piscataway, NJ, USA, v. 18, n. 1, p. 178–183, jan. 1988. ISSN 0018-9472.
- LU, Y. et al. Formalizing workflow with location constraints by colored petri nets. In: **IEEE. e-Business Engineering (ICEBE), 2014 IEEE 11th International Conference on**. USA, 2014. p. 226–231.
- MALAQUIAS, F. F. de O. et al. Sistema de gerenciamento de workflow baseado em redes de petri e em algoritmos genéticos. In: **SBSI. Simpósio Brasileiro de Sistemas de Informação**. João Pessoa, PB, 2013. p. 415–426.
- MARTOS-SALGADO, M.; ROSA-VELARDO, F. Dynamic soundness in resource-constrained workflow nets. In: BRUNI, R.; DINGEL, J. (Ed.). **FMOODS/FORTE**. Reykjavik, Iceland: Springer, 2011. (Lecture Notes in Computer Science, v. 6722), p. 259–273. ISBN 978-3-642-21460-8.
- MEDEIROS, F. F. **Análise sob restrições baseada em um raciocínio energético para o problema de escalonamento tempo real de sistemas de gerenciamento de workflow**. Dissertação (Mestrado) — Universidade Federal de Uberlândia, Faculdade de Computação, Uberlândia, 2009.
- MERLIN, P. M. **A Study of the Recoverability of Computing Systems**. Tese (Doutorado) — University of California, Irvine, 1974. AAI7511026.
- MIKA, M.; WALIGORA, G.; WEGLARZ, J. Modelling and solving grid resource allocation problem with network resources for workflow applications. **Journal of Scheduling**, Springer US, v. 14, n. 3, p. 291–306, 2011. ISSN 1094-6136.
- MURATA, T. Petri nets: Properties, analysis and applications. **Proceedings of the IEEE**, v. 77, n. 4, p. 541–580, April 1989.
- \_\_\_\_\_. Temporal uncertainty and fuzzy-timing high-level petri nets. In: **Application and Theory of Petri Nets**. Osaka, Japan: Springer, 1996. p. 11–28.
- OLIVEIRA, G. W. de; JULIA, S.; PASSOS, L. M. S. Game modeling using workflow nets. In: **SMC**. Anchorage, Alaska, USA: IEEE, 2011. p. 838–843.
- OMG. **Business Process Model and Notation (BPMN), Version 2.0**. USA, 2011. Disponível em: <<http://www.omg.org/spec/BPMN/2.0>>.
- \_\_\_\_\_. **OMG Unified Modeling Language (OMG UML), Superstructure, Version 2.4.1**. USA, 2011. Disponível em: <<http://www.omg.org/spec/UML/2.4.1>>.
- PASSOS, L. M. S.; JULIA, S. Deadlock-freeness scenarios detection in web service composition. In: **IEEE. Information Technology-New Generations (ITNG), 2015 12th International Conference on**. Las Vegas, NV, 2015. p. 780–783.

PEGDEN, C. D.; SADOWSKI, R. P.; SHANNON, R. E. **Introduction to Simulation Using SIMAN**. 2nd. ed. New York, NY, USA: McGraw-Hill, Inc., 1995. ISBN 0070493200.

PETERSON, J. L. **Petri Net Theory and the Modeling of Systems**. Upper Saddle River, NJ, USA: Prentice Hall PTR, 1981. ISBN 0136619835.

PLA, A. et al. Petri net-based process monitoring: a workflow management system for process modelling and monitoring. **Journal of Intelligent Manufacturing**, Springer, v. 25, n. 3, p. 539–554, 2014.

RAMCHANDANI, C. **Analysus of Asynchronous Concurrent Systems by Timed Petri nets**. Cambridge, MA, USA, 1974.

RATZER, A. V. et al. Cpn tools for editing, simulating, and analysing coloured petri nets. In: **Proceedings of the 24th International Conference on Applications and Theory of Petri Nets**. Berlin, Heidelberg: Springer-Verlag, 2003. (ICATPN'03), p. 450–462. ISBN 3-540-40334-5.

REZENDE, L. P. de; JULIA, S. Deadlock avoidance in interorganizational business processes using a possibilistic workflow net. In: **ICEIS 2015 - Proceedings of the 17th International Conference on Enterprise Information Systems, Volume 1, 27-30 April, 2015**. Barcelona, Spain: SciTPress, 2015. p. 429–439.

REZENDE, L. P. de; JULIA, S.; CARDOSO, J. Possibilistic interorganizational workflow net for the recovery problem concerning communication failures. In: **ICEIS 2014 - Proceedings of the 16th International Conference on Enterprise Information Systems, Volume 1, Lisbon, Portugal, 27-30 April, 2014**. Lisbon, Portugal: SciTPress, 2014. p. 432–439.

ROZINAT, A. et al. Discovering colored petri nets from event logs. **International Journal on Software Tools for Technology Transfer**, Springer, v. 10, n. 1, p. 57–74, 2008.

SHARP, A.; MCDERMOTT, P. **Workflow Modeling: Tools for Process Improvement and Application Development**. 1st. ed. Norwood, MA, USA: Artech House, Inc., 2001. ISBN 1580530214.

SIBERTIN-BLANC, C. CoOperative Objects: Principles, Use and Implementation. In: AGHA, G.; CINDIO, F. D. (Ed.). **Concurrent Object-Oriented Programming and Petri Nets**. <http://www.springerlink.com/>: Springer-Verlag, 2001, (LNCS, v. 2001). p. 210–241. Disponível em: <[http://www.irit.fr/publis/SMAC/DOCUMENTS/PUBLIS/SocLab/COOP\\_PN.pdf](http://www.irit.fr/publis/SMAC/DOCUMENTS/PUBLIS/SocLab/COOP_PN.pdf)>.

SILVA, M.; VALETTE, R. Petri nets and flexible manufacturing. In: **Advances in Petri Nets 1989, Covers the 9th European Workshop on Applications and Theory in Petri Nets-selected Papers**. London, UK, UK: Springer-Verlag, 1990. p. 374–417. ISBN 3-540-52494-0.

SU, X. et al. A novel approach to filter design for t–s fuzzy discrete-time systems with time-varying delay. **Fuzzy Systems, IEEE Transactions on, IEEE**, v. 20, n. 6, p. 1114–1129, 2012.

- TANG, Y. et al. Temporal workflow modeling and its application. In: **Temporal Information Processing Technology and Its Application**. Tsinghua University Press: Springer, 2010. p. 293–309.
- TSAI, J. J. P.; YANG, S. J.; CHANG, Y.-H. Timing constraint petri nets and their application to schedulability analysis of real-time system specifications. **IEEE transactions on Software Engineering**, IEEE, v. 21, n. 1, p. 32–49, 1995.
- TÜYSÜZ, F.; KAHRAMAN, C. Modeling a flexible manufacturing cell using stochastic petri nets with fuzzy parameters. **Expert Systems with Applications**, Elsevier, v. 37, n. 5, p. 3910–3920, 2010.
- VALETTE, R.; ATABAKHCHE, H. Petri nets for sequence constraint propagation in knowledge based approaches. In: **Concurrency and Nets**. Germany: Springer, 1987. p. 555–569.
- VALETTE, R.; CARDOSO, J.; DUBOIS, D. Monitoring manufacturing systems by means of Petri nets with imprecise markings. In: **Proc. IEEE Inter. Symp. on Intelligent Control, Albany, N.Y.**, . Albany, NY: IEEE, 1989. p. 233–238.
- VANDERFEESTEN, I.; AALST, W. van der; REIJERS, H. A. Modelling a product based workflow system in cpn tools. In: CITESEER. **Proceedings of the Sixth Workshop on the Practical Use of Coloured Petri Nets and CPN Tools (CPN 2005)**. Aarhus, Denmark, 2005. v. 576, p. 99–118.
- VASCONCELOS, E. M.; DIAS, K. L. A colored petri net model of simulation for performance evaluation for ieee 802.22 based network. **arXiv preprint arXiv:1608.08682**, 2016.
- VENERO, M. L. F.; SILVA, F. S. C. da. Modeling and simulating interaction protocols using nested petri nets. In: **Software Engineering and Formal Methods**. Madrid, Spain: Springer, 2014. p. 135–150.
- WANG, J.; LI, D. Resource oriented workflow nets and workflow resource requirement analysis. **International Journal of Software Engineering and Knowledge Engineering**, v. 23, n. 5, p. 677–694, 2013.
- WANG, J.; TEPFENHART, W. M.; ROSCA, D. Emergency response workflow resource requirements modeling and analysis. **IEEE Transactions on Systems, Man, and Cybernetics, Part C**, v. 39, n. 3, p. 270–283, 2009.
- WESTERGAARD, M.; SLAATS, T. Cpn tools 4: A process modeling tool combining declarative and imperative paradigms. In: **BPM (Demos)**. [S.l.: s.n.], 2013.
- WFMC. Glossary. **WfMC: Terminology and Glossary**. 1999. 65 p. Disponível em: <[http://www.wfmc.org/standards/docs/TC-1011\\_term\\_glossary\\_v3.pdf](http://www.wfmc.org/standards/docs/TC-1011_term_glossary_v3.pdf)>.
- XIAO, Z.; MING, Z. A method of workflow scheduling based on colored petri nets. **Data & Knowledge Engineering**, Elsevier, v. 70, n. 2, p. 230–247, 2011.
- XU, T.; ZENG, J.-J.; LU, Z.-L. Study on scheduling of main resources in airport based on cpn tools. **Zhongguo Minhang Daxue Xuebao/ Journal of Civil Aviation University of China**, Zhongguo Minhang Daxue, v. 31, n. 2, p. 36–39, 2013.

- ZADEH, L. A. Fuzzy sets. **Information and Control**, v. 8, p. 338–353, 1965.
- \_\_\_\_\_. Fuzzy logic. **IEEE Computer**, v. 21, n. 4, p. 83–93, 1988.
- ZHOU, F. et al. User experience modeling and simulation for product ecosystem design based on fuzzy reasoning petri nets. **Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on**, IEEE, v. 42, n. 1, p. 201–212, 2012.
- ZHOU, M. **Petri nets in Flexible and Agile Automation**. New York, NY: Springer Science & Business Media, 2012. v. 310.
- ZHOU, Y.; MURATA, T. Petri net model with fuzzy timing and fuzzy-metric temporal logic. **Int. J. Intell. Syst.**, v. 14, n. 8, p. 719–745, 1999.
- ZIMMERMANN, H. **Fuzzy Set Theory and Its Applications**. Netherlands: Springer Netherlands, 2001. ISBN 9780792374350.
- ZUBEREK, W. Timed petri nets definitions, properties, and applications. **Microelectronics Reliability**, v. 31, n. 4, p. 627 – 644, 1991. ISSN 0026-2714.



# APÊNDICE **A**

## Relatórios Completos da Simulação no CPN Tools

Neste Apêndice, os relatórios completos gerados pela ferramenta CPN Tools serão apresentados.

### A.1 Capítulo 3

No capítulo 3, três diferentes modelos foram simulados. O relatório completo dos resultados apresentados na seção 3.3.1 é mostrado na Figura 81.

Statistics							
Name	Avrg	90% Half Length	95% Half Length	99% Half Length	StD	Min	Max
Duracao_Total							
count_iid	100.000000	0.000000	0.000000	0.000000	0.000000	100	100
max_iid	6255.550000	115.018723	139.233191	190.323057	297.501079	5805	6714
min_iid	87.300000	6.476739	7.840263	10.717149	16.752376	60	115
sum_iid	240543.250000	4643.529561	5621.114731	7683.712014	12010.697227	223783	263418
avrg_iid	2405.432500	46.435296	56.211147	76.837120	120.106972	2237.830000	2634.180000

Figura 81 – Resultado da Simulação: Mecanismo de Alocação de Recurso Discreto.

O relatório completo dos resultados apresentados na seção 3.3.2 é mostrado na Figura 82 e o relatório dos resultados apresentados na seção 3.3.3 é mostrado na Figura 83.

### A.2 Capítulo 4

O capítulo 4 apresenta diferentes cenários de simulação. O relatório com os resultados da simulação do jogador com possibilidade conjunta usando distribuição normal mostrado na seção 4.3.3 é apresentado na Figura 84.

Statistics							
Name	Avrg	90% Half Length	95% Half Length	99% Half Length	StD	Min	Max
Duracao							
count_iid	100.000000	0.000000	0.000000	0.000000	0.000000	100	100
max_iid	1751.000000	138.905144	168.148332	229.848245	359.284376	1235	2479
min_iid	62.650000	2.035275	2.463753	3.367797	5.264329	56	72
sum_iid	45753.900000	2930.155257	3547.030048	4848.568069	7578.977836	33790	59425
avrg_iid	457.539000	29.301553	35.470300	48.485681	75.789778	337.900000	594.250000

Figura 82 – Resultado da Simulação: Mecanismo de Alocação de Recurso Contínuo e Discreto.

Statistics							
Name	Avrg	90% Half Length	95% Half Length	99% Half Length	StD	Min	Max
Duracao							
count_iid	100.000000	0.000000	0.000000	0.000000	0.000000	100	100
max_iid	108.550000	4.326093	5.236849	7.158445	11.189633	98	138
min_iid	59.600000	0.644496	0.780179	1.066456	1.667018	56	62
sum_iid	8066.350000	78.750835	95.329959	130.310087	203.692564	7730	8613
avrg_iid	80.663500	0.787508	0.953300	1.303101	2.036926	77.300000	86.130000

Figura 83 – Resultado da Simulação: Mecanismo de Alocação de Recurso *Fuzzy* e Discreto.

O relatório com os resultados da simulação do jogador com possibilidade conjunta usando distribuição exponencial: cenário 1 mostrado na seção 4.3.5.1 é apresentado na Figura 85.

O relatório com os resultados da simulação do jogador com possibilidade conjunta usando distribuição exponencial: cenário 2 mostrado na seção 4.3.5.2 é apresentado na Figura 86.

O relatório com os resultados da simulação do jogador com possibilidade conjunta usando distribuição exponencial: cenário 3 mostrado na seção 4.3.5.3 é apresentado na Figura 87.

O relatório com os resultados da simulação do jogador com possibilidade conjunta usando distribuição exponencial: cenário 4 mostrado na seção 4.3.5.4 é apresentado na Figura 88.



Statistics							
Name	Avrg	90% Half Length	95% Half Length	99% Half Length	Std	Min	Max
Duracao_A1							
count_iid	100.000000	0.000000	0.000000	0.000000	0.000000	100	100
max_iid	10.000000	0.000000	0.000000	0.000000	0.000000	10	10
min_iid	5.000000	0.000000	0.000000	0.000000	0.000000	5	5
sum_iid	750.150000	5.666095	6.858957	9.375765	14.655608	731	776
avrg_iid	7.501500	0.056661	0.068890	0.093758	0.146556	7.310000	7.760000
Duracao_A2							
count_iid	100.000000	0.000000	0.000000	0.000000	0.000000	100	100
max_iid	35.000000	0.000000	0.000000	0.000000	0.000000	35	35
min_iid	25.000000	0.000000	0.000000	0.000000	0.000000	25	25
sum_iid	3001.150000	12.421557	15.040253	20.559085	32.136674	2934	3052
avrg_iid	30.011500	0.124246	0.150403	0.205591	0.321367	29.340000	30.520000
Duracao_A3							
count_iid	100.000000	0.000000	0.000000	0.000000	0.000000	100	100
max_iid	30.000000	0.000000	0.000000	0.000000	0.000000	30	30
min_iid	20.000000	0.000000	0.000000	0.000000	0.000000	20	20
sum_iid	2511.100000	10.382263	12.568002	17.179672	26.854187	2479	2575
avrg_iid	25.111000	0.103823	0.125680	0.171797	0.268542	24.790000	25.750000
Duracao_A5							
count_iid	100.000000	0.000000	0.000000	0.000000	0.000000	100	100
max_iid	25.000000	0.000000	0.000000	0.000000	0.000000	25	25
min_iid	15.000000	0.000000	0.000000	0.000000	0.000000	15	15
sum_iid	1992.100000	9.397071	11.375102	15.519162	21.305915	1950	2031
avrg_iid	19.921000	0.093971	0.113751	0.155195	0.213059	19.500000	20.310000
Duracao_A6							
count_iid	50.750000	1.691627	2.047759	2.799159	4.375170	43	60
max_iid	14.950000	0.088450	0.104850	0.143050	0.223807	14	15
min_iid	5.000000	0.000000	0.000000	0.000000	0.000000	5	5
sum_iid	506.350000	17.135616	20.743151	26.354589	44.322116	434	597
avrg_iid	9.981500	0.121028	0.146508	0.200267	0.313045	9.098039	10.413043
Duracao_A7							
count_iid	49.200000	1.657450	2.006386	2.742604	4.287067	40	56
max_iid	30.000000	0.000000	0.000000	0.000000	0.000000	30	30
min_iid	20.150000	0.141636	0.171454	0.234367	0.366348	20	21
sum_iid	1239.950000	39.785092	48.160901	65.832937	102.905923	1002	1381
avrg_iid	25.212804	0.155486	0.188219	0.257284	0.402171	24.660714	26.125000
Quantidade_de_Casos							
count_iid	99.550000	0.342911	0.415113	0.567175	0.887011	97	100
max_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
min_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
sum_iid	99.550000	0.342911	0.415113	0.567175	0.887011	97	100
avrg_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1.000000	1.000000
R2_sobrecarregado_em_A3							
count_iid	100.000000	0.000000	0.000000	0.000000	0.000000	100	100
max_iid	79.400000	1.739886	2.108177	2.879013	4.500292	71	85
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	3958.100000	102.363202	123.913350	169.381794	264.767008	3409	4351
avrg_iid	39.581000	1.023632	1.239133	1.693818	2.647670	34.090000	43.510000
R2_sobrecarregado_em_A7							
count_iid	49.200000	1.657450	2.006386	2.742604	4.287067	40	56
max_iid	47.250000	1.710128	2.070155	2.829773	4.423323	38	54
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	1172.100000	81.152166	98.236832	134.283601	209.903712	712	1527
avrg_iid	23.638946	0.999773	1.097989	1.497211	2.324716	19.550000	27.267857
R2_sobregarado_em_A2							
count_iid	100.000000	0.000000	0.000000	0.000000	0.000000	100	100
max_iid	93.850000	0.578566	0.700369	0.957361	1.496487	91	97
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	4687.150000	42.493962	51.440060	70.315342	109.912537	4508	4898
avrg_iid	46.871500	0.424940	0.514401	0.703153	1.099125	45.080000	48.980000

Figura 84 – Resultado da Simulação - Distribuição Normal.

Statistics							
Name	Avrg	90% Half Length	95% Half Length	99% Half Length	Std	Min	Max
<b>Duracao_A1</b>							
count_iid	100.000000	0.000000	0.000000	0.000000	0.000000	100	100
max_iid	41.550000	5.135902	6.217145	8.498448	13.284241	23	81
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	754.050000	41.287213	49.979257	68.318517	106.791225	579	922
avrg_iid	7.540500	0.412872	0.499793	0.683185	1.067912	5.790000	9.220000
<b>Duracao_A2</b>							
count_iid	58.650000	2.140769	2.591457	3.542360	5.537195	48	68
max_iid	138.500000	11.492529	13.912009	19.016845	29.725941	75	194
min_iid	0.350000	0.226999	0.274788	0.375818	0.587143	0	2
sum_iid	1791.150000	137.526552	166.179510	227.567070	355.718587	1104	2585
avrg_iid	30.423056	1.758344	2.128521	2.909555	4.548034	22.530612	41.693548
<b>Duracao_A3</b>							
count_iid	60.650000	2.310437	2.796845	3.823112	5.978049	48	71
max_iid	113.000000	10.751603	13.015098	17.790825	27.809502	76	172
min_iid	0.300000	0.220851	0.267346	0.365445	0.571241	0	2
sum_iid	1514.450000	106.897534	129.402278	176.884815	276.495260	994	2081
avrg_iid	24.895911	1.237846	1.498442	2.048281	3.201745	19.460317	30.062500
<b>Duracao_A5</b>							
count_iid	25.250000	1.462137	1.769955	2.419418	3.781882	16	32
max_iid	84.550000	10.561491	12.784963	17.476244	27.317769	33	136
min_iid	1.050000	0.539151	0.652656	0.892141	1.394538	0	5
sum_iid	538.750000	47.075928	56.986529	77.897018	121.769739	305	783
avrg_iid	21.396961	1.564351	1.893688	2.588554	4.046265	14.925926	32.434783
<b>Duracao_A6</b>							
count_iid	8.600000	1.067302	1.291998	1.766080	2.760625	5	15
max_iid	27.250000	4.679703	5.661904	7.743569	12.104262	8	59
min_iid	0.850000	0.360830	0.436791	0.597070	0.933302	0	3
sum_iid	88.950000	16.722151	20.242604	27.670373	43.252593	24	173
avrg_iid	10.200556	1.446126	1.750574	2.392925	3.740470	4.400000	19.222222
<b>Duracao_A7</b>							
count_iid	8.200000	1.449844	1.755075	2.399078	3.750088	3	17
max_iid	76.100000	14.359907	17.383045	23.761534	37.142542	29	190
min_iid	3.900000	2.654659	3.213534	4.392700	6.866394	0	31
sum_iid	222.550000	50.502828	61.135002	83.567721	130.627826	87	504
avrg_iid	26.920447	3.576577	4.329540	5.918210	9.250975	14.750000	48.625000
<b>Quantidade de Casos</b>							
count_iid	11.900000	1.630170	1.973364	2.697465	4.216509	5	23
max_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
min_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
sum_iid	11.900000	1.630170	1.973364	2.697465	4.216509	5	23
avrg_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1.000000	1.000000
<b>R2_sobrecarregado_em_A3</b>							
count_iid	60.650000	2.310437	2.796845	3.823112	5.978049	48	71
max_iid	35.950000	3.463626	4.192810	5.731309	8.958824	18	55
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	1105.550000	149.508624	180.984124	247.393970	386.710755	368	1981
avrg_iid	17.883530	1.878199	2.273609	3.107881	4.858045	7.215686	27.901408
<b>R2_sobrecarregado_em_A7</b>							
count_iid	8.200000	1.449844	1.755075	2.399078	3.750088	3	17
max_iid	3.950000	1.053018	1.274706	1.742443	2.723678	1	11
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	20.300000	8.551708	10.352141	14.150728	22.119532	1	93
avrg_iid	1.982472	0.503349	0.609317	0.832899	1.301935	0.333333	5.470588
<b>R2_sobregarredo em A2</b>							
count_iid	58.650000	2.140769	2.591457	3.542360	5.537195	48	68
max_iid	32.950000	2.985442	3.613956	4.940051	7.721978	19	53
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	988.700000	117.908092	142.730849	195.104137	304.974563	443	1711
avrg_iid	16.600502	1.524112	1.844977	2.521969	3.942183	9.040816	25.161765

Figura 85 – Resultado da Simulação - Distribuição Exponencial: Cenário 1.

Statistics							
Name	Avrg	90% Half Length	95% Half Length	99% Half Length	Std	Min	Max
<b>Duracao_A1</b>							
count_iid	100.000000	0.000000	0.000000	0.000000	0.000000	100	100
max_iid	47.450000	5.099007	6.172483	8.437397	13.188811	30	77
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	788.500000	34.890847	42.236288	57.734362	90.246738	631	976
avrg_iid	7.885000	0.348908	0.422363	0.577344	0.902467	6.310000	9.760000
<b>Duracao_A2</b>							
count_iid	92.600000	1.618061	1.958705	2.677427	4.185187	84	98
max_iid	147.350000	14.773261	17.883422	24.445518	38.211703	100	230
min_iid	0.200000	0.158664	0.192067	0.262543	0.410391	0	1
sum_iid	2808.100000	142.038935	171.841869	235.033773	387.390069	2172	3785
avrg_iid	30.276380	1.254328	1.518397	2.075554	3.244376	25.657143	39.631579
<b>Duracao_A3</b>							
count_iid	94.250000	1.639674	1.984868	2.713191	4.241090	85	100
max_iid	132.550000	10.835040	13.116102	17.928990	28.025318	98	199
min_iid	0.100000	0.118998	0.144050	0.196908	0.307794	0	1
sum_iid	2333.400000	120.447655	145.805056	199.306386	311.543255	1833	2927
avrg_iid	24.720225	1.065197	1.289449	1.762597	2.755180	19.784946	29.670103
<b>Duracao_A5</b>							
count_iid	57.500000	2.365037	2.862940	3.913460	6.117275	44	68
max_iid	87.650000	6.191293	7.494723	10.241817	16.014056	68	119
min_iid	0.200000	0.158664	0.192067	0.262543	0.410391	0	1
sum_iid	1181.950000	76.139102	92.168387	125.988417	196.937199	829	1477
avrg_iid	20.571507	1.115912	1.350811	1.846515	2.886357	16.732143	27.769231
<b>Duracao_A6</b>							
count_iid	21.500000	2.115726	2.561141	3.500920	5.472419	10	30
max_iid	41.300000	4.475581	5.417809	7.405808	11.578292	23	89
min_iid	0.300000	0.181772	0.220040	0.300781	0.470162	0	1
sum_iid	219.200000	30.469464	36.884087	50.418239	78.810632	101	392
avrg_iid	10.160182	0.967836	1.171591	1.601491	2.503351	6.965517	16.400000
<b>Duracao_A7</b>							
count_iid	21.000000	2.049622	2.481122	3.391538	5.301440	11	28
max_iid	72.600000	8.577763	10.383607	14.193741	22.186767	34	129
min_iid	1.550000	0.633104	0.766389	1.047006	1.637553	0	6
sum_iid	532.450000	76.413963	92.501113	126.443232	197.648139	231	865
avrg_iid	24.933543	2.054438	2.486951	3.399507	5.313896	14.437500	32.791667
<b>Quantidade_de_Casos</b>							
count_iid	32.200000	2.551070	3.091769	4.226255	6.606217	22	47
max_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
min_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
sum_iid	32.200000	2.551070	3.091769	4.226255	6.606217	22	47
avrg_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1.000000	1.000000
<b>R2_sobrecarregado_em_A3</b>							
count_iid	94.250000	1.639674	1.984868	2.713191	4.241090	85	100
max_iid	73.250000	2.970848	3.596048	4.815572	7.683715	61	86
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	3486.400000	183.151255	221.709414	303.062892	473.728925	2692	4251
avrg_iid	36.875283	1.413767	1.711402	2.339380	3.656772	28.638298	42.510000
<b>R2_sobrecarregado_em_A7</b>							
count_iid	21.000000	2.049622	2.481122	3.391538	5.301440	11	28
max_iid	15.500000	1.819886	2.203020	3.011391	4.707217	8	23
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	174.450000	34.127839	41.312648	56.471804	88.273185	43	347
avrg_iid	7.799949	0.919084	1.112575	1.520821	2.377252	3.009091	12.992857
<b>R2_sobregarredo_em_A2</b>							
count_iid	92.600000	1.618061	1.958705	2.677427	4.185187	84	98
max_iid	70.350000	3.380405	4.092069	5.593603	8.743589	54	83
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	3243.000000	200.422351	242.616530	331.641611	518.401389	2115	4056
avrg_iid	34.893782	1.735373	2.100715	2.871546	4.488620	25.178571	41.814433

Figura 86 – Resultado da Simulação - Distribuição Exponencial: Cenário 2.

Statistics							
Name	Avrg	90% Half Length	95% Half Length	99% Half Length	Std	Min	Max
<b>Duracao_A1</b>							
count_iid	100.000000	0.000000	0.000000	0.000000	0.000000	100	100
max_iid	39.000000	3.384417	4.096926	5.600241	8.753946	26	58
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	739.500000	30.788697	37.270528	50.946479	79.636345	604	897
avrg_iid	7.395000	0.307887	0.372705	0.509465	0.796363	6.040000	8.970000
<b>Duracao_A2</b>							
count_iid	61.900000	2.094799	2.535810	3.466293	5.418293	51	74
max_iid	145.650000	10.978433	13.289883	18.166164	28.396210	89	215
min_iid	0.450000	0.265347	0.321210	0.439074	0.686333	0	2
sum_iid	1934.250000	128.148369	155.128973	212.048882	331.481497	1373	2517
avrg_iid	31.162788	1.616155	1.956398	2.674274	4.180257	24.087719	39.952381
<b>Duracao_A3</b>							
count_iid	62.100000	2.225894	2.694504	3.683218	5.757375	51	76
max_iid	112.700000	12.389672	14.997903	20.501195	32.046184	81	204
min_iid	0.300000	0.181772	0.220040	0.300781	0.470162	0	1
sum_iid	1508.050000	94.562772	114.470724	156.474315	244.590845	1089	1940
avrg_iid	24.278217	1.263513	1.529516	2.090752	3.268133	18.290323	30.312500
<b>Duracao_A5</b>							
count_iid	32.800000	1.604390	1.942156	2.654805	4.149826	23	41
max_iid	87.350000	11.017775	13.337307	18.231263	28.497969	52	141
min_iid	0.300000	0.181772	0.220040	0.300781	0.470162	0	1
sum_iid	676.250000	50.939347	61.663420	84.290036	131.756903	409	959
avrg_iid	20.660043	1.307692	1.592993	2.163839	3.392377	15.078947	27.827586
<b>Duracao_A6</b>							
count_iid	12.550000	1.232042	1.491420	2.038677	3.186732	4	19
max_iid	33.700000	5.241488	6.344932	8.873125	13.557288	19	88
min_iid	1.350000	1.262320	1.528071	2.088778	3.265046	0	15
sum_iid	126.150000	13.638586	16.509644	22.567923	35.276763	82	206
avrg_iid	10.755070	2.053479	2.485790	3.397919	5.311414	6.666867	32.250000
<b>Duracao_A7</b>							
count_iid	10.100000	0.929404	1.125068	1.537897	2.403944	5	13
max_iid	69.850000	11.751628	14.225655	19.445590	30.396113	35	166
min_iid	3.600000	1.717129	2.078630	2.841367	4.441432	0	20
sum_iid	248.250000	34.488167	41.748834	57.068043	89.205189	83	407
avrg_iid	24.923758	3.470473	4.201099	5.742639	8.976534	11.583333	46.800000
<b>Quantidade_de_Casos</b>							
count_iid	16.200000	1.444408	1.748494	2.390082	3.736027	9	24
max_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
min_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
sum_iid	16.200000	1.444408	1.748494	2.390082	3.736027	9	24
avrg_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1.000000	1.000000
<b>R2_sobrecarregado_em_A3</b>							
count_iid	62.100000	2.225894	2.694504	3.683218	5.757375	51	76
max_iid	36.650000	2.714368	3.285814	4.491501	7.020834	28	55
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	1128.250000	125.206525	151.565793	207.180953	323.852285	716	2000
avrg_iid	17.983825	1.449568	1.754741	2.398621	3.749374	12.437500	26.315789
<b>R2_sobrecarregado_em_A7</b>							
count_iid	10.100000	0.929404	1.125068	1.537897	2.403944	5	13
max_iid	5.100000	0.638364	0.772756	1.056309	1.651156	2	7
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	28.550000	5.485498	6.640340	9.076929	14.188487	4	53
avrg_iid	2.859815	0.379289	0.460138	0.627613	0.981045	0.800000	4.076923
<b>R2_sobregarredo_em_A2</b>							
count_iid	61.900000	2.094799	2.535810	3.466293	5.418293	51	74
max_iid	34.650000	3.113908	3.769465	5.152623	8.054257	16	50
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	1087.950000	124.028009	150.139169	205.230847	320.804002	406	1856
avrg_iid	17.339590	1.534218	1.857211	2.538691	3.988323	7.960784	25.081081

Figura 87 – Resultado da Simulação - Distribuição Exponencial: Cenário 3.

Statistics							
Name	Avrg	90% Half Length	95% Half Length	99% Half Length	StD	Min	Max
<b>Duracao_A1</b>							
count_iid	100.000000	0.000000	0.000000	0.000000	0.000000	100	100
max_iid	43.050000	5.343105	6.467969	8.841309	13.820180	24	77
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	767.100000	37.918665	45.901542	62.744534	98.078325	590	982
avrg_iid	7.671000	0.379187	0.459015	0.627445	0.980783	5.900000	9.820000
<b>Duracao_A2</b>							
count_iid	99.500000	0.443479	0.536843	0.733831	1.147079	96	100
max_iid	168.550000	21.280141	29.391719	40.176682	62.801672	94	337
min_iid	0.200000	0.158664	0.192067	0.262543	0.410391	0	1
sum_iid	2995.500000	90.853324	109.980340	150.336241	234.996193	2601	3464
avrg_iid	30.114707	0.948172	1.147787	1.568953	2.452190	26.010000	34.640000
<b>Duracao_A3</b>							
count_iid	99.700000	0.356994	0.432150	0.590723	0.923381	96	100
max_iid	137.400000	11.602147	14.044704	19.198231	30.009472	93	195
min_iid	0.200000	0.158664	0.192067	0.262543	0.410391	0	1
sum_iid	2538.750000	105.515450	127.729229	174.597862	272.920439	2131	3101
avrg_iid	25.462477	1.049570	1.270532	1.736737	2.714759	21.310000	31.010000
<b>Duracao_A5</b>							
count_iid	72.650000	2.350941	2.845876	3.890135	6.080815	59	81
max_iid	108.300000	9.305118	11.264090	15.397307	24.068105	68	147
min_iid	0.150000	0.141636	0.171454	0.234367	0.366348	0	1
sum_iid	1507.200000	85.528684	103.534723	141.525486	221.223773	1118	2045
avrg_iid	20.719973	0.857384	1.037886	1.418725	2.217662	16.831169	26.907895
<b>Duracao_A6</b>							
count_iid	26.250000	2.010384	2.433623	3.326611	5.199949	19	37
max_iid	38.050000	4.860459	5.883713	8.042667	12.571794	19	61
min_iid	0.250000	0.212685	0.257461	0.351933	0.550120	0	2
sum_iid	250.800000	33.659684	40.745909	55.897107	87.062228	141	437
avrg_iid	9.418122	0.764239	0.925131	1.261596	1.976738	5.423077	12.636364
<b>Duracao_A7</b>							
count_iid	27.250000	1.658754	2.007966	2.744763	4.290442	20	34
max_iid	80.450000	15.486746	18.747114	25.626131	40.057163	33	204
min_iid	0.350000	0.189195	0.229025	0.313063	0.469360	0	1
sum_iid	627.500000	91.787584	111.111286	151.882174	237.412699	241	1095
avrg_iid	22.438630	2.155028	2.608718	3.565955	5.574077	10.478261	33.181818
<b>Quantidade de Casos</b>							
count_iid	43.100000	2.292062	2.774602	3.792707	5.928522	31	52
max_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
min_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
sum_iid	43.100000	2.292062	2.774602	3.792707	5.928522	31	52
avrg_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1.000000	1.000000
<b>R2_sobrecarregado_em_A3</b>							
count_iid	99.700000	0.356994	0.432150	0.590723	0.923381	96	100
max_iid	75.500000	1.780556	2.155410	2.946311	4.605489	69	85
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	3734.550000	98.070998	118.717522	162.279421	253.865025	3225	4125
avrg_iid	37.450407	0.930650	1.126576	1.539960	2.407168	32.250000	41.250000
<b>R2_sobrecarregado_em_A7</b>							
count_iid	27.250000	1.658754	2.007966	2.744763	4.290442	20	34
max_iid	21.650000	1.853191	2.243337	3.066501	4.793362	14	30
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	304.350000	42.395358	51.320697	70.152180	109.657493	159	501
avrg_iid	10.850964	0.888754	1.075860	1.470634	2.298803	7.608696	15.181818
<b>R2_sobregarredo em A2</b>							
count_iid	99.500000	0.443479	0.536843	0.733831	1.147079	96	100
max_iid	70.650000	2.023645	2.419676	3.348554	5.234250	62	81
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	3536.150000	107.862025	130.569820	178.480771	278.989959	3223	4255
avrg_iid	35.541941	1.082731	1.310674	1.791610	2.800532	32.290000	42.550000

Figura 88 – Resultado da Simulação - Distribuição Exponencial: Cenário 4.