

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



MECANISMO DE ALOCAÇÃO DE RECURSO *FUZZY*
PARA SISTEMAS DE GERENCIAMENTO DE
WORKFLOW

Joslaine Cristina Jeske

Setembro

2006

Joslaine Cristina Jeske

Mecanismo de Alocação de Recurso *Fuzzy* para Sistemas de Gerenciamento de *Workflow*

Submetido em atendimento parcial dos requisitos para a obtenção do grau de Mestre em Ciência da Computação junto à Universidade Federal de Uberlândia, Minas Gerais.

©Todos os direitos reservados

Dados Internacionais de Catalogação na Publicação (CIP)

J58m Jeske, Joslaine Cristina, 1973-

Mecanismo de alocação de recurso fuzzy para sistemas de gerenciamento de workflow / Joslaine Cristina Jeske. - 2006.

96 f. : il.

Orientador: Stéphane Julia.

Dissertação (mestrado) - Universidade Federal de Uberlândia, Programa de Pós-Graduação em Ciência da Computação.

Inclui bibliografia.

1. Engenharia de software - Teses. 2. Inteligência Artificial - Teses. I. Julia, Stéphane. II. Universidade Federal de Uberlândia. Programa de Pós-Graduação em Ciência da Computação. III. Título.

CDU: 681.3.06

Elaborada pelo Sistema de Bibliotecas da UFU / Setor de Catalogação e Classificação

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE COMPUTAÇÃO

Os abaixo assinados, por meio deste, certificam que leram e recomendam para a Faculdade de Computação a aceitação da dissertação intitulada “**Mecanismo de Alocação de Recurso *Fuzzy* para Sistemas de Gerenciamento de *Workflow*** ” por **Joslaine Cristina Jeske** como parte dos requisitos exigidos para a obtenção do título de **Mestre em Ciência da Computação**.

Uberlândia, 29 de setembro de 2006

Orientador:

Prof. Dr. Stéphane Julia
Universidade Federal de Uberlândia UFU/MG

Banca Examinadora:

Prof. Dr. Ricardo Lüders
Universidade Tecnológica Federal do Paraná UTFPR/PR

Prof. Dr. João Nunes de Souza
Universidade Federal de Uberlândia UFU/MG

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Data: Setembro de 2006

Autor: **Joslaine Cristina Jeske**

Título: **Mecanismo de Alocação de Recurso *Fuzzy*
para Sistemas de Gerenciamento de *Workflow***

Faculdade: **Faculdade de Computação**

Grau: **Mestre** Convocação: **Setembro** Ano: **2006**

A Universidade Federal de Uberlândia possui permissão para distribuir e ter cópias desse documento para propósitos exclusivamente acadêmicos, desde que a autoria seja devidamente divulgada.

Joslaine Cristina Jeske

O AUTOR RESERVA OS DIREITOS DE PUBLICAÇÃO E ESSE DOCUMENTO NÃO PODE SER IMPRESSO OU REPRODUZIDO DE OUTRA FORMA, SEJA NA TOTALIDADE OU EM PARTES SEM A PERMISSÃO ESCRITA DO AUTOR.

“ Alguns homens vêm as coisas como são, e dizem: Por quê? Eu sonho com as coisas que nunca foram e digo: Por que não?” George Bernard Shaw

Agradecimentos

A Deus por tudo que fez e faz por mim.

Aos meus pais, Paulo e Neusa, pelo apoio, carinho e amizade. Pelas madrugadas que me buscaram na rodoviária, por estarem sempre ao meu lado incondicionalmente.

Ao meu irmão Emerson, minha cunhada Regina e minhas sobrinhas Nayla e Nathália. Família é o bem maior que uma pessoa pode possuir. Sem o apoio de todos, meus sonhos não se tornariam realidade.

Ao meu orientador Stéphane Julia pela imensa capacidade de direcionar o trabalho. Sempre me senti segura com sua orientação.

A todos os professores que contribuíram para o meu aprendizado.

Aos meus queridos amigos que sempre estiveram presentes na minha vida. Principalmente ao Gustavo do Carmo que nunca mediu esforços para me ajudar.

Resumo

O objetivo deste trabalho é o de propor um modelo de rede de Petri p-temporal com recursos híbridos *fuzzy* como solução para o problema de alocação de recursos humanos em sistemas de gerenciamento de *workflow*. Inicialmente, uma rede de Petri ordinária é usada para mostrar as principais atividades do sistema e as diferentes rotas do processo de *workflow*. Mecanismos de alocação de recursos híbridos são modelados por redes de Petri híbridas *fuzzy* com transições discretas onde recursos discretos representam equipamentos e recursos contínuos representam a disponibilidade de um funcionário.

Para expressar de forma mais realista o mecanismo de alocação de recursos onde o comportamento humano é considerado, conjuntos *fuzzy* delimitados pela distribuição de possibilidade na forma triangular são associados com as marcações dos lugares que representam a disponibilidade humana. Novas regras de disparo e evolução das marcações *fuzzy* do novo modelo são definidas. O modelo de alocação de recurso *fuzzy* é aplicado para o mecanismo de alocação de recursos de um "Serviço de Reclamação". Finalmente, um jogador de redes de Petri é aplicado ao modelo afim de obter um cenário admissível correspondente a uma seqüência específica de atividades que respeite as restrições temporais.

Palavras-chave: Rede de Petri, Conjuntos Fuzzy, Teoria da Possibilidade, Sistema de Gerenciamento de Workflow.

Abstract

In this paper, an approach based on a fuzzy hybrid Petri net model is proposed to solve the resource allocation problem of Workflow Management Systems. Initially, an ordinary Petri net model is used to show the main activities of the system and the different routings of the Workflow Process. Hybrid resource allocation mechanisms are modeled by hybrid Petri net with discrete transitions where discrete resources represent equipment and continuous resources represent employees availability. To express in a more realistic way the resource allocation mechanisms when human behavior is considered, fuzzy sets delimited by possibility distributions of the triangular form are associated with the marking of the places which represent human availability. New firing rules and the fuzzy marking evolution of the new model are defined. Such a fuzzy resource allocation model is applied to an example of resource allocation mechanism of a “Handle Complaint Process”. Finally, a token player algorithm is applied to the model in order to obtain an acceptable scenario corresponding to a specific sequence of activities which respects the time constraints.

Key-words: Petri Nets, Fuzzy Set, Possibility Theory, Workflow Management Systems.

Conteúdo

1	Introdução	1
2	Fundamentos Teóricos	4
2.1	Sistemas de Gerenciamento de <i>Workflow</i>	4
2.1.1	Conceitos sobre <i>Workflow</i>	4
2.1.2	Modelagem de <i>Workflow</i>	8
2.2	Redes de Petri	12
2.2.1	Componentes conservativos, invariante de lugar	22
2.2.2	Componentes Repetitivos, Invariantes de Transição	24
2.3	Conjuntos <i>Fuzzy</i> e Teoria das Possibilidades	24
2.3.1	Conjuntos <i>Fuzzy</i>	24
2.3.2	Teoria das Possibilidades	27
2.4	Problema do Escalonamento	30
2.4.1	Definição	30
2.4.2	Abordagem para o Problema do Escalonamento	32
3	Modelagem de Sistemas de Gerenciamento de <i>Workflow</i>	35
3.1	Modelagem de Workflow	35
3.2	Modelo de Alocação de Recurso	42
3.2.1	Mecanismo de Alocação de Recurso Discreto	43
3.2.2	Mecanismos de Alocação de Recurso Contínuo	44
3.2.3	Mecanismo de Alocação de Recurso Contínuo <i>Fuzzy</i>	46
4	Problema de Escalonamento em Sistemas de Gerenciamento de <i>Workflow</i>	57
4.1	Mecanismo de Propagação de Restrição Relativo aos Prazos de Entrega . .	57

4.2	Definição de Resolução de Situações de Conflito	58
4.3	Jogador de Rede de Petri p-temporal	61
4.3.1	Simulação Usando Recursos Discretos	63
4.3.2	Simulação Usando Recursos Discretos e Contínuos	72
4.3.3	Simulação Usando Recursos Discretos e Contínuos <i>Fuzzy</i>	79
5	Conclusão	88

Lista de Figuras

2.1	Itens da Representação Gráfica de <i>Workflow</i> - Modelo Casati	9
2.2	Tipos de Encadeamento de Atividades - Modelo Casati	10
2.3	Itens da Representação Gráfica - Diagrama de Atividades UML	10
2.4	Exemplo de uma <i>WF-net</i>	11
2.5	Exemplo de uma Rede de Petri Marcada	13
2.6	Exemplo de Disparo de uma Transição	14
2.7	Representação de uma Operação	14
2.8	Seqüência de Operações	15
2.9	Paralelismo	15
2.10	Caminhos Alternativos	16
2.11	Junção de Caminhos Alternativos	16
2.12	Sincronização do tipo Rendez-Vous	16
2.13	Comunicação Assíncrona tipo Semáforo	17
2.14	Compartilhamento de Recursos	17
2.15	Capacidade Limitada	17
2.16	Evolução de uma Rede p-temporal	19
2.17	Rede de Petri Contínua - Transferência de um Lote para um Reator.	20
2.18	Evolução da Marcação - Rede de Petri Contínua	20
2.19	Representação de Lugares e Transições em uma Rede de Petri Híbrida	21
2.20	Rede de Petri Híbrida - Transferência de um Lote para um Reator.	21
2.21	Exemplo de uma Rede de Petri Marcada	23
2.22	Representação de um Conjunto <i>Fuzzy</i> para Tamanhos Médios	25
2.23	Representação de um Conjunto <i>Fuzzy</i>	25
2.24	Exemplo de Soma Fuzzy	26
2.25	Exemplo de Subtração <i>Fuzzy</i>	27

2.26	Exemplo de Multiplicação <i>Fuzzy</i>	27
2.27	Distribuição de Possibilidade de Passageiros em um Vagão de Metrô	29
2.28	Cálculo da Possibilidade para que $a \leq b$	29
2.29	Cálculo da Necessidade para que $a \leq b$	29
2.30	Cálculo de Possibilidade entre dois Eventos	30
2.31	Cálculo de Necessidade entre dois Eventos	30
2.32	Escalonamento Flexível	33
3.1	Representação de uma Atividade	35
3.2	Representação de uma Seqüência de Atividades	36
3.3	Representação de um Roteiro Alternativo	36
3.4	Junção de um Roteiro Alternativo	37
3.5	Representação de duas Atividades Concorrentes	37
3.6	Sincronização de duas Atividades Concorrentes	37
3.7	Serviço de Reclamações	38
3.8	Bloco Bem Formado onde a Atividade1 está contida na Atividade1'	39
3.9	Modelo p-temporal - Intervalos Estáticos	40
3.10	Modelo p-temporal com Intervalos Estáticos para o “Serviço de Reclamações”	41
3.11	Alocação de Recurso Discreto	44
3.12	Alocação de Recurso Contínuo	46
3.13	Recurso Contínuo <i>Fuzzy</i>	48
3.14	Cálculo de Possibilidade Associado a t3	48
3.15	Marcação após Disparo de t3	50
3.16	Cálculo de Possibilidade Associado a t4	50
3.17	Marcação após o Disparo de t4	51
3.18	Cálculo de Possibilidade Associado a t14	51
3.19	Marcação após o Disparo de t14	52
3.20	Cálculo de Possibilidade Associado a t5	53
3.21	Marcação após Disparo de t5	53
3.22	Cálculo da Possibilidade Associado a t6	54
3.23	Marcação após o Disparo de t6	54
3.24	Cálculo da Possibilidade Associado a t16	55
3.25	Marcação após Disparo de t16	55

4.1	Modelo p-temporal para o “Serviço de Reclamações” após Aplicação dos Mecanismos de Propagação de Restrição	59
4.2	Conflito em uma Rede de Petri p-temporal	60
4.3	Jogador de Rede de Petri p-temporal	62
4.4	Jogador de Rede de Petri p-temporal com Recursos Discretos e Contínuos <i>Fuzzy</i>	64
4.5	Modelo p-temporal para o “Serviço de Reclamações” com Recursos Discretos	65
4.6	Seqüência de Atividades - Recursos Discretos	71
4.7	Modelo p-temporal para o “Serviço de Reclamações” com Recursos Discretos e Contínuos	73
4.8	Seqüência de Atividades - Recursos Contínuos	78
4.9	Modelo p-temporal para o “Serviço de Reclamações” com Recursos Discretos e Contínuos <i>Fuzzy</i>	80
4.10	Cálculo de Possibilidade Associado a t4	81
4.11	Cálculo de Possibilidade Associado a t3	81
4.12	Cálculo de Possibilidade Associado a t4	81
4.13	Cálculo de Possibilidade Associado a t3	82
4.14	Cálculo de Possibilidade Associado a t4	83
4.15	Cálculo de Possibilidade Associado a t14	84
4.16	Cálculo de Possibilidade Associado a t14	85
4.17	Cálculo de Possibilidade Associado a t14	86
4.18	Seqüência de Atividades - Recursos <i>Fuzzy</i>	87

Lista de Acrônimos

BPR - *Business Process Re-engineering*

FAPEMIG - *Fundação de Amparo à Pesquisa do Estado de Minas Gerais*

MIT - *Massachusetts Institute of Technology*

UML - *Unified Modeling Language*

WfMC - *Workflow Management System*

Capítulo 1

Introdução

Atualmente, as pessoas estão exigindo um nível cada vez maior de qualidade, além de agilidade e confiabilidade na execução de serviços. Com isso, surge a necessidade de se tratar o trabalho em equipe, o gerenciamento e reuso da informação e o aperfeiçoamento constante dos processos de negócio.

A fim de enfrentar esses novos desafios, as empresas têm adotado sistemas e ferramentas tecnológicas que lhes permitem lidar com a informação necessária garantindo qualidade e exatidão. O desenvolvimento de pacotes de *software* genéricos para gerenciamento de processo de negócios - denominados sistemas de gerenciamento de *workflow*- são particularmente importantes neste contexto. Os sistemas de gerenciamento de *workflow* ajudam as empresas a atingir seus objetivos de coordenação, comunicação e cooperação entre os membros da organização.

É indiscutível o fato de que a tecnologia de informação é sobretudo concretizada pelas funcionalidades que são implementadas no *software*. No entanto, problemas resultantes do processo de desenvolvimento podem ter sérios impactos em uma organização. Para evitar que tais problemas ocorram torna-se necessário o uso dos princípios da Engenharia de *Software*, que têm como principal objetivo aplicar técnicas de engenharia (modelar antes de realizar; estimar diversos fatores antes de avançar; medir antes, durante e depois do produto realizado; analisar fatores de risco) para o desenvolvimento de *software*.

Dentro do contexto de desenvolvimento de *software*, umas das técnicas fundamentais é a modelagem. Através desta, é possível construir modelos que representem a estrutura e o comportamento desejado para o sistema. Se esta fase for suprimida, a probabilidade de erros ou da construção de itens errados aumentará.

De todas as notações usadas para a modelagem de processos de *workflow*, redes de Petri são as mais adaptadas [Aalst e Hee, 2002] já que representam os roteiros básicos presentes em processos de *workflow*. Além disso, redes de Petri podem ser usadas para especificar características de tempo real em sistemas de gerenciamento de *workflow* e mecanismos de alocação de recursos. De fato, em uma organização, entregas atrasadas, se devem, geralmente, ao problema de recursos sobrecarregados e o modelo usado no sistema de gerenciamento de *workflow* deve considerar o mecanismo de alocação de recursos.

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 são representados 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

Em [Francielle, 2005], uma proposta para o escalonamento em tempo real dos sistemas de gerenciamento de *workflow* foi apresentado, cujo principal objetivo foi a obtenção de uma seqüência de atividades admissível que respeitasse as restrições temporais e as restrições de alocação dos recursos compartilhados. Em particular, recursos discretos foram usados para representar equipamentos (recursos disjuntivos) e recursos contínuos foram usados para representar funcionários (recursos cumulativos).

A idéia central deste trabalho está baseada no fato de usar um modelo baseado em rede de Petri, associado a conjuntos *fuzzy* a fim de representar o mecanismo de alocação de recursos do tipo humano de forma mais próxima da realidade, inclusive mostrando que em algumas situações, para que os prazos sejam mantidos, um funcionário pode trabalhar acima da sua capacidade normal de trabalho.

O capítulo 2 será dedicado aos principais fundamentos teóricos que serão utilizados neste trabalho. Os conceitos sobre sistemas de gerenciamento de *workflow*, redes de Petri, conjuntos *fuzzy* e teoria das possibilidades, e por fim o problema do escalonamento serão abordados.

O capítulo 3 apresentará uma nova forma de representação de mecanismo de alocação

de recursos do tipo humano. Para tanto, os tipos de alocação de recursos apresentados em [Francielle, 2005] serão formalizados e, baseados nestes, um novo modelo de alocação de recurso *fuzzy* será definido.

O capítulo 4 abordará o problema do escalonamento do sistema de gerenciamento de *workflow*. Um algoritmo do jogador de rede de petri p-temporal será apresentado e diversas simulações considerando recursos discretos, contínuos e *fuzzy* serão realizadas.

Para concluir este trabalho, as contribuições e algumas sugestões para futuros trabalhos, além dos resultados obtidos serão apresentados.

Capítulo 2

Fundamentos Teóricos

Este capítulo apresenta os fundamentos teóricos que serão utilizados neste trabalho.

2.1 Sistemas de Gerenciamento de *Workflow*

2.1.1 Conceitos sobre *Workflow*

Num contexto de grandes e rápidas mudanças, as organizações estão adotando novos modelos de gestão de organização social do trabalho mais flexíveis, mais participativos e com estruturas hierárquicas mais reduzidas. As organizações estão tomando também consciência da importância cada vez maior da necessidade de aprendizagem e da gestão do seu conhecimento. Como resposta a estas exigências, uma constante reengenharia e melhoria de cada processo de negócio torna-se necessária. No entanto, a reengenharia dos processos de negócio (BPR - *Business Process Re-engineering*) é um processo de mudança muito complexo que requer várias capacidades em nível tecnológico e organizacional.

A adoção de novos modelos de trabalho é acompanhada pela adoção de tecnologias e sistemas de informação, tais como os processadores de texto, correio eletrônico, sistemas que usam *data warehouses* conectados a ferramentas para análises estatísticas, sistemas de suporte a decisão, sistemas de gerenciamento de *workflow*. Dentre os sistemas citados, destacam-se os sistemas de gerenciamento de *workflow* que se tornaram, nos últimos anos, um componente padrão para sistemas de informação empresariais.

Os sistemas de gerenciamento de *workflow* são pacotes de software genéricos para gerenciamento de processos de negócio [Aalst e Hee, 2002], [Alonso et al, 1997], [Eshuis, 2002], [Hollingsworth, 1985]. 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 e a alocação dos recursos apropriados associados com as diversas atividades. Esses sistemas permitem o gerenciamento das tarefas fazendo com que a informação certa chegue até a pessoa certa no momento certo. Um sistema de gerenciamento de *workflow*, no entanto, não executa nenhuma das tarefas em um processo.

Vários conceitos são atribuídos a *workflow* mas, em geral, todos contemplam a idéia que *workflow* é direcionado a processos de negócios. Segundo [Eshuis, 2002], por exemplo, *workflow* é um processo de negócios operacional. Um processo de negócios consiste em um conjunto de atividades relacionadas que, coletivamente, visam atingir um objetivo, dentro do contexto de uma estrutura organizacional [Hollingsworth, 1985].

Segundo a Coalizão para Especificações de Gerenciamento de *Workflow* (WfMC), *workflow* é a automação de processos de negócios, no todo ou em parte, no qual documentos, informações ou atividades são passadas de um participante para outro, de acordo com um conjunto de regras [WFMC, 2006].

Vale ressaltar que apesar da definição de *workflow* ser tradicionalmente ligada a processos de negócios, seus princípios e ferramentas podem ser aplicados a diversas atividades onde a coordenação do trabalho seja exigida [Plesums, 2002]. A distribuição das atividades pode ser feita automaticamente para pessoas, equipamentos ou sistemas computacionais. Sistemas que manipulam *workflow* são ferramentas poderosas para a organização de tarefas complexas. Podem ser usados desde o sequenciamento de DNA [Meidanis e Weske, 1996] até pesquisas científicas para comprovação dos princípios da teoria da relatividade - como a detecção de ondas gravitacionais [Gil, 2004].

Workflow pode ser classificado de diversas formas. Destaca-se a classificação quanto a estruturação dos processos. Esta abordagem propõe três categorias para os diferentes tipos de aplicações baseados no grau de frequência e fluxos de trabalho que apóiam [Chaffey, 1998], [Freitas Filho, 2000], [Georgakopoulos e Shet, 2004], [Marshak, 1995], [Nicolao, 1998], [Usinoro, 2003].

Ad-hoc*: são os que possuem um fluxo de trabalho pouco estruturado. As tarefas e o fluxo de interação entre elas são normalmente imprevisíveis ou desconhecidas até o momento da execução. Não há um padrão pré-determinado de movimentação das atividades

*Ad Hoc é uma expressão em latim que significa “para isto”, “para um fim específico”.

entre pessoas. A ordenação e a coordenação de tarefas em um *workflow* do tipo *ad-hoc* não são automatizadas, necessitam da coordenação humana ou co-decisão [Shael, 1991]. Estes sistemas estão voltados para grupos dinâmicos que executam processos únicos e altamente individualizados. Tipicamente envolvem pequenos grupos de profissionais que têm a intenção de apoiar pequenas atividades que requerem uma solução rápida. Exemplos destes tipos de processos são aqueles que envolvem a produção de conhecimento como a elaboração de um relatório, revisão de um livro ou artigo, ou a construção de um sistema.

Administrativos: estes gerenciam processos com maior grau de estruturação. São processos repetitivos que possuem maior previsibilidade na interação de suas tarefas. Sua ordenação e coordenação podem ser facilmente automatizadas. Geralmente apóiam processos administrativos que podem mudar bastante de uma organização para outra. São concebidos para o roteamento inteligente de formulários através da organização e praticamente todos os indivíduos podem ser seus usuários. Um exemplo pode ser um pedido de compra de materiais, que geralmente é solicitado por uma pessoa uma vez por semana e tramita por diversas áreas que vão gradativamente preenchendo as informações no documento eletrônico até a finalização do processo.

Produção: seguem uma estruturação rígida e bem definida. As regras de interação do processo são previamente levantadas através de uma análise básica do processo. Estes processos são previsíveis e repetem-se com alta frequência e geralmente possuem uma forte dependência de outros sistemas da organização como banco de dados. *Workflow* de produção difere do administrativo por um processamento de informações complexas envolvendo múltiplos sistemas. Parte das tarefas é executada por agentes de *software* e não por atores humanos. Processos de atendimento a clientes em um *Call Center* são um exemplo típico deste tipo de *workflow*.

É importante salientar a diferença que existe entre sistema de *workflow* e sistema de gerenciamento de *workflow*. O primeiro refere-se a sistemas que suportam processos específicos, gerenciando os fluxos de trabalho e o encadeamento das atividades para cada instância do processo. Cada instância é um caso particular da execução de um processo. Um sistema de *workflow*, geralmente envolve um sistema de gerenciamento de *workflow*, definições de processo, infra-estrutura como banco de dados, comunicação e aplicativos. O segundo refere-se à ferramenta básica que provê as características e funcionalidades para a implementação e controle do processo. Da mesma maneira que sistemas de bancos de

dados são criados e administrados usando-se um sistema gerenciador de bando de dados, os sistemas de *workflow* são criados e administrados pelo sistema de gerenciamento de *workflow* [Aalst e Hee, 2002].

Os sistemas que manipulam *workflow* são baseados nos seguintes conceitos fundamentais [Aalst e Hee, 2002], [Moreno e Meziat, 2000]:

- **caso:** o principal objetivo de um sistema de *workflow* é lidar com casos. Um paciente em um hospital, um pedido, ou uma reclamação são exemplos de casos. Cada caso tem uma única identidade, um tempo de vida limitado e envolve a execução de um processo;
- **processo:** um processo consiste em um número de tarefas que precisam ser executadas e um conjunto de condições que determinam a ordem das tarefas. Um processo também pode ser chamado *procedimento*;
- **tarefa:** uma tarefa é uma unidade lógica de trabalho que é executada de uma só vez por um recurso, o que significa que uma tarefa é indivisível e não pode ser interrompida;
- **recurso:** um recurso é um nome genérico para uma pessoa, uma máquina ou um grupo de pessoas ou máquinas que podem executar tarefas específicas;
- **atividade:** uma atividade é a execução de uma tarefa realizada por um recurso. Uma atividade pode ser manual ou automatizada;
- **evento:** algo que ocorre em um determinado instante de tempo específico, por exemplo, a ocorrência de um sinistro, acidente de carro;
- **gatilho:** disparo de uma tarefa por um evento. Pode ser visto como uma regra que é avaliada em função da ocorrência de um evento. Disparo significa dar início à execução.
- **roteamento:** as tarefas podem ser opcionais, ou seja, podem haver tarefas que só precisam ser executadas para alguns casos. A ordem em que as tarefas são executadas também varia de caso para caso. O roteamento das tarefas determina quais tarefas podem ser executadas e em que ordem. Um processo pode seguir um roteiro sequencial (quando duas ou mais tarefas são executadas em seqüência), pode seguir

um roteiro paralelo (quando duas ou mais tarefas são executadas simultaneamente) ou pode seguir um roteiro seletivo (quando uma escolha precisa ser feita entre duas ou mais tarefas). Alguns processos também podem incluir iteração ou repetição.

Os sistemas de gerenciamento de *workflow*, de um modo geral, precisam suportar as seguintes funcionalidades [Hollingsworth, 1985]:

- funções referentes à definição e, possivelmente, modelagem dos processos de negócio e das atividades que constituem estes processos;
- funções de controle em tempo de execução referentes ao gerenciamento dos processos de *workflow* em um ambiente operacional e o sequenciamento das várias atividades de cada processo;
- interações em tempo de execução entre usuários humanos e ferramentas de Tecnologia da Informação para processar as diversas atividades.

Pela sua natureza, os sistemas de gerenciamento de *workflow* implicam que os projetos relativos a sua adoção e utilização apresentem uma coordenação perfeita entre os processos de negócios e a tecnologia. No entanto, para que o projeto seja bem sucedido, torna-se obrigatório que toda a metodologia de implementação seja conduzida cuidadosamente prestando atenção em diferentes aspectos de ordem institucional. As organizações devem apostar não só no desenvolvimento e implementação, mas também na modelagem dos processos de negócios.

2.1.2 Modelagem de *Workflow*

Uma empresa ou organização que consegue distinguir bem seus principais processos de negócios, desenvolver uma modelagem bem estruturada destes processos, e, por fim, executar a implementação do controle do fluxo de informações através de um sistema de *workflow* poderá extrair disso tudo diversas vantagens e benefícios que vão desde a confiança nos dados até um controle geral dos processos.

As técnicas de modelagem de sistemas de *workflow* devem objetivar a minimização dos problemas de coordenação do trabalho nos processos de negócios, baseando-se no comportamento dinâmico do processo e oferecendo recursos para representação do fluxo de trabalho ao longo do processo modelado.

Muitos trabalhos estão sendo relacionados à modelagem de sistemas de *workflow*. Mas cada um utiliza sua própria técnica de modelagem, isto é, não há um modelo conceitual comum, aceito e usado por todos na área de *workflow*. Alguns [Gerrit, 2000] reconhecem inclusive, o problema de adaptabilidade inerente aos modelos de sistemas de *workflow* devido às mudanças constantes e/ou radicais das regras dos processos de negócios.

Um modelo de *workflow* é uma representação gráfica ou textual de um conjunto de atividades e o relacionamento existente entre estas, que responde as questões de quando, como, onde cada atividade é executada e por quem [Araujo e Borges, 2001].

Um dos padrões usados para representação gráfica de *workflow* é o modelo de Casati [Casati, 1996], representado na figura 2.1. Este padrão possibilita a representação das atividades e o encadeamento entre as mesmas. Os tipos de encadeamento estão representados na figura 2.2. Diagramas de atividades da UML [Booch et al., 1986] também podem ser usados para modelagem de sistemas de *workflow* (figura 2.3)[Eshuis, 2002]. Tanto o modelo de Casati quanto diagramas de atividades têm a vantagem de mostrar os principais roteiros existentes em um *workflow*. No entanto, não é possível representar mecanismos de alocação de recursos e restrições temporais quando são utilizados para a especificação de características de tempo real dos sistemas de gerenciamento de *workflow*.

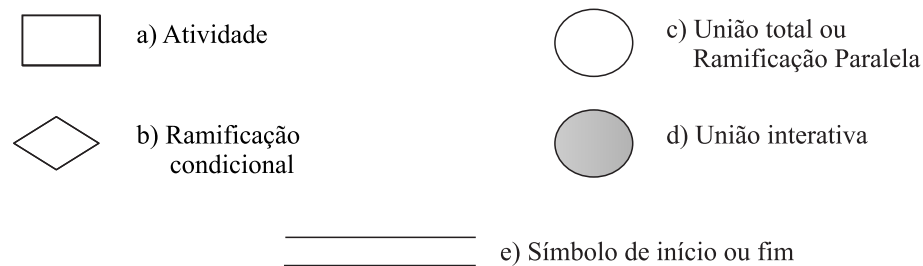
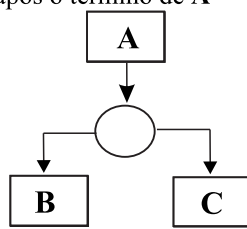


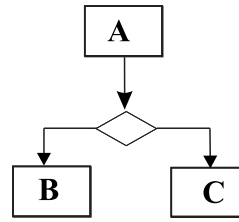
Figura 2.1: Itens da Representação Gráfica de *Workflow* - Modelo Casati

As redes de Petri são adequadas para modelar sistemas de *workflow* já que permitem uma boa representação de situações de conflito, compartilhamento de recursos, restrições de precedência, comunicações síncronas e assíncronas e explícitas restrições temporais, no caso das redes de Petri temporais. A proposta apresentada em [Wirtz e Giese, 2000], por exemplo, integra uma estrutura de modelagem padrão orientada a objetos, usando UML e as técnicas de redes de Petri para especificar o comportamento a fim de prover um suporte adequado para modelar todos os aspectos dos sistemas de *workflow*. Em [Covès et al., 1998], um modelo de empresa foi traduzido automaticamente para uma

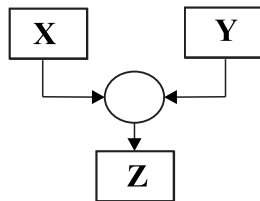
a) **Paralelo:** Tanto **B** e **C** são ativadas simultaneamente após o término de **A**



b) **Condicional:** Apenas **B** ou **C** são ativadas após o término de **A**



c) **União total:** **Z** só pode ser executado após a finalização de **X** e **Y**



c) **União interativa:** **Z** só pode ser executado após o término de **X** ou **Y**

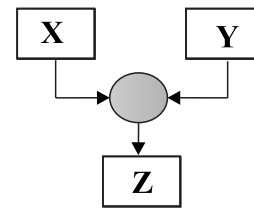


Figura 2.2: Tipos de Encadeamento de Atividades - Modelo Casati

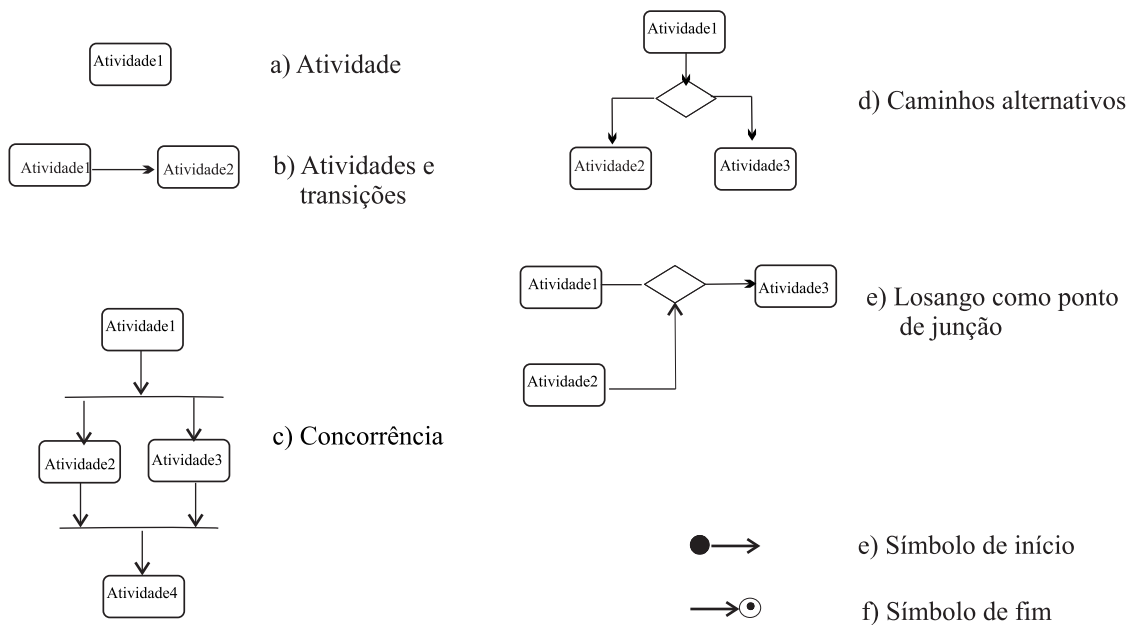


Figura 2.3: Itens da Representação Gráfica - Diagrama de Atividades UML

rede de Petri equivalente a fim de se verificar boas propriedades. Em [Aalst e Hee, 2002] foram utilizadas redes de Petri de alto nível para a modelagem de *workflow* e de sistemas de gerenciamento de *workflow* a fim de formalizar alguns conceitos e de poder analisá-los. A proposta apresentada em [del Foyo e Silva, 2003] utiliza um modelo de rede de Petri Orientada a Objetos na análise de *workflow*.

Existem modelos baseados em redes de Petri que foram definidos exclusivamente para a representação de *workflow*: as *WF-nets* [Li et al.,2003, Aalst e Hee, 2002], com um lugar de entrada (ε) e um lugar de saída (θ) indicando o início e o fim do processo de negócio modelado. As *WF-nets* requerem que todas as transições e lugares estejam situados no caminho entre (ε) e (θ). Um exemplo de uma *WF-net* é mostrado na 2.4. Essas redes são adequadas para representação, validação e para verificação de *workflow*. As *WF-nets* podem ser extendidas com informações de tempo para representar as restrições temporais, dando origem às redes de *workflow* com restrições temporais (*TCWF-nets*) [Li et al.,2003].

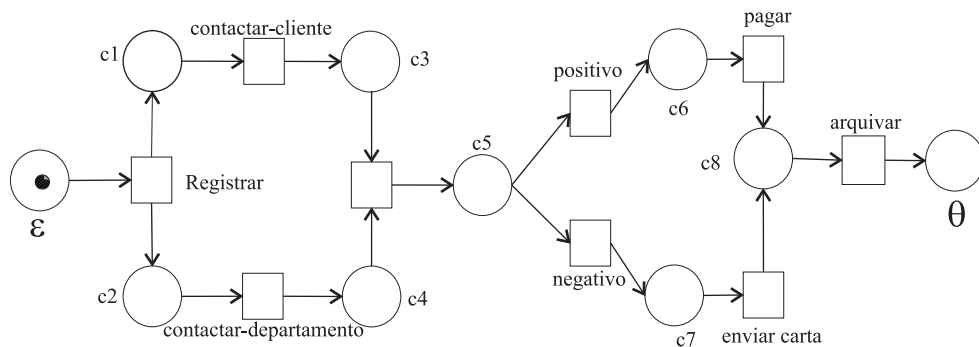


Figura 2.4: Exemplo de uma *WF-net*

Apesar de existirem muitos trabalhos sobre modelagem de *workflow* utilizando redes de Petri, a maioria considera a dimensão dos processos relacionados ao sequenciamento das atividades ou da análise Markoviana [Francielle, 2005]. Estes trabalhos, no entanto possuem algumas limitações no que se refere ao problema de alocação de recursos. Em alguns deles, como por exemplo em [Aalst e Hee, 2002], a modelagem da alocação de recursos é feita apenas de modo informal.

2.2 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 Darmsdadt. É a partir deste trabalho que *Anatol W. Holt*, com a ajuda de pesquisadores do MIT (*Massachusetts Institute of Technology*), desenvolveu as bases da teoria das redes de Petri entre 1968 e 1976. As redes de Petri são consideradas como uma ferramenta gráfica e matemática de representação formal que permite a modelagem, a análise e o controle de sistemas a eventos discretos que comportam atividades paralelas, concorrentes e assíncronas. A definição das redes de Petri e suas propriedades podem ser vistas em [Cardoso et al, 1999, David e Alla, 2004, Murata, 1989, Peterson, 1981, Petri, 1966].

Os elementos básicos que permitem a definição de uma rede de Petri são os lugares e as transições conectadas por arcos. Os arcos são rotulados por seus pesos.

A definição de uma rede de Petri autônoma é a seguinte [Cardoso et al, 1999]:

Definição 2.2.1 *Uma rede de Petri pode ser formalmente definida como uma quádrupla $R = \langle P, T, Pre, Post \rangle$*

onde:

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

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

Definição 2.2.2 *Uma rede marcada N é uma dupla $N = \langle R, M \rangle$ onde:*

- R é uma rede de Petri,
- 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 2.5 onde a marcação é dada por $M^T = [1 \ 0 \ 3 \ 0 \ 1]$ (M^T é o vetor transposto).

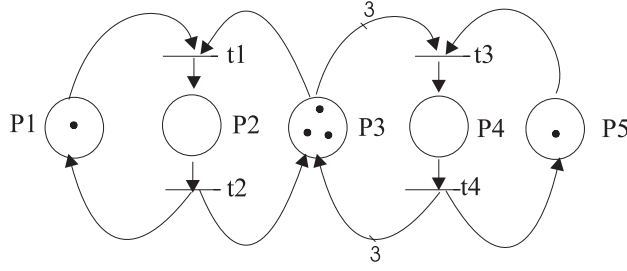


Figura 2.5: Exemplo de uma Rede de Petri Marcada

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

Definição 2.2.3 *Uma transição t está sensibilizada ou habilitada se e somente se:*

$$\forall p \in P, \quad M(p) \geq \text{Pre}(p, t) \quad (2.1)$$

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.

Definição 2.2.4 *Se t está sensibilizada por uma marcação M , uma nova marcação M' é obtida através do disparo de t de maneira que:*

$$\forall p \in P, \quad M'(p) = M(p) - \text{Pre}(p, t) + \text{Post}(p, t) \quad (2.2)$$

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 2.6.

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.

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

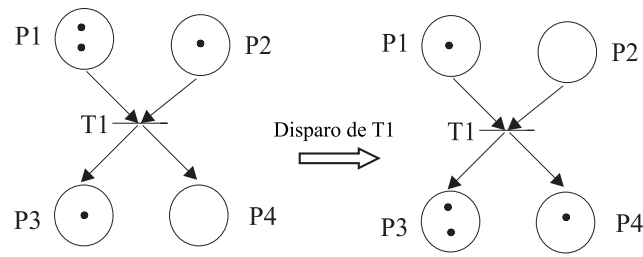


Figura 2.6: Exemplo de Disparo de uma Transição

como condições ou estados, mas também entidades físicas como peças ou depósito. Uma transição pode ser associada a um evento que ocorre no sistema, uma operação. 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 autoriza procedimentos de análise.

A característica marcante encontrada nas redes de Petri é a capacidade de representar graficamente relações e visualizar vários conceitos como a representação de uma operação, (figura 2.7), uma seqüência de operações (figura 2.8), evoluções paralelas (figura 2.9), caminhos alternativos (figura 2.10), junção de caminhos alternativos (figura 2.11), sincronizações do tipo *Rendez-vous* (figura 2.12), comunicações assíncronas do tipo semáforo (figura 2.13), compartilhamento de recursos (figura 2.14), e a noção de capacidade (figura 2.15).

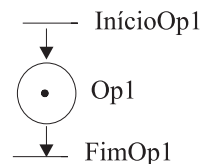


Figura 2.7: Representação de uma Operação

As redes de Petri têm sido utilizadas no domínio das mais diversas aplicações. Dependendo da complexidade do sistema a ser modelado, pode ser necessário a utilização de outros modelos de redes de Petri que ofereçam mais recursos do que o simples modelo autônomo. Alguns tipos de redes de Petri de alto nível são apresentadas a seguir.

- **Redes de Petri Coloridas** [David e Alla, 2004, Jensen, 1990]: nessas redes são associadas cores às fichas com o objetivo de diferenciá-las. Em [Moncelet et al., 1998],

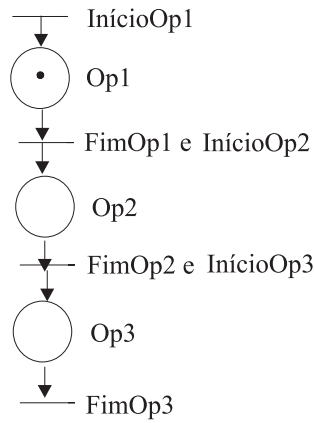


Figura 2.8: Sequência de Operações

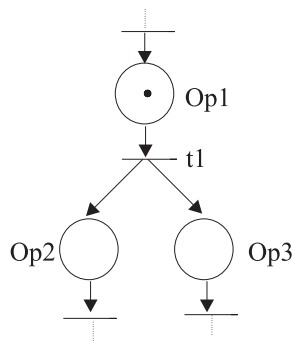


Figura 2.9: Paralelismo

por exemplo, foi utilizado um modelo de rede de Petri Colorida para a modelagem de sistemas de produção híbridos

- **Redes de Petri Predicado/Transição:** [Cardoso et al, 1999, Genrich, 1987]: são uma maneira estruturada de descrever o conjunto *controle* e *dados*. As redes de Petri predicado/transição foram usadas, por exemplo, em [Champagnat et al., 1998] onde foram integradas a uma sequência de equações diferenciais e algébricas para a modelagem de uma estocagem de gás.
- **Redes de Petri a Objetos** [Silbertin, 1985, Cardoso et al, 1999]: a rede de Petri a objetos pode ser considerada como uma utilização da rede de Petri predicado/transição no contexto de uma abordagem a objetos. Em [Sanders, 1998], as redes de Petri a objetos foram usadas para a modelagem de sistemas de produção complexos com problemas baseados em restrições. A vantagem dessa modelagem é a possibilidade de se aproveitar a capacidade das redes de Petri para represen-

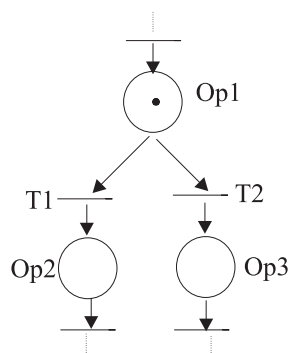


Figura 2.10: Caminhos Alternativos

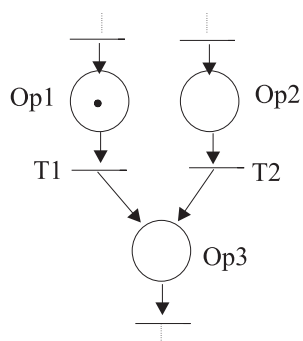


Figura 2.11: Junção de Caminhos Alternativos

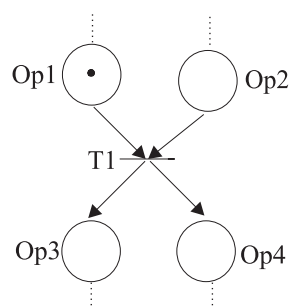


Figura 2.12: Sincronização do tipo Rendez-Vous

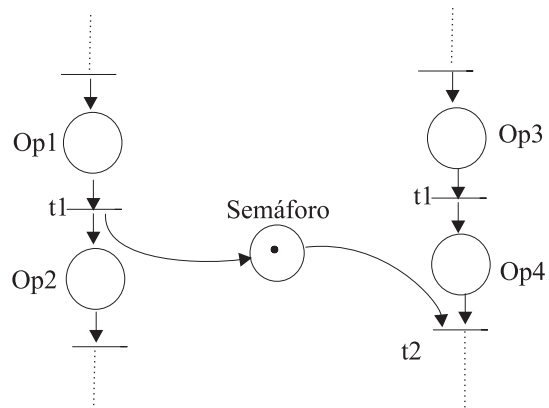


Figura 2.13: Comunicação Assíncrona tipo Semáforo

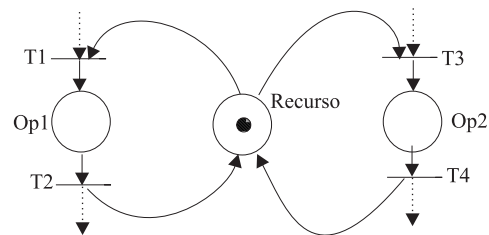


Figura 2.14: Compartilhamento de Recursos

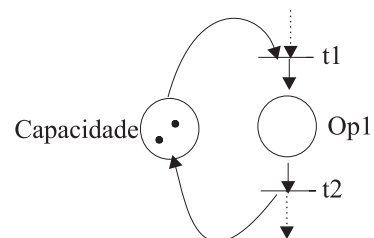


Figura 2.15: Capacidade Limitada

tar concorrência, controle de fluxo e restrições e ao mesmo tempo, beneficiar-se da modularidade da orientação a objetos.

- **Redes de Petri temporais/temporizadas:** esses modelos são empregados em sistemas onde o tempo é um fator importante. De um modo geral, as redes de Petri temporais são bastante adequadas para serem utilizadas em aplicações de simulação, diagnóstico e supervisão, análise de desempenho, mas também podem ser usadas para resolver o problema do escalonamento de sistemas de produção [Julia, 1998]. Nas redes de Petri p-temporizadas o tempo é representado por durações associadas aos lugares do modelo [Sifakis, 1977, Tazza, 1987], enquanto nas redes de Petri t-temporizadas, o tempo é representado por durações (números racionais positivos ou nulos) associadas às transições [Ramchandani, 1974]. Nas redes de Petri t-temporais [Merlin, 1974, Menasche, 1982] o tempo é representado por um intervalo $[\theta_{min}, \theta_{max}]$ associado a cada transição ; a duração de sensibilização deve ser maior que θ_{min} e menor que θ_{max} , ou seja, a transição só pode ser disparada dentro do intervalo de tempo $[\theta_{min}, \theta_{max}]$. Já nas redes de Petri p-temporais, o tempo é representado por um intervalo de tempo $[\theta_{min}, \theta_{max}]$ associado ao lugar p . θ_{min} representa a data mínima depois da qual uma ficha que está no lugar p torna-se disponível para o disparo de uma transição. θ_{max} representa a data máxima antes da qual a ficha em p deve ser utilizada para o disparo de uma transição. Depois de θ_{max} a ficha torna-se “morta”, o que caracteriza uma violação de restrição e um funcionamento anormal do sistema. A principal vantagem das redes de Petri p-temporais é a capacidade de expressar um intervalo mínimo e máximo entre dois eventos.

O comportamento dinâmico de uma rede de Petri p-temporal depende da marcação da rede e da situação temporal das fichas que é dada pelo intervalo de visibilidade cuja definição é a seguinte:

Definição 2.2.5 *Um intervalo de visibilidade $[(\delta_p)_{min}, (\delta_p)_{max}]$ associado a uma ficha em um lugar p de uma rede de Petri p-temporal define:*

- a data mínima $((\delta_p)_{min})$ em que a ficha se torna disponível em p para o disparo de uma transição de saída de p ;
- a data máxima $((\delta_p)_{max})$ após a qual a ficha se torna indisponível (morta) e não pode ser utilizada para o disparo de nenhuma transição.

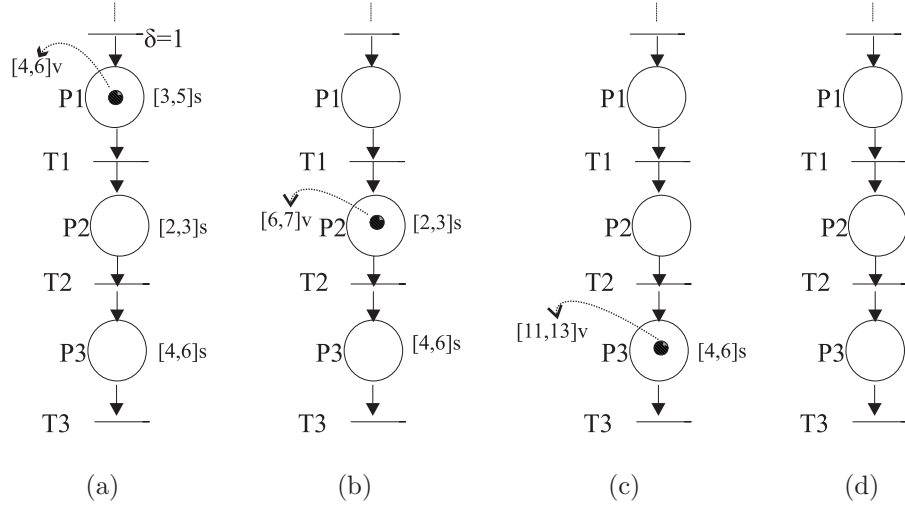


Figura 2.16: Evolução de uma Rede p-temporal

Um exemplo de evolução de uma rede de Petri p-temporal é dado pela figura 2.16. Os intervalos associados aos lugares representam as durações das operações. Supondo que a execução do modelo inicie na data $\delta = 1$. (figura 2.16(a)). Como a operação associada a $p1$ possui uma duração que pertence ao intervalo $[3, 5]_s$, obtém-se então, um intervalo dinâmico (intervalos de data) $[1 + 3, 1 + 5]_v = [4, 6]_v$ para a ficha produzida em $p1$. Isto significa que a transição $t1$ pode ser disparada entre as datas $\delta = 4$ e $\delta = 6$. Disparando $t1$ o mais cedo possível (figura 2.16(b)), por exemplo, uma nova ficha será produzida em $p2$ com o intervalo dinâmico $[4 + 2, 4 + 6]_v = [6, 7]_v$ o que representa o intervalo em que $t2$ deve ser disparada. Disparando $t2$ na data mais tarde (figura 2.16(c)) por exemplo, uma nova ficha será produzida em $p3$ com intervalo dinâmico $[7 + 4, 7 + 6]_v = [11, 13]_v$. A figura 2.16(d) mostra o disparo de $t3$ na data $\delta = 12$, finalizando a execução.

- **Redes de Petri Estocásticas** [Florin e Natkin, 1984]: nas redes de Petri estocásticas, um tempo aleatório é associado ao disparo de uma transição. Esse modelo é usado para sistemas cujos eventos não podem ser bem definidos, como, por exemplo, o tempo entre a falha de uma máquina e outra. Para esse modelo, geralmente, o tempo é definido através de uma distribuição que segue uma lei exponencial, o que permite associar a rede a um processo Markoviano equivalente.
- **Redes de Petri Contínuas e Redes de Petri Híbridas** [David e Alla, 2004]: no caso de uma rede de Petri contínua, a marcação de um lugar corresponde a uma

variável contínua (número real não negativo) e o disparo de uma transição também se torna contínuo, seguindo o modelo de funcionamento de uma ampulheta ou do fluxo de água liberado por uma torneira. A marcação contínua é progressivamente transferida de um lugar para outro respeitando uma certa velocidade de disparo (fluxo de marcas contínuo transferido de um lugar para outro). Um exemplo é apresentado na figura 2.17. A marcação de $p1$ representa o volume (em litros) de um lote de produto que se encontra em um estoque. O disparo da transição $t1$ representa a transferência contínua deste lote para um reator, por exemplo, representado pelo lugar $p2$. A inscrição $v1$ representa a velocidade de disparo de $t1$ em quantidade de marcas por unidade de tempo. Em função do tempo, considerando que inicialmente a marcação $m2$ de $p2$ é nula e a marcação $m1$ de $p1$ é igual a $M1$, a evolução das marcações de $p1$ e $p2$ é dada pela figura 2.18 .

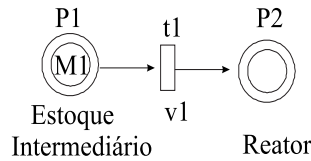


Figura 2.17: Rede de Petri Contínua - Transferência de um Lote para um Reator.

As redes de Petri híbridas são modelos que apresentam tanto uma parte discreta quanto uma parte contínua. Uma rede de Petri híbrida pode conter lugares discretos e contínuos e transições discretas e contínuas. A fim de distinguir lugares discretos de lugares contínuos e transições discretas de transições contínuas, a representação da figura 2.19 é utilizada.

No caso do exemplo da figura 2.17, deseja-se representar o fato de que o reator tem uma capacidade de enchimento V_{max} que representa o volume máximo de produto

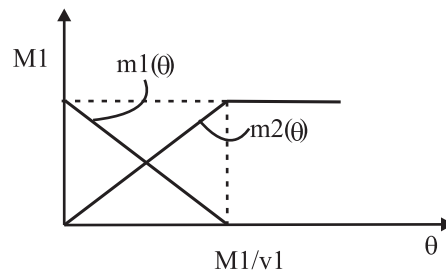


Figura 2.18: Evolução da Marcação - Rede de Petri Contínua



Figura 2.19: Representação de Lugares e Transições em uma Rede de Petri Híbrida

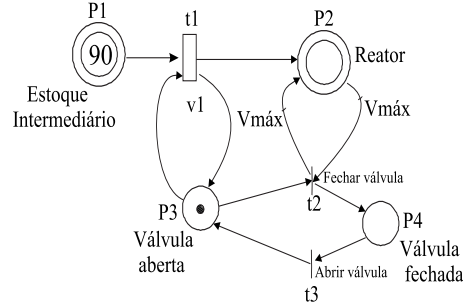


Figura 2.20: Rede de Petri Híbrida - Transferência de um Lote para um Reator.

que o reator pode conter. Após receber o volume V_{max} , a operação de enchimento deve ser interrompida, mesmo que ainda haja produto no estoque intermediário. Tal comportamento discreto pode ser representado pela rede de Petri híbrida da figura 2.20. Neste modelo, quando a marcação em $p2$ se torna igual a V_{max} , a transição $t2$ é disparada e a marca em $p3$ é consumida. Como consequência disso, a transição contínua $t1$ não está mais habilitada e a transferência de produto é interrompida.

- **Redes de Petri Fuzzy** [Scarpelli e Gomide, 1993, Cardoso et al, 1999]: é um formalismo que combina a teoria de conjunto *fuzzy* e a teoria de rede de Petri. Sendo assim, é uma ferramenta que representa a incerteza do conhecimento sobre um estado. Desde 1988, Looney [Scarpelli e Gomide, 1993] e outros autores de rede de Petri e da comunidade de Inteligência Artificial propuseram diferentes tipos de redes de Petri *fuzzy*. Os diferentes tipos de redes de Petri *fuzzy* estão relacionados a ferramenta *fuzzy* utilizada (lógica *fuzzy*, teoria da possibilidade).

Como foi mostrado, existem vários modelos de redes de Petri, cada um apropriado para modelar um tipo particular de aplicação. Mas, independentemente do modelo escolhido, é possível usar meios analíticos para fazer a verificação das propriedades de uma rede de Petri autônoma. Algumas propriedades são dependentes da marcação inicial e são agrupadas sob o nome genérico de *boas propriedades* que são [Cardoso et al, 1999]:

Definição 2.2.6 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.

Definição 2.2.7 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.

Definição 2.2.8 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*).

Definição 2.2.9 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 et al, 1999] 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.

2.2.1 Componentes conservativos, invariante de lugar

Considerando a figura 2.21 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 $\forall M \in A(R, M_0), M(p1) + M(p2) = M_0(p1) + M_0(p2)$.[†]

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.

[†]Todos os estados que podem ser atingidos em uma rede R com estado inicial M_0 são denotados como $A(R, M_0)$.

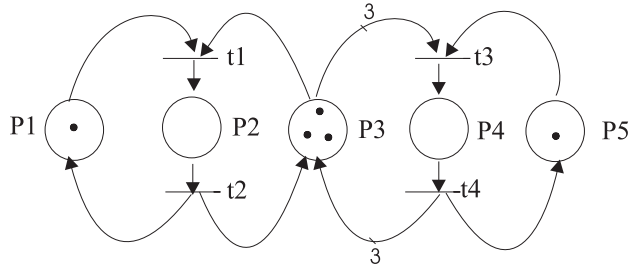


Figura 2.21: Exemplo de uma Rede de Petri Marcada

Definição 2.2.10 *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 (2.2). Para se obter uma soma ponderada das marcações, pré-multiplica-se a equação por um vetor $\mathbf{f}^T$:

$$\mathbf{f}^T M'(p) = \mathbf{f}^T M(p) + \mathbf{f}^T C s \quad (2.3)$$

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

Definição 2.2.11 *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 $\mathbf{f}$, solução da equação:*

$$\mathbf{f}^T C = 0 \quad (2.4)$$

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.

2.2.2 Componentes Repetitivos, Invariantes de Transição

Considerando novamente a figura 2.21 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 (2.2). É fácil verificar que para obter-se $M' = M$, a seqüência s deve ser tal que o vetor s verifique:

$$Cs = 0 \quad (2.5)$$

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 [Cardoso et al, 1999].

2.3 Conjuntos *Fuzzy* e Teoria das Possibilidades

2.3.1 Conjuntos *Fuzzy*

A noção de conjunto *fuzzy* foi introduzida por Zadeh [Zadeh, 1965] a fim de representar a incerteza nas informações. Por exemplo, a altura de um homem pode ser considerada 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 anexada a cada interpretação possível de uma informação simbólica e pode simplesmente ser formalizada por um conjunto *fuzzy* F de

um conjunto de referência X que pode ser definido pela função $\mu_F : X \rightarrow [0, 1]$

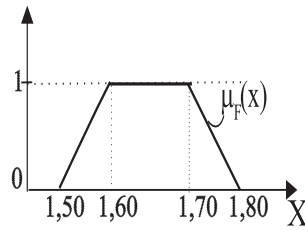


Figura 2.22: Representação de um Conjunto *Fuzzy* para Tamanhos Médios

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 certo grau de pertinência. Por exemplo, considerando que a figura 2.22 representa o conjunto *fuzzy* 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 2.23, 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.

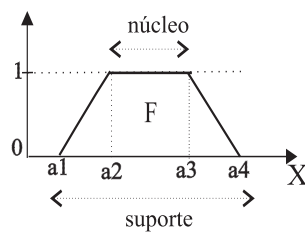


Figura 2.23: Representação de um Conjunto *Fuzzy*

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

1. forma triangular, $a_2 = a_3$,
2. caso impreciso, $a_1 = a_2$ e $a_3 = a_4$,

3. 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 e Yuan, 1995]:

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

$$[a_1, a_2, a_3, a_4] \oplus [b_1, b_2, b_3, b_4] = [a_1 + b_1, a_2 + b_2, a_3 + b_3, a_4 + b_4] \quad (2.6)$$

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

$$[a_1, a_2, a_3, a_4] \ominus [b_1, b_2, b_3, b_4] = [a_1 - b_4, a_2 - b_3, a_3 - b_2, a_4 - b_1] \quad (2.7)$$

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

$$[a_1, a_2, a_3, a_4] \otimes [b_1, b_2, b_3, b_4] = [a_1.b_1, a_2.b_2, a_3.b_3, a_4.b_4] \quad (2.8)$$

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 2.24(a)), 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 2.24(b)). 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 o produto A e B seriam gastos no mínimo R\$15,00 e no máximo R\$35,00. Esta operação pode ser visualizada na figura 2.24(c).

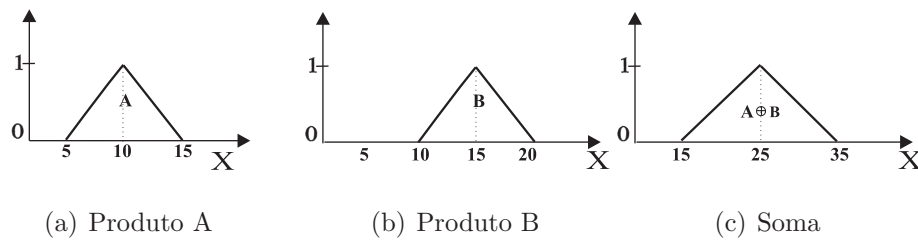


Figura 2.24: Exemplo de Soma Fuzzy

Supondo que seja necessário 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 2.25(a)). Neste caso, usando a expressão 2.7, o resultado será $R =$

$[-20, -5, -5, 10]$ (figura 2.25(b)). 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).

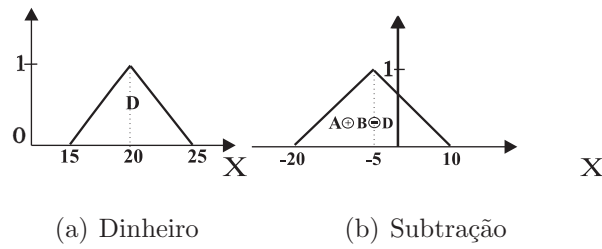


Figura 2.25: Exemplo de Subtração *Fuzzy*

No caso em que se deseja comprar quatro (4) produtos A , pode-se fazer este cálculo usando a expressão 2.8, onde a quantidade é um conjunto preciso $P = [4, 4, 4, 4]$; Assim o resultado será $R = [20, 40, 40, 60]$ (figura 2.26). No caso, para comprar quatro produtos A , pode-se dizer que serão gastos em torno de R\$40,00.

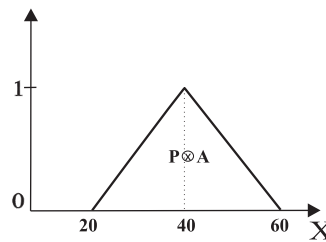


Figura 2.26: Exemplo de Multiplicação *Fuzzy*

2.3.2 Teoria das Possibilidades

A teoria das possibilidades fornece um método 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 à 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, Prade, 1988, Cardoso et al, 1999] de tal forma que:

$$\forall x \in X, \quad \Pi_f(x) = \mu_F(x) \quad (2.9)$$

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

Definição 2.3.1 *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:*

$$\Pi(S) = \sup_{x \in S} \Pi_f(x) \quad (2.10)$$

e

$$N(S) = \inf_{x \notin S} (1 - \Pi_f(x)) = 1 - \Pi(\bar{S}) \quad (2.11)$$

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.

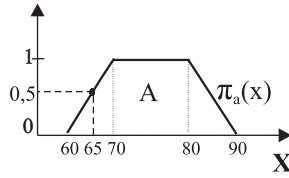


Figura 2.27: Distribuição de Possibilidade de Passageiros em um Vagão de Metrô

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 2.27. 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:

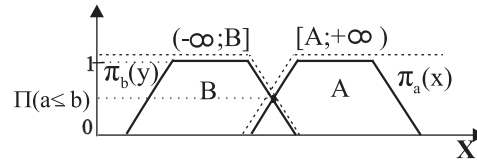


Figura 2.28: Cálculo da Possibilidade para que $a \leq b$

$$\Pi(a \leq b) = \sup_{x \leq y} \{\min(\Pi_a(x), \Pi_b(y))\} = \max\{[A, +\infty) \cap (-\infty, B]\} \quad (2.12)$$

como mostra a figura 2.28.

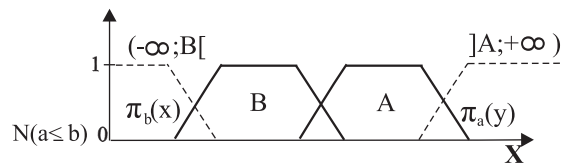


Figura 2.29: Cálculo da Necessidade para que $a \leq b$

A necessidade é então definida por:

$$N(a \leq b) = 1 - \Pi(a > b) = 1 - \sup_{x > y} \{\min(\Pi_a(x), \Pi_b(y))\} = \max\{]A, +\infty) \cap (-\infty, B[\} \quad (2.13)$$

como mostra a figura 2.29.

Por exemplo, seja X o conjunto de datas para a realização de um evento. As figuras 2.30(a) e 2.30(b) 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 2.30(c).

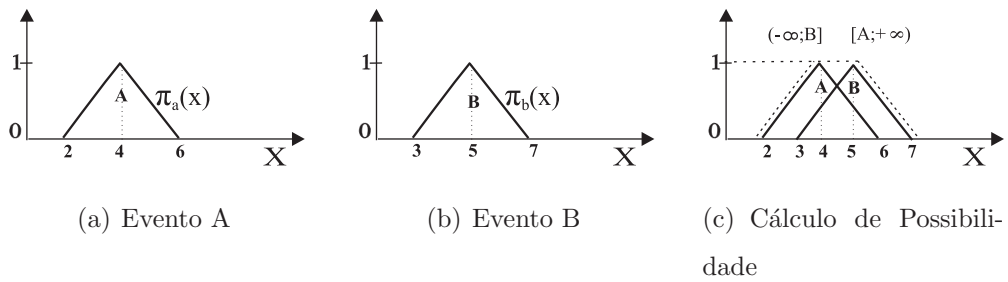


Figura 2.30: Cálculo de Possibilidade entre dois Eventos

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 2.31.

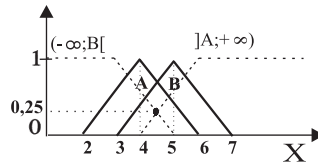


Figura 2.31: Cálculo de Necessidade entre dois Eventos

2.4 Problema do Escalonamento

2.4.1 Definição

O problema do escalonamento [Esquirol e Lopez, 1995] consiste em organizar no tempo a realização de tarefas considerando restrições temporais (intervalos de tempo) e restrições de utilização dos recursos compartilhados necessários à execução das tarefas.

As restrições temporais descrevem as relações de ordem relativa entre as diversas tarefas e são, geralmente derivadas de imperativos relacionados à gestão de projetos, às datas limites das tarefas (prazos de entrega, disponibilidade de abastecimento) ou à duração total do projeto. Já as restrições de utilização dos recursos exprimem a natureza e a quantidade dos recursos utilizados para as tarefas assim como as características de utilização destes recursos.

Existem recursos renováveis que depois de terem sido alocados para uma ou diversas tarefas, ficam disponíveis novamente na mesma quantidade (os homens, as máquinas) e recursos consumíveis (matéria bruta, dinheiro) cuja utilização no tempo é limitada.

A disponibilidade de um recurso pode variar no tempo. No caso de recursos renováveis é possível considerar dois tipos de recursos: os recursos disjuntivos, que só podem executar uma tarefa por vez (máquinas-ferramenta, robôs-manipuladores) e os recursos cumulativos, que podem ser utilizados por diversas tarefas simultaneamente, mas em número limitado (equipe de funcionários).

Do ponto de vista tradicional da Engenharia de *Software*, o problema do escalonamento é similar à atividade de execução de cenários. Uma execução de cenários se torna um tipo de simulação que mostra o comportamento do sistema em tempo real.

Para resolver um problema de escalonamento, pode se optar por obter uma solução ótima ou uma solução admissível. O problema do escalonamento ótimo supõe que as soluções candidatas a um problema possam ser ordenadas de maneira racional segundo um ou mais critérios de avaliação numérica que são vistos como indicadores de desempenho. Procura-se, então, minimizar ou maximizar tais critérios. Os critérios podem ser relacionados ao tempo (tempo total de execução, ou tempo médio de finalização de um conjunto de tarefas), aos recursos (quantidade de recursos necessários para realizar um conjunto de tarefas), aos custos de produção [Moraes, 2000].

O problema de escalonamento pode ser estático ou dinâmico. [Esquirol e Lopez, 1995]. Considera-se um escalonamento estático quando o conjunto de informações necessárias à sua resolução é fixo (conjunto de tarefas, de recursos e suas características). A solução é um plano previsional onde o conjunto dos intervalos de execução das atividades caracteriza o horizonte temporal do plano. A solução possui uma duração de validade limitada devido ao aparecimento de novas informações que ocorrem na execução do plano, como por exemplo, o aparecimento de novas tarefas, panes, atrasos e outros.

O monitoramento em tempo real da execução de um plano previsional ou da solução de um problema de escalonamento estático pode ser considerado como um problema de escalonamento dinâmico onde as situações de conflito serão resolvidas por uma função de decisão em tempo real que poderá realizar localmente ajustes devido a alguns acontecimentos (eventos) dificilmente previsíveis durante a elaboração do plano previsional.

2.4.2 Abordagem para o Problema do Escalonamento

Em [Lee e DiCesare, 1994] um modelo de rede de Petri t-temporizada foi apresentado como uma opção de modelagem do problema de escalonamento de sistemas flexíveis de manufatura. Baseado neste tipo de modelo, um algoritmo do tipo *Branch and Bound*, cuja função é minimizar o tempo global de execução do sistema, foi apresentado como uma solução possível do problema de escalonamento dos sistemas flexíveis de manufatura.

Uma outra solução que parece melhor adaptada à natureza dos sistemas de comportamento flexível é a elaboração de um conjunto de regras heurísticas que definem o que deve ser feito a cada vez que uma máquina se torna disponível para realizar a operação seguinte. Estas regras substituem o algoritmo *Branch and Bound* e podem ser aplicadas diretamente, em tempo real, à rede de Petri t-temporizada no nível de coordenação global do sistema. Fala-se então de escalonamento implícito. A limitação desta abordagem é a dificuldade de se obter um bom desempenho do sistema, ou, em outras palavras, de comparar o desempenho obtido com o da solução ótima.

Um exemplo de modelagem de escalonamento flexível é dado na figura 2.32. Neste modelo são representados em particular as durações das operações associadas as transições, os possíveis roteiros de produção e, finalmente, os recursos utilizados para a realização das operações.

Outras abordagens para o problema do escalonamento são baseados em jogadores de rede de Petri. O funcionamento normal de um jogador de redes de Petri é o disparo das transições assim que elas se tornam habilitadas. Esta estratégia permite o escalonamento das atividades de um sistema em tempo real e dá a necessária flexibilidade para se levar em conta possíveis perturbações (como atrasos, por exemplo).

Em [Julia et al, 1995] foi apresentado uma abordagem baseada em uma rede de Petri temporizada com as durações de espera associadas com os lugares (tempo de espera de uma peça em um estoque intermediário entre duas operações) e com as durações das

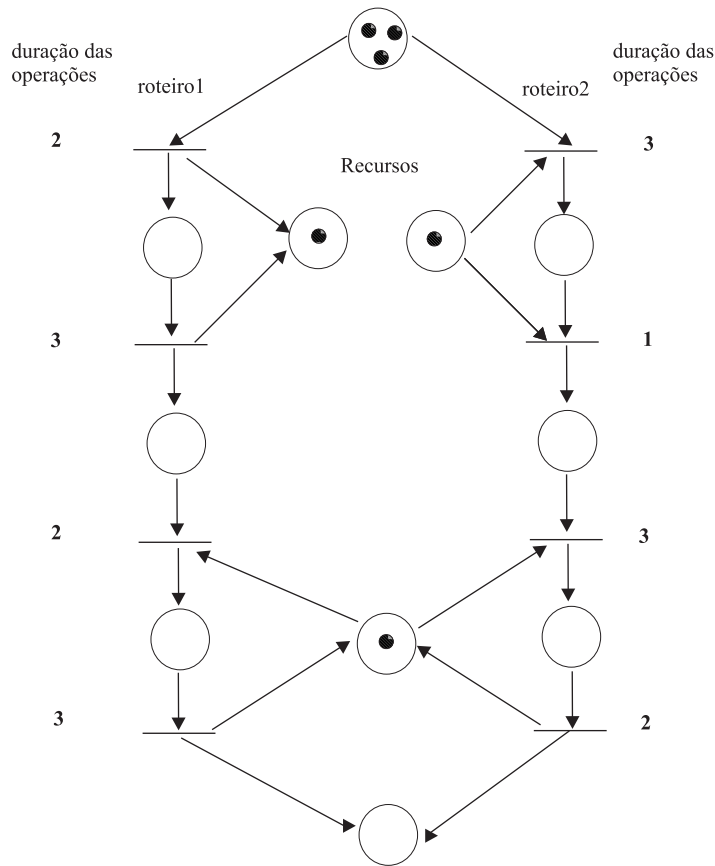


Figura 2.32: Escalonamento Flexível

operações associadas com as transições. Em uma primeira fase, a programação linear [Minoux, 1983] foi utilizada para calcular bordas mínimas e máximas que representam o funcionamento mais ou menos flexível do sistema. Em uma segunda fase, um algoritmo baseado em um jogador de rede de Petri (mecanismo de inferência especializado) associado a um mecanismo de retrocesso (*backtrack*) foi usado para calcular um escalonamento admissível que respeita as bordas obtidas a partir da programação linear. O objetivo era obter um bom equilíbrio entre flexibilidade e eficácia a fim de poder utilizar a solução obtida em tempo real usando-se um jogador de rede de Petri sem mecanismo de retrocesso, de maneira tal a suportar as possíveis perturbações (chegada atrasada das peças nos estoques de entradas das operações seguintes) durante o funcionamento real do sistema.

No caso dos sistemas de gerenciamento de *workflow*, até o momento não existem muitas propostas para o problema do escalonamento. Alguns trabalhos recentes [Aalst e Hee, 2002], por exemplo, apontam a habilidade limitada dos atuais sistemas de gerenciamento de *workflow* 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.

O problema do escalonamento é bastante complexo e segundo [Tramontina et al, 2004], não há resultados na literatura sobre o escalonamento que cubram uma proporção significativa das características de sistemas de gerenciamento de *workflow*. Em particular, uma solução para o problema do escalonamento de sistemas de gerenciamento de *workflow* deverá considerar os diferentes tipos de recursos utilizados, o compartilhamento de recursos, além das restrições temporais relativas aos prazos de entrega para os casos.

Em [Francielle, 2005], uma proposta para o escalonamento em tempo real dos sistemas de gerenciamento de *workflow* foi apresentado, cujo principal objetivo foi a obtenção de uma seqüência de atividades admissível que respeitasse as restrições temporais e as restrições de alocação dos recursos compartilhados. Em particular, recursos discretos foram usados para representar equipamentos (recursos disjuntivos) e recursos contínuos foram usados para representar funcionários (recursos cumulativos)

A proposta deste trabalho é expressar de forma mais próxima da realidade o mecanismo de alocação de recursos quando o comportamento humano é considerado. Para isto, conjuntos *fuzzy* delimitados por distribuições de possibilidade serão associados ao esforço de recursos do tipo humano para a realização de atividades e mecanismos de inferências especializados (jogador de rede de Petri p-temporal *fuzzy*) serão aplicados aos novos modelos a fim de se obter seqüências admissíveis (escalonamento admissível).

Capítulo 3

Modelagem de Sistemas de Gerenciamento de *Workflow*

Neste capítulo, uma abordagem baseada em um modelo de rede de Petri com recursos contínuos *fuzzy* será utilizada para a modelagem de sistemas de gerenciamento de *workflow*.

3.1 Modelagem de Workflow

As especificações de sistemas de gerenciamento de *workflow* devem considerar duas importantes dimensões: a dimensão dos processos ou dos fluxos de controle e a dimensão dos recursos. A dimensão dos processos se refere à ordem de execução das atividades utilizadas para a realização das tarefas no tempo (quais atividades devem ser executadas e quando). Uma atividade é uma etapa em um processo, onde um trabalho está sendo realizado. Como pode ser visto na figura 3.1 uma atividade pode ser representada por um lugar em uma rede de Petri [David e Alla, 2004] com uma transição de entrada que mostra o início da atividade e uma transição de saída que mostra o fim da atividade

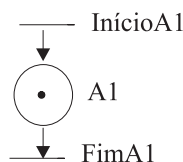


Figura 3.1: Representação de uma Atividade

Quando atividades sequenciais em redes de Petri são representadas, torna-se necessário representar lugares de espera entre duas atividades sequenciais. De fato, ao final de uma atividade a próxima só pode ser iniciada se o recurso correspondente estiver imediatamente disponível, o que não é necessariamente o caso. Um exemplo de uma seqüência de atividades pode ser visto na figura 3.2

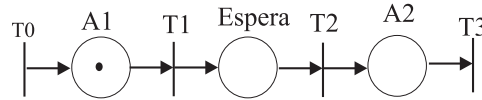


Figura 3.2: Representação de uma Seqüência de Atividades

Atividades sequenciais são comuns em *workflow*, mas não são o único tipo de roteiro encontrado. A modelagem dos fluxos de controle de *workflow* deve considerar tanto caminhos alternativos que são tomados com base em alguma expressão booleana, quanto processos concorrentes (simultâneos).

A representação de um roteiro alternativo com uma escolha entre duas atividades pode ser visto na figura 3.3. Supondo-se que o roteiro 1 seja escolhido, a junção dos dois caminhos alternativos é mostrado na figura 3.4.

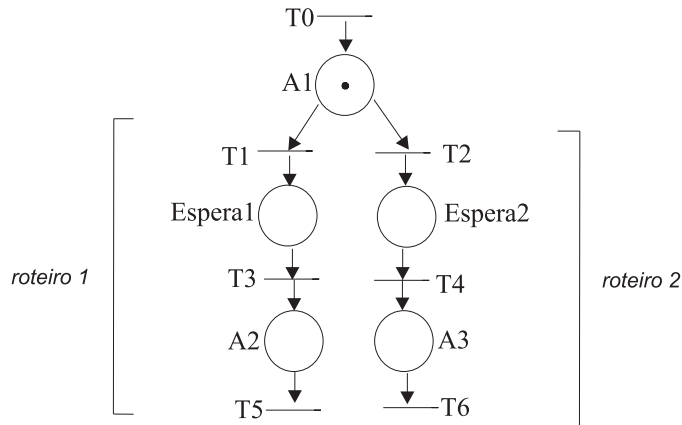


Figura 3.3: Representação de um Roteiro Alternativo

É possível também modelar fluxos concorrentes. A representação de duas atividades concorrentes pode ser vista na figura 3.5, e a sincronização de duas atividades concorrentes é mostrada ser vista na figura 3.6

Com o objetivo de ilustrar a modelagem de sistemas de *workflow* usando redes de Petri, o processo de um “Serviço de Reclamações” apresentado em [Aalst e Hee, 2002]

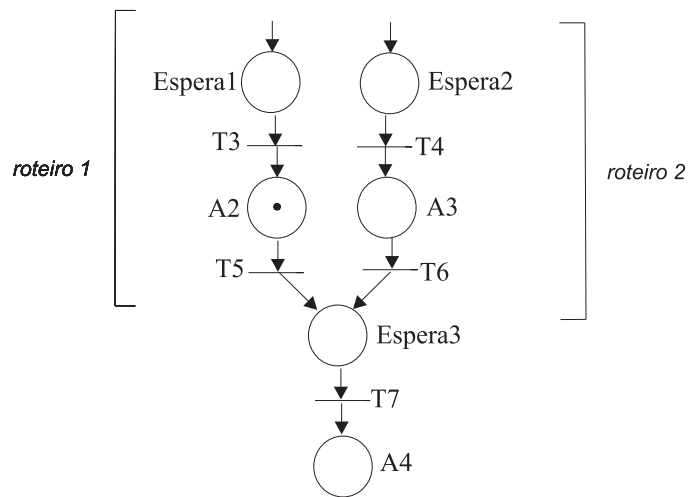


Figura 3.4: Junção de um Roteiro Alternativo

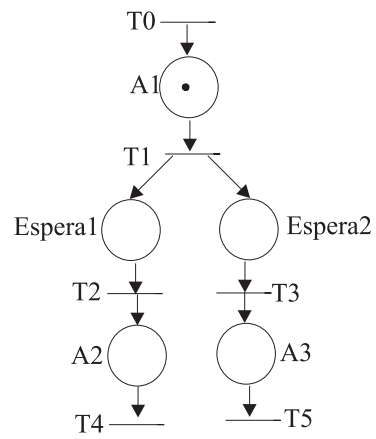


Figura 3.5: Representação de duas Atividades Concorrentes

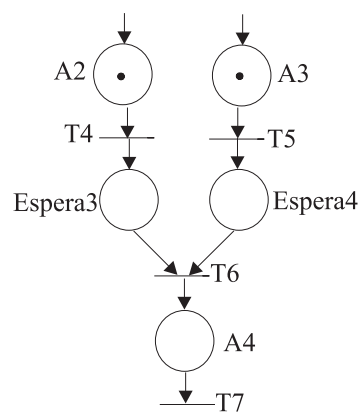


Figura 3.6: Sincronização de duas Atividades Concorrentes

será utilizado. Neste processo, primeiro uma reclamação é registrada. Depois, o cliente que fez a reclamação e o departamento afetado pela reclamação são contactados. O cliente é contactado a fim de se obter mais informações. O departamento é informado sobre a reclamação e é questionado sobre uma reação inicial. Essas duas atividades devem ser executadas em paralelo. Depois disso, as informações são coletadas e uma decisão é tomada. Dependendo da decisão, ou um pagamento é feito ou uma carta é enviada ao cliente. Finalmente, a reclamação é arquivada.

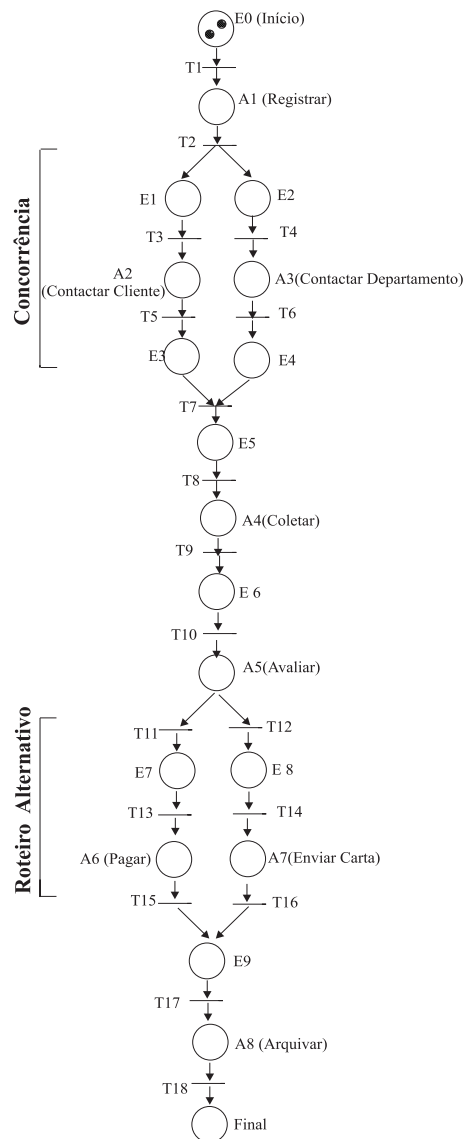


Figura 3.7: Serviço de Reclamações

A figura 3.7 mostra a rede de Petri para o “Serviço de Reclamações”, onde as atividades são representadas pelo lugares A_i (para $i=1$ até 8) e os tempos de espera são representados pelos lugares E_j (para $j=0$ até 9). Os lugares de espera “ E_3 ” e “ E_4 ” são necessários para

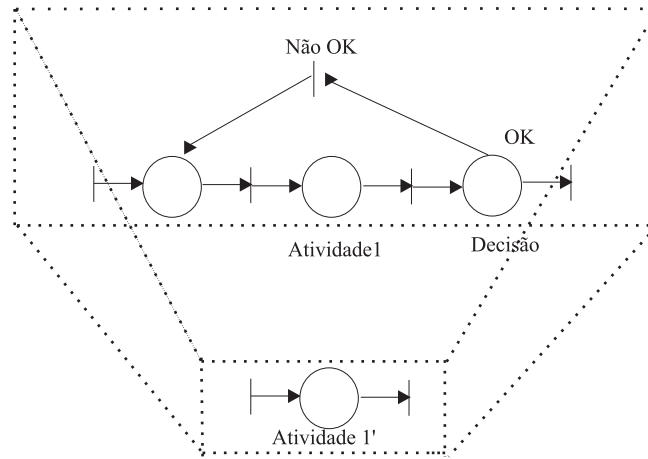


Figura 3.8: Bloco Bem Formado onde a Atividade1 está contida na Atividade1'

sincronizar o final das atividades “Contactar Cliente” e “Contactar Departamento”.

Até o momento, não foram considerados os roteiros iterativos. Estes roteiros representam a situação em que uma atividade pode ser executada mais de uma vez para um mesmo caso.

Embora o ideal seja uma atividade ser executada apenas uma vez para um determinado caso, algumas vezes pode ocorrer de uma atividade precisar ser executada diversas vezes até que o resultado de um teste subsequente seja positivo.

Na abordagem proposta, os roteiros iterativos serão substituídos por uma atividade global. Na prática, um sistema de *workflow* deve respeitar um prazo máximo e não pode repetir uma atividade indefinidamente. A estrutura hierárquica das redes de Petri baseada no conceito de “blocos bem formados” [Valette, 1979], pode ser utilizada para representar roteiros iterativos através de uma simples atividade como mostrado na figura 3.8

Posteriormente, a fim de especificar de uma maneira implícita o número de vezes que a atividade Atividade1 do bloco bem formado poderá ser executada, uma duração máxima será associada à Atividade1'.

Como o tempo real necessário para a execução de uma atividade em um sistema de *workflow* é não-determinístico e não é facilmente previsível, um intervalo de tempo pode ser atribuído a cada atividade. Como foi mostrado em [Julia, Valette, 2000] restrições de tempo explícitas que existem em sistemas de tempo real podem ser formalmente especificadas usando um modelo de rede de Petri p-temporal

Definição 3.1.1 Uma rede de Petri p-temporal [Khansa, 1997] é dada pela dupla $<$

$R; IE > onde:$

- R é uma rede de Petri marcada

- $IE : \rightarrow (Q^+ \cup 0) \times (Q^+ \cup \infty)$

$p_i \rightarrow IE_i = [a_i, b_i]$ com $0 \leq a_i \leq b_i$

IE_i representa o intervalo estático de tempo de permanência da ficha no lugar p_i

(Q^+ representa o conjunto dos números racionais positivos.)

A definição estática de uma rede de Petri p-temporal [Khansa e Denat, 1996] é baseada em intervalos estáticos que representam o intervalo de permanência das fichas nos lugares do ponto de vista das durações de permanência das fichas.

Uma ficha pertencente a um lugar p_i participará da sensibilização de uma transição de saída de p_i se ela tiver permanecido lá pelo menos durante a duração especificada pela borda mínima e no máximo durante a duração máxima especificada pela borda máxima do intervalo associado a p_i . Enquanto a ficha não alcança o tempo mínimo no lugar de permanência onde se encontra, ela fica indisponível para o disparo de outras transições. Dentro do intervalo $[a_i, b_i]$ a ficha fica disponível e após o limite máximo de permanência, a ficha se torna morta e não pode mais disparar nenhuma transição. Em termos práticos, a morte de uma ficha indica a violação de uma restrição temporal.

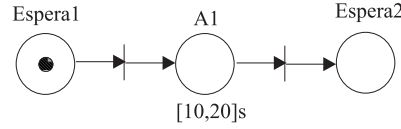


Figura 3.9: Modelo p-temporal - Intervalos Estáticos

Na rede de Petri da figura 3.9, o intervalo estático é associado ao lugar que representa a atividade. Por exemplo, o intervalo estático $[10, 20]_s$ associado ao lugar que representa a atividade “A1”, mostra que o recurso que executará esta atividade precisará de no mínimo 10 unidades de tempo e no máximo 20 unidades de tempo.

O modelo p-temporal para o “Serviço de Reclamações” é apresentado na figura 3.10. Os intervalos estáticos associados às atividades “Coletar Informações” em A_4 e “Arquivar” em A_8 são iguais a $[0, 0]$ porque a duração destas atividades é desprezível se comparada com as outras atividades do “Serviço de Reclamações”. Os intervalos estáticos associados às atividades foram estipulados apenas a título de exemplificação.

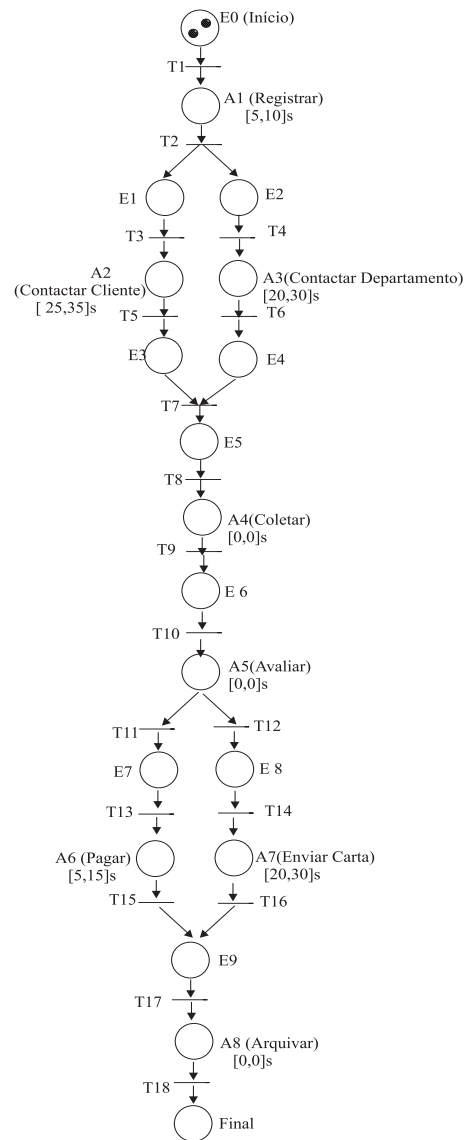


Figura 3.10: Modelo p-temporal com Intervalos Estáticos para o “Serviço de Reclamações”

3.2 Modelo de Alocação de Recurso

O próximo passo da atividade de modelagem é a alocação dos recursos utilizados para tratar as atividades. Existem diferentes tipos de recursos usados para realizar as atividades em sistemas de gerenciamento de *workflow*. 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 a capacidades computacionais. Às vezes, o conceito de grupo também é utilizado. Um grupo é um conjunto de recursos que podem desempenhar um mesmo papel.

Alguns dos recursos usados em um sistema de gerenciamento de *workflow* 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 do 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.

Em [Francielle, 2005] foi proposto um mecanismo de alocação de recursos híbridos tal que recursos humanos foram representados como recursos contínuos e foram modelados por um número real equivalente a sua disponibilidade, por exemplo, 100% quando não estão realizando nenhuma tarefa. Da mesma forma, aos pesos dos arcos foram atribuídos valores correspondentes à previsão da disponibilidade exigida para a realização da tarefa.

Apesar da forma inovadora de representar recursos humanos, ainda persiste um problema. Não é possível prever ao certo o quanto cada funcionário pode gastar para realizar uma tarefa. Neste trabalho, uma nova forma de representar recursos humanos através de intervalos *fuzzy* será apresentada. Desta forma, os valores associados ao recurso e ao peso do arco são valores aproximados, tornando, assim, o modelo mais próximo da realidade.

Os diferentes tipos de mecanismos de alocação serão formalizados nas seções seguintes.

3.2.1 Mecanismo de Alocação de Recurso Discreto

Definição 3.2.1 *O mecanismo de alocação de recurso discreto pode ser definido formalmente por um modelo de rede de Petri ordinária marcada [David e Alla, 2004] $C_{DR} = \langle A_{DR}, T_{DR}, Pre_{DR}, Pos_{DR}, M_{DR} \rangle$ com :*

- $A_{DR} = \bigcup_{\alpha=1}^{N_{DR}} A_{(DR)_\alpha} \cup \{R_D\}$ onde R_D representa o lugar do recurso discreto , $A_{(DR)_\alpha}$ o lugar da atividade α e N_{DR} o número das atividades que estão conectadas com o lugar do recurso discreto R_D .
- $T_{DR} = \bigcup_{\alpha=1}^{N_{DR}} T_{in_\alpha} \cup \bigcup_{\alpha=1}^{N_{DR}} T_{out_\alpha}$ onde T_{in_α} representa a transição de entrada da atividade $A_{(DR)_\alpha}$ e T_{out_α} representa a transição de saída da atividade $A_{(DR)_\alpha}$.
- $Pre_{DR} : A_{DR} \times T_{DR} \rightarrow \{0, 1\}$ 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).

- $Pos_{DR} : A_{DR} \times T_{DR} \rightarrow \{0, 1\}$ 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).

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

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

e

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

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:

$$M_{DR}(R_D) + (M_{DR}(A_1)) + (M_{DR}(A_2)) + \dots + (M_{DR}(A_{n_{DR}})) = m_D \quad (3.1)$$

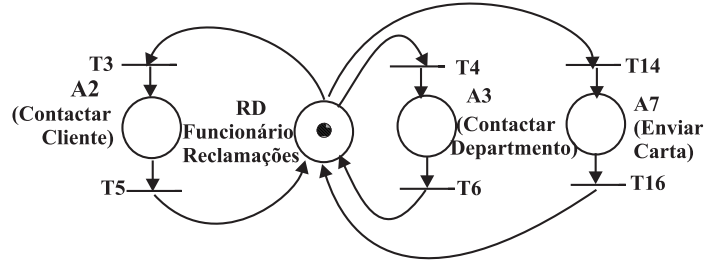


Figura 3.11: Alocação de Recurso Discreto

Supondo-se que no “Serviço de Reclamações” apresentado anteriormente um único funcionário trata as atividades “Contactar-Cliente”, “Contactar-Departamento” e “Enviar Carta”, então o mecanismo de alocação de recurso discreto é dado pela figura 3.11.

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.

Como o “Funcionário-Reclamações” (R_C) é um recurso renovável, a propriedade do invariante de lugar deverá ser respeitada. Usando a equação 3.1 o invariante de lugar terá neste caso a forma:

$$M_{DR}(R_D) + (M_{DR}(A_2)) + (M_{DR}(A_3)) + (M_{DR}(A_7)) = 1$$

3.2.2 Mecanismos de Alocação de Recurso Contínuo

Definição 3.2.2 *O mecanismo de alocação de recurso contínuo pode ser definido formalmente por um modelo de rede de Petri híbrida marcada com transições discretas [David e Alla, 2004] $C_{CR} = \langle A_{CR}, T_{CR}, Pre_{CR}, Pos_{CR}, M_{CR} \rangle$ com:*

- $A_{CR} = \bigcup_{\alpha=1}^{N_{CR}} A_{(CR)_\alpha} \cup \{R_C\}$ onde R_C representa o lugar de recurso contínuo, $A_{(CR)_\alpha}$ o lugar da atividade α e N_{CR} o número de atividades que estão conectados com o lugar do recurso contínuo R_C .

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

- $Pre_{CR} : A_{CR} \times T_{CR} \rightarrow R^+$ 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).

- $Pos_{CR} : A_{CR} \times T_{CR} \rightarrow R^+$ 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).

- $M_{CR} : A_{CR} \rightarrow R^+$ é a marcação inicial tal que

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

e

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

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:

$$M_{CR}(R_C) + (X_1 \times M_{CR}(A_1)) + (X_2 \times M_{CR}(A_2)) + \dots + (X_{N_{CR}} \times M_{CR}(A_{N_{CR}})) = m_C \quad (3.2)$$

Supondo-se que no “Serviço 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 3.12.

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

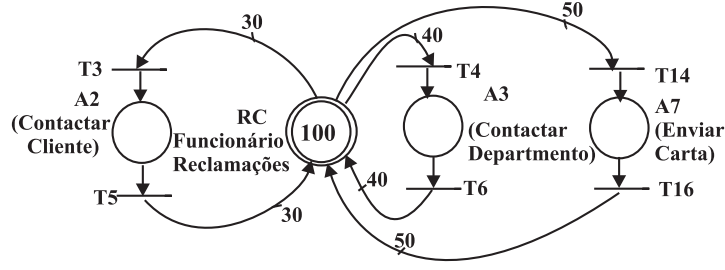


Figura 3.12: Alocação de Recurso Contínuo

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 3.2 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 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*) aproxima-se mais do que acontece na prática e será apresentada no parágrafo a seguir.

3.2.3 Mecanismo de Alocação de Recurso Contínuo *Fuzzy*

Definição 3.2.3 *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 $C_{FCR} = \langle A_{FCR}, T_{FCR}, Pre_{FCR}, Pos_{FCR}, M_{FCR} \rangle$ com:*

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

$$Pre_{FCR}(R_{FC}, T_{in_\alpha}) = [w1, w2, w3, w4] \text{ com } w2 = w3 \text{ e}$$

$$Pre_{FCR}(A_{(FCR)_\alpha}, T_{out_\alpha}) = [1, 1, 1, 1]$$

$$\text{onde } (w1, w2, w3) \in \mathfrak{R} \text{ e } w4 \in \mathfrak{R}^+$$

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

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

$$Pos_{FCR}(R_{FC}, T_{out_\alpha}) = [w1, w2, w3, w4] \text{ com } w2 = w3 \text{ e}$$

$$Pos_{FCR}(A_{(FCR)_\alpha}, T_{in_\alpha}) = [1, 1, 1, 1]$$

$$\text{onde } (w1, w2, w3) \in \mathfrak{R} \text{ e } w4 \in \mathfrak{R}^+$$

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

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

$$M_{FCR}(R_{FC}) = [m1, m2, m3, m4] \text{ com } (m1, m2, m3) \in \mathfrak{R} \text{ e } m4 \in \mathfrak{R}^+$$

e

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

representa a disponibilidade fuzzy (em porcentagem) do recurso contínuo fuzzy.

Supondo-se que no “Serviço 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 3.13.

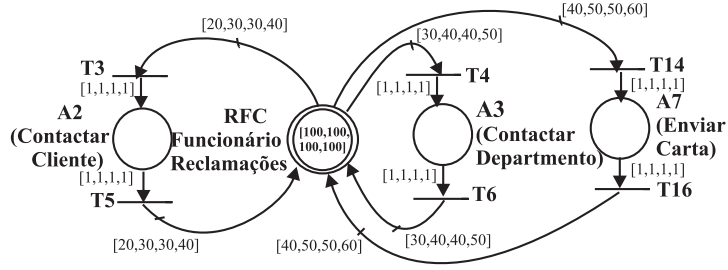


Figura 3.13: Recurso Contínuo *Fuzzy*

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

$$\Pi_t = \Pi(Pre_{FCR}(p, t) \leq M_{FCR}(p)) > 0 \quad (3.3)$$

Por exemplo, a transição $t3$ na figura 3.13 está habilitada porque:

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

como mostra a figura 3.14 com $A = Pre_{FCR}(R_{FC}, t3)$ e $B = M_{FCR}(R_{FC})$.

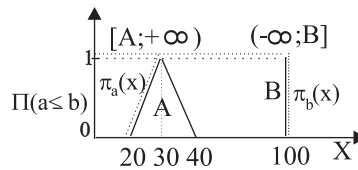


Figura 3.14: Cálculo de Possibilidade Associado a $t3$

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:

$$M'(p) = M(p) - Pre(p, t) + Pos(p, t) \quad (3.4)$$

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

$$M'_{FCR}(p) = M_{FCR}(p) \ominus Pre_{FCR}(p, t) \boxplus Pos_{FCR}(p, t) \quad (3.5)$$

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

$$[a1, a2, a3, a4] \boxplus [b1, b2, b3, b4] = [a1 + b4, a2 + b3, a3 + b2, a4 + b1] \quad (3.6)$$

O uso da operação $\boxplus$ vem do fato de que a operação *fuzzy* “ $\oplus$ ” não mantém o invariante de lugar associado a alocação de recursos, já que se trata de um recurso renovável. De fato, 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 3.13

- Disparo de t_3 :

o cálculo da possibilidade associado a t_3 é :

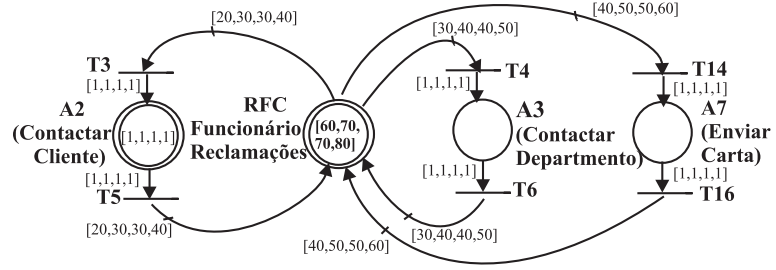


Figura 3.15: Marcação após Disparo de t_3

$$\Pi_{t_3} = \Pi(Pre_{FCR}(R_{FC}, t_3) \leq M_{FCR}(R_{FC})) = 1 > 0$$

como mostra a figura 3.14 com $A = Pre_{FCR}(R_{FC}, t_3)$ e $B = M_{FCR}(R_{FC})$.

Depois do disparo de t_3 , as novas marcações de R_{FC} e A2 são (figura 3.15):

$$\begin{aligned} M'_{FCR}(R_{FC}) &= M_{FCR}(R_{FC}) \ominus Pre_{FCR}(R_{FC}, t_3) = \\ &= [100, 100, 100, 100] \ominus [20, 30, 30, 40] = [60, 70, 70, 80] \end{aligned}$$

e

$$\begin{aligned} M'_{FCR}(A2) &= M_{FCR}(A2) \boxplus Pos_{FCR}(A2, t_3) = \\ &= [0, 0, 0, 0] \boxplus [1, 1, 1, 1] = [1, 1, 1, 1] \end{aligned}$$

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

$$\begin{aligned} &M_{FCR}(R_{FC}) \boxplus (Pre_{FCR}(R_{FC}, t_3) \otimes M_{FCR}(A2)) \boxplus (Pre_{FCR}(R_{FC}, t_4) \otimes \\ &M_{FCR}(A3)) \boxplus (Pre_{FCR}(R_{FC}, t_{14}) \otimes M_{FCR}(A7)) = \\ &= [60, 70, 70, 80] \boxplus ([20, 30, 30, 40] \otimes [1, 1, 1, 1]) \boxplus ([30, 40, 40, 50] \otimes [0, 0, 0, 0]) \boxplus \\ &([40, 50, 50, 60] \otimes [0, 0, 0, 0]) = [60, 70, 70, 80] \boxplus [20, 30, 30, 40] = [100, 100, 100, 100] \end{aligned}$$

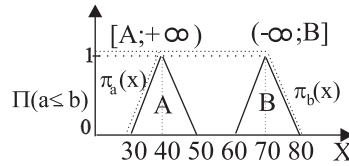


Figura 3.16: Cálculo de Possibilidade Associado a t_4

- Disparo de t_4 :

o cálculo de possibilidade associado a t_4 é:

$$\Pi_{t_4} = \Pi(Pre_{FCR}(R_{FC}, t_4) \leq M_{FCR}(R_{FC})) = 1 > 0$$

como mostra a figura 3.16 com $a = Pre_{FCR}(R_{FC}, t_4)$ e $b = M_{FCR}(R_{FC})$.

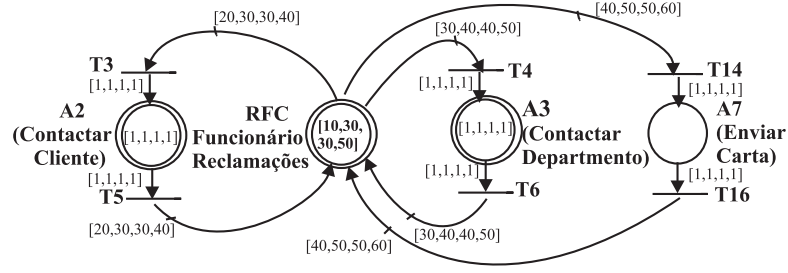


Figura 3.17: Marcação após o Disparo de t_4

Depois do disparo de t_4 , as novas marcações de R_{FC} e A3 são (figura 3.17):

$$\begin{aligned}
 M'_{FCR}(R_{FC}) &= M_{FCR}(R_{FC}) \ominus Pre_{FCR}(R_{FC}, t_4) = \\
 &[60, 70, 70, 80] \ominus [30, 40, 40, 50] = [10, 30, 30, 50] \\
 &\text{e} \\
 M'_{FCR}(A3) &= M_{FCR}(A3) \boxplus Pos_{FCR}(A3, t_4) = \\
 &[0, 0, 0, 0] \boxplus [1, 1, 1, 1] = [1, 1, 1, 1]
 \end{aligned}$$

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

$$\begin{aligned}
 &M_{FCR}(R_{FC}) \boxplus (Pre_{FCR}(R_{FC}, t_3) \otimes M_{FCR}(A2)) \boxplus (Pre_{FCR}(R_{FC}, t_4) \otimes \\
 &M_{FCR}(A3)) \boxplus (Pre_{FCR}(R_{FC}, t_{14}) \otimes M_{FCR}(A7)) = \\
 &[10, 30, 30, 50] \boxplus ([20, 30, 30, 40] \otimes [1, 1, 1, 1]) \boxplus ([30, 40, 40, 50] \otimes [1, 1, 1, 1]) \boxplus \\
 &([40, 50, 50, 60] \otimes [0, 0, 0, 0]) = \\
 &[10, 30, 30, 50] \boxplus [20, 30, 30, 40] \boxplus [30, 40, 40, 50] = \\
 &[50, 60, 60, 70] \boxplus [30, 40, 40, 50] = [100, 100, 100, 100]
 \end{aligned}$$

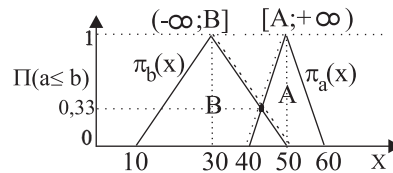


Figura 3.18: Cálculo de Possibilidade Associado a t_{14}

- Disparo de t_{14} :

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

$$\Pi_{t_4} = \Pi(Pre_{FCR}(R_{FC}, t_{14}) \leq M_{FCR}(R_{FC})) = 0,33 > 0$$

como mostra a figura 3.18 com $A = Pre_{FCR}(R_{FC}, t_{14})$ e $B = M_{FCR}(R_{FC})$.

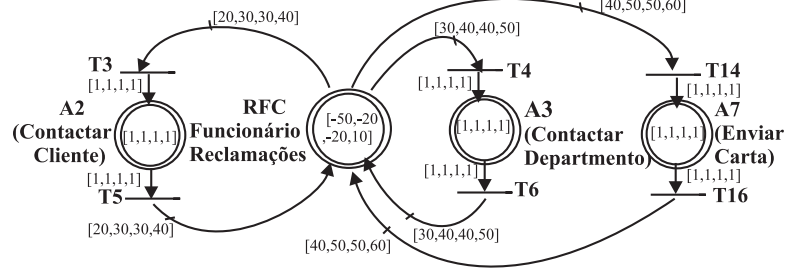


Figura 3.19: Marcação após o Disparo de t_{14}

Depois do disparo de t_{14} , as novas marcações de R_{FC} e A3 são (figura 3.19):

$$\begin{aligned} M'_{FCR}(R_{FC}) &= M_{FCR}(R_{FC}) \ominus Pre_{FCR}(R_{FC}, t_{14}) = \\ &[10, 30, 30, 50] \ominus [40, 50, 50, 60] = [-50, -20, -20, 10] \end{aligned}$$

e

$$\begin{aligned} M'_{FCR}(A7) &= M_{FCR}(A7) \boxplus Pos_{FCR}(A7, t_{14}) = \\ &[0, 0, 0, 0] \boxplus [1, 1, 1, 1] = [1, 1, 1, 1] \end{aligned}$$

A parte negativa da marcação *fuzzy* de R_{FC} que aparece depois do disparo de t_{14} 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:

$$\begin{aligned} &M_{FCR}(R_{FC}) \boxplus (Pre_{FCR}(R_{FC}, t_3) \otimes M_{FCR}(A2)) \boxplus (Pre_{FCR}(R_{FC}, t_4) \otimes \\ &M_{FCR}(A3)) \boxplus (Pre_{FCR}(R_{FC}, t_{14}) \otimes M_{FCR}(A7)) = \\ &[-50, -20, -20, 10] \boxplus ([20, 30, 30, 40] \otimes [1, 1, 1, 1]) \boxplus ([30, 40, 40, 50] \otimes [1, 1, 1, 1]) \boxplus \\ &([40, 50, 50, 60] \otimes [1, 1, 1, 1]) = \end{aligned}$$

$$\begin{aligned}
& [-50, -20, -20, 10] \boxplus [20, 30, 30, 40] \boxplus [30, 40, 40, 50] \boxplus ([40, 50, 50, 60] = \\
& [-10, 10, 10, 30] \boxplus [30, 40, 40, 50] \boxplus ([40, 50, 50, 60] = \\
& [40, 50, 50, 60] \boxplus ([40, 50, 50, 60] = [100, 100, 100, 100]
\end{aligned}$$

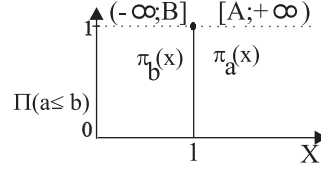


Figura 3.20: Cálculo de Possibilidade Associado a t_5

- Disparo t_5 :

o cálculo de possibilidade associado a t_5 é :

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

como mostra a figura 3.20 com $A = Pre_{FCR}(A2, t_5)$ e $B = M_{FCR}(A2)$.

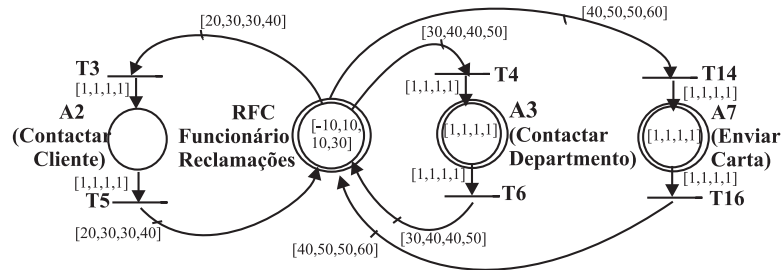


Figura 3.21: Marcação após Disparo de t_5

Depois do disparo de t_5 , as novas marcações de R_{FC} e A2 são (figura 3.21):

$$\begin{aligned}
M'_{FCR}(R_{FC}) &= M_{FCR}(R_{FC}) \boxplus Pos_{FCR}(R_{FC}, t_5) = \\
& [-50, -20, -20, 10] \boxplus [20, 30, 30, 40] = [-10, 10, 10, 30]
\end{aligned}$$

e

$$\begin{aligned}
M'_{FCR}(A2) &= M_{FCR}(A2) \ominus Pre_{FCR}(A2, t_5) = \\
& [1, 1, 1, 1] \ominus [1, 1, 1, 1] = [0, 0, 0, 0]
\end{aligned}$$

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

$$\begin{aligned}
& M_{FCR}(R_{FC}) \boxplus (Pre_{FCR}(R_{FC}, t_3) \otimes M_{FCR}(A_2)) \boxplus (Pre_{FCR}(R_{FC}, t_4) \otimes \\
& M_{FCR}(A_3)) \boxplus (Pre_{FCR}(R_{FC}, t_{14}) \otimes M_{FCR}(A_7)) = \\
& [-10, 10, 10, 30] \boxplus ([20, 30, 30, 40] \otimes [0, 0, 0, 0]) \boxplus ([30, 40, 40, 50] \otimes [1, 1, 1, 1]) \boxplus \\
& ([40, 50, 50, 60] \otimes [1, 1, 1, 1]) = \\
& [-10, 10, 10, 30] \boxplus [30, 40, 40, 50] \boxplus ([40, 50, 50, 60] = \\
& [40, 50, 50, 60] \boxplus ([40, 50, 50, 60] = [100, 100, 100, 100]
\end{aligned}$$

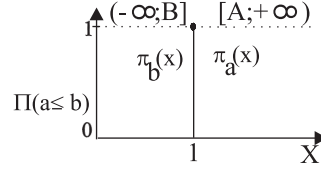


Figura 3.22: Cálculo da Possibilidade Associado a t_6

- Disparo de t_6 :

o cálculo da possibilidade associado a t_6 é :

$$\Pi_{t_6} = \Pi(Pre_{FCR}(A_3, t_6) \leq M_{FCR}(A_3)) = 1 > 0$$

como mostra a figura 3.22 com $A = Pre_{FCR}(A_3, t_6)$ e $B = M_{FCR}(A_3)$.

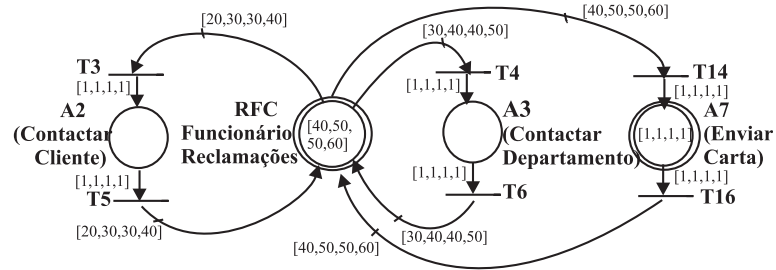


Figura 3.23: Marcação após o Disparo de t_6

Depois do disparo de t_6 , as novas marcações de R_{FC} e A_3 são (figura 3.23:

$$\begin{aligned}
M'_{FCR}(R_{FC}) &= M_{FCR}(R_{FC}) \boxplus Pos_{FCR}(R_{FC}, t_6) = \\
& [-10, 10, 10, 30] \boxplus [30, 40, 40, 50] = [40, 50, 50, 60]
\end{aligned}$$

e

$$\begin{aligned}
M'_{FCR}(A_3) &= M_{FCR}(A_3) \ominus Pre_{FCR}(A_3, t_6) = \\
& [1, 1, 1, 1] \ominus [1, 1, 1, 1] = [0, 0, 0, 0]
\end{aligned}$$

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

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

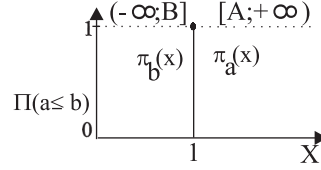


Figura 3.24: Cálculo da Possibilidade Associado a t_{16}

- 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 3.24 com $A = Pre_{FCR}(A7, t_{16})$ e $B = M_{FCR}(A7)$.

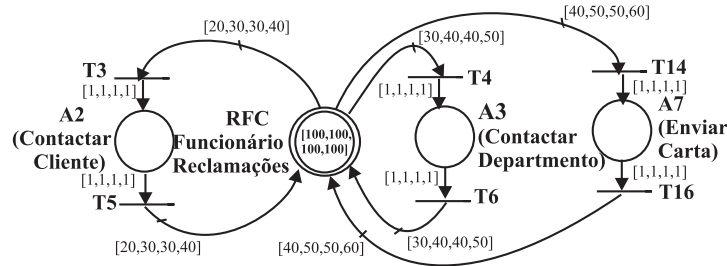


Figura 3.25: Marcação após Disparo de t_{16}

Depois do disparo de t_{16} , as novas marcações de R_{FC} e $A7$ são (figura 3.25):

$$\begin{aligned}
M'_{FCR}(R_{FC}) &= M_{FCR}(R_{FC}) \boxplus Pos_{FCR}(R_{FC}, t_{16}) = \\
& [40, 50, 50, 60] \boxplus [40, 50, 50, 60] = [100, 100, 100, 100]
\end{aligned}$$

e

$$\begin{aligned}
M'_{FCR}(A7) &= M_{FCR}(A7) \ominus Pre_{FCR}(A7, t_{16}) = \\
& [1, 1, 1, 1] \ominus [1, 1, 1, 1] = [0, 0, 0, 0]
\end{aligned}$$

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

$$\begin{aligned}
& M_{FCR}(R_{FC}) \boxplus (Pre_{FCR}(R_{FC}, t_3) \otimes M_{FCR}(A_2)) \boxplus (Pre_{FCR}(R_{FC}, t_4) \otimes \\
& M_{FCR}(A_3)) \boxplus (Pre_{FCR}(R_{FC}, t_{14}) \otimes M_{FCR}(A_7)) = \\
& [100, 100, 100, 100] \boxplus ([20, 30, 30, 40] \otimes [0, 0, 0, 0]) \boxplus ([30, 40, 40, 50] \otimes [0, 0, 0, 0]) \boxplus \\
& ([40, 50, 50, 60] \otimes [0, 0, 0, 0]) = [100, 100, 100, 100]
\end{aligned}$$

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 3.13 após o disparo de t_3 , as novas marcações de R_{FC} e A_2 seriam as mesmas apresentadas anteriormente. No entanto, após o disparo de t_5 , a nova marcação de R_{FC} seria:

$$\begin{aligned}
M'_{FCR}(R_{FC}) &= M_{FCR}(R_{FC}) \oplus Pos_{FCR}(R_{FC}, t_5) = \\
& [60, 70, 70, 80] \oplus [20, 30, 30, 40] = [80, 100, 100, 120]
\end{aligned}$$

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.

Capítulo 4

Problema de Escalonamento em Sistemas de Gerenciamento de *Workflow*

A resolução do problema do escalonamento em sistemas de gerenciamento de *workflow* 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.

A abordagem proposta neste trabalho é baseada na combinação de um mecanismo de propagação de restrição relativo aos prazos de entrega e em um jogador de rede de Petri, cujo papel principal é a resolução das situações de conflito em tempo real.

4.1 Mecanismo de Propagação de Restrição Relativo aos Prazos de Entrega

O mecanismo de propagação de restrição proposto neste trabalho foi inicialmente apresentado em [Francielle, 2005].

Em um sistema de gerenciamento de *workflow*, os tempos de espera existentes entre as atividades podem ser representados por um intervalo de visibilidade cujas bordas mínimas

e máximas irão depender dos prazos de entrega dos casos considerados.

Quando se conhece a data de início e o prazo máximo de um caso, é possível calcular os intervalos de visibilidade associados às fichas que serão produzidas nos diversos lugares de espera do processo estudado usando técnicas de propagação de restrições classicamente utilizadas em problemas de escalonamento baseados em grafos sem circuitos [Esquirol e Lopez, 1995].

Na rede de Petri da figura 4.1, quatro casos representados por quatro fichas em $E0$, são considerados. Dois casos devem ser iniciados na data 0 que é representada como a borda mínima do intervalo de visibilidade. Como a máxima duração permitida para cada caso é de 105 unidades de tempo, a borda máxima do intervalo de visibilidade associado ao último lugar “Final” do processo para os casos será 105 (data de início do caso + máxima duração do caso). Os outros dois casos devem ser iniciados na data 10, e conseqüentemente, devem terminar no máximo na data 115. As bordas dos outros intervalos de visibilidade associados aos casos nos lugares de espera foram calculadas aplicando-se as técnicas de propagação de restrição apresentados em [Francielle, 2005]. Por exemplo, a borda mínima do intervalo de visibilidade do caso 1 em $E1$ é obtida considerando a mínima duração da atividade “Registrar”: 0 (borda mínima do intervalo de visibilidade associado a $E0$) + 5 (borda mínima do intervalo estático em $A1$) = 5 (mínima borda do intervalo de visibilidade $v1$ em $E1$ correspondente ao caso 1). Da mesma forma, a borda máxima do intervalo de visibilidade de $v1$ em $E9$ é obtida considerando a duração máxima da atividade “Arquivar”: 105 (borda máxima do intervalo de visibilidade $v1$ em $Final$) - 0 (borda máxima do intervalo estático em $A8$) = 105 (borda máxima do intervalo de visibilidade de $E9$). Aplicando cálculo semelhantes a todos os lugares de espera, as bordas mínimas e máximas dos lugares de espera para os quatro casos são obtidos e mostrados na figura 4.1.

4.2 Definição de Resolução de Situações de Conflito

A definição de situações de conflito, em uma rede de Petri p-temporal, é mais complexa do que para uma simples rede de Petri autônoma [Julia, 1997]. No modelo autônomo, a informação qualitativa do tempo apenas indica que duas operações são executadas em paralelo, ou que estão em exclusão mútua. No modelo p-temporal, a representação

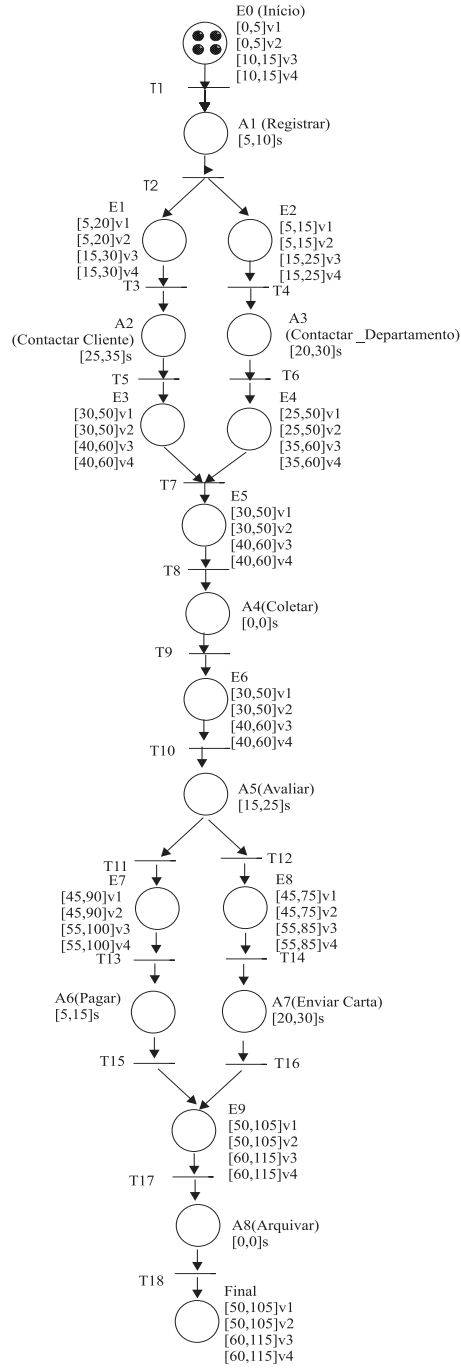


Figura 4.1: Modelo p-temporal para o “Serviço de Reclamações” após Aplicação dos Mecanismos de Propagação de Restrição

explícita informa não só que as operações estão em conflito, mas também, em qual intervalo de tempo elas entram em conflito.

As definições necessárias para entender a noção de conflito de uma rede de Petri p-temporal são as seguintes:

Definição 4.2.1 Intervalo de sensibilização de uma transição: o intervalo de sensibilização de uma transição t é o intervalo resultante da intersecção dos intervalos de visibilidade das fichas que participam do disparo de t .

Definição 4.2.2 Conflito estrutural: duas transições t_i e t_j estão em conflito estrutural se somente se elas têm pelo menos um lugar de entrada em comum.

Definição 4.2.3 Intervalo de conflito: se duas transições t_i e t_j estão em conflito estrutural, então o intervalo de conflito dessas transições é obtido através da intersecção dos intervalos de sensibilização de t_i e t_j . Se a intersecção dos intervalos é diferente de vazio, então as transições estão em conflito efetivo durante o intervalo calculado.

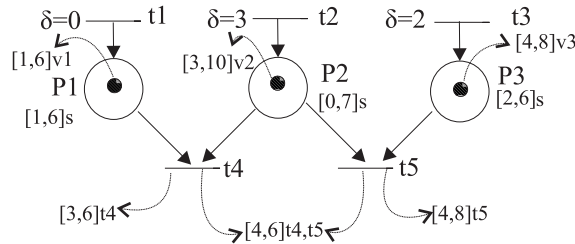


Figura 4.2: Conflito em uma Rede de Petri p-temporal

Na figura 4.2 as datas de chegadas de fichas nos lugares $P1$, $P2$ e $P3$ são, respectivamente, $\delta = 0$, $\delta = 3$ e $\delta = 2$. Os intervalos estáticos associados a cada um desses lugares são, respectivamente, $[1, 6]_s$, $[0, 7]_s$ e $[2, 6]_s$. Conseqüentemente, pode-se calcular os seguintes intervalos de visibilidade associados a cada ficha produzida. Sendo o intervalo de visibilidade $[1 + 0, 6 + 0]_{v1} = [1, 6]_{v1}$ associado a ficha em $P1$, $[0 + 3, 7 + 3]_{v2} = [3, 10]_{v2}$ associado a ficha em $P2$ e $[2 + 2, 6 + 2]_{v3} = [4, 8]_{v3}$ associado a ficha em $P3$.

O intervalo de sensibilização de $t4$ é dado por $[1, 6]_{v1} \cap [3, 10]_{v2} = [3, 6]_{t4}$ e o intervalo de sensibilização de $t5$ é dado por $[3, 10]_{v2} \cap [4, 8]_{v3} = [4, 8]_{t5}$. Portanto o intervalo de conflito do par $(t4, t5)$ é $[3, 6]_{t4} \cap [4, 8]_{t5} = [4, 6]_{t4, t5}$. Isso significa que existe uma situação de conflito efetivo na rede de Petri p-temporal da figura 4.2 dentro do intervalo $[4, 6]$.

4.3 Jogador de Rede de Petri p-temporal

Uma das técnicas que permite a execução dinâmica das redes de Petri p-temporais é o algoritmo do jogador de rede de Petri p-temporal [Julia, Valette, 2000], que pode ser usado para a resolução do problema do escalonamento de sistemas em tempo real. O jogador de rede de Petri p-temporal trata-se de um mecanismo de inferência especializado que permite a simulação de uma rede de Petri p-temporal. Quando o modelo é baseado no modelo de rede de Petri p-temporal, o algoritmo do jogador deve resolver as situações de conflito em tempo real de forma a não permitir a possibilidade de violação de restrição temporal que pode ser causada pela morte de uma ficha. A figura 4.3 mostra o funcionamento básico deste algoritmo.

O jogador possui um calendário dos eventos (bordas mínimas e máximas de todos os intervalos de visibilidade) escalonados no tempo. A cada vez que uma borda mínima é alcançada, uma ficha torna-se disponível. Se essa ficha sensibiliza uma transição e se não houver conflito efetivo, a transição é disparada; caso contrário, o conflito é isolado e o mecanismo de tomada de decisões é ativado para verificar se a transição pode ser disparada na data mínima. Se a decisão for o disparo, uma nova marcação é gerada e novos intervalos de visibilidade são definidos; caso contrário, é tratado o próximo evento do calendário. O mecanismo de tomada de decisão é baseado na geração de um grafo das classes e foi apresentado em [Francielle, 2005]. Se um limite máximo for atingido, então a ficha morre, representando um comportamento não permitido (violação de restrição). A única solução, neste caso, é relaxar uma restrição (aumentar o valor da borda máxima do intervalo de visibilidade correspondente.) Procedendo assim, não haverá garantia de que os prazos de entrega serão respeitados.

Uma particularidade do jogador apresentado neste trabalho é que em casos em que os recursos são representados usando intervalos *fuzzy*, existe a possibilidade de sobrecarregar um recurso em caso de violação de um prazo (borda máxima atingida). O cálculo da possibilidade de sensibilização de uma transição t (Π_t) em situações normais de trabalho é 1. Quando este valor é inferior a 1, sabe-se que o recurso poderá trabalhar além da sua capacidade normal. Quando o jogador é iniciado, escolhe-se qual o valor da possibilidade (Π_t) que será aceitável para disparar uma transição. No entanto, quando a borda máxima de um intervalo de visibilidade é atingida existe a escolha de simplesmente estender o intervalo de visibilidade para evitar a morte da ficha, ou, então, sobrecarregar o recurso

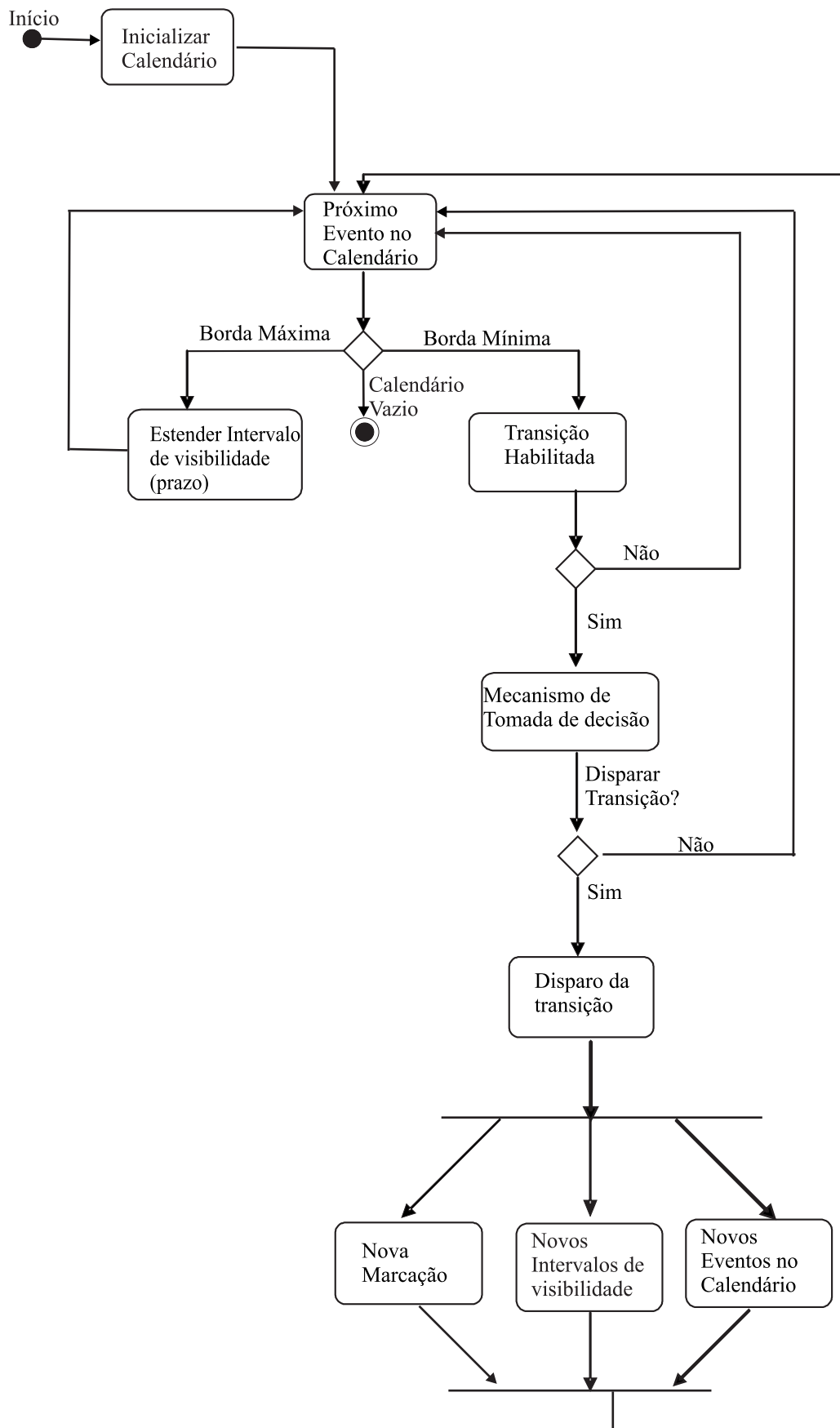


Figura 4.3: Jogador de Rede de Petri p-temporal

correspondente, ou seja, diminuir o valor de (Π_t) especificado para autorizar o disparo da transição.

O algoritmo do jogador para recursos discretos e contínuos *fuzzy* é apresentado na figura 4.4

Para ilustrar o funcionamento do jogador, o “Serviço de Reclamações”, apresentado no capítulo anterior, será simulado usando as diferentes formas de alocação de recursos. A fim de comparar os resultados o cenário de execução será sempre o mesmo e compreende as atividades de registrar a reclamação, contactar cliente e contactar departamento, em seguida coletar e avaliar os casos, enviar uma carta e arquivar o caso.

4.3.1 Simulação Usando Recursos Discretos

A figura 4.5 apresenta todos os recursos alocados de forma discreta. Sendo assim, cada recurso só poderá realizar uma única tarefa de cada vez (recurso disjuntivo). Neste caso, muitos conflitos podem ocorrer assim como atrasos. A fim de evitá-los ou pelo menos diminuí-los é usada uma heurística: o caso com o maior atraso, ou seja, o intervalo que possuir a menor borda máxima, será considerado primeiro quando houver situações de conflito.

Aplicando o algoritmo do jogador de rede de Petri p-temporal ao modelo global da figura 4.5 a seqüência de eventos é a seguinte:

- Na data $\delta = 0$:
 - duas fichas (casos 1 e 2) se tornam disponíveis em $E0$, sensibilizando a transição $t1$. O recurso $R1$ só pode tratar um caso de cada vez. Assim, o mecanismo de tomada de decisão escolhe a transição $t1$ que é disparada para o caso 1 e uma ficha (caso 1) é produzida em $A1$ com o intervalo de visibilidade $[0, 5]$.
- Na data $\delta = 5$:
 - a atividade $A1$ (caso 1) é finalizada, a ficha (caso 1) se torna disponível em $A1$, a transição $t2$ é disparada e o recurso é devolvido para $R1$. Uma ficha (caso 1) é produzida em $E1$ e outra em $E2$. As transições $t3$ e $t4$ são sensibilizadas mas não é possível disparar as duas simultaneamente já que o recurso $R2$ só possui uma ficha. Neste caso, o mecanismo de tomada de decisão escolhe o disparo

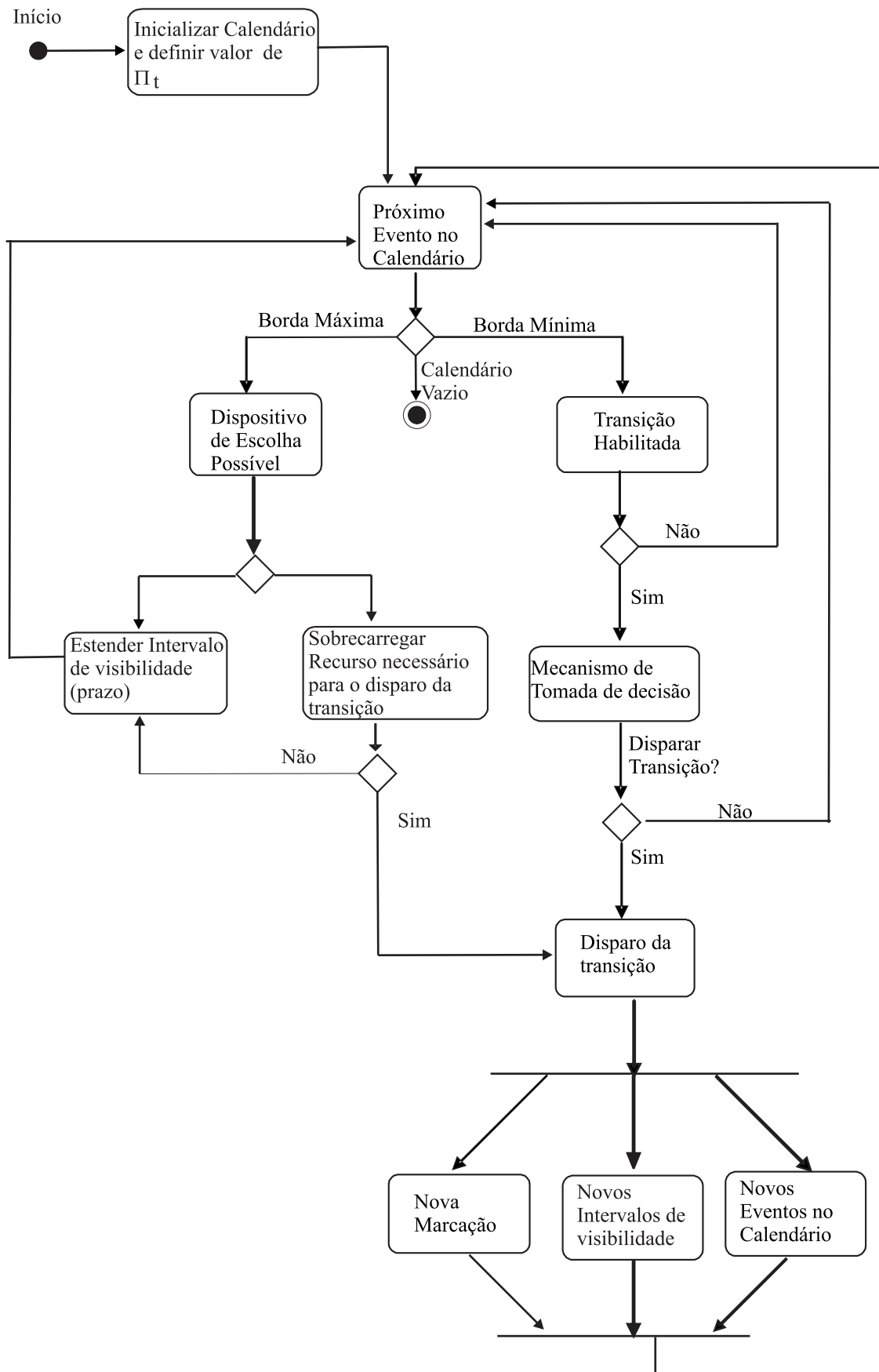


Figura 4.4: Jogador de Rede de Petri p-temporal com Recursos Discretos e Contínuos Fuzzy

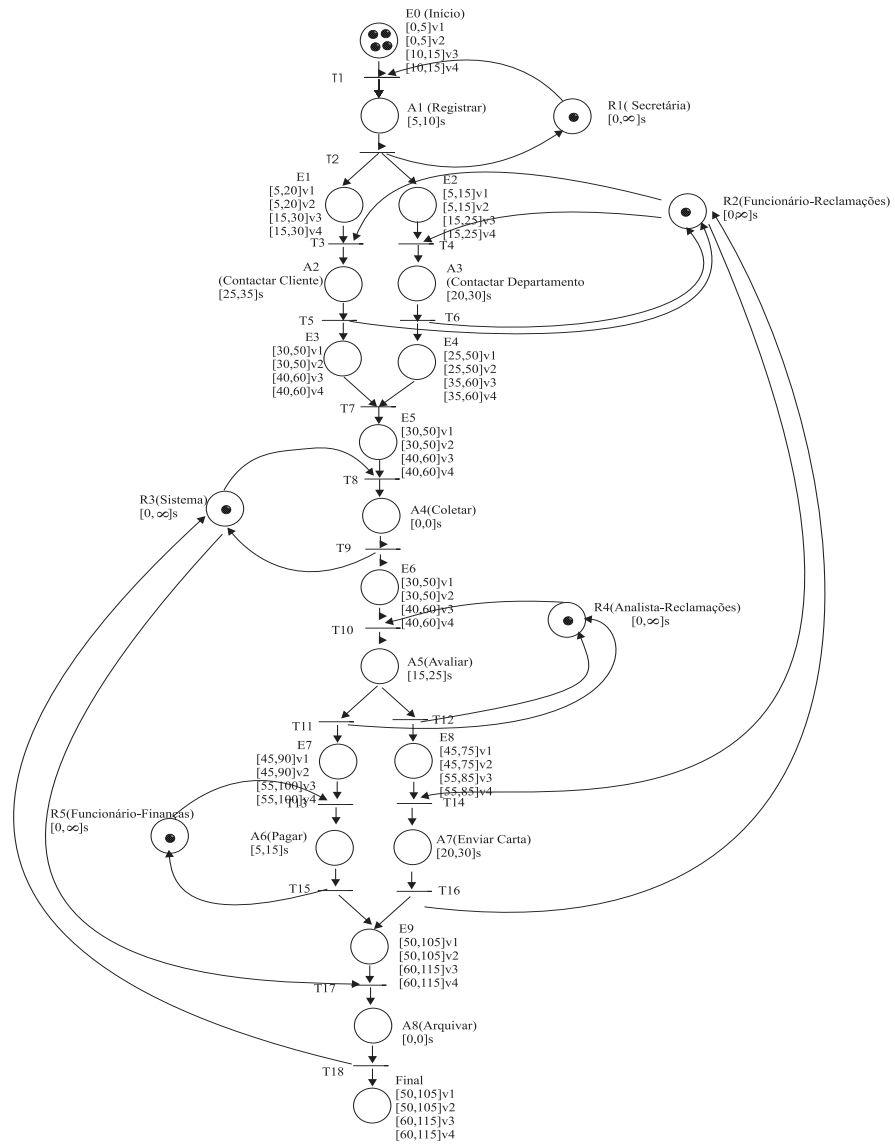


Figura 4.5: Modelo p-temporal para o “Serviço de Reclamações” com Recursos Discretos

da transição $t4$ (pois a ficha que está em $E2$ possui a menor borda máxima). Uma ficha é produzida em $A3$ com intervalo de visibilidade $[5, 25]$;

- a transição $t1$ (caso 2) é sensibilizada e disparada. Uma ficha (caso 2) é produzida em $A1$ com intervalo de visibilidade $[5, 10]$.

- Na data $\delta = 10$:

- a atividade $A1$ (caso 2) é finalizada, a ficha (caso 2) se torna disponível em $A1$, a transição $t2$ é disparada e o recurso é devolvido para $R1$. Uma ficha (caso 2) é produzida em $E1$ e outra em $E2$;
- duas fichas (casos 3 e 4) se tornam disponíveis em $E0$, sensibilizando a transição $t1$. O recurso $R1$ só pode tratar um caso de cada vez. O mecanismo de tomada de decisão escolhe o disparo da transição $t1$ (caso 3). Uma ficha é produzida em $A1$ com o intervalo de visibilidade $[10, 15]$.

- Na data $\delta = 15$:

- a atividade $A1$ (caso 3) é finalizada, a ficha (caso 3) se torna disponível em $A1$, a transição $t2$ é disparada e o recurso é devolvido para $R1$. Uma ficha (caso 3) é produzida em $E1$ e outra em $E2$;
- a transição $t1$ (caso 4) é sensibilizada e disparada. Uma ficha (caso 4) é produzida em $A1$ com intervalo de visibilidade $[15, 20]$.

- Na data $\delta = 20$:

- a atividade $A1$ (caso 4) é finalizada, a ficha (caso 4) se torna disponível em $A1$, a transição $t2$ é disparada e o recurso é devolvido para $R1$. Uma ficha (caso 4) é produzida em $E1$ e outra em $E2$.

- Na data $\delta = 25$:

- a atividade $A3$ (caso 1) é finalizada, a ficha (caso 1) se torna disponível em $A3$, a transição $t6$ é disparada e o recurso é devolvido para $R2$. Uma ficha é produzida em $E4$;

- neste momento, existem quatro fichas em $E1$ (casos 1, 2, 3, e 4) e três fichas em $E2$ (casos 2, 3, 4). A ficha do caso 2 que está em $E2$ está mais atrasada (possui o menor intervalo de visibilidade). O mecanismo de tomada de decisão escolhe o disparo da transição $t4$ (caso 2) para ser disparada, então, uma ficha é produzida em $A3$ (caso 2) com intervalo de visibilidade $[25, 45]$.
- Na data $\delta = 45$:
 - a atividade $A3$ (caso 2) é finalizada, a ficha (caso 2) se torna disponível em $A3$, a transição $t6$ é disparada e o recurso é devolvido para $R2$. Uma ficha é produzida em $E4$;
 - neste momento, existem quatro fichas em $E1$ (Caso 1, 2, 3, e 4) e duas fichas em $E2$ (caso 3 e 4). O mecanismo de tomada de decisão escolhe o disparo da transição $t3$. Uma ficha é produzida em $A2$ (caso 1) com intervalo de visibilidade $[45, 70]$.
- Na data $\delta = 70$:
 - a atividade $A2$ (caso 1) é finalizada, a ficha (caso 1) se torna disponível em $A2$, a transição $t5$ é disparada e o recurso é devolvido para $R2$. Uma ficha é produzida em $E3$. A transição $t7$ é sensibilizada e disparada, fazendo com que uma ficha se torne disponível em $E5$, que sensibiliza e dispara a transição $t8$. A ficha (caso 1) se torna disponível em $A4$, que sensibiliza e dispara a transição $t9$ tornando disponível a ficha em $E6$ que sensibiliza e dispara a transição $t10$. Uma ficha (caso 1) é produzida em $A5$ com intervalo de visibilidade $[70, 85]$;
 - em $E1$ existem três fichas a espera do recurso $R2$ (casos 2, 3 e 4) e em $E2$ existem duas fichas (casos 3 e 4). Como a ficha que está em $E2$ (caso 2) é a mais atrasada, o mecanismo de tomada de decisão escolhe o disparo da transição $t3$. Uma ficha é produzida em $A2$ com intervalo de visibilidade $[70, 95]$.
- Na data $\delta = 85$:
 - a atividade $A5$ (caso 1) é finalizada, a ficha (caso 1) se torna disponível em $A5$, a transição $t12$ é disparada e o recurso é devolvido para $R4$. Uma ficha (caso 1) é produzida em $E8$.

- Na data $\delta = 95$:
 - a atividade $A2$ (caso 2) é finalizada, a ficha (caso 2) se torna disponível em $A2$, a transição $t5$ é disparada e o recurso é devolvido para $R2$. A transição $t7$ é sensibilizada e disparada, fazendo com que uma ficha se torne disponível em $E5$, que sensibiliza e dispara a transição $t8$. A ficha (caso 1) se torna disponível em $A4$, que sensibiliza e dispara a transição $t9$ tornando disponível a ficha em $E6$ que sensibiliza e dispara a transição $t10$. Uma ficha (caso 2) é produzida em $A5$ com intervalo de visibilidade $[95, 110]$;
 - em $E1$ existem duas fichas a espera do recurso $R2$ (caso 3 e 4), em $E2$ (caso 3 e 4) e em $E8$ (caso 1) existe somente uma ficha. O mecanismo de tomada de decisão escolhe o disparo da transição $t4$ (caso 3). Neste caso, uma ficha é produzida em $A3$ com intervalo de visibilidade $[95, 115]$.
- Na data $\delta = 110$:
 - a atividade $A5$ (caso 2) é finalizada, a ficha (caso 2) se torna disponível em $A5$, a transição $t12$ é disparada e o recurso é devolvido para $R4$. Uma ficha (caso 2) é produzida em $E8$.
- Na data $\delta = 115$:
 - $A3$ (caso 3) é finalizada, a ficha (caso 3) se torna disponível em $A3$, a transição $t6$ é disparada e o recurso é devolvido para $R2$. Uma ficha é produzida em $E4$;
 - neste momento, existem duas fichas em $E1$ (casos 3 e 4), uma ficha em $E2$ (caso 4) e duas fichas em $E8$ (casos 1 e 2) a espera do recurso $R2$. O mecanismo de tomada de decisão escolhe o disparo da transição $t4$ (já que a ficha que se encontra em $E2$ (caso 4) é a mais atrasada). Uma ficha é produzida em $A3$ (caso 4) com intervalo de visibilidade $[115, 135]$.
- Na data $\delta = 135$:
 - $A3$ (caso 4) é finalizada, a ficha (caso 4) se torna disponível em $A3$, a transição $t6$ é disparada e o recurso é devolvido para $R2$. Uma ficha é produzida em $E4$;

- neste momento, existem duas fichas em $E1$ (casos 3, e 4) e duas fichas em $E8$ (casos 1 e 2) a espera do recurso $R2$. O mecanismo de tomada de decisão escolhe o disparo da transição $t3$. Uma ficha é produzida em $A2$ (caso 3).
- Na data $\delta = 160$:
 - a atividade $A2$ (caso 3) é finalizada, a ficha (caso 3) se torna disponível em $A2$, a transição $t5$ é disparada e o recurso é devolvido para $R2$. A transição $t7$ é sensibilizada e disparada, fazendo com que uma ficha se torne disponível em $E5$, que sensibiliza e dispara a transição $t8$. A ficha (caso 3) se torna disponível em $A4$, que sensibiliza e dispara a transição $t9$ tornando disponível a ficha em $E6$ que sensibiliza e dispara a transição $t10$. Uma ficha (caso 3) é produzida em $A5$ com intervalo de visibilidade $[160, 175]$;
 - em $E1$ existe uma ficha a espera do recurso $R2$ (caso 4), e em $E8$ existem três fichas (casos 1, 2 e 3). O mecanismo de tomada de decisão escolhe o disparo da transição $t3$ (caso 4) (já que a ficha em $E1$ (caso 1) é a mais atrasada). Neste caso, uma ficha é produzida em $A2$ com intervalo de visibilidade $[160, 185]$.
- Na data $\delta = 175$:
 - a atividade $A5$ (caso 3) é finalizada, a ficha (caso 3) se torna disponível em $A5$, a transição $t12$ é disparada e o recurso é devolvido para $R4$. Uma ficha (caso 3) é produzida em $E8$.
- Na data $\delta = 185$:
 - a atividade $A2$ (caso 4) é finalizada, a ficha (caso 4) se torna disponível em $A2$, a transição $t5$ é disparada e o recurso é devolvido para $R2$. A transição $t7$ é sensibilizada e disparada, fazendo com que uma ficha se torne disponível em $E5$, que sensibiliza e dispara a transição $t8$. A ficha (caso 4) se torna disponível em $A4$, que sensibiliza e dispara a transição $t9$ tornando disponível a ficha em $E6$ que sensibiliza e dispara a transição $t10$. Uma ficha (caso 4) é produzida em $A5$ com intervalo de visibilidade $[185, 200]$;
 - em $E8$ existem quatro fichas (casos 1, 2 ,3 e 4) a espera do recurso $R2$. O mecanismo de tomada de decisão escolhe o disparo da transição $t14$ (caso 1)

(já que a ficha em $E8$ (caso 1) é a mais atrasada). Neste caso, uma ficha é produzida em $A7$ com intervalo de visibilidade $[185, 205]$.

- Na data $\delta = 200$:
 - a atividade $A5$ (caso 4) é finalizada, a ficha (caso 4) se torna disponível em $A5$, a transição $t12$ é disparada e o recurso é devolvido para $R4$. Uma ficha (caso 4) é produzida em $E8$.
- Na data $\delta = 205$:
 - a atividade $A7$ (caso 1) é finalizada, a ficha (caso 1) se torna disponível em $A7$, a transição $t16$ é disparada e o recurso é devolvido para $R2$. Uma ficha (caso 1) se torna disponível em $E9$ sensibilizando a transição $t17$ que é disparada. Uma ficha se torna disponível em $A8$ que sensibiliza e dispara a transição $t18$. O processo para o caso 1 é encerrado;
 - das três fichas que estão em $E8$ (caso 2, 3 e 4), a ficha do caso 2 é a mais atrasada. Neste caso, o mecanismo de tomada de decisão escolhe o disparo da transição $t14$ (caso 2). Uma ficha é produzida em $A7$ com intervalo de visibilidade $[205, 225]$.
- Na data $\delta = 225$:
 - a atividade $A7$ (caso 2) é finalizada, a ficha (caso 2) se torna disponível em $A7$, a transição $t16$ é disparada e o recurso é devolvido para $R2$. Uma ficha (caso 2) se torna disponível em $E9$ sensibilizando a transição $t17$ que é disparada. Uma ficha se torna disponível em $A8$ que sensibiliza e dispara a transição $t18$. O processo para o caso 2 é encerrado;
 - o mecanismo de tomada de decisão escolhe o disparo da transição $t14$ (caso 3). Uma ficha é produzida em $A7$ (caso 3) com intervalo de visibilidade $[225, 245]$.
- Na data $\delta = 245$:
 - a atividade $A7$ (caso 3) é finalizada, a ficha (caso 3) se torna disponível em $A7$, a transição $t16$ é disparada e o recurso é devolvido para $R2$. Uma ficha (caso 3) se torna disponível em $E9$ sensibilizando a transição $t17$ que é disparada.

Uma ficha se torna disponível em $A8$ que sensibiliza e dispara a transição $t18$.

O processo para o caso 3 é encerrado;

- a transição $t14$ (caso 4) é disparada. Uma ficha é produzida em $A7$ com intervalo de visibilidade $[245, 265]$.

- Na data $\delta = 265$:

- a atividade $A7$ (caso 4) é finalizada, a ficha (caso 4) se torna disponível em $A7$, a transição $t16$ é disparada e o recurso é devolvido para $R2$. Uma ficha (caso 4) se torna disponível em $E9$ sensibilizando a transição $t17$ que é disparada. Uma ficha se torna disponível em $A8$ que sensibiliza e dispara a transição $t18$. O processo para o caso 4 é encerrado.

A figura 4.6 mostra um diagrama de Gantt que fornece a solução do escalonamento no final da execução do jogador de rede de Petri p-temporal. É importante destacar que os prazos máximos não foram respeitados. O caso 1 deveria ter terminado na data mais tarde 105 e terminou na data 205. O caso 2 deveria ter terminado na data mais tarde 105 e terminou na data 225. O caso 3 deveria ter terminado na data mais tarde 115 e terminou na data 245. O caso 4 deveria ter terminado na data mais tarde 115 e terminou na data 265.

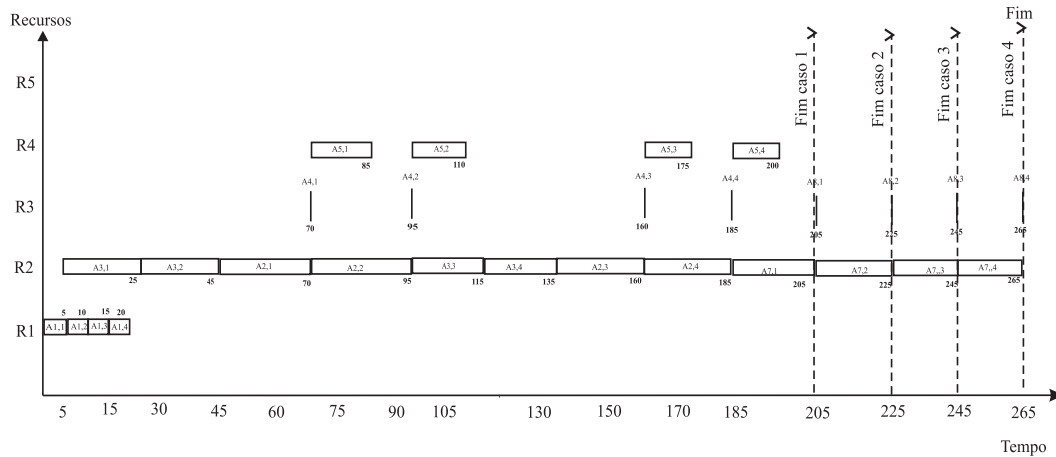


Figura 4.6: Sequência de Atividades - Recursos Discretos

4.3.2 Simulação Usando Recursos Discretos e Contínuos

A figura 4.7 apresenta $R2$ como um recurso contínuo. Neste caso, desde que o pesos dos arcos permitam, mais de uma atividade pode ser realizada ao mesmo tempo (recurso cumulativo).

Aplicando o algoritmo do jogador de rede de Petri p-temporal ao modelo global da figura 4.7 a seqüência de eventos é a seguinte:

- Na data $\delta = 0$:
 - duas fichas (casos 1 e 2) se tornam disponíveis em $E0$, sensibilizando a transição $t1$. O recurso $R1$ só pode tratar um caso de cada vez. O mecanismo de tomada de decisão escolhe o disparo da transição $t1$ (caso 1). Uma ficha (caso 1) é produzida em $A1$ com o intervalo de visibilidade $[0, 5]$.
- Na data $\delta = 5$:
 - a atividade $A1$ (caso 1) é finalizada, a ficha (caso 1) se torna disponível em $A1$, a transição $t2$ é disparada e o recurso é devolvido para $R1$. Uma ficha (caso 1) é produzida em $E1$ e outra em $E2$. As transições $t3$ e $t4$ são sensibilizadas e disparadas. Uma ficha é produzida em $A2$ com intervalo de visibilidade $[5, 30]$ e outra em $A3$ com intervalo de visibilidade $[5, 25]$. O valor do recurso $R2$ neste momento é de 30%;
 - a transição $t1$ (caso 2) está sensibilizada e é então disparada. Uma ficha (caso 2) é produzida em $A1$ com intervalo de visibilidade $[5, 10]$.
- Na data $\delta = 10$:
 - a atividade $A1$ (caso 2) é finalizada, a ficha (caso 2) se torna disponível em $A1$ a transição $t2$ é disparada e o recurso é devolvido para $R1$. Uma ficha (caso 2) é produzida em $E1$ e outra em $E2$. A transição $t3$ está sensibilizada e é disparada, visto que o valor disponível do recurso $R2$ é de somente 30%. Uma ficha é produzida em $A2$ com intervalo de visibilidade $[10, 35]$. Após o disparo, o valor de $R2$ é 0%;
 - a transição $t1$ (caso 3) é disparada. Uma ficha é produzida em $A1$ com intervalo de visibilidade $[10, 15]$.

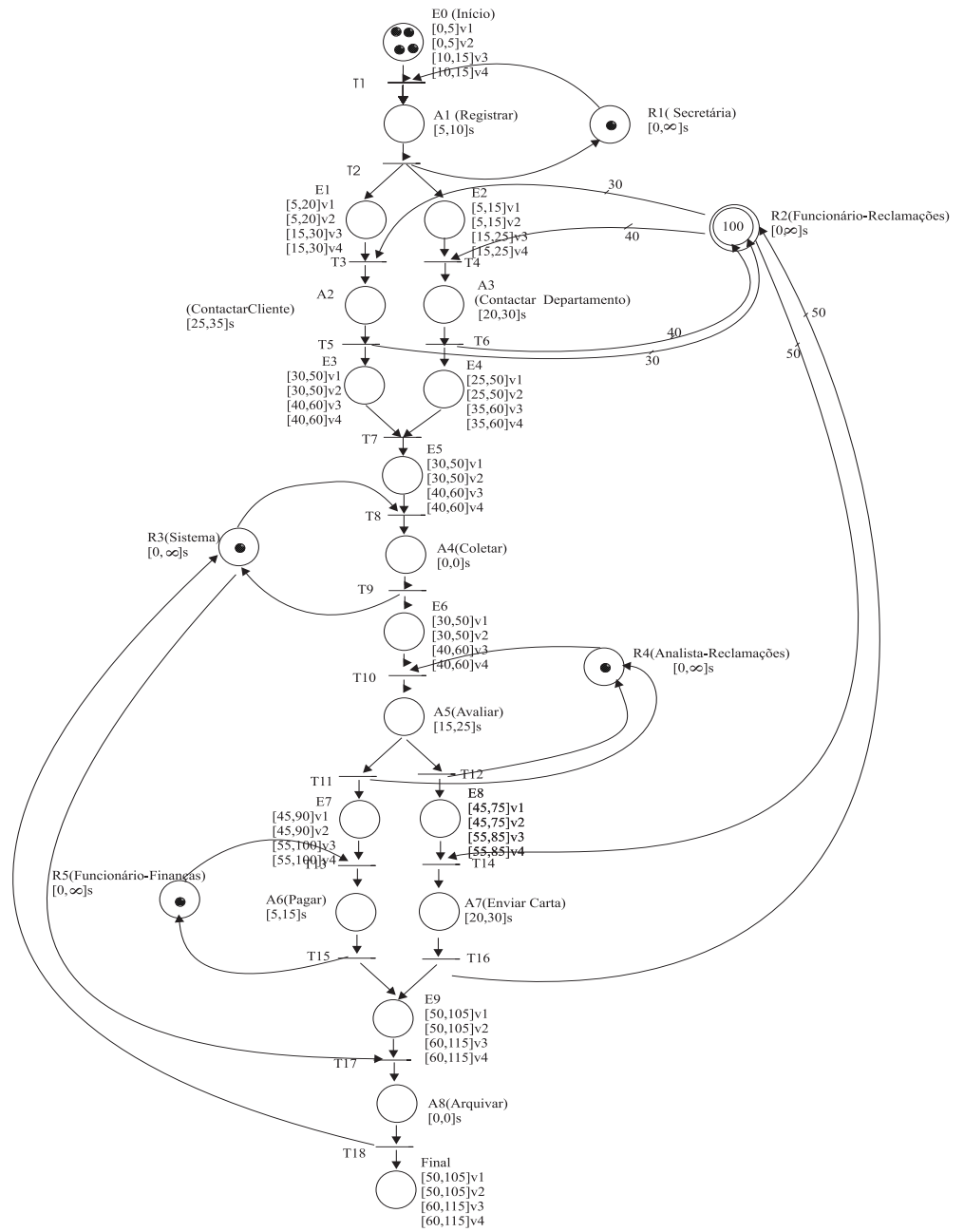


Figura 4.7: Modelo p-temporal para o “Serviço de Reclamações” com Recursos Discretos e Contínuos

- Na data $\delta = 15$:
 - a atividade $A1$ (caso 3) é finalizada, a ficha (caso 3) se torna disponível em $A1$, a transição $t2$ é disparada e o recurso é devolvido para $R1$. Uma ficha (caso 3) é produzida em $E1$ e outra em $E2$;
 - a transição $t1$ (caso 4) é disparada. Uma ficha é produzida em $A1$ com intervalo de visibilidade $[15, 20]$.
- Na data $\delta = 20$:
 - a atividade $A1$ (caso 4) é finalizada, a ficha (caso 4) se torna disponível em $A1$, a transição $t2$ é disparada e o recurso é devolvido para $R1$. Uma ficha (caso 4) é produzida em $E1$ e outra em $E2$.
- Na data $\delta = 25$:
 - a atividade $A3$ (caso 1) é finalizada, a ficha (caso 1) se torna disponível em $A3$, a transição $t6$ é disparada e o recurso é devolvido para $R2$. Neste momento $R2 = 40\%$. Uma ficha é produzida em $E4$;
 - o mecanismo de tomada de decisão escolhe o disparo da transição $t4$ (caso 2). Uma ficha é produzida em $A3$ com intervalo de visibilidade $[25, 45]$. O valor associado a $R2$ é 0% .
- Na data $\delta = 30$:
 - a atividade $A2$ (caso 1) é finalizada, a ficha (caso 1) se torna disponível em $A2$, a transição $t5$ é disparada e o recurso é devolvido para $R2$. Neste momento $R2 = 30\%$. Uma ficha é produzida em $E3$. A transição $t7$ é sensibilizada e disparada, fazendo com que uma ficha se torne disponível em $E5$, que sensibiliza e dispara a transição $t8$. A ficha (caso 1) se torna disponível em $A4$, que sensibiliza e dispara a transição $t9$ tornando disponível a ficha em $E6$ que sensibiliza e dispara a transição $t10$. Uma ficha é produzida em $A5$ com intervalo de visibilidade $[30, 45]$;
 - como o valor associado ao recurso $R2$ é 30% a transição $t3$ para o caso 3 é sensibilizada e disparada. Uma ficha é produzida em $A2$ com intervalo de visibilidade $[30, 55]$ e o valor de $R2$ após o disparo de $t3$ é 0% .

- Na data $\delta = 35$:
 - a atividade $A2$ (caso 2) é finalizada, a ficha (caso 2) se torna disponível em $A2$, a transição $t5$ é disparada e o recurso é devolvido para $R2$. Neste momento $R2 = 30\%$. Uma ficha é produzida em $E3$.
 - como o valor associado ao recurso $R2$ é 30% a transição $t3$ (caso 4) é sensibilizada e disparada. Uma ficha é produzida em $A2$ e o valor de $R2$ após o disparo de $t3$ é 0% .
- Na data $\delta = 45$:
 - a atividade $A5$ (caso 1) é finalizada, a ficha (caso 1) se torna disponível em $A5$, a transição $t12$ é disparada e o recurso é devolvido para $R4$. Uma ficha (caso 1) é produzida em $E8$;
 - a atividade $A2$ (caso 2) é finalizada, a ficha (caso 2) se torna disponível em $A2$, a transição $t5$ é disparada e o recurso é devolvido para $R2$. Neste momento o valor de $R2 = 40\%$. Uma ficha é produzida em $E3$. A transição $t7$ é sensibilizada e disparada, fazendo com que uma ficha se torne disponível em $E5$, que sensibiliza e dispara a transição $t8$. A ficha (caso 1) se torna disponível em $A4$, que sensibiliza e dispara a transição $t9$ tornando disponível a ficha em $E6$ que sensibiliza e dispara a transição $t10$. Uma ficha é produzida em $A5$ com intervalo de visibilidade $[45, 60]$;
 - o mecanismo de tomada de decisão escolhe o disparo da transição $t4$ (caso 3). Uma ficha é produzida em $A3$ com intervalo de visibilidade $[45, 65]$. O valor de $R2$ é 0% .
- Na data $\delta = 55$:
 - a atividade $A2$ (caso 3) é finalizada, a ficha (caso 3) se torna disponível em $A2$, a transição $t5$ é disparada e o recurso é devolvido para $R2$. Neste momento $R2 = 30\%$. Uma ficha é produzida em $E3$ (caso 3).
- Na data $\delta = 60$:

- a atividade $A5$ (caso 2) é finalizada, a ficha (caso 2) se torna disponível em $A5$, a transição $t12$ é disparada e o recurso é devolvido para $R4$. Uma ficha (caso 2) é produzida em $E8$;
 - a atividade $A2$ (caso 4) é finalizada, a ficha (caso 4) se torna disponível em $A2$ a transição $t5$ é disparada e o recurso é devolvido para $R2$. Neste momento $R2 = 60\%$. Uma ficha é produzida em $E3$ (caso 4);
 - o mecanismo de tomada de decisão escolhe o disparo da transição $t4$ (caso 4). Uma ficha é produzida em $A3$ com intervalo de visibilidade $[60, 80]$. Neste momento, o valor de $R2 = 20\%$.
- Na data $\delta = 65$:
 - a atividade $A3$ (caso 3) é finalizada, a ficha (caso 3) se torna disponível em $A3$, a transição $t6$ é disparada e o recurso é devolvido para $R2$. Neste momento $R2 = 60\%$. Uma ficha é produzida em $E4$. A transição $t7$ é sensibilizada e disparada, fazendo com que uma ficha se torne disponível em $E5$, que sensibiliza e dispara a transição $t8$. A ficha (caso 3) se torna disponível em $A4$, que sensibiliza e dispara a transição $t9$ tornando disponível a ficha em $E6$ que sensibiliza e dispara a transição $t10$. Uma ficha é produzida em $A5$ com intervalo de visibilidade $[65, 80]$;
 - o mecanismo de tomada de decisão escolhe o disparo da transição $t14$ (caso 1). Uma ficha é produzida em $A7$ com intervalo de visibilidade $[65, 85]$. Após o disparo de $t14$ o valor de $R2 = 10\%$.
 - Na data $\delta = 80$:
 - a atividade $A5$ (caso 3) é finalizada, a ficha (caso 3) se torna disponível em $A5$ e a transição $t12$ é disparada, o recurso é devolvido para $R4$. Uma ficha (caso 3) é produzida em $E8$;
 - a atividade $A3$ (caso 4) é finalizada, a ficha (caso 4) se torna disponível em $A3$, a transição $t5$ é disparada e o recurso é devolvido para $R2$. Neste momento $R2 = 50\%$. Uma ficha é produzida em $E4$ (caso 4). A transição $t7$ é sensibilizada e disparada, fazendo com que uma ficha se torne disponível em $E5$, que sensibiliza e dispara a transição $t8$. A ficha (caso 3) se torna disponível em

$A4$, que sensibiliza e dispara a transição $t9$ tornando disponível a ficha em $E6$ que sensibiliza e dispara a transição $t10$. Uma ficha é produzida em $A5$ com intervalo de visibilidade $[80, 95]$;

- o mecanismo de tomada de decisão escolhe o disparo da transição $t14$ (caso 2). Uma ficha é produzida em $A7$ com intervalo de visibilidade $[80, 100]$. Neste momento, o valor de $R2$ é 0%.

- Na data $\delta = 85$:

- a atividade $A7$ (caso 1) é finalizada, a ficha (caso 1) se torna disponível em $A7$, a transição $t16$ é disparada e o recurso é devolvido para $R2$. Neste momento, o valor de $R2 = 50\%$. Uma ficha (caso 1) se torna disponível em $E9$ sensibilizando a transição $t17$ que é disparada. Uma ficha se torna disponível em $A8$ que sensibiliza e dispara a transição $t18$. O processo para o caso 1 é encerrado;
- o mecanismo de tomada de decisão escolhe o disparo da transição $t14$ (caso 3). Uma ficha é produzida em $A7$ com intervalo de visibilidade $[85, 105]$. Neste momento, o valor de $R2$ é 0%.

- Na data $\delta = 95$:

- a atividade $A5$ (caso 4) é finalizada, a ficha (caso 4) se torna disponível em $A5$ e a transição $t12$ é disparada, o recurso é devolvido para $R4$. Uma ficha (caso 4) é produzida em $E8$.

- Na data $\delta = 100$:

- a atividade $A7$ (caso 2) é finalizada, a ficha (caso 2) se torna disponível em $A7$, a transição $t16$ é disparada e o recurso é devolvido para $R2$. Neste momento, o valor de $R2 = 50\%$. Uma ficha (caso 2) se torna disponível em $E9$ sensibilizando a transição $t17$ que é disparada. Uma ficha se torna disponível em $A8$ que sensibiliza e dispara a transição $t18$. O processo para o caso 2 é encerrado;
- a transição $t14$ (caso 4) é disparada e uma ficha é produzida em $A7$ com intervalo de viabilidade $[100, 120]$. O valor de $R2$ é 0%.

- Na data $\delta = 105$:

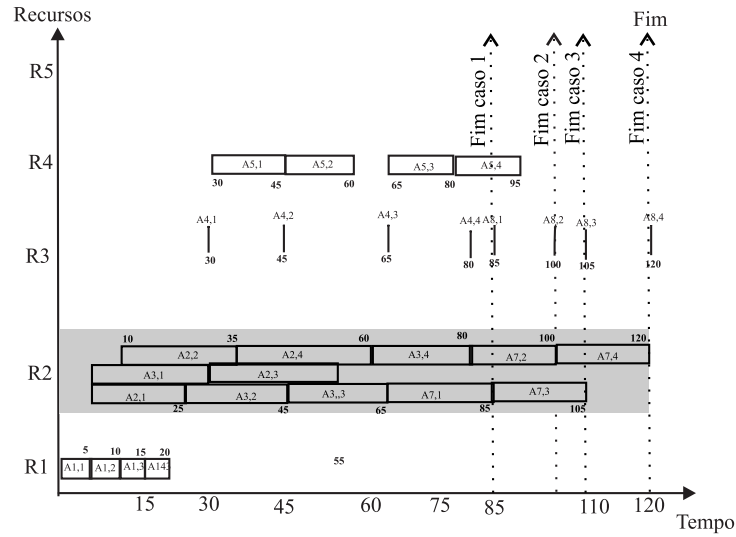


Figura 4.8: Sequência de Atividades - Recursos Contínuos

- a atividade $A7$ (caso 3) é finalizada, a ficha (caso 3) se torna disponível em $A7$, a transição $t16$ é disparada e o recurso é devolvido para $R2$. Neste momento, o valor de $R2 = 50\%$. Uma ficha (caso 3) se torna disponível em $E9$ sensibilizando a transição $t17$ que é disparada. Uma ficha se torna disponível em $A8$ que sensibiliza e dispara a transição $t18$. O processo para o caso 3 é encerrado.
- Na data $\delta = 120$
 - a atividade $A7$ (caso 4) é finalizada, a ficha (caso 4) se torna disponível em $A7$, a transição $t16$ é disparada e o recurso é devolvido para $R2$. Neste momento, o valor de $R2 = 100\%$. Uma ficha (caso 4) se torna disponível em $E9$ sensibilizando a transição $t17$ que é disparada. Uma ficha se torna disponível em $A8$ que sensibiliza e dispara a transição $t18$. O processo para o caso 4 é encerrado.

A figura 4.8 mostra um diagrama de Gantt que fornece a solução do escalonamento no final da execução do jogador de rede de Petri p-temporal com recursos discretos e contínuos. Desta vez, os prazos máximos não foram respeitados também, mas os atrasos são menores. O caso 1 deveria ter terminado na data mais tarde 105 e terminou na data 85. O caso 2 deveria ter terminado na data mais tarde 105 e terminou na data 100. O caso 3 deveria ter terminado na data mais tarde 115 e terminou na data 105. O caso 4 deveria ter terminado na data mais tarde 115 e terminou na data 120.

4.3.3 Simulação Usando Recursos Discretos e Contínuos *Fuzzy*

A figura 4.9 apresenta $R2$ como um recurso contínuo *fuzzy*. Como no caso de recursos contínuos, desde que os pesos dos arcos permitam, mais de uma atividade pode ser realizada ao mesmo tempo (recurso cumulativo).

Aplicando o algoritmo do jogador de rede de Petri p-temporal fuzzy ao modelo global da figura 4.9, a sequência de eventos é a seguinte:

- Na data $\delta = 0$:
 - duas fichas (casos 1 e 2) se tornam disponíveis em $E0$, sensibilizando a transição $t1$. O recurso $R1$ só pode tratar um caso de cada vez. O mecanismo de tomada de decisão escolhe o disparo da transição $t1$ (caso 1), uma ficha (caso 1) é produzida em $A1$ com o intervalo de visibilidade $[0, 5]$.
- Na data $\delta = 5$:
 - a atividade $A1$ (caso 1) é finalizada, a ficha (caso 1) se torna disponível em $A1$, a transição $t2$ é disparada e o recurso é devolvido para $R1$. Uma ficha (caso 1) é produzida em $E1$ e outra em $E2$. A possibilidade de disparo de $t3$ é $\Pi_{t3} = 1$ como mostrado na figura 3.14. A possibilidade de disparo de $t4$ é $\Pi_{t4} = 1$ como mostrado na figura 3.16. Sendo assim, $t3$ e $t4$ são disparadas. O valor de $R2$ após os disparos é $[10, 30, 30, 50]$;
 - a transição $t1$ (caso 2) é disparada. Uma ficha é produzida em $A1$ com intervalo de visibilidade $[5, 10]$.
- Na data $\delta = 10$:
 - a atividade $A1$ (caso 2) é finalizada, a ficha (caso 2) se torna disponível em $A1$, a transição $t2$ é disparada e o recurso é devolvido para $R1$. Uma ficha (caso 2) é produzida em $E1$ e outra em $E2$. As transições $t3$ e $t4$ estão em conflito. O mecanismo de tomada de decisão escolhe a transição $t4$ (o bordo máxima da ficha em $E2$ é menor e $\Pi_{t4} = 0,66$ (figura 4.10)) para disparar. Uma ficha é produzida em $A3$ com intervalo de visibilidade $[10, 30]$. O valor de $R2$ após o disparo é $[-40, -10, -10, 20]$. É possível perceber que neste momento, o recurso $R2$ está trabalhando com sobrecarga;

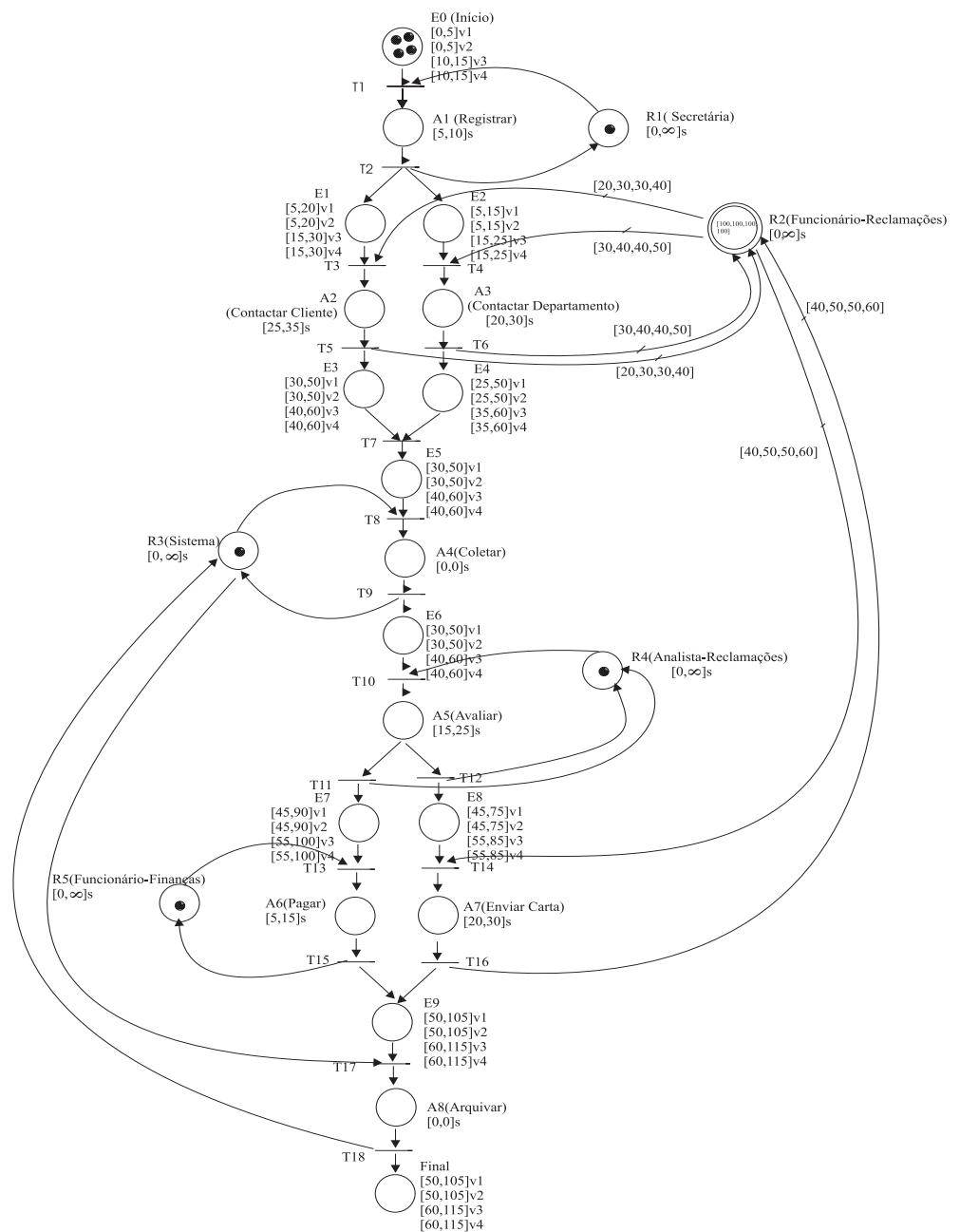


Figura 4.9: Modelo p-temporal para o “Serviço de Reclamações” com Recursos Discretos e Contínuos *Fuzzy*

- a transição $t1$ (caso 3) é disparada. Uma ficha é produzida em $A1$ com intervalo de visibilidade $[10, 15]$.

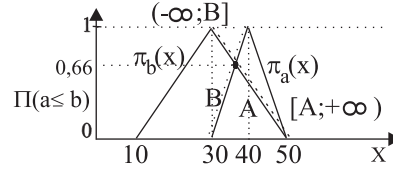


Figura 4.10: Cálculo de Possibilidade Associado a $t4$

- Na data $\delta = 15$:
 - a atividade $A1$ (caso 3) é finalizada, a ficha (caso 3) se torna disponível em $A1$, a transição $t2$ é disparada e o recurso é devolvido para $R1$. Uma ficha (caso 3) é produzida em $E1$ e outra em $E2$. As transições $t3$ e $t4$ não são sensibilizadas pois $\Pi_{t3} = 0$ e $\Pi_{t4} = 0$, figuras 4.11 e 4.12, respectivamente;

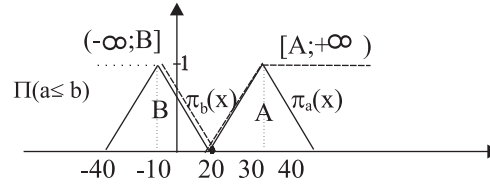


Figura 4.11: Cálculo de Possibilidade Associado a $t3$

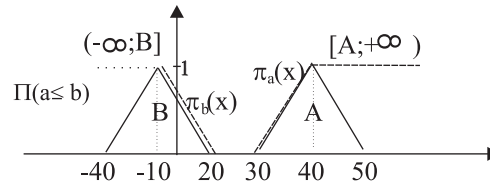


Figura 4.12: Cálculo de Possibilidade Associado a $t4$

- a transição $t1$ (caso 4) é disparada. Uma ficha é produzida em $A1$ com intervalo de visibilidade $[15, 20]$.
- Na data $\delta = 20$:
 - a atividade $A1$ (caso 4) é finalizada, a ficha (caso 4) se torna disponível em $A1$, a transição $t2$ é disparada e o recurso é devolvido para $R1$. Uma ficha (caso 4)

é produzida em $E1$ e outra em $E2$. As transições $t3$ e $t4$ não são sensibilizadas pois $\Pi_{t3} = 0$ e $\Pi_{t4} = 0$, figuras 4.11 e 4.12, respectivamente.

- Na data $\delta = 25$:

- a atividade $A3$ (caso 1) é finalizada, a ficha (caso 1) se torna disponível em $A3$, a transição $t6$ é disparada e o recurso é devolvido para $R2$. Neste momento $R2 = [10, 30, 30, 50]$. Uma ficha é produzida em $E4$;
- em $E1$ três fichas (casos 2, 3 e 4) esperando o recurso $R2$ e em $E2$ também existem duas fichas (casos 3 e 4). Como $\Pi_{t3} = 1$ (caso 2)(figura 4.13) e $E1$ (caso 2) possui a menor borda máxima, o mecanismo de tomada de decisão escolhe o disparo da transição $t3$ (caso 2). Uma ficha é produzida em $A2$ com intervalo de visibilidade $[25, 50]$. O valor associado a $R2 = [-30, 0, 0, 30]$.

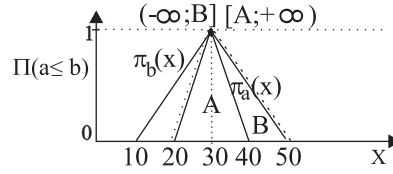


Figura 4.13: Cálculo de Possibilidade Associado a $t3$

- Na data $\delta = 30$:

- a atividade $A2$ (caso 1) é finalizada, a ficha (caso 1) se torna disponível em $A2$, a transição $t5$ é disparada e o recurso é devolvido para $R2$. Neste momento $R2 = [10, 30, 30, 50]$. Uma ficha é produzida em $E3$. A transição $t7$ é sensibilizada e disparada, fazendo com que uma ficha se torne disponível em $E5$, que sensibiliza e dispara a transição $t8$. A ficha (caso 1) se torna disponível em $A4$, que sensibiliza e dispara a transição $t9$ tornando disponível a ficha em $E6$ que sensibiliza e dispara a transição $t10$. Uma ficha é produzida em $A5$ com intervalo de visibilidade $[30, 45]$;
- a atividade $A3$ (caso 2) é finalizada, a ficha (caso 2) se torna disponível em $A3$, a transição $t6$ é disparada e o recurso é devolvido para $R2$. Neste momento $R2 = [60, 70, 70, 80]$. Uma ficha é produzida em $E4$;

- existem duas fichas em $E1$ (casos 3 e 4) e duas fichas em $E2$ a espera do recurso $R2$. $\Pi_{t4} = 1$ (caso 3) (figura 4.14), então o mecanismo de tomada de decisão escolhe o disparo da transição $t4$ (caso 3). Uma ficha se torna disponível em $A3$ com intervalo de visibilidade $[30, 50]$ e o valor de $R2 = [10, 30, 30, 50]$;

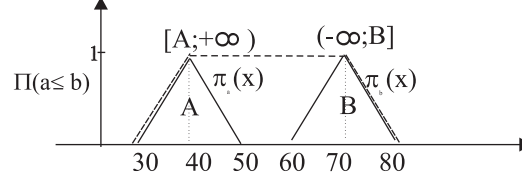


Figura 4.14: Cálculo de Possibilidade Associado a $t4$

- $\Pi_{t3} = 1$ (caso 3) (figura 4.13), então o mecanismo de tomada de decisão escolhe o disparo da transição $t3$. Uma ficha é produzida em $A2$ com intervalo de visibilidade $[30, 55]$ e o valor de $R2 = [-30, 0, 0, 30]$.
- Na data $\delta = 45$:
 - a atividade $A5$ (caso 1) é finalizada, a ficha (caso 1) se torna disponível em $A5$, a transição $t12$ é disparada e o recurso é devolvido para $R4$. Uma ficha (caso 1) é produzida em $E8$.
- Na data $\delta = 50$:
 - a atividade $A2$ (caso 2) é finalizada, a ficha (caso 2) se torna disponível em $A2$, a transição $t5$ é disparada e o recurso é devolvido para $R2$. Neste momento $R2 = [10, 30, 30, 50]$. Uma ficha é produzida em $E3$. A transição $t7$ é sensibilizada e disparada, fazendo com que uma ficha se torne disponível em $E5$, que sensibiliza e dispara a transição $t8$. A ficha (caso 1) se torna disponível em $A4$, que sensibiliza e dispara a transição $t9$ tornando disponível a ficha em $E6$ que sensibiliza e dispara a transição $t10$. Uma ficha é produzida em $A5$ com intervalo de visibilidade $[50, 65]$;
 - a atividade $A3$ (caso 3) é finalizada, a ficha (caso 3) se torna disponível em $A3$, a transição $t6$ é disparada e o recurso é devolvido para $R2$. Neste momento $R2 = [60, 70, 70, 80]$. Uma ficha é produzida em $E4$;

- a transição $t4$ (caso 4) é disparada , pois $\Pi_{t4} = 1$ (figura 4.14). Uma ficha é produzida em $A3$ (caso 4) com intervalo de visibilidade $[50, 70]$. O valor de $R2$ após o disparo é $[10, 30, 30, 50]$;
- a transição $t3$ (caso 4) é sensibilizada e disparada ,pois $\Pi_{t3} = 1$ (figura 4.13). Uma ficha é produzida em $A2$ (caso 4) com intervalo de visibilidade $[50, 75]$. O valor de $R2$ após o disparo é $[-30, 0, 0, 50]$.
- Na data $\delta = 55$:
 - a atividade $A2$ (caso 3) é finalizada, a ficha (caso 3) se torna disponível em $A2$, a transição $t5$ é disparada e o recurso é devolvido para $R2$. Neste momento $R2 = [10, 30, 30, 50]$. Uma ficha é produzida em $E3$. A transição $t7$ é sensibilizada e disparada, fazendo com que uma ficha se torne disponível em $E5$, que sensibiliza e dispara a transição $t8$. A ficha (caso 1) se torna disponível em $A4$, que sensibiliza e dispara a transição $t9$ produzindo uma ficha em $E6$;
 - sabendo que $\Pi_{t14} = 0,33$ (caso 1) conforme mostra figura 4.15 a transição $t14$ é disparada e uma ficha é produzida em $A7$ com intervalo de visibilidade $[55, 75]$. O valor associado a $R2$ é $[-50, -20, -20, 10]$.

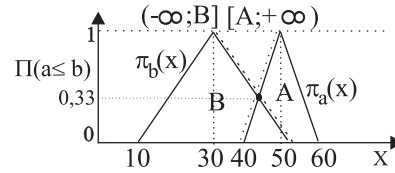


Figura 4.15: Cálculo de Possibilidade Associado a $t14$

- Na data $\delta = 65$:
 - a atividade $A5$ (caso 2) é finalizada, a ficha (caso 2) se torna disponível em $A5$, a transição $t12$ é disparada e o recurso é devolvido para $R4$. Uma ficha (caso 2) é produzida em $E8$;
 - a transição $t10$ (caso 3) é disparada. Uma ficha é produzida em $A5$ com intervalo de visibilidade $[65, 80]$.
- Na data $\delta = 70$:

- a atividade $A3$ (caso 4) é finalizada, a ficha (caso 4) se torna disponível em $A3$ a transição $t6$ é disparada e o recurso é devolvido para $R2$. Neste momento $R2 = [0, 20, 20, 40]$. Uma ficha é produzida em $E4$.
- Na data $\delta = 75$:
 - a atividade $A2$ (caso 4) é finalizada, a ficha (caso 4) se torna disponível em $A2$, a transição $t5$ é disparada e o recurso é devolvido para $R2$. Neste momento $R2 = [40, 50, 50, 60]$. Uma ficha é produzida em $E3$. A transição $t7$ é sensibilizada e disparada, fazendo com que uma ficha se torne disponível em $E5$, que sensibiliza e dispara a transição $t8$. A ficha (caso 1) se torna disponível em $A4$, que sensibiliza e dispara a transição $t9$ produzindo uma ficha em $E6$;
 - a atividade $A7$ (caso 1) é finalizada, a ficha (caso 1) se torna disponível em $A7$, a transição $t16$ é disparada e o recurso é devolvido para $R2$. Neste momento, o valor de $R2 = [100, 100, 100, 100]$. Uma ficha (caso 1) se torna disponível em $E9$ sensibilizando a transição $t17$ que é disparada. Uma ficha se torna disponível em $A8$ que sensibiliza e dispara a transição $t18$. O processo para o caso 1 é encerrado;
 - a transição $t14$ (caso 2) possui $\Pi_{t14} = 1$ conforme figura 4.16, assim, é disparada. Uma ficha é produzida em $A7$ com intervalo de visibilidade $[75, 95]$. O valor de $R2$ é $[40, 50, 50, 60]$.

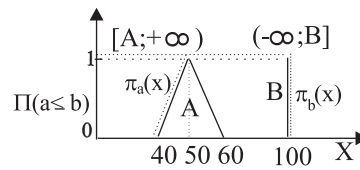


Figura 4.16: Cálculo de Possibilidade Associado a $t14$

- Na data $\delta = 80$:
 - a atividade $A5$ (caso 3) é finalizada, a ficha (caso 3) se torna disponível em $A5$, a transição $t12$ é disparada e o recurso é devolvido para $R4$. Uma ficha (caso 3) é produzida em $E8$;

- a transição t_{10} (caso 4) é disparada. Uma ficha é produzida em $A5$ com intervalo de visibilidade $[80, 95]$;
- a transição t_{14} (caso 3) possui $\Pi_{t_{14}} = 1$ conforme figura 4.17, assim, é disparada. Uma ficha se torna disponível em $A7$ com intervalo de visibilidade $[80, 100]$. O valor de $R2$ é $[-20, 0, 0, 20]$.

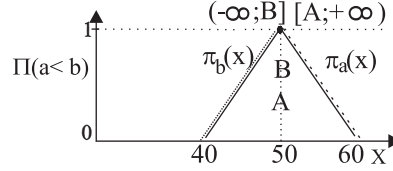


Figura 4.17: Cálculo de Possibilidade Associado a t_{14}

- Na data $\delta = 85$:
 - a atividade $A7$ (caso 2) é finalizada, a ficha (caso 2) se torna disponível em $A7$, a transição t_{16} é disparada e o recurso é devolvido para $R2$. Neste momento, o valor de $R2 = [40, 50, 50, 60]$. Uma ficha (caso 2) se torna disponível em $E9$ sensibilizando a transição t_{17} que é disparada. Uma ficha se torna disponível em $A8$ que sensibiliza e dispara a transição t_{18} . O processo para o caso 2 é encerrado.
- Na data $\delta = 95$:
 - a atividade $A5$ (caso 4) é finalizada, a ficha (caso 4) se torna disponível em $A5$, a transição t_{12} é disparada e o recurso é devolvido para $R4$. Uma ficha (caso 4) se torna disponível em $E8$. A transição t_{14} (caso 4) possui $\Pi_{t_{14}} = 1$ conforme figura 4.17, assim, é disparada. Uma ficha se torna disponível em $A7$ com intervalo de visibilidade $[95, 115]$. O valor de $R2$ é $[-20, 0, 0, 20]$.
- Na data $\delta = 100$:
 - a atividade $A7$ (caso 3) é finalizada, a ficha (caso 3) se torna disponível em $A7$, a transição t_{16} é disparada e o recurso é devolvido para $R2$. Neste momento, o valor de $R2 = [40, 50, 50, 60]$. Uma ficha (caso 3) se torna disponível em $E9$ sensibilizando a transição t_{17} que é disparada. Uma ficha se torna disponível

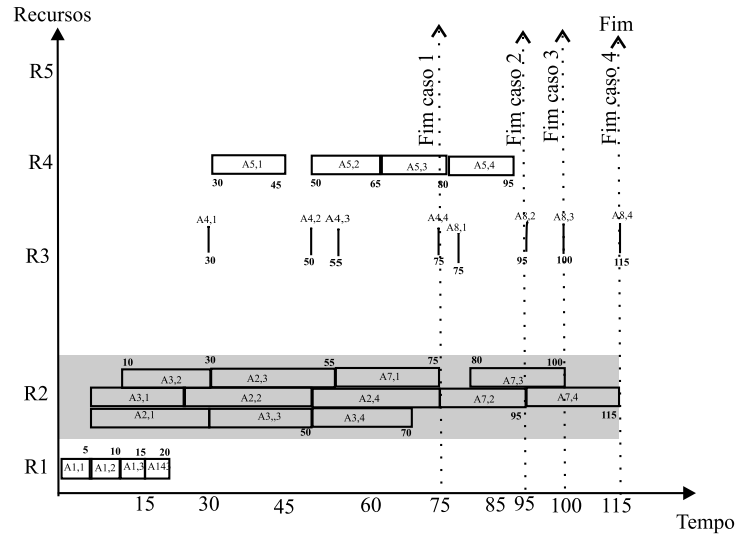


Figura 4.18: Sequência de Atividades - Recursos *Fuzzy*

em $A8$ que sensibiliza e dispara a transição $t18$. O processo para o caso 3 é encerrado.

- Na data $\delta = 115$:
 - a atividade $A7$ (caso 4) é finalizada, a ficha (caso 4) se torna disponível em $A7$, a transição $t16$ é disparada e o recurso é devolvido para $R2$. Neste momento, o valor de $R2 = [100, 100, 100, 100]$. Uma ficha (caso 4) se torna disponível em $E9$ sensibilizando a transição $t17$ que é disparada. Uma ficha se torna disponível em $A8$ que sensibiliza e dispara a transição $t18$. O processo para o caso 4 é encerrado.

A figura 4.18 mostra um diagrama de Gantt que fornece a solução do escalonamento no final da execução do jogador de rede de Petri p-temporal fuzzy. Desta vez, os prazos máximos foram respeitados. O caso 1 deveria terminar na data mais tarde 105 e terminou na data 75. O caso 2 deveria ter terminado na data mais tarde 105 e terminou na data 85. O caso 3 deveria ter terminado na data mais tarde 115 e terminou na data 100. O caso 4 deveria ter terminar na data mais tarde 115 e terminou exatamente na data 115.

Capítulo 5

Conclusão

Em sistemas 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.

O correspondente a definição de “transição habilitada” e “equação fundamental” foram definidas para o novo modelo. Além disso, uma expressão linear da marcação fuzzy foi proposta a fim de garantir que o novo modelo seja consistente com a teoria geral das redes de Petri.

Posteriormente, uma abordagem baseada na combinação de um mecanismo de propagação de restrições relativos aos prazos de entrega e em um jogador de redes de Petri p-temporal foi apresentada. Para ilustrar esta abordagem, o “Serviço de Reclamações” foi simulado usando as diferentes formas de alocação de recursos. Pôde-se notar que somente no caso em que se usam recursos discretos e contínuos *fuzzy* os prazos foram respeitados. Isso se deve ao fato de que usando recursos contínuos *fuzzy* é possível, quando os prazos estão atrasados, sobrecarregar funcionários, como acontece geralmente na realidade.

A principal contribuição deste trabalho é a de propor, de forma bem próxima ao que acontece nas empresas, uma nova abordagem para a representação de recursos humanos em um sistema de gerenciamento de *workflow*.

Este trabalho teve como resultados, além da escrita desta dissertação, a publicação e apresentação do artigo “Fuzzy Continuous Resource Allocation Mechanisms in Workflow

Management Systems ”, no congresso IEEE IRI 2006: Heuristic Systems Engineering - “2006 International Conference on Information Reuse and Integration” realizado em Waikoloa-Hawaii.

A implementação do jogador de rede de Petri p-temporal e do jogador de rede de Petri p-temporal *fuzzy* está sendo implementado por um aluno de iniciação científica, como parte deste projeto financiado pela FAPEMIG (Fundação de Amparo à Pesquisa do Estado de Minas Gerais).

Como proposta de trabalhos futuros, acredita-se que um estudo sobre o cálculo da necessidade para o disparo das transições seja interessante, podendo, talvez, aproximar mais o mecanismo de alocação de recursos humanos da realidade nas empresas.

Além disso, a utilização de intervalos *fuzzy* para representar os intervalos p-temporais e aplicar um mecanismo de propagação de restrição *fuzzy* para uma melhor flexibilização dos prazos em cada atividade.

E por fim, a aplicação deste mecanismo de alocação de recurso contínuo *fuzzy* associado a intervalos p-temporais fuzzy para monitoramento de prazos em gestão de projetos.

Bibliografia

- [Aalst e Hee, 2002] Aalst, W.v.d., Hee, K.v. (2002). Workflow Management: Models, Methods, and Systems. The MIT Press Cambridge, Massachusetts. London, England.
- [Alonso et al, 1997] Alonso G.; Agrawal, D.; Abbadi, A. et al. (1997). Functionality and Limitations of current Workflow Management Systems. *Submitted to IEEE Expert*.
- [Araujo e Borges, 2001] Araujo, R. M. ; Borges, M. R.(2001) *Sistemas de Workflow* In: *XX Jornada de Atualização em Informática - Congresso SBC*, Fortaleza - CE.
- [Berry e Drabble, 1999] Berry P. M.; Drabble B. Swim (1999): *An AI-based Systems for Workflow Enabled Reactive Control* Proceedings of the IJCAI Workshop on Workflow and Process Management, IJCAI-99.
- [Booch et al., 1986] Booch, G. (1986) Object Oriented development. *IEEE Transactions on Software Engineering*. Vol 12, n.2, p. 211-221.
- [Cardoso et al, 1999] Cardoso, J., Valette, R., Dubois, D. (1999). Possibilistic Petri Nets. *IEEE Trans. on Systems, Man, and Cybernetics - Part B*. Vol. 29, No. 5. P. 573-582.
- [Casati, 1996] Casati. F. et al.(1996), *Workflow Evolution*. In *Proceedings of ER'06*. Springer Verlag. Cottbus, Germany.
- [Chaffey, 1998] Chaffey, D. (1998) *Groupware, Workflow and Intranets - Reengineering the Enterprise with Collaborative Software*. Digital Press.
- [Champagnat et al., 1998] Champagnat, R.; Pingaud, H.; Alla, H. et al. (1998). A gas storage example as a benchmark for hybrid modeling. *APII-JESA, Special Issue on Automation of mixed process and hybrid dynamical systems*. Vol 32, N. 9-10, p. 1233-1253.

- [Covès et al., 1998] Covès, C.; Crestani, D.; Prunet, F. (1998). Design and Analysis of Workflow Process with Petri nets. *In: 1998 IEEE Internacional Conference on Systems, Man and Cybernetics Proceedings of IEEE SMC'1998*. vol.1. p. 101-106.
- [David e Alla, 2004] David, R., Alla, H. (2004). Discrete, Continuous, and Hybrid Petri Nets. Springer.
- [del Foyo e Silva, 2003] del Foyo, P. G., Silva, J. R. (2003) Towards a unified view of Petri nets and object oriented modeling. *In: 17th International Congress of Mechanical Engineering. Proceedings of COBEM 2003*. São Paulo, Brasil.
- [Dubois, Prade, 1988] Dubois, D., Prade, (1988). Possibility theory. Springer.
- [Elmaghraby, 1977] Elmaghraby, S., E.;(1977) Activity network. *John Wiley and Sons*.
- [Eshuis, 2002] Eshuis, H.; (2002). *Semantic and Verification of UML Activity Diagrams for Workflow Modeling*. PhD thesis, University of Twente, Enschede, The Netherlands.
- [Esquirol e Lopez, 1995] Esquirol, P., Huguet, M.J., Lopez, P. (1995). Modeling and managing disjunctions in scheduling problems. *Journal of Intelligent Manufacturing* 6. pp. 133-144.
- [Florin e Natkin, 1984] Florin, G., Natkin, S. (1984) *Définition formelle des Réseaux de Petri Stochstiques*. Technical Report CNAM -Paris. Springer-Verlag.
- [Francielle, 2005] Francielle, F. (2005) *Problema do Escalonamento em Tempo Real dos Sistemas de Gerenciamento de Workflow Baseado em um Modelo de Rede de Petri Híbrida p-Temporal*. Tese (Mestrado em Computação) - Faculdade de Computação, Universidade Federal de Uberlândia, Minas Gerais - 2005.
- [Freitas Filho, 2000] Freitas Filho, L. (2000) *Sistemas Workflow em Processos Empresariais aplicando técnicas de Processos Produtivos Fabris e Modelagem em Redes de Petri*. Dissertação de Mestrado, Universidade Mackenzie, São Paulo.
- [Genrich, 1987] Genrich, h. J.; (1987) Predicate-Transition Nets. *Lecture Notes in Computer Science*, Springer-Verlag.

- [Georgakopoulos e Shet, 2004] Georgakopoulos, D.; Hornick, M.; Shet, A. (2004) *An Overview of Workflow Management: From Process Modeling to Workflow Automation Infrastructure* Distributed and Parallel Databases, p.26-33.
- [Gerrit, 2000] Gerrit, K.; Verelst, J. Jan; Weyn, B. Bart et al. (2000). Techniques for modeling workflows and their support of reuse. *Business Process Management*, LNCS 1806, (W. van der Aalst et al., Eds.) 2000, p. 1-15.
- [Gil, 2004] Gil, Y. et al. (2004). *Artificial Intelligence and Grids: Workflow Planning and Beyond*. IEEE Intelligent Systems, v. 19, n. 1. p.26-33.
- [Gondran e Minoux, 1984] Gondran, M, M. Minoux (1984) Graphes et algorithmes. *Eyrolles*.
- [Gotha, 1993] Gotha (1993). Les problèmes d'ordonnement. *Recherche opérationnelle*.
- [Hollingsworth, 1985] Hollingsworth, D. (1985). *The Workflow Reference Model*. *Workflow Management Coalition* Document Number TC00-1003. Document Status - Issue 1.1.
- [Jensen, 1990] Jensen, K. (1990). Coloured Petri nets: a high level language or system design and analysis. Advances in Petri Nets (G. Rozemberg, Ed), *Lecture Notes in Computer Science*. Springer Verlag.
- [Jeske e Julia, 2006] Jeske, J. Julia, S. Fuzzy Continuous Resource Allocation Mechanisms in Workflow Management System. *In: IEEE IRI 2006: Heuristic Systems Engineering*. Waikoloa, Hawaii, USA.
- [Julia, 1997] Julia, S.; (1997). "Conception et pilotage de cellules flexibles à fonctionnement répétitif modélisées par réseaux de Petri". Tese de doutorado, Université Paul Sabatier. Toulouse, França.
- [Julia et al, 1995] Julia, S., Valette; R. Tazza, M. (1995). Computing a flexible schedule under a set of cyclical constraints. *In: Second International Conference on Industrial Automation*, Paris, França.
- [Julia, 1998] Julia, S. (1998). Da concepção ao controle em tempo real de sistemas flexíveis de manufatura usando as redes de Petri. *Segundo SBAI*. p.243-248. Curitiba, Brasil.

- [Julia e Francielle, 2004] Julia, S., Francielle, F. (2004). A p-time hybrid Petri net model for the scheduling problem of Workflow Management Systems. IEEE International Conference on Systems, Man and Cybernetics. The Hague.
- [Julia, Valette, 2000] Julia, S., Valette, R. (2000). Real Time Scheduling of Batch Systems. *Simulation Practice and Theory*. Elsevier Science. pp. 307-319.
- [Julia e Kanacilo, 2002] Julia, S.; Kanacilo, E. M.; (2002) An approach based on dynamic UML diagrams and on a token player algorithm for the scenario verification of real time systems. 14th *European Simulation Symposium, Simulation in Industry* Dresden, German. p.377-381.
- [Khansa, 1997] Khansa, W. 1997. *Réseaux de Petri p-temporels Contribution a l'étude des systemes a événements discrets*. Thèse de Docteur. Université de Savoie, França.
- [Khansa e Denat, 1996] Khansa, W., Aygaline, P., Denat, J. P. (1996). Structural analysis of p-time Petri Nets. *Symposium on discrete events and manufacturing systems. CESA'96 IMACS Multiconference*. Lille, France.
- [Klir e Yuan, 1995] Klir, G.J., Yuan, B. (1995). *Fuzzy Sets and Fuzzy Logic - Theory and Applications*. Imp Upper Saddle River: Prentice Hal.
- [Lee e DiCesare, 1994] Lee, D.Y., DiCesare, F. (1994). *Scheduling flexible manufacturing systems using Petri nets and heuristic search*. IEEE Trans. on Robotics and Automation 10 (3). p. 123-132.
- [Li et al.,2003] Li, J.; Fan, Y.; Zhou, M. (2003) Timing Constraint Workflow Nets for Workflow Analysis. *IEEE Transactions on Systems, Man and Cybernetics Part A: Systems and Humans*. Vol. 33, n. 2, March.
- [Marshak, 1995] Marshak. R. T. (1995) *Workflow Applying Automation to Group Processes*. In: Coleman, D; Khana, R. *Groupware Technology and Applications*, Upper Saddle River, Nj, USA: Prentice Hall.
- [Meidanis e Weske, 1996] Meidanis, J.; Vossen, G.; Weske, M. (1996) *Using Workflow Management in DNA Sequencing*In: *Proceedings: The First International Conference on Cooperative Information Systems*. Los Alamitos, California, USA.

- [Menasche, 1982] Menasche, M. (1982) *Analyse des Réseaux de Petri Temporisés et Application aux Systèmes Distribués* Thèse de Docteur. Université Paul Sabatier, Toulouse, França.
- [Merlin, 1974] Merlin, P. (1974) *A Study of Recoverability of Computer Systems*. PhD Thesis. University of California. Irvine, EUA.
- [Minoux, 1983] *Programmation Mathématique: Théorie et Algorithme*. Dunod.
- [Moncelet et al., 1998] Moncelet, G.; Christensen, S.; Paudeto M et al. (1998). *Dependability evaluation a simple mechatronic system using colored Petri Nets*. Workshop on practical use of colored Petri nets and Design CPN.
- [Moraes, 2000] Moraes, C.M. (2000). Escalonamento de sistemas de produção híbridos usando uma rede de Petri p-temporal com mecanismo de retrocesso inteligente. Dissertação de Mestrado, Faculdade de Computação. Universidade Federal de Uberlândia.
- [Moreno e Meziat, 2000] R-Moreno, M.D.; Kearney, P.; Meziat, D.A (2000) *A Case Study: Using Workflow and AI planners*. In: Nineteenth Workshop of the UK Planning and Scheduling Special Interest Group, Reino Unido: p. 175-184.
- [Murata, 1989] Murata, T. (1989). *Petri Nets: Properties, analysis and applications*. Proceedings of the IEEE 77(4). p. 541-580.
- [Myers, 1998] Myers, K. L. ; Berry, P. M.(1998). *Workflow Management Systems* In. *AI Perspective*, Technical Report, Artificial Intelligence Center. Menlo Park, CA (USA): SRI International.
- [Nicolao, 1998] Nicolao, M.; (1998). *Modelagem de Workflow utilizando um Modelo de Dados Temporal Orientado a Objetos com Papéis*. Dissertação de Mestrado, Universidade Federal do Rio Grande do Sul, Porto Alegre.
- [Peterson, 1981] Peterson, J. L. (1981). *Petri Net Theory and the Modeling of Systems* Prentice Hall.
- [Petri, 1966] Petri, C. A (1966) *Communication with automata*. Technical Report RADC-TR-65-377. Darmstadt University.

- [Plesums, 2002] Plesums, C.(2002) *Introduction to Workflow* . In Fischer, L. (Ed): *Workflow Handbook 2002*, Lighthouse Point (FL): Future Strategies Inc pp, 19-28.
- [Prins, 1994] Prins, C. (1994). Algorithmes de Graphes (avec programmes en pascal) *Eyrolles*.
- [Ramchandani, 1974] Ramchandani, C.:(1974) *Analysis of asynchronous concurrent systems by timed Petri net Models* PhD Thesis. MIT. Project MAC TR-120.
- [Sanders, 1998] Sanders, M. J. (1998). Constraint Programming with object oriented Petri nets. *In: IEEE Internacional Conference on Systems, Man and Cybernetics Proceedings of IEEE SMC'1998*. p. 289-294.
- [Scarpelli e Gomide, 1993] H. Scarpelli, F. Gomide - Fuzzy Reasoning and Fuzzy Petri Nets in *Manufacturing Systems Modeling, Journal of Intelligent and Fuzzy Systems*, v.1, n.3, 1993, pp.225-241.
- [Shael, 1991] Shael, T. et al. (1991) *Design Principles for Cooperative Office Support Systems in Distributed Process Management* In: Support Functionality in the Office Environment, A. Verrijn-Stuart (ed), Noth Holland.
- [Sifakis, 1977] Sifakis, J.:(1977). Use of Petri nets for Performance evaluation. In *3rd Internacional Symposium on Modeling and Performance Evaluation of Computer Systems*. (H. Beilner and E. Gelembe, Eds.) North Holland.
- [Silbertin, 1985] Silbertin, c.:(1985). High-Level Petri nets with Data Structures. In *European Workflow on Application and Theory of Petri Nets*. Ed. Digital Systems Laboratory. Helsink, Finland.
- [Tazza, 1987] Tazza, M. (1987). Quantitative Analysis of a resource allocation problem: a net theory based proposal. *Concurrency and nets*. p.511-532. Springer-Verlag.
- [Tramontina et al, 2004] Tramontina, G.B.; Wainer, J.; Ellis, C. (2004). Applying Scheduling Techniques to Minimize the Number of Late Jobs in Workflow Systems. *In: 2004 ACM Symposium on Applied Computing*.
- [Usinoro, 2003] Usinoro, C.H. (2003). *Tecnologia Workflow: O impacto de sua utilização nos processos de negócio. Um estudo de casos múltiplos*. Dissertação de Mestrado, Departamento de Administração, Universidade de São Paulo.

- [Valette, 1979] Vallete, R. (1979) Analysis of Petri Nets by Stepwise Refinements. *Journal of Computer and System Sciences*. Academic Press, New York and London.
- [WFMC, 2006] Workflow Management Coalition, 2004, *The Workflow reference model*. 2006. Disponível em < [http : //www.wfmc.org](http://www.wfmc.org) >. Acesso em 08/05/2006.
- [Wirtz e Giese, 2000] Wirtz, G.; Giese, H. (2000) Using UML and Object-Coordination-Nets for Workflow Specification. In: *2000 IEEE International Conference on Systems, Man and Cybernetics. Proceedings of IEEE SMC'2000*. vol.5, p. 3159-3164.
- [Zadeh, 1965] Zadeh, L.A. (1965). Fuzzy sets. *Information and Control.*, vol. 8. pp. 338-353.