



Universidade Federal de Ouro Preto  
Instituto de Ciências Exatas e Aplicadas  
Departamento de Engenharia Elétrica

## Trabalho de Conclusão de Curso

# Planejamento Ótimo Baseado na Teoria de Controle Supervisório Aplicado a um Sistema de Triagem

Franklin Couto Marinho

João Monlevade, MG  
2019

Franklin Couto Marinho

# Planejamento Ótimo Baseado na Teoria de Controle Supervisório Aplicado a um Sistema de Triagem

Trabalho de Conclusão de curso apresentado à Universidade Federal de Ouro Preto como parte dos requisitos para obtenção do Título de Bacharel em Engenharia Elétrica pelo Instituto de Ciências Exatas e Aplicadas da Universidade Federal de Ouro Preto.

Orientador: Prof. Hugo Jerzy Bravo Cipriano

Coorientador: Prof. Rodrigo Augusto Ricco

**João Monlevade**  
**2019**

## SISBIN - SISTEMA DE BIBLIOTECAS E INFORMAÇÃO

M338p Marinho, Franklin Couto .

Planejamento ótimo baseado na teoria de controle supervísório aplicado a um sistema de triagem. [manuscrito] / Franklin Couto Marinho. - 2019.

67 f.: il.: color., tab.. + Apêndice.

Orientador: Prof. Me. Hugo Jerzy Bravo Cipriano.

Coorientador: Prof. Dr. Rodrigo Augusto Ricco.

Monografia (Bacharelado). Universidade Federal de Ouro Preto. Instituto de Ciências Exatas e Aplicadas. Graduação em Engenharia Elétrica .

1. Planejamento. 2. Sistemas de controle supervísório. 3. Controle de processo. 4. Engenharia elétrica. I. Cipriano, Hugo Jerzy Bravo. II. Ricco, Rodrigo Augusto. III. Universidade Federal de Ouro Preto. IV. Título.

CDU 681.51

Bibliotecário(a) Responsável: Flavia Reis - CRB6- 2431



#### ANEXO IV - ATA DE DEFESA

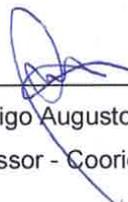
Aos 03 dias do mês de dezembro de 2019, às 13:30 horas, no bloco H deste instituto, foi realizada a defesa de monografia pelo (a) formando (a) **Franklin Couto Marinho**, matrícula 14.1.8256, sendo a comissão examinadora constituída pelos professores: Wendy Yadira Eras Herrera, Rodrigo Augusto Ricco, Marcelo Moreira Tiago, Elton Máximo Cardoso, e Hugo Jerzy Bravo Cipriano. O (a) candidato (a) apresentou a monografia intitulada: **Planejamento Ótimo Baseado na Teoria de Controle Supervisório Aplicado a um Sistema de Triagem**. A comissão examinadora deliberou, por unanimidade, pela APROVAÇÃO do candidato, com a nota média 8,0, de acordo com a tabela 1. Na forma regulamentar foi lavrada a presente ata que é assinada pelos membros da comissão examinadora e pelo (a) formando(a).

Tabela 1 – Notas de avaliação da banca examinadora

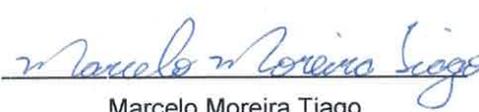
Banca Examinadora	Nota
Hugo Jerzy Bravo Cipriano	8,0
Rodrigo Augusto Ricco	8,0
Wendy Yadira Eras Herrera	8,0
Marcelo Moreira Tiago	8,0
Elton Máximo Cardoso	8,0
<b>Média</b>	<b>8,0</b>

João Monlevade, 03 de dezembro de 2019

  
Hugo Jerzy Bravo Cipriano  
Professor - Orientador

  
Rodrigo Augusto Ricco  
Professor - Coorientador

  
Wendy Yadira Eras Herrera  
Professora

  
Marcelo Moreira Tiago  
Professor

  
Elton Máximo Cardoso  
Professor

  
Franklin Couto Marinho  
Aluno

## Agradecimentos

Agradeço ao meu orientador, ao meu co-orientador, aos meus professores, aos técnicos, aos meus amigos e à minha família. Todos foram de grande importância em minha caminhada até então, não há palavras que possam descrever o quanto sou grato pela ajuda, motivação e força que vocês me deram no decorrer desses anos. Novamente agradeço a todos pela ajuda que me deram.

### *Resumo*

Os sistemas de triagem são sistemas complexos que demandam coordenação de diversas ações utilizando-se apenas sensores como sinais de entrada. Neste trabalho, realiza-se a correção e otimização de controle de um sistema de triagem, cujo supervisor não descrevem corretamente o comportamento do sistema. Utilizando a abordagem modular local da teoria de controle supervisão, o sistema foi decomposto em sete equipamentos cujo comportamento é coordenado por controle supervisão. O sistema teve seu comportamento descrito por autômatos de estados finitos e a modelagem foi realizada no Visual Studio, com auxílio da biblioteca UltraDES, desenvolvida especialmente para modelagem de sistema a eventos discretos. As plantas e especificações foram remodeladas para corrigir os erros de funcionamento e otimizar os supervisores responsáveis pelo direcionamento das caixas para os armazéns. Em seguida os supervisores foram sintetizados pela abordagem modular local, permitindo comparar a complexidade computacional de cada supervisor. Finalmente verificou-se que o sistema não apresentara conflito entre os supervisores modulares locais, validando a modelagem aplicada.

**Palavras-chave:** Sistemas a Eventos Discretos, Sistema de Triagem, Controle Supervisão Modular Local, UltraDES.

*Abstract*

Screening systems are complex systems that require coordination of several actions using only sensors as input signals. In this work, the correction and optimization of control of a screening system is performed, whose supervisors do not correctly describe the behavior of the system. Using the local modular approach of supervisory control theory, the system was decomposed into seven equipment whose behavior is coordinated by supervisory control. The system had its behavior described by finite state automata and modeling was performed at Visual Studio, with the aid of the UltraDES library, specially developed for system modeling to discrete events. Plants and specifications have been remodeled to correct malfunctions and optimize the supervisors responsible for targeting the boxes to the warehouses. Then the supervisors were synthesized by the local modular approach, allowing to compare the computational complexity of each supervisor. Finally, it was found that the system had not presented conflict between local modular supervisors, validating the applied modeling.

**Keywords:** Discrete Event Systems, Screening System, Local Modular Supervisory Control, UltraDES.

# Lista de Figuras

2.1	Autômato $G$ do Exemplo 2.1. . . . .	10
2.2	Autômato $G$ do Exemplo 2.2. . . . .	12
2.3	Autômato $Trim(G)$ do Exemplo 2.2. . . . .	12
2.4	Exemplo 2.3: (a) $G_1$ ; (b) $G_2$ e (c) $G_1 \parallel G_2$ . . . . .	13
2.5	Sistema em malha fechada. . . . .	14
2.6	Linha de Transferência. . . . .	17
2.7	Máquinas $M_1$ e $M_2$ . . . . .	18
2.8	Modelo da planta $G$ . . . . .	18
2.9	Especificação para <i>overflow</i> e <i>underflow</i> . . . . .	19
2.10	Autômato $R$ . . . . .	19
2.11	Autômato $S$ . . . . .	19
2.12	Comportamento de um Supervisor. . . . .	20
2.13	Célula de Manufatura Didática. Adaptado de Curzel e Leal (2008) . . . .	23
2.14	Autômato para a esteira. . . . .	24
2.15	Autômato para o robô 1 . . . . .	25
2.16	Autômato para a mesa giratória e o sensor de teste. . . . .	26
2.17	Autômato para o robô 2. . . . .	26
2.18	Especificação de funcionamento sobre o robô 1 e a esteira. . . . .	27
2.19	Especificação de funcionamento sobre o robô 1 e a mesa giratória. . . .	27

2.20	Especificação de funcionamento sobre a mesa giratória e o robô 2. . . . .	28
2.21	Supervisor sobre a esteira e o robô 1. . . . .	29
2.22	Supervisor sobre o robô 1 e a mesa giratória. . . . .	29
2.23	Supervisor sobre a mesa giratória e o robô 2. . . . .	30
3.1	Sistema de Triagem. . . . .	32
3.2	Sistema de Triagem. . . . .	34
3.3	Planta do Tapete Alimentador. Adaptado de de Aguiar et al. (2012). . . . .	35
3.4	Ilustração da Mesa Transportadora de Entrada. . . . .	35
3.5	Planta da Mesa Transportadora de Entrada. Adaptado de de Aguiar et al. (2012). . . . .	36
3.6	Ilustração da Mesa Rotatória. . . . .	36
3.7	Planta da Mesa Rotatória. Adaptado de de Aguiar et al. (2012). . . . .	37
3.8	Ilustração das Mesas Transportadoras de Saída. . . . .	38
3.9	Planta das Mesas Transportadoras de Saída. Adaptado de de Aguiar et al. (2012). . . . .	38
3.10	Planta dos Elevadores. Adaptado de de Aguiar et al. (2012). . . . .	38
3.11	Restrição de funcionamento <i>ROpl</i> . Adaptado de de Aguiar et al. (2012). . . . .	39
3.12	Especificação de carga da Mesa Transportadora de Entrada. Adaptado de de Aguiar et al. (2012). . . . .	40
3.13	Especificação de funcionamento da Mesa Rotatória. Adaptado de de Aguiar et al. (2012). . . . .	40
3.14	Especificação de carga das Mesas Transportadoras de Saída. Adaptado de de Aguiar et al. (2012). . . . .	41
3.15	Especificação de carga dos Elevadores. Adaptado de de Aguiar et al. (2012). . . . .	41
3.16	Especificação de Roteamento. Adaptado de de Aguiar et al. (2012). . . . .	42

3.17	Esquemático para Síntese dos Supervisores. . . . .	43
3.18	Ilustração da Ativação <i>Fanstasma</i> dos Sensores. . . . .	45
4.1	Planta de TA. . . . .	48
4.2	Planta de MTE. . . . .	48
4.3	Planta da MR. . . . .	49
4.4	Planta de MTS1 e MTS2. . . . .	49
4.5	Planta de ED1 e ED2. . . . .	50
4.6	Especificação de carga de MTE. . . . .	51
4.7	Especificação de correção de funcionamento sobre TA e MTE. . . . .	51
4.8	Especificação de funcionamento sobre a Mesa Rotatória. . . . .	52
4.9	Especificação de Funcionamento sobre a Mesa Transportadora de Saída 1. . . . .	53
A.1	Comando para criação de estados. . . . .	65
A.2	Comando para criação de eventos. . . . .	66
A.3	Comando para criação de eventos. . . . .	66
A.4	Exemplificação do uso dos comandos para criação de autômatos: . . . . .	67
A.5	Comando para criação de supervisores. . . . .	67

# Lista de Tabelas

2.1	Exemplo 2.1: Tabela da função de transição de estados. . . . .	10
2.2	Tabela de função dos sensores. Adaptado de Curzel e Leal (2008). . . . .	23
2.3	Tabela de função dos atuadores. Adaptado de Curzel e Leal (2008) . . . .	24
2.4	Tabela de função dos sensores dos supervisores do Exemplo 2.5. . . . .	28
3.1	Tabela de função dos sensores. Adaptado de de Aguiar et al. (2012). . . .	33
3.2	Tabela de função dos atuadores. Adaptado de de Aguiar et al. (2012). . .	33
3.3	Especificações e Plantas Locais Associadas. Adaptado de de Aguiar et al. (2012). . . . .	43
3.4	Resultados para síntese de Supervisores Modulares. Adaptado de de Aguiar et al. (2012). . . . .	44
4.1	Especificações e Plantas Otimizadas Associadas. . . . .	53
4.2	Resultados para síntese de Supervisores Modulares Otimizados. . . . .	54
5.1	Especificações e Plantas Locais Associadas. Adaptado de de Aguiar et al. (2012). . . . .	56
5.2	Resultados para síntese de Supervisores Modulares. Adaptado de de Aguiar et al. (2012). . . . .	56
5.3	Especificações e Plantas Otimizadas Associadas. . . . .	57
5.4	Resultados para síntese de Supervisores Modulares Otimizados. . . . .	57

# Capítulo 1

## Introdução

### 1.1 Contextualização

Atualmente, a tecnologia moderna tem produzido diversos sistemas com o objetivo de executar tarefas de alta complexidade, de forma eficiente e automatizada. Estes sistemas estão presentes em uma série de aplicações, como por exemplo a robótica, controle de estoque, organização de serviços, automação e otimização da manufatura. Os sistemas são um conjunto de elementos que interagem entre si com o intuito de realizar uma tarefa específica. Esses percebem as alterações do ambiente ao seu redor, através da recepção de estímulos, denominados eventos([Cury, 2001a](#)). São exemplos de eventos o apertar de um botão, o início e término de uma tarefa e a percepção de uma mudança na leitura de um sensor. Estes eventos são considerados instantâneos, o que denota um caráter discreto no tempo, logo, a evolução destes sistemas depende apenas da ocorrência de eventos e não de intervalos de tempo definidos ([Cassandras e Lafortune, 2009](#)). Sistemas que apresentam essas características são denotados como sistemas a eventos discretos (SEDs).

O comportamento de um SED pode ser descrito pelo conjunto de todas as possíveis seqüências de eventos geradas por ele. Esse conjunto é denominado "linguagem" do

sistema (Alves, 2014). Entretanto, a linguagem gerada por um SED pode alcançar comportamentos indesejáveis para uma dada tarefa específica. Logo, é possível alterá-los através da adição de um controlador, que possui o propósito de evitar a geração desses comportamentos. Os SEDs são sistemas cujo o controle e a modelagem não podem ser realizados utilizando os métodos convencionais baseados em equações diferenciais, pois devido a natureza discreta destes sistemas, a teoria de controle clássica se torna inadequada à sua representação (Cury, 2001a). Ramadge e Wonham foram os pioneiros na modelagem e controle de sistemas a eventos discretos, baseando-se nos conceitos de linguagens formais e autômatos finitos (Ramadge e Wonham, 1989a), conhecido como modelo RW (Ramadge e Wonham, 1987a). Este modelo é dividido entre o sistema a ser controlado, denominado planta, e a entidade que o controla, chamado de supervisor (Cury, 2001a). A planta é um modelo que representa o comportamento de um SED, ou seja, todas as ações que este sistema realizar sem a presença de uma unidade de controle, como por exemplo, uma esteira transportando caixas de um armazém e um braço robô retirando as caixas e as colocando em um baú de um caminhão. Entretanto, dependendo das ações que um SED realiza, para uma determinada tarefa, algumas destas ações podem ser "indesejadas", retomando o exemplo da esteira e do braço robô, uma ação indesejada seria o robô não retirar uma caixa da esteira e a caixa cair fora do sistema. Essas ações ditas indesejáveis são evitadas com a utilização do supervisor, cujo papel é restringir o comportamento da planta desabilitando eventos indesejáveis (Cury, 2001a).

Um supervisor é modelado com base na teoria de controle supervisorio (TCS), em que sua atuação sobre a planta consiste na habilitação e desabilitação de seus eventos controláveis. Alguns desses eventos podem ser não controláveis, ou seja, dependendo do sistema em que o controle é realizado, o supervisor não pode habilitar ou desabilitar eventos em toda a extensão da planta (Pena, 2007). A forma como o supervisor irá controlar um sistema depende das tarefas que o sistema deve cumprir, sendo assim,

a modelagem é realizada sobre uma especificação (Cury, 2001a), por exemplo, se um forno industrial suporta temperaturas de até 1000<sup>o</sup> celcius, um controlador para esse sistema é modelado de forma a impedir que o sistema ultrapasse a temperatura limite especificada.

Em sistemas industriais, geralmente a planta a ser controlada é formada por um conjunto de subsistemas distintos que operam conjuntamente para a realização de uma tarefa ou confecção de um produto. Estes subsistemas trabalham de forma concorrente e possuem comportamentos distintos entre si, responsáveis por tarefas particulares, como por exemplo, transportar, montar e armazenar subprodutos (Pena, 2007). Devido a esses subsistemas, a planta como um todo apresenta grande complexidade e a modelagem de um controlador para esse sistema se torna extremamente essencial para garantir qualidade e eficiência na confecção. Em resumo, a teoria de controle supervi-sório considera a planta como bloco único, e o supervisor age sobre ela, entretanto, o supervisor de sistemas como estes apresentam uma grande complexidade e são de difícil implementação (de Aguiar et al., 2012). Para resolver esse empecilho, foi desenvolvidas diversas abordagens para a modelagem de supervisores, dentre elas estão o controle hierárquico, o controle descentralizado, o controle modular e o controle modular local.

O controle Hierárquico consiste da divisão do sistema em diversos níveis de abstração, e a informação desses níveis é utilizada para a modelagem de um controlador para um nível superior (Muler, 2018). A modelagem de supervisores através do controle descentralizado é realizada somente se o sistema observado pode ser dividido em subtarefas, sendo a modelagem dos supervisores dessas subtarefas mais simples, e cada um desses supervisores é responsável por um subconjunto de eventos da planta (Lin e Wonham, 1991).

O controle modular modela o supervisor de forma que o sistema obedeça um conjunto de restrições que são divididas em subtarefas e que são realizadas por diferentes supervisores (Pena, 2007). É uma forma de modelagem parecida com o controle modu-

lar clássico, denominada controle modular local (CML), apresenta a mesma ideia do controle modular, entretanto, ao invés de utilizar a linguagem da planta como um todo para a modelagem, a mesma é dividida em subplantas que são restringidas por especificações locais (Pena, 2007). Logo, menos autômatos são projetados, as linguagens que representam esses autômatos são menores em comparação com as abordagens hierárquica, descentralizada e modular clássica e a complexidade na síntese dos supervisores é reduzida.

## 1.2 Formulação do Problema

O trabalho em questão trata da síntese de supervisores para um sistema de triagem, um tipo de sistema de manufatura que consiste do transporte de caixas de dois tamanhos, grandes e pequenas, e dependendo dessa característica, o sistema as destina à armazéns diferentes. A planta é modelada e seus supervisores são sintetizados através da teoria de controle modular local, pelo fato da redução de complexidade de sua modelagem, assim como é apresentado no artigo de [de Aguiar et al. \(2012\)](#).

O trabalho de [de Aguiar et al. \(2012\)](#) divide o sistema de triagem em sete equipamentos, sendo eles o tapete alimentador (TA), a mesa transportadora de entrada (MTE), a mesa rotatória (MR), as mesas transportadoras de saída um e dois (MTS1 e MTS2) e os elevadores de carga um e dois (ED1 e ED2). O tapete alimentador é responsável por inserir caixas pequenas e grandes no sistema. A mesa transportadora de entrada rotula as caixas pelo tamanho para serem corretamente direcionadas aos armazéns. A mesa rotatória é responsável por direcionar as caixas pequenas à mesa transportadora de saída 1 e as caixas grandes à mesa transportadora de saída 2. As mesas transportadoras de saída transportam as caixas até os elevadores, responsáveis por armazená-las adequadamente nos depósitos.

Os eventos associados ao sistema estão relacionados com o acionamento de sensores.

A modelagem dos equipamentos está associada às especificações de limite de carga de cada um. O tapete alimentador possui capacidade indefinida. A mesa transportadora de entrada possui capacidade de 8 caixas. A mesa rotatória possui limite de carga de 1 caixa. As mesas transportadoras de saída possuem carga máxima de 4 caixas. Finalmente os elevadores possuem carga de 1 caixa. Com base nessas especificações, os supervisores são modelados por meio da teoria de controle modular local (Pena, 2007).

Após a verificação da modelagem realizada no trabalho de [de Aguiar et al. \(2012\)](#), identificou-se o mau funcionamento de alguns dos supervisores, causado pela generalização na modelagem das plantas e ausência de especificações sobre o comportamento conjunto das plantas dos equipamentos. Esse mau funcionamento está relacionado à leitura “fanstasma” dos sensores de presença e de identificação de caixas. Dessa forma, o conjunto de supervisores modelado no artigo não descreve o comportamento do sistema de triagem e sua aplicação pode gerar gastos com energia e manutenção de equipamentos.

Em sistemas industriais, é de suma importância um funcionamento eficiente de todos os componentes do sistema, e problemas de mal funcionamento e de generalização na modelagem das plantas são inaceitáveis. Portanto, este trabalho consiste em solucionar o problema de mal funcionamento dos supervisores remodelando as plantas e especificações do sistema, considerando as especificações de carga e o funcionamento conjunto de cada equipamento, visando reduzir os supervisores em número e complexidade.

### 1.3 Organização do Texto

Após este capítulo introdutório, no capítulo 2 são apresentados os conceitos básicos para modelagem dos supervisores, como linguagens e teoria de autômatos, a teoria de controle supervísório e a teoria de controle modular local. O capítulo 3 apresenta a modelagem dos supervisores do sistema de triagem presente no trabalho de [de Aguiar et al. \(2012\)](#).

O capítulo 4 apresenta a modelagem dos supervisores otimizada e o capítulo 5 apresenta a discussão dos resultados deste trabalho comparados aos resultados do trabalho de [de Aguiar et al. \(2012\)](#).

# Capítulo 2

## Preliminares

Neste capítulo serão apresentadas definições, notações e terminologias fundamentais para o desenvolvimento teórico do presente trabalho. Tem-se como objetivo principal proporcionar ao leitor as noções chaves para um sólido entendimento dos formalismos teóricos apresentados nos capítulos posteriores. Grande parte dos assuntos tratados neste capítulo pertencem a conceitos estabelecidos na literatura. Para uma exposição detalhada desses conceitos sugere-se consultar ([Ramadge e Wonham, 1989a](#); [Hopcroft, 2008](#); [Cassandras e Lafortune, 2009](#); [Berstel et al., 2009](#); [Wonham, 2018](#)).

### 2.1 Linguagens Formais e Autômatos

Esta seção apresenta os fundamentos da teoria linguagens e autômatos, utilizada como meio para a modelagem da dinâmica dos SEDs.

#### 2.1.1 Eventos e Linguagens

Seja  $\Sigma$  um conjunto finito de *eventos*, também denominado *alfabeto*, assim uma *palavra*  $s$  sobre  $\Sigma$  é uma sequência finita de eventos de  $\Sigma$ , ou seja,

$$s = \sigma_1\sigma_2 \dots \sigma_n,$$

tal que  $\sigma_i \in \Sigma$ , para todo  $i = 1, \dots, n$ . Entende-se eventos como incidentes que alteram o comportamento de um SED. A sequência vazia chama-se *palavra vazia*, representada por  $\varepsilon$ . O *comprimento* de uma palavra  $s = \sigma_1\sigma_2\dots\sigma_n$ , dito  $|s|$ , é o número  $n$  de eventos que a compõem. Claramente, observa-se que  $|\varepsilon| = 0$ . O fechamento de Kleene  $\Sigma^*$  é o conjunto de todas as palavras sobre  $\Sigma$ , incluindo a palavra vazia. Por exemplo, o fechamento de Kleene do alfabeto binário  $\Sigma = \{0,1\}$  é o conjunto:  $\Sigma^* = \{\varepsilon, 0, 1, 00, 01, 10, 11, 000, 001, \dots\}$ . A *concatenação* é uma operação binária associativa em  $\Sigma^*$  designada por  $\bullet$ . Assim

$$\begin{aligned} \bullet : \Sigma^* \times \Sigma^* &\rightarrow \Sigma^* \\ (s, u) &\mapsto s \bullet u \end{aligned}$$

onde  $s, u \in \Sigma^*$ . De forma geral,  $s \bullet u$  é usualmente denotado por  $su$ . Considere  $s, u$  e  $v$  palavras sobre  $\Sigma^*$ . Se  $s$  concatenada com  $u$  forma a palavra  $v$ , então diz-se que  $s$  é *prefixo* de  $v$ , e denota-se como  $s \sqsubseteq v$ .

**Definição 2.1** (*Linguagem*) *Seja  $\Sigma$  um alfabeto. Uma linguagem  $L$  definida sobre  $\Sigma$  é qualquer subconjunto de  $\Sigma^*$ , isto é,  $L \subseteq \Sigma^*$ .*

O *prefixo fechamento* de  $L$  é o conjunto de todos os prefixos de palavras em  $L$ . Ou seja,  $\bar{L} = \{s \in \Sigma^* \mid s \sqsubseteq v \text{ para algum } v \in L\}$ . De maneira óbvia,  $L \subseteq \bar{L}$ . Por exemplo, o prefixo fechamento de  $L = \{abb\}$  é  $\bar{L} = \{a, ab, abb\}$ . Uma linguagem  $L$  chama-se *prefixo fechado* quando  $L = \bar{L}$ .

Nem sempre uma linguagem pode ser representada de forma compreensível. Nesse sentido, torna-se necessária a utilização de estruturas compactas que permitam definir linguagens arbitrárias e, ao mesmo tempo, facilitem a manipulação e análises das mesmas. A seguir, apresenta-se o conceito de autômato tido como uma abordagem formal

para representar, analisar e manipular linguagens.

### 2.1.2 Autômatos de Estados Finitos

Um autômato determinístico de estados finitos (ADEF) é uma estrutura capaz de representar uma linguagem formal.

**Definição 2.2** (*Autômatos Determinísticos Finitos*) *Um autômato determinístico de estados finitos é uma quintupla  $G = \langle \Sigma, X, \rightarrow_G, x^\circ, X^m \rangle$ , em que  $\Sigma$  é um alfabeto,  $X$  é o conjunto finito de estados,  $\rightarrow_G \subseteq X \times \Sigma \times X$  é a relação de transição entre os estados,  $x^\circ \in X$  é o estado inicial, e  $X^m \subseteq X$  é o conjunto de estados marcados.*

Uma transição em  $G$  pode ser representada pela expressão  $x_1 \xrightarrow{\sigma} x_2$ , na qual  $x_1$  é o estado de origem,  $x_2$  o é estado de destino e  $\sigma$  é o rótulo da transição. A relação de transição pode ser estendida para reconhecer palavras sobre  $\Sigma^*$ , ou seja, formalmente  $x_1 \xrightarrow{\epsilon} x_1$  para todo  $x_1 \in X$ , e  $x_1 \xrightarrow{s\sigma} x_3$  se  $x_1 \xrightarrow{s} x_2$  e  $x_2 \xrightarrow{\sigma} x_3$  para algum  $x_2 \in X$ . No restante deste trabalho, utiliza-se o termo *autômato* para referir-se a um autômato determinístico de estados finitos.

Um autômato pode ser representado como um grafo orientado, em que os vértices representam os estados, as arestas representam as transições de estados e os rótulos das arestas são os eventos que determinam as transições. O estado inicial é identificado com uma seta que se origina do nada e os estados marcados são representados por círculos duplos. Para ilustrar o conceito de autômato, o Exemplo 2.1 é apresentado.

**Exemplo 2.1** (*Cassandras e Lafortune, 2009*) *Dado um autômato  $G = \langle \Sigma, X, \rightarrow_G, x^\circ, X^m \rangle$  da Figura 2.1 distingue-se:  $X = \{x, y, z\}$  o conjunto de estados;  $\Sigma = \{a, b, g\}$  o alfabeto;  $x^\circ = x$  o estado inicial e  $X^m = \{x, z\}$  o conjunto de estados marcados. As transições entre estados são apresentadas na Tabela 2.1, em que a partir dos estados*

apresentados na primeira coluna, a ocorrência de um dos eventos de  $\Sigma$  encaminha o autômato a um novo estado.

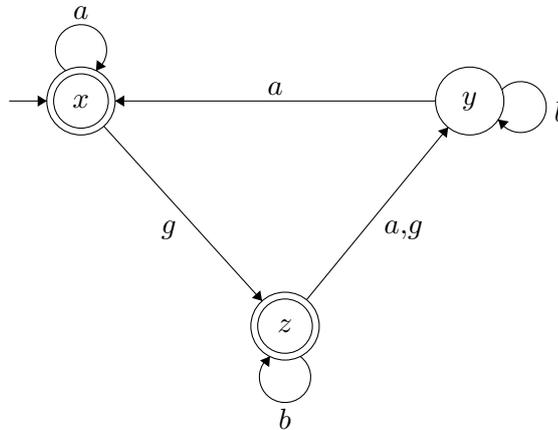


Figura 2.1: Exemplo 2.1: Autômato  $G$ .

Tabela 2.1: Exemplo 2.1: Tabela da função de transição de estados.

$\rightarrow_G$	a	b	g
x	x	$\emptyset$	z
y	x	y	$\emptyset$
z	y	z	y

### 2.1.3 Linguagens Representadas por Autômatos

A linguagem *reconhecida* (ou gerada) por um autômato  $G = \langle \Sigma, X, \rightarrow_G, x^\circ, X^m \rangle$  representa o conjunto de todas as palavras que podem ser geradas a partir do estado inicial, e escreve-se

$$\mathcal{L}(G) = \{s \in \Sigma^* \mid x^\circ \xrightarrow{s} x \text{ tal que } x \in X\}.$$

Por definição, a linguagem  $\mathcal{L}(G)$  é prefixo fechada, ou seja,  $\mathcal{L}(G) = \overline{\mathcal{L}(G)}$ . A *linguagem marcada* de  $G$  é um subconjunto de  $\mathcal{L}(G)$  composto por palavras que terminam em

estados marcados, e é definida como

$$\mathcal{L}_m(G) = \{s \in \Sigma^* \mid x^\circ \xrightarrow{s} x \text{ tal que } x \in X^m\}.$$

Pelo teorema de Kleene, linguagens (regulares) podem ser reconhecidas por autômatos.

Dado um autômato  $G = \langle \Sigma, X, \rightarrow_G, x^\circ, X^m \rangle$ , afirma-se que um estado  $x \in X$  é *acessível* em  $G$  se, para algum  $s \in \mathcal{L}(G)$ , tem-se  $x^\circ \xrightarrow{s} x$ , e *coacessível* se, para algum  $x^m \in X^m$  e  $s \in \mathcal{L}_m(G)$ , tem-se  $x \xrightarrow{s} x^m$ . Um autômato chama-se acessível se todos seus estados são acessíveis. A componente acessível de  $G$  é obtida a partir de  $G$  pela eliminação dos estados não acessíveis e transições associadas, indicada por  $Ac(G)$ . De forma semelhante, um autômato chama-se coacessível se todos seus estados são coacessíveis. A componente coacessível de  $G$  é obtida a partir de  $G$  pela eliminação dos estados não coacessíveis e transições associadas, manifesta por  $CoAc(G)$ . Um autômato é dito bloqueante se  $\overline{\mathcal{L}(G)} \subset \mathcal{L}(G)$ , ou seja, existem uma ou mais sequências de eventos que não conseguem alcançar um estado marcado. Finalmente, um autômato é considerado *trim* se for acessível e coacessível, denotando-se  $trim(G)$ .

**Exemplo 2.2** (*Cassandras e Lafortune, 2009*)

*Nota-se que no autômato  $G$  da Figura 2.2, o estado 6 não é acessível, pois não é possível alcançá-lo a partir do estado inicial. O estado 3 configura um tipo de bloqueio chamado de “deadlock”, pois uma vez alcançado, não é possível alcançar outros estados. Os estados 4 e 5 configuram um tipo de bloqueio chamado de “livelock”, pois uma vez alcançados o autômato entra em um “loop” entre esses estados.*

*Usando a definição de componente acessível elimina-se o estado 6, uma vez que não é possível alcançá-lo a partir do estado inicial, e usando a definição de componente coacessível elimina-se os estados 3, 4 e 5, pois uma vez alcançados não é possível alcançar um estado marcado. O resultado do processo da eliminação de estados bloqueantes dá*

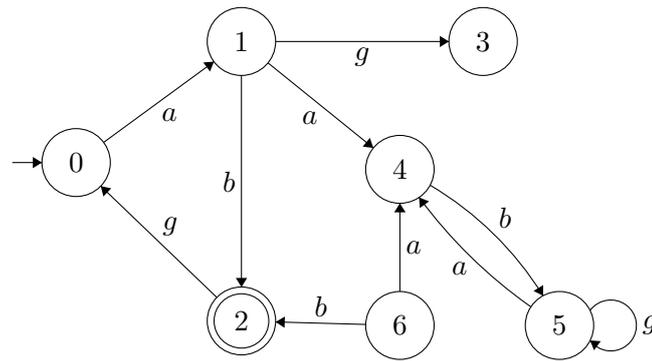


Figura 2.2: Exemplo 2.2: Autômato  $G$ .

como resultado um autômato *Trim*, como ilustrado na Figura 2.3.

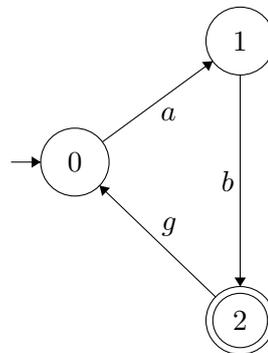


Figura 2.3: Autômato  $Trim(G)$  do Exemplo 2.2.

### 2.1.4 Composição Síncrona de Autômatos

Quando autômatos convergem para evoluir de forma concorrente, a sincronização só pode ocorrer quando eventos, pertencentes a um alfabeto comum, apareçam sincronizadamente. Os demais eventos se dão assincronamente, ou seja, de modo independente em cada um dos autômatos. Este procedimento denomina-se composição síncrona de autômatos.

**Definição 2.3** *Sejam  $G_1 = \langle \Sigma_1, X_1, \rightarrow_{G_1}, x_1^o, X_1^m \rangle$  e  $G_2 = \langle \Sigma_2, X_2, \rightarrow_{G_2}, x_2^o, X_2^m \rangle$ . A*

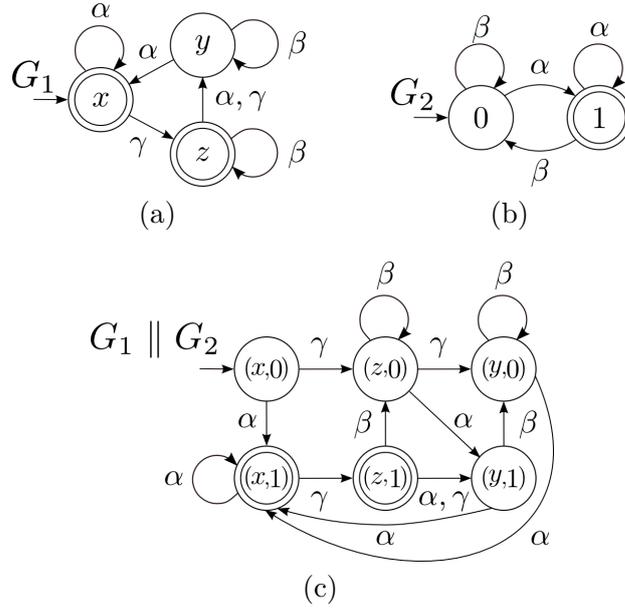


Figura 2.4: Exemplo 2.3: (a)  $G_1$ ; (b)  $G_2$  e (c)  $G_1 \parallel G_2$ .

composição síncrona de  $G_1$  e  $G_2$  é definida como

$$G_1 \parallel G_2 = (\Sigma_1 \cup \Sigma_2, X_1 \times X_2, \rightarrow, (x_1^{\circ}, x_2^{\circ}), X_1^m \times X_2^m)$$

na qual

$$\begin{aligned} (x_1, x_2) &\xrightarrow{\sigma} (x'_1, x'_2) && \text{se } \sigma \in \Sigma_1 \cap \Sigma_2, x_1 \xrightarrow{\sigma}_{G_1} x'_1, x_2 \xrightarrow{\sigma}_{G_2} x'_2; \\ (x_1, x_2) &\xrightarrow{\sigma} (x'_1, x_2) && \text{se } \sigma \in \Sigma_1 \setminus \Sigma_2, x_1 \xrightarrow{\sigma}_{G_1} x'_1; \\ (x_1, x_2) &\xrightarrow{\sigma} (x_1, x'_2) && \text{se } \sigma \in \Sigma_2 \setminus \Sigma_1, x_2 \xrightarrow{\sigma}_{G_2} x'_2. \end{aligned}$$

**Exemplo 2.3** Para ilustrar o uso da composição síncrona, considere os autômatos  $G_1$  e  $G_2$ , como mostrado nas Figuras 2.4a e 2.4b, respectivamente. O autômato  $G_1 \parallel G_2$ , mostrado na Figura 2.4c, é o resultado da composição síncrona de  $G_1$  e  $G_2$ .

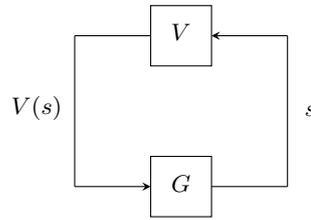


Figura 2.5: Sistema em malha fechada.

## 2.2 Teoria de Controle Supervisório

Revisam-se nesta seção alguns conceitos básicos da Teoria de Controle Supervisório iniciada por [Ramadge e Wonham \(1989a\)](#).

### 2.2.1 Controle Supervisório

No método proposto por [Ramadge e Wonham \(1989a\)](#) define-se distintamente o sistema a controlar, denominado *planta*, e o elemento controlador, denominado *supervisor*. A planta representa o modelo do sistema a ser controlado, isto é, o comportamento do sistema sem que nenhuma ação de controle seja realizada. O supervisor tem como propósito regular o comportamento da planta delimitando-a a certas especificações desejadas por meio da observação dos eventos gerados pela planta e, exercendo assim, uma ação de controle na forma de desabilitação, ou inibição, da ocorrência de certos eventos. Em TCS, considera-se que o supervisor  $S$  interage com a planta  $G$ , numa estrutura de malha fechada (Fig. 2.5) onde  $S$  observa os eventos ocorridos em  $G$  e define que eventos, dentre os fisicamente possíveis de ocorrerem no estado atual, são permitidos de ocorrerem a seguir.

A planta é modelada por um autômato  $G = \langle \Sigma, X, \rightarrow_G, x^\circ, X^m \rangle$ . Com o objetivo de realizar controle,  $\Sigma$  é particionado em dois subconjuntos disjuntos, o conjunto de eventos controláveis  $\Sigma_c$  e o conjunto de eventos não controláveis  $\Sigma_{nc}$ . O supervisor para  $G$  é uma função  $V : \mathcal{L}(G) \rightarrow 2^\Sigma$ , tal que  $V(s) \supseteq \Sigma_{nc}$  para todo  $s \in \mathcal{L}(G)$ , e pode ser implementada por um autômato  $S = \langle \Sigma, Z, \rightarrow_S, z^\circ, Z^m \rangle$ .

### 2.2.2 Controlabilidade

Seja  $E = \langle \Sigma, Y, \rightarrow_E, y^\circ, Y^m \rangle$  o autômato que modela a especificação de controle sobre a planta  $G = \langle \Sigma, X, \rightarrow_G, x^\circ, X^m \rangle$ . O *comportamento desejado*  $K$  do sistema pode ser alcançado.  $K$  é definido pela linguagem  $K = \mathcal{L}_m(G||E) \subset \mathcal{L}(G)$ , em que  $K \neq \emptyset$  e sua controlabilidade é determinada pela propriedade apresentada em seguida.

**Definição 2.4** (*Controlabilidade de  $K$* )  $K$  será controlável com respeito a  $G$  se, e somente se:

$$\overline{K}\Sigma_{nc} \cap \mathcal{L}(G) \subseteq \overline{K}$$

Dada uma planta  $G = \langle \Sigma, X, \rightarrow_G, x^\circ, X^m \rangle$  e uma especificação  $E = \langle \Sigma, Y, \rightarrow_E, y^\circ, Y^m \rangle$  sobre  $G$ , existe um supervisor não bloqueante  $S$  para  $G$  tal que  $\mathcal{L}_m(G||S) = K$ , se  $K$  for controlável. Se  $K$  não for controlável, é possível calcular uma linguagem controlável suprema  $\sup C = (K, \mathcal{L}_m(G))$ .

Se as condições de controlabilidade da Definição 2.4 forem satisfeitas,  $\sup C$  representa o comportamento menos restritivo possível implementado por supervisão para uma planta  $G$ , sobre uma especificação  $E$ .

### 2.2.3 Síntese de supervisores

A metodologia de obtenção de um supervisor para uma planta  $G$  de especificação  $E$  se baseia em três passos (Cury, 2001a):

1. Obtenção de um modelo para a *planta*  $G$  a ser controlada;
2. Obtenção de um modelo de representação para as especificações a serem respeitadas;

3. Síntese de uma lógica de controle não bloqueante e ótima.

### **Obtenção de um modelo para a planta**

O modelo de uma planta  $G$  pode ser obtido através dos seguintes passos:

1. Identificar o conjunto de recursos ou equipamentos que compõem o sistema;
2. Modelar o autômato  $G_i$ , de cada recurso  $i$  que compõe sistema;
3. Obter o modelo da planta a ser controlada, através da composição síncrona de todos os autômatos  $G_i$ ;

### **Obtenção de um modelo para a especificação**

Os passos para obtenção da especificação global  $E$  são os seguintes:

1. Construir autômatos  $E_j$  para cada restrição de coordenação  $j$ , do sistema a ser controlado;
2. Realizar a composição dos autômatos obtidos no passo anterior, compor o autômato resultante com a planta  $G$ , gerando o autômato  $R$ ;
3. Atualizar  $R$  pela eliminação de estados considerados proibidos;
4. Atualizar  $R$  pelo cálculo de sua componente coacessível, a fim de obter um autômato não bloqueante.

### **Síntese de um supervisor não bloqueante ótimo**

O supervisor  $sup C$  é obtido por um processo iterativo que identifica “maus estados”, ou seja, estados bloqueantes no autômato  $R$ . Os passos para realizar a identificação desses estados é apresentados a seguir:

1. Identificar maus estados em  $R$ ; Caso não existam,  $S = R$ , fim;

2. Caso existam, atualizar  $R$  por eliminação dos maus estados;
3. Calcular a componente Trim de  $R$  e voltar ao primeiro passo.

Para ilustrar o processo de síntese de supervisores, o exemplo da pequena fábrica é apresentado a seguir:

**Exemplo 2.4** *A pequena fábrica Cury (2001a).*

Seja uma linha de transferência composta por duas máquinas  $M_1$  e  $M_2$  e um armazém de capacidade 1, ilustrada pela Figura 2.6. O comando  $\alpha_i$  carrega a máquina  $M_i$  que em seguida realiza uma operação sobre a mesma. Ao fim da operação sobre a peça (sinal  $\beta_i$ ), a máquina descarrega automaticamente a peça. A modelagem de cada equipamento ( $M_1$  e  $M_2$ ) e de uma especificação de capacidade para o buffer  $B$  deve ser realizada, para que não ocorra underflow ( $M_2$  inicia o processo de carga com o armazém vazio) e overflow ( $M_1$  descarrega peça no armazém já cheio). A partir dessa modelagem projeta-se um supervisor  $S$  sobre as plantas de  $M_1$ ,  $M_2$  e a especificação do buffer.

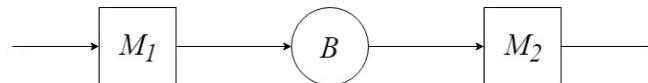


Figura 2.6: Linha de Transferência.

Os eventos controláveis são representados pelo sinal de comando que carrega uma peça nas máquinas,  $\alpha_i$  e os eventos não controláveis são representados pela finalização de uma peça por uma das máquinas,  $\beta_i$ . Sendo assim, as máquinas  $M_1$  e  $M_2$  são modeladas como autômatos, apresentados na Figura 2.7 e apresentam dois estados, o estado “0”, inicial e marcado, em que a máquina está pronta para realizar o processamento de uma peça, e o estado “1”, em que a máquina está processando uma peça.

A planta  $G$  do sistema é dada pela composição síncrona das máquinas  $M_1$  e  $M_2$ , apresentada na Figura 2.8. A notação “0,0” representa a máquina  $M_1$  e a máquina  $M_2$  no estado inicial, respectivamente. Nota-se que a ocorrência dos eventos alteram o comportamento das máquinas de forma sincronizada e assincronizada.

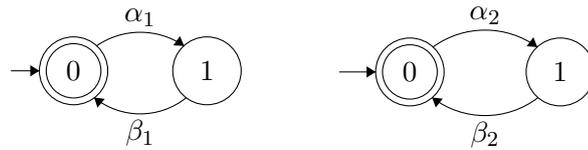


Figura 2.7: Máquinas  $M_1$  e  $M_2$ .

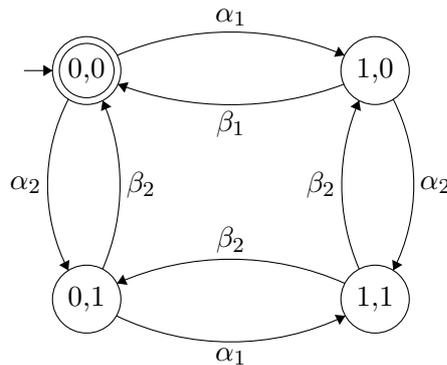


Figura 2.8: Modelo da planta  $G$ .

Analisando a Figura 2.6, após a máquina  $M_1$  processar uma peça, essa é descarregada no buffer. A máquina  $M_2$  só deve entrar em funcionamento se o buffer se encontra cheio. Dessa forma a especificação do buffer é realizada sobre os eventos  $\beta_1$  e  $\alpha_2$  das plantas  $M_1$  e  $M_2$ , pois após a ocorrência da saída de uma peça da máquina  $M_1$ , obrigatoriamente o evento  $\beta_1$  não pode ocorrer até que a máquina  $M_2$  receba uma peça. Caso contrário, ocorreria um overflow no buffer. Da mesma forma, se ocorre a entrada de uma peça na máquina  $M_2$ , obrigatoriamente o evento  $\alpha_2$  não pode ocorrer até que a máquina  $M_1$  entregue uma peça, caso contrário, ocorreria um underflow no buffer. A partir disso, a especificação é modelada de forma a alternar os eventos  $\beta_1$  e  $\alpha_2$ , como mostrado na Figura 2.9.

O autômato  $R$  é gerado através da composição síncrona entre  $G$  e  $E_1$  e é apresentado na Figura 2.10. A notação “ $0,0,0$ ” representam a máquina  $M_1$ , a máquina  $M_2$  e a especificação no estado inicial, respectivamente. A medida que os eventos ocorrem os estados das máquinas se alteram de forma sincronizada e assincronizada. A partir de

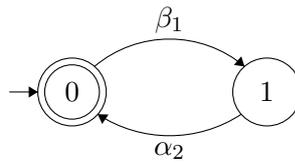


Figura 2.9: Especificação para *overflow* e *underflow*.

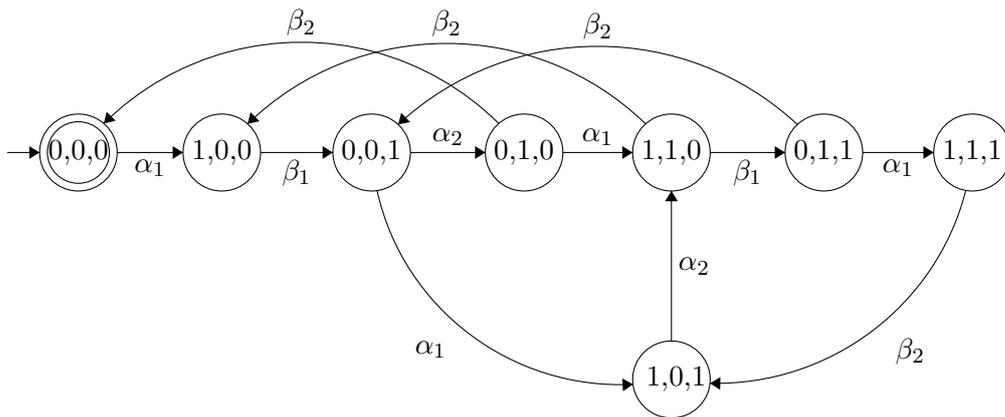


Figura 2.10: Autômato  $R$ .

$R$  calcula-se o supervisor  $S$  como na Figura 2.11.

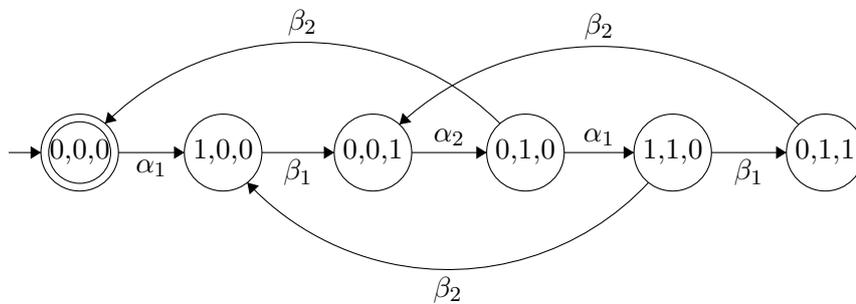


Figura 2.11: Autômato  $S$ .

Diferente do autômato  $R$ ,  $S$  realiza uma ação de controle para impedir a máquina  $M_1$  de iniciar uma operação quando o armazém estiver cheio, desabilitando o evento  $\alpha_1$  nos estados “0,0,1” e “0,1,1”. O esquemático da figura 2.12 ilustra o comportamento do supervisor.

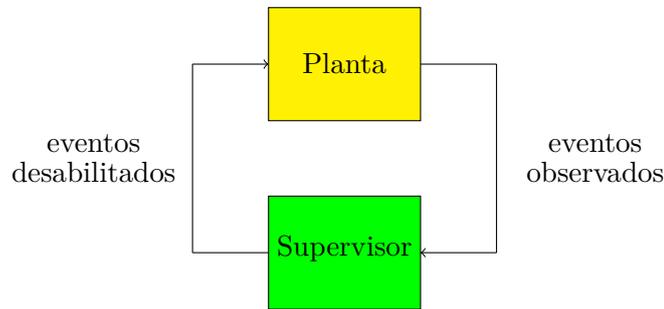


Figura 2.12: Comportamento de um Supervisor.

*Observa-se que este evento não está desabilitado no autômato  $R$ . Essa ação é necessária para impedir de forma indireta a ocorrência do evento não controlável  $\beta_1$ , que gera um estado de overflow no sistema.*

## 2.3 Controle Modular Local

Em [Ramadge e Wonham \(1989b\)](#), observa-se que a complexidade computacional dos algoritmos para síntese de supervisores é de ordem polinomial em relação ao número de estados dos modelos da planta e das especificações. Não obstante, o número de estados da planta e das especificações varia de forma exponencial com o número de seus componentes. Desta forma, revela-se um problema fundamental desse método: a explosão combinatória dos estados dos modelos em função do número de componentes.

A explosão combinatória dos estados também ocasiona um crescimento exponencial, em relação ao número de componentes da planta e das especificações, do número de estados do supervisor sintetizado. Ao tentar implementar um supervisor resultante em algum sistema de controle discreto, por exemplo, um computador de processos, um sistema de controle, supervisão e aquisição de dados (SCADA), um controlador lógico programável (CLP), ou um microcontrolador, quando há restrições de tempo de execução ou de espaço de memória, esse processo pode tornar-se problemático. A complexidade dos supervisores acarreta a dificuldade de uma maior aceitação do método

proposto por Ramadge e Wonham na engenharia de sistemas de controle discreto (Cury, 2001b). Pode-se adicionar que os algoritmos existentes para a redução do número de estados de um supervisor são de complexidade exponencial em relação ao número de estados do mesmo (Vaz e Wonham, 1986; Sivoletta et al., 2006).

De acordo com Wonham (1991), uma alternativa para aliviar o problema da explosão combinatória dos estados é por meio da decomposição da arquitetura do sistema (Wong e Wonham, 1996). Dentre elas, no sentido de uma decomposição horizontal da arquitetura têm-se o controle supervisorio *modular e descentralizada* (Ramadge e Wonham, 1987b; Wonham e Ramadge, 1988; Lin e Wonham, 1988; Rudie e Wonham, 1992; Wong e Wonham, 1998; de Queiroz e Cury, 2000). No sentido de uma decomposição vertical têm-se o controle supervisorio *hierárquico* (Zhong e Wonham, 1990; Wong e Wonham, 1996; Hubbard e Caines, 2002; Torrico e Cury, 2004; Leduc et al., 2005; Ma e Wonham, 2006; Schmidt et al., 2006; Hill e Tilbury, 2006; Cunha e Cury, 2007; Feng e Wonham, 2008). No trabalho de Eyzell e Cury (2001), aborda-se a exploração da simetria da arquitetura aplicada à redução da complexidade.

O controle modular local permite explorar a estrutura naturalmente descentralizada de sistemas de manufatura automatizados, reduzindo o esforço computacional no processo de síntese e implementação de supervisores. Os trabalhos utilizados para desenvolver esta seção foram os trabalhos de Pena (2007), Queiroz e Cury (2000), Max H. de Queiroz (2002) e Max H. de Queiroz (2014).

### 2.3.1 Representação por sistema produto

Seja a representação por sistema produto (RSP) de uma planta  $G$  formada por subplantas  $G_i = (X_i, \Sigma_i, \delta_i, \Gamma_i, x_{0i}, X_{mi})$ ,  $i \in I = \{1, \dots, n\}$ . Para  $j = l, \dots, m$ , sejam agora as especificações locais  $E_j$  para cada  $G_i$ , definidas em subconjuntos de eventos  $\Sigma_j \subseteq \Sigma$ . A planta local  $G_{loc,j}$  associada à especificação  $E_j$  é definida por  $G_{loc,j} = \parallel_{i \in I_{loc,j}} G_i$ , com  $I_{loc,j} = \{k \in I \mid \Sigma_k \cap \Sigma_j \neq \emptyset\}$

Assim, a planta local  $G_{loc,j}$  é composta pelas subplantas da planta original  $G$ , que estão diretamente ou indiretamente restringidas por  $E_j$ . A planta enxuta  $G_e = \|\|_{j=1}^m G_{loc,j}$  é composta apenas pelos subsistemas relevantes (Max H. de Queiroz, 2002).

### 2.3.2 Controle modular

Sejam as linguagens  $L_i \subseteq \Sigma^*$ ,  $i = 1, \dots, n$ . O conjunto de linguagens dos subsistemas  $L_i$  é modular se  $\cap_{i=1}^n \overline{L_i} = \overline{\cap_{i=1}^n L_i}$ , ou seja, sempre que um prefixo for aceito por todo o conjunto de linguagens, todo o conjunto deve aceitar uma palavra contendo esse prefixo. Em outras palavras, a linguagem não gera conflito (Max H. de Queiroz, 2002).

### 2.3.3 Controle Modular Local

Sejam as linguagens  $L_i \subseteq \Sigma^*$ ,  $i = 1, \dots, n$ . O conjunto de linguagens dos subsistemas  $L_i$  é localmente modular se  $\|\|_{i=1}^n \overline{L_i} = \overline{\|\|_{i=1}^n L_i}$ . Se para  $i = 1, \dots, n$ , as linguagens  $L_i$  forem marcadas por plantas  $G_i$ , a modularidade local é verificada se, e somente se  $G = \|\|_{i=1}^n G_i$ . Portanto, a síntese da máxima linguagem controlável de múltiplas especificações pode ser executada diretamente a partir das especificações locais, desde que a modularidade local seja válida.

Sendo assim, o seguinte teorema fundamenta a abordagem de controle modular local proposta por Queiroz e Cury (2000).

**Teorema 2.1** (Queiroz e Cury (2000)) *Dado um sistema com RSP formada por  $G_j$ ,  $j = 1, \dots, m$ , e as especificações  $E_{gen,i}$ ,  $i = 1, \dots, n$ . Sejam  $E_{loc,i}$ ,  $K_{glo,i}$ ,  $G_{loc,i}$  e  $G$  definidos. Se  $Sup C(E_{loc,i}, K_{glo,i}, G_{loc,i})$ ,  $i = 1, \dots, n$  for localmente modular, então,  $Sup C(\cap_{i=1}^n K_{glo,i}, G) = \|\|_{i=1}^n Sup C(E_{loc,i}, G_{loc,i})$ .*

Sejam  $n$  especificações locais sobre um sistema, essas podem ser expressas em termos de subsistemas  $G_{loc,i}$ ,  $i = 1, \dots, n$  restringidas por elas. Portanto, o cálculo dos

supervisores máximos locais ( $Sup C(E_{loc,i}, G_{loc,i}), i = 1, \dots, n$ ) é realizado sobre essas restrições e plantas.

**Exemplo 2.5** (*Célula de Manufatura (Curzel e Leal, 2008)*)

A célula de manufatura apresentada na Figura 2.13 é composta de dois robôs, uma mesa giratória, uma estação de teste, uma esteira e um depósito (buffer) para peças rejeitadas.

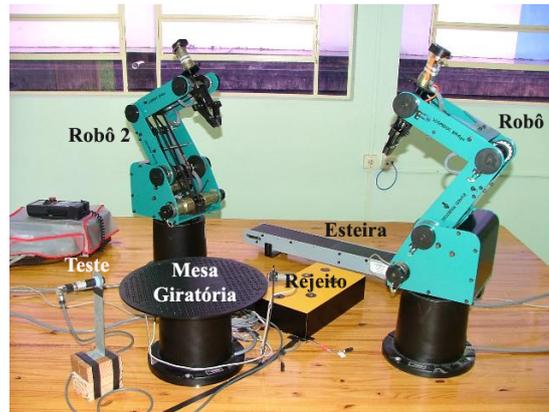


Figura 2.13: Célula de Manufatura Didática. Adaptado de Curzel e Leal (2008)

O funcionamento da célula prevê que as peças posicionadas ao final da esteira de entrada são transportadas pelo robô 1 até a mesa giratória. As peças são manufaturadas e testadas, e se apresentarem algum problema, o robô 2 as transporta para o depósito de rejeitos, caso contrário, as retira da célula.

Tabela 2.2: Tabela de função dos sensores. Adaptado de Curzel e Leal (2008).

Sensor	Acionamento
S0	Presença no fim da esteira
S1	Presença na mesa giratória
S2	Mesa em posição para recebimento
S3	Mesa em posição para teste

O teste de qualidade das peças é realizado por um sensor fotoelétrico, que detecta a

Tabela 2.3: Tabela de função dos atuadores. Adaptado de [Curzel e Leal \(2008\)](#)

Atuador	Elemento Movimentado
A0	Esteira
A1	Robô 1
A2	Mesa giratória
A3	Sensor fotoelétrico
A4	Robô 2

diferença de coloração das peças (pretas ou brancas). Os sinais dos sensores e atuadores dos equipamentos da célula são apresentados nas Tabelas 2.2 e 2.3 e são interconectados a um módulo de entradas e saídas digitais de um CLP, responsável pelo controle da sequência de operações de cada dispositivo. A esteira tem capacidade de carga infinita e os robôs e a esteira tem capacidade de uma peça.

### 1. Modelagem dos Dispositivos da planta

O funcionamento da esteira é modelado pelo autômato da Figura 2.14, em que o estado inicial “0” representa a esteira desligada e o estado “1” representa a esteira ligada. Os eventos  $E_{liga}$  e  $E_{desliga}$  são eventos controláveis uma vez que é possível ligar e desligar a esteira através do atuador A0.

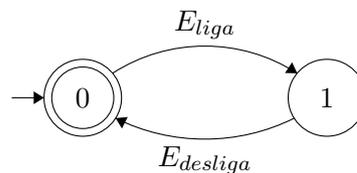


Figura 2.14: Autômato para a esteira.

O funcionamento do robô 1 é modelado pelo autômato da Figura 2.15, “0” é o estado inicial, marcado e indica que o robô está pronto para receber uma peça e “1” indica que o robô retirou a peça da esteira e a transportou para a mesa giratória. O evento  $T_M$  representa o transporte de uma peça da esteira à mesa giratória e é controlável, uma vez que é possível ligar e desligar o robô 1 pelo atuador A1 e

$F_{rb1}$  indica que a máquina está pronta para realizar outra tarefa e é um evento não controlável, uma vez que é a resposta da máquina para a ativação do atuador A1.

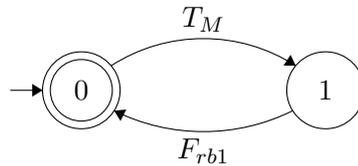


Figura 2.15: Autômato para o robô 1

O funcionamento da mesa giratória e do sensor podem ser modelados juntos, como é apresentado na Figura 2.16. O estado inicial “0” é marcado e indica que a mesa está pronta para receber uma peça. O estado “1” indica que há uma peça sobre a mesa, o estado “2” indica que a peça está pronta para teste, o estado “3” indica que o teste foi iniciado e o estado “4” indica que o teste foi finalizado e a peça retirada da mesa.

O evento  $s_{mesa}$  representa a ativação do sensor de presença S1 e é controlável, uma vez que a mesa pode ser desligada pelo atuador A2.  $I_{giro}$  representa o posicionamento da peça para realização do teste e não é um evento controlável, pois é uma resposta do sistema para a ativação do sensor S1. O início do teste de qualidade sobre a peça é representado pelo evento  $I_{giro}$  e é controlável, uma vez que o atuador A3 é responsável por ativar e desativar o sensor fotoelétrico. Os eventos  $T_S$  e  $T_R$  são referentes a aceitação ou rejeição de uma peça, seguida da sua retirada da mesa, e são controláveis, devido aos atuadores A3 e A4. O último evento  $F_{giro}$  representa o giro da mesa para o recebimento de uma nova peça e não é controlável, uma vez que é a resposta do sistema à desativação do sensor S1 enquanto o sensor S3 se encontra ativado.

O comportamento do robô 2 é representado pelo autômato da Figura 2.17, o estado

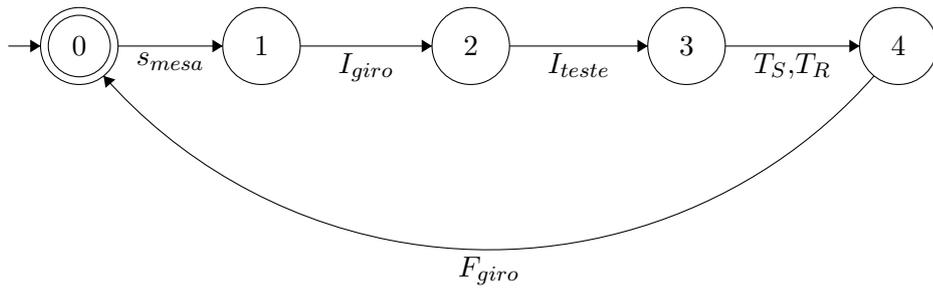


Figura 2.16: Autômato para a mesa giratória e o sensor de teste.

“0”, inicial e marcado, indica que o robô está pronto para a retirada de uma peça da mesa giratória e o estado “1” indica que uma peça foi retirada da mesa e transportada adequadamente dependendo do resultado do teste de qualidade. Os eventos  $T_S$  e  $T_R$  indicam que o robô 2 retirou a peça da mesa, a transportando para fora da célula ou para a mesa de rejeitos, respectivamente e  $F_{rb2}$  representa o retorno do robô à posição de retirada de peças e é um evento não controlável, uma vez que é a resposta da máquina para a ativação do atuador A4.

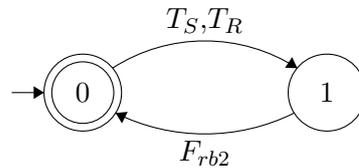


Figura 2.17: Autômato para o robô 2.

2. Modelagem das especificações dos dispositivos

A modelagem sobre as especificações foi realizada em cima dos eventos de entrada e saída de cada dispositivo, resultando em três especificações, uma especificação sobre a esteira e o robô 1, uma especificação sobre o robô 1 e a mesa giratória e uma especificação sobre a mesa giratória e o robô 2.

A especificação de funcionamento sobre a esteira e o robô 1 apresentada na Figura 2.18 é modelada sobre o sensor de presença na saída da esteira. O estado “0”,

inicial e marcado, é referente ao sensor desativado e o estado “1” refere-se ao sensor ativado. O auto-laço  $E_{liga}$  permite que a esteira seja ligada apenas se o sensor de presença estiver desativado. O evento  $E_{desliga}$  é a ativação do sensor de presença e o evento  $T_M$  é a retirada da peça ao final da esteira.

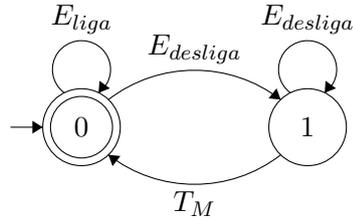


Figura 2.18: Especificação de funcionamento sobre o robô 1 e a esteira.

A especificação de funcionamento sobre o robô 1 e a mesa giratória apresentada na Figura 2.19 é modelada sobre o sensor  $S_1$  da mesa giratória. O estado “0”, inicial e marcado, refere-se a ausência de peças sobre a mesa e o estado “1” refere-se a presença de peças. O auto-laço  $S_{mesa}$  permite que o sensor de presença só ative após o robô 1 entrar em funcionamento, indicado pelo evento  $T_M$ . O robô 1 só poderá entregar outra peça ao sistema após a ocorrência do evento  $F_{giro}$ .

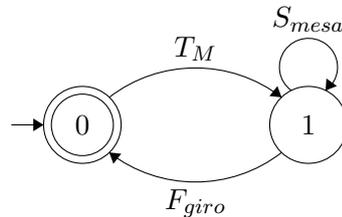


Figura 2.19: Especificação de funcionamento sobre o robô 1 e a mesa giratória.

A especificação de funcionamento sobre a mesa giratória e o robô 2 apresentada na Figura 2.20 é modelada sobre o teste das peças. O estado inicial “0” é marcado e indica que nenhuma peça foi testada e o estado “1” indica que a peça foi testada e retirada da mesa adequadamente pelo robô 2. Os eventos  $T_S$ ,  $T_R$  indicam que o robô 2 retirou a peça da mesa e o evento  $F_{rb2}$  só permite que uma peça seja

testada após o robô 2 estar pronto para a retirada de peças.

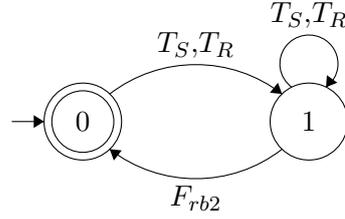


Figura 2.20: Especificação de funcionamento sobre a mesa giratória e o robô 2.

### 3. Síntese dos supervisores

A síntese dos supervisores sobre as especificações de funcionamento é realizada em três passos, primeiro a RSP é calculada para determinar quais plantas  $G_{ilocais}$  estão associadas a quais especificações  $E_{ilocais}$ , em seguida aplica-se composição síncrona entre cada especificação com suas respectivas plantas associadas, para a obtenção do autômato  $R_{local}$  e por fim os supervisores são obtidos pelo cálculo da máxima linguagem controlável  $sup C(R_{local})$ . A Tabela 2.4 apresenta a associação entre especificações e plantas obtida pela RSP.

Tabela 2.4: Tabela de função dos sensores dos supervisores do Exemplo 2.5.

Especificação	Plantas associadas
$E_{esteira,rob\hat{o}1}$	$G_{esteira}    G_{rob\hat{o}1}$
$E_{rob\hat{o}1,mesa}$	$G_{rob\hat{o}1}    G_{mesa}$
$E_{mesa,rob\hat{o}2}$	$G_{mesa}    G_{rob\hat{o}2}$

A Figura 2.21 apresenta o supervisor sobre a esteira e o robô 1, cada estado indica em que estado se encontra a esteira, o robô 1 e a especificação, respectivamente, assim como o comportamento desejado das plantas.

A Figura 2.22 apresenta o supervisor sobre o robô 1 e a mesa giratória, cada estado indica em que estado se encontra o robô 1, a mesa giratória e a especificação, respectivamente.

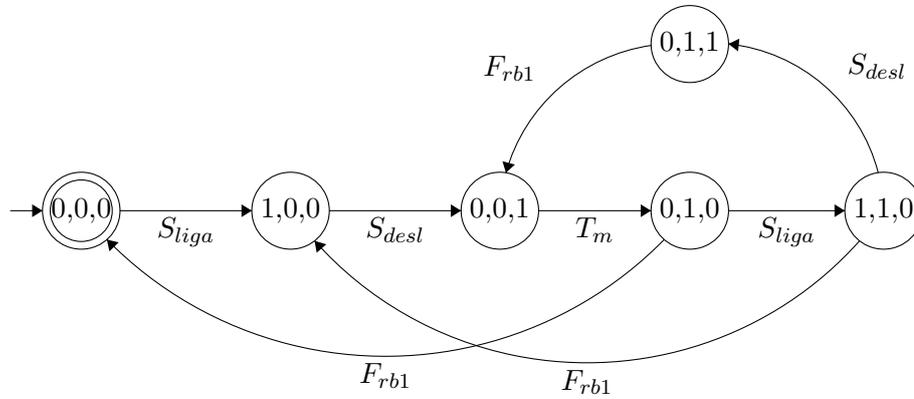


Figura 2.21: Supervisor sobre a esteira e o robô 1.

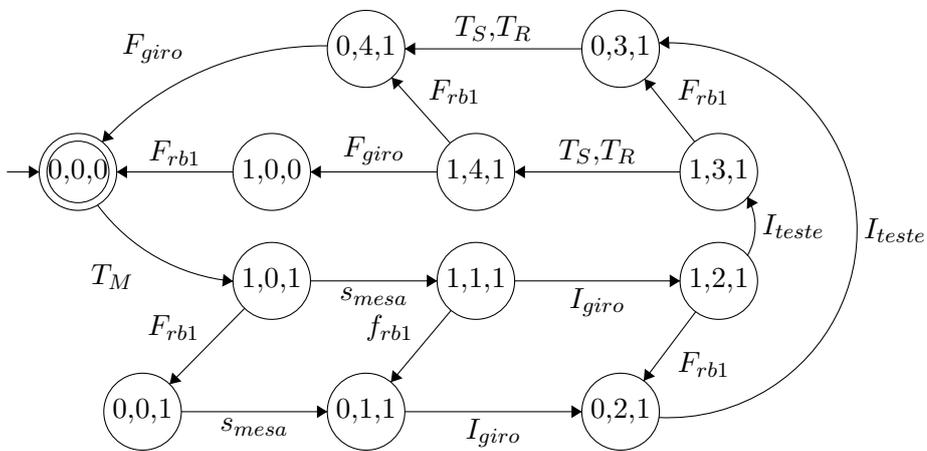


Figura 2.22: Supervisor sobre o robô 1 e a mesa giratória.

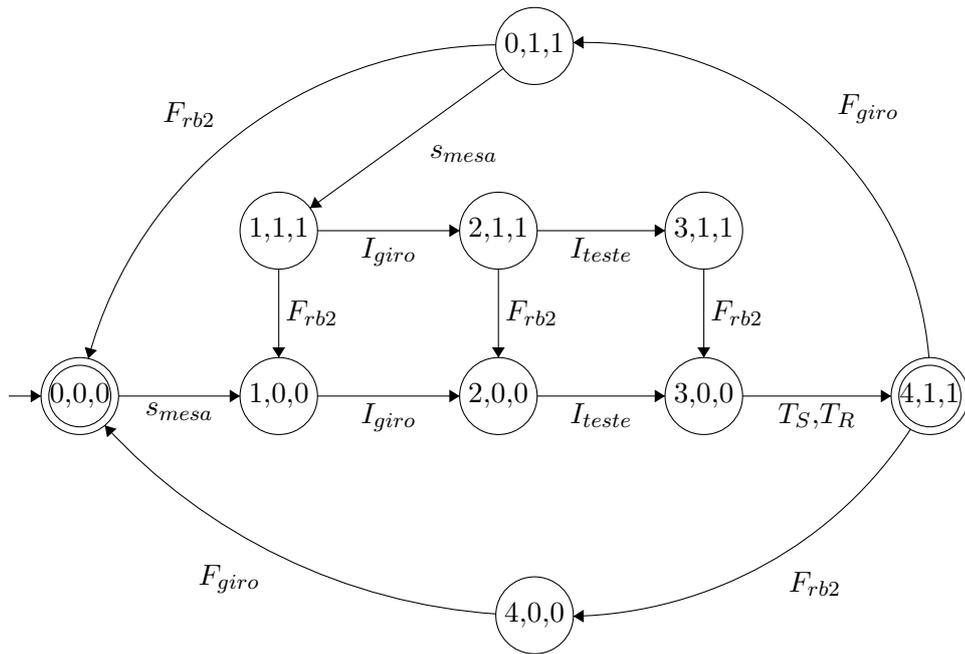


Figura 2.23: Supervisor sobre a mesa giratória e o robô 2.

A Figura 2.23 apresenta o supervisor sobre a mesa giratória e o robô 2, cada estado indica em que estado se encontra a mesa giratória, o robô 2 e a especificação, respectivamente.

## Capítulo 3

# Modelagem do Sistema de Triagem

Neste capítulo apresenta-se a modelagem do sistema de triagem presente em ([de Aguiar et al., 2012](#)). O objetivo deste capítulo é detalhar a modelagem realizada no artigo de [de Aguiar et al. \(2012\)](#) e discutir os resultados obtidos. A organização do capítulo é como se segue. Na Seção 3.1 o sistema é apresentado. Na Seção 3.2 apresenta-se a modelagem da planta, na Seção 3.3 as especificações são apresentadas. Na seção 3.4 apresenta-se os supervisores sintetizados e a seção 3.5 apresenta uma análise sobre os resultados apresentados.

Todos os autômatos para as plantas, especificações e supervisores do sistema apresentados neste capítulo foram modelados utilizando o *visual studio 2017* e a biblioteca *UltraDES* desenvolvida pelo Laboratório de Análise e Controle de Sistemas a Eventos Discretos (LACSED).

### 3.1 O sistema de triagem

O sistema a ser modelado é apresentado na Figura 3.1 e é composto de cinco esteiras, uma mesa rotatória, dez sensores (Tabela 3.1) e dois elevadores. As esteiras e a mesa rotatória se movimentam após a ativação de seus respectivos atuadores (Tabela 3.2).

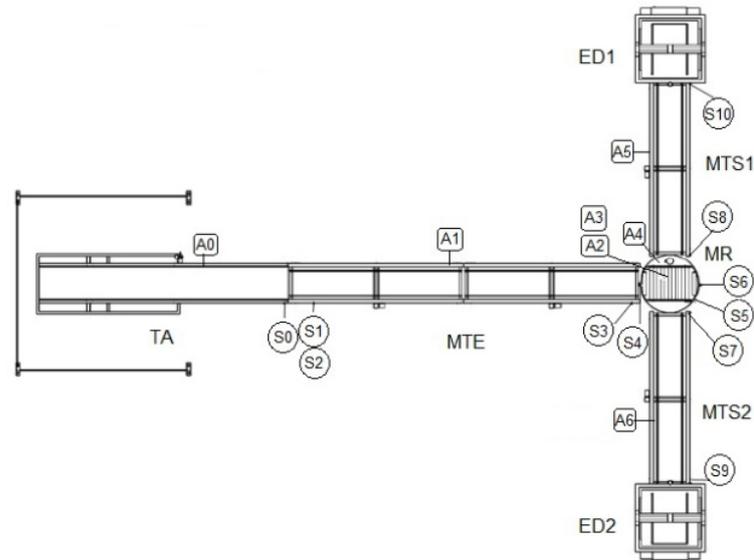


Figura 3.1: Planta do Tapete Alimentador, Adaptado de [de Aguiar et al. \(2012\)](#).

O sistema é alimentado aleatoriamente com caixas grandes ou pequenas pelo tapete alimentador (TA). Essas caixas são transferidas para a mesa transportadora de entrada (MTE), onde os sensores S1 e S2 identificam os seus tamanhos. Em seguida, as caixas são entregues à mesa rotatória (MR) que deve estar em posição para recebimento de caixas (sensor S4 ativado). A mesa gira no sentido anti horário, permitindo o direcionamento das caixas (sensor S5 ativado). Se a caixa for pequena, ela é designada para a mesa transportadora de saída 1 (MTS1), caso contrário, a caixa é direcionada para a mesa transportadora de saída 2 (MTS2). Por fim, as caixas são encaminhadas até os elevadores de descarga (ED1 e ED2).

O controle e modelagem do sistema são realizados sobre as limitações de capacidade de cada elemento da planta. TA possui capacidade infinita, MTE suporta transportar

Tabela 3.1: Tabela de função dos sensores. Adaptado de [de Aguiar et al. \(2012\)](#).

Sensor	Acionamento
S0	Presença no fim de TA
S1	Caixa pequena e grande
S2	Caixa grande
S3	Presença no fim de MTE
S4	MR em posição de recebimento
S5	MR em posição de entrega
S6	Presença na esteira MR
S7	Presença na entrada de MTS1
S8	Presença na entrada de MTS2
S9	Presença na saída de MTS1
S10	Presença na saída de MTS2

Tabela 3.2: Tabela de função dos atuadores. Adaptado de [de Aguiar et al. \(2012\)](#).

Atuador	Elemento Movimentado
A0	TA
A1	MTE
A2	Esteira de MR sentido MTS1 e carga de MR
A3	Esteira de MR sentido MTS2
A4	MR-posição de entrega de caixas
A5	MTS1
A6	MTS2

um total de oito caixas simultaneamente, MTS1 e MTS2 comportam quatro caixas e ED1, ED2 e MR possuem capacidade de uma caixa (de Aguiar et al., 2012). Devido a complexidade do sistema, todas as análises foram realizadas com abstrações para as plantas e especificações para carga máxima de duas caixas.

## 3.2 Planta

Nesta etapa é realizada a modelagem de cada elemento do sistema de triagem, sendo atribuída a controlabilidade dos eventos. Para simplificação de notação, será utilizado o sufixo  $x$  no nome dos eventos que se refiram genericamente à manipulação de caixas pequenas ou grandes.

### 3.2.1 Tapete Alimentador

A ilustração do tapete alimentador é apresentado na Figura 3.2. A modelagem da planta TA é apresentada na Figura 3.3.

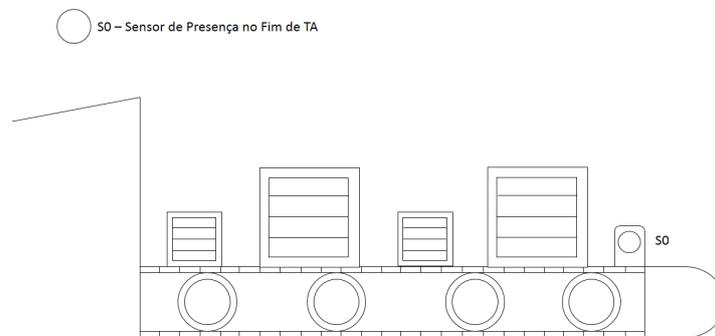


Figura 3.2: Ilustração do Tapete Alimentador.

Esse sistema possui dois estados, “0” indicando que não há caixas sobre o tapete e “1” indicando que há uma caixa. O evento  $eta$  representa a entrada de uma caixa no sistema e o evento  $sta$  representa a ativação do sensor S0. Ambos os eventos são

controláveis, uma vez que o tapete pode ser desligado pelo atuador A0, impedindo a ocorrência de ambos os eventos.

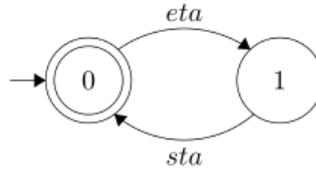


Figura 3.3: Planta do Tapete Alimentador. Adaptado de [de Aguiar et al. \(2012\)](#).

### 3.2.2 Mesa Transportadora de Entrada

A ilustração da mesa transportadora de entrada é apresentada na Figura 3.4. A planta MTE possui quatro eventos, dois eventos de entrada, a entrada de uma caixa pequena, *emte1*, a entrada de uma caixa grande, *emte2*, a saída de uma caixa pequena, *smtte1* e a saída de uma caixa grande, *smtte2*.

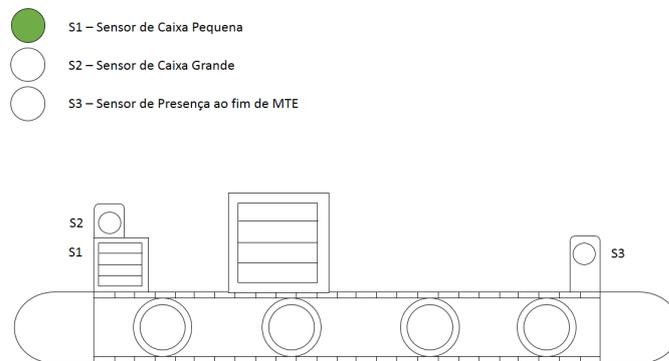


Figura 3.4: Ilustração da Mesa Transportadora de Entrada.

Todos os eventos são controláveis, pois a esteira pode ser desligada pelo atuador A1. Os estados do sistema representam a ordem de caixas presentes na esteira, o estado inicial é o estado sem caixas e o estado marcado é o próprio estado inicial, pois é o estado final desejado para o sistema.

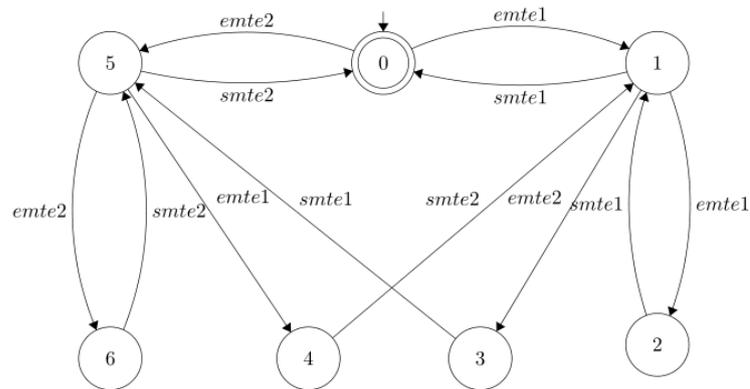


Figura 3.5: Planta da Mesa Transportadora de Entrada. Adaptado de [de Aguiar et al. \(2012\)](#).

A MTE é um sistema do tipo FIFO, sendo assim, a primeira caixa a entrar é a primeira a sair, o que gera uma explosão de estados. Por causa do tamanho e complexidade da planta MTE para oito caixas, a Figura 3.5 apresenta a planta MTE para duas caixas. A planta completa possui um total de 511 estados e 1020 transições.

### 3.2.3 Mesa Rotatória

A ilustração da mesa rotatória é apresentada na Figura 3.6.

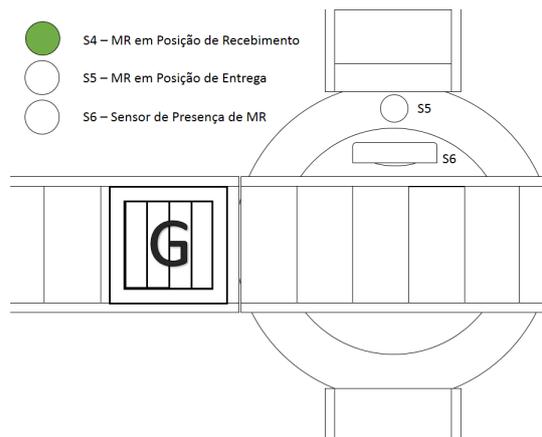


Figura 3.6: Ilustração da Mesa Rotatória.

O autômato que modela seu comportamento é apresentado na Figura 3.7. O primeiro evento a ocorrer a partir do estado inicial é o evento de entrada de uma caixa *emr* (sensor S4 ativado e ativação do sensor S6). Com isso, ocorre o giro da mesa para a posição de entrega de caixas (ativação do sensor S5), representado pelo evento *giro1*, então a esteira de MR direciona a caixa para MTS1 (evento *smr1*) se for pequena, ou MTS2 (evento *smr2*) se for grande. Por fim o evento *giro2* ocorre, retornando a planta para a posição de recebimento de caixas. Os eventos *emr* e *smr<sub>x</sub>* são controláveis, já que a esteira MR pode ser desligada através dos atuadores A2 e A3 e os eventos *giro1* e *giro2* não são controláveis, pois são uma resposta do sistema à ativação e desativação do sensor S6, respectivamente.

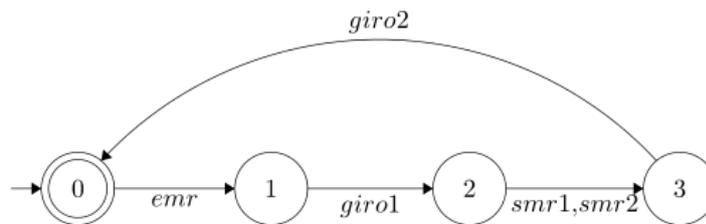


Figura 3.7: Planta da Mesa Rotatória. Adaptado de [de Aguiar et al. \(2012\)](#).

### 3.2.4 Mesas Transportadoras de Saída

As plantas de MTS1 e MTS2 são idênticas e são apresentadas na Figura 3.8.

Os eventos são os de entrada de caixas nos sistemas *emts<sub>x</sub>*, identificados pelos sensores S7 e S8, e os de saída de caixas dos sistemas *smts<sub>x</sub>*, identificados pelos sensores S9 e S10. Esses eventos são controláveis, uma vez que as mesas podem ser desligadas pelos atuadores A5 e A6. Os estados representam o número de caixas em cada esteira e o estado inicial é marcado, indicando que não há caixas sobre as esteiras. A Figura 3.9 apresenta o autômato de MTS1 e MTS2.

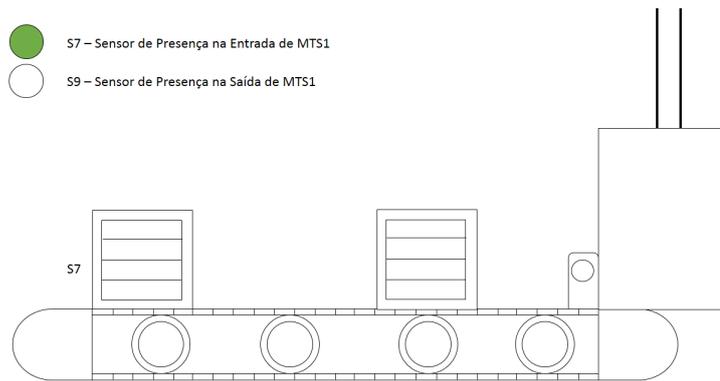


Figura 3.8: Ilustração das Mesas Transportadoras de Saída.

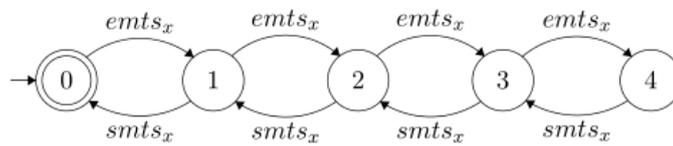


Figura 3.9: Planta das Mesas Transportadoras de Saída. Adaptado de [de Aguiar et al. \(2012\)](#).

### 3.2.5 Elevadores

As plantas dos elevadores ED1 e ED2 são idênticas, e apresentam um estado em que ocorre o evento não controlável  $\tau_{au1}$ , representando a descida do elevador após o descarregamento de uma caixa. A modelagem desses elevadores é apresentada na Figura 3.10.

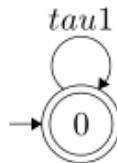


Figura 3.10: Planta dos Elevadores. Adaptado de [de Aguiar et al. \(2012\)](#).

O funcionamento dos elevadores possuem uma restrição de operação com as respectivas mesas transportadoras de saída ( $ROpl$ ), cuja a saída de uma caixa da  $MTS_x$  só pode

ocorrer após  $ED_x$  estiver em posição para recebimento. Essa restrição é apresentada na Figura 3.11. Os estados do elevador são ocupado (estado “1”), devido ao carregamento e transporte de uma caixa para fora do sistema, e ocioso (estado “0”, inicial e marcado), indicando que o elevador está pronto para receber outra caixa. Os eventos associados a restrição são o de carregamento dos elevadores,  $smts_x$ , e o movimento de descida dos elevadores,  $tau_1$ . O evento  $smts_x$  é controlável.

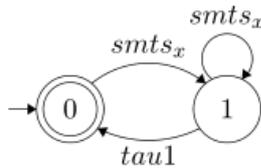


Figura 3.11: Restrição de funcionamento  $ROpl$ . Adaptado de [de Aguiar et al. \(2012\)](#).

### 3.3 Especificações

Esta seção apresenta a modelagem das especificações de cada planta do sistema de triagem, restringidas à carga máxima de cada recurso e a uma especificação de roteamento para definição adequada do destino de cada caixa.

#### 3.3.1 Especificação de carga do Tapete Alimentador

O tapete alimentador não necessita especificação, pois o recebimento de caixas é automático e o único processo interessante a ser observado é o de saída de caixas para a mesa transportadora de entrada.

#### 3.3.2 Especificação de carga da Mesa Transportadora de Entrada

A planta MTE suporta até oito caixas, sendo assim, toda vez que o evento  $sta$  ocorre, MTE é carregada com uma caixa (evento  $emte_x$ ). Esse evento pode ocorrer até oito

vezes, sem que ocorra a saída de uma caixa de MTE, evento  $smtex$ . A Figura 3.12 ilustra a especificação de carga de MTE.

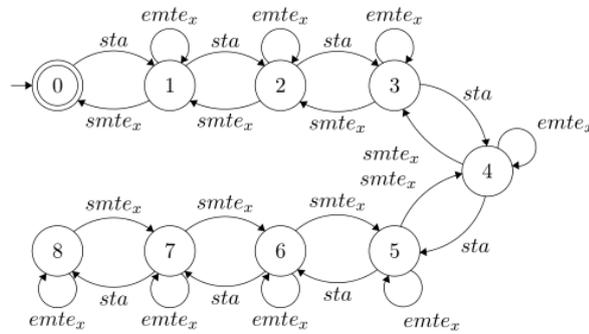


Figura 3.12: Especificação de carga da Mesa Transportadora de Entrada. Adaptado de de Aguiar et al. (2012).

### 3.3.3 Especificação de funcionamento da Mesa Rotatória

A planta de MR apresenta capacidade de uma caixa, logo, uma vez que ocorre a saída de uma caixa da esteira MTE, pelo evento  $smtex$ , carregando MR, esse evento só poderá ocorrer novamente após a ocorrência do evento  $giro2$ , indicando que a a planta está pronta para receber uma outra caixa. A Figura 3.13 apresenta a especificação de carga de MR.

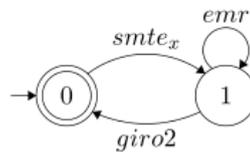


Figura 3.13: Especificação de funcionamento da Mesa Rotatória. Adaptado de de Aguiar et al. (2012).

### 3.3.4 Especificação de Carga das Mesas Transportadoras de Saída

A especificação de carga de MTS2 é idêntica a de MTS1, e são ilustradas na Figura 3.14. A especificação das mesas se assemelham muito à especificação de carga de MTE, sendo o carregamento da esteira com uma caixa dado pelo evento  $smr_x$  e a saída dada pelo evento  $smts_x$ . Como as esteiras possuem carga máxima de quatro caixas, essa especificação apresenta menos estados.

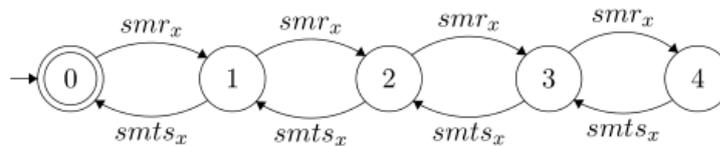


Figura 3.14: Especificação de carga das Mesas Transportadoras de Saída. Adaptado de de Aguiar et al. (2012).

### 3.3.5 Especificação de Carga dos Elevadores

A especificação de carga dos elevadores, apresentada na Figura 3.15, restringe a saída de caixas das mesas transportadoras de saída. Uma vez que uma caixa deixa uma das mesas (evento  $smts_x$ ), outra caixa só poderá sair do sistema se os elevadores se encontram em posição para recebimento de caixas (evento  $tau1$ ).

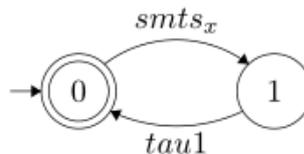


Figura 3.15: Especificação de carga dos Elevadores. Adaptado de de Aguiar et al. (2012).

### 3.3.6 Especificação de Roteamento

A especificação de roteamento, apresentada na Figura 3.16, é responsável por determinar para qual MTS a caixa em MR deve ser enviada. A partir dos eventos  $smtex$ , a esteira

MR identifica se a caixa é pequena ou grande. Dessa forma após a ocorrência dos eventos *smt**e*1 ou *smt**e*2, esses eventos só poderão ocorrer após *smr*1 ou *smr*2, respectivamente.

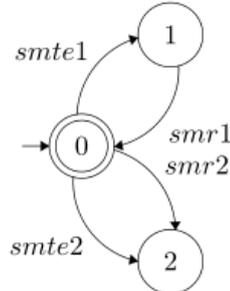


Figura 3.16: Especificação de Roteamento. Adaptado de [de Aguiar et al. \(2012\)](#).

### 3.4 Síntese dos supervisores

A síntese dos supervisores sobre as especificações de carga depende da associação entre as plantas e cada uma das especificações. Todas as plantas, especificações e supervisores foram modelados utilizando o software *Visual Studio* e a biblioteca *UltraDES*. No Apêndice A encontra-se uma breve explicação dos comandos utilizados na parte computacional para a obtenção dos resultados apresentados neste capítulo e nos próximos. A partir do cálculo da RSP, que consiste em associar uma especificação às plantas de cada equipamento por meio dos eventos em comum entre elas, obteve-se essa associação, que é apresentada na Tabela 3.3.

Para cada especificação e respectivas plantas da Tabela 3.3, realizou-se a sincronização entre as especificações locais  $E_{local}$  e as respectivas plantas locais associadas  $G_{local}$ , para obtenção do sistema sob supervisão  $R_{local}$ . A partir de  $R_{local}$  extraiu-se os supervisores pela máxima linguagem controlável  $\mathcal{L}(S||G)$ , ou  $sup C(R_{local})$ . O esquemático da Figura 3.17 ilustra esse processo.

O número de especificações delimita o número de supervisores, resultando em um total de 7. O número de estados de cada um dos autômatos que compõem os supervisores

Tabela 3.3: Especificações e Plantas Locais Associadas. Adaptado de [de Aguiar et al. \(2012\)](#).

Especificação	Planta Local
Carga de MTE	$G_{TA}  G_{MTE}$
Carga de MR	$G_{MTE}  G_{MR}$
Carga de MTSx	$G_{MR}  G_{MTSx}  G_{EDx}  G_{ROpl}$
Carga de EDx	$G_{MTSx}  G_{EDx}  G_{ROpl}$
Roteamento	$G_{MTE}  G_{MR}$

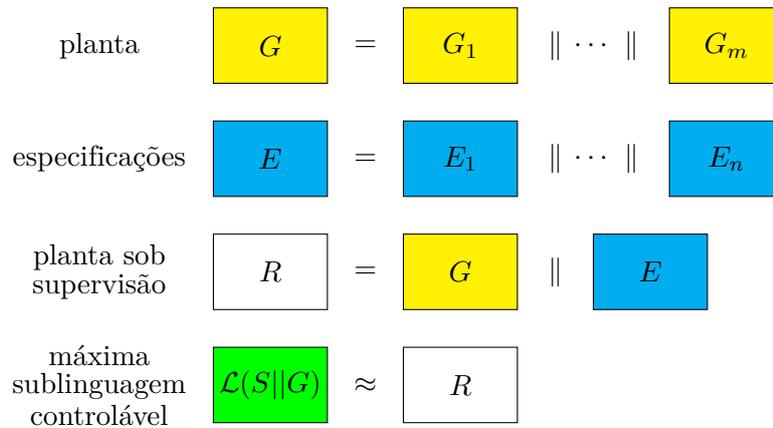


Figura 3.17: Esquemático para Síntese dos Supervisores.

locais  $Z_{local}$  são apresentados na tabela 3.4.

Conforme realiza-se a inspeção da Tabela 3.4, observa-se que a linguagem alvo é controlável em relação a planta, dessa forma as especificações podem ser utilizadas como supervisores reduzidos para o sistema.

A síntese dos supervisores se torna satisfatória após a análise de conflito. Para isso, obtém-se o supervisor monolítico e o compara com a sincronização de todos os supervisores modulares. O teste para OP-abstrações ([Pena, 2007](#)) demonstrou a ausência de bloqueio, validando a abordagem modular local implementada.

A complexidade computacional do algoritmo de obtenção dos supervisores é polinomial em  $mn$ , sendo  $m$  o número de estados da planta e  $n$  o número de estados da especi-

Tabela 3.4: Resultados para síntese de Supervisores Modulares. Adaptado de [de Aguiar et al. \(2012\)](#).

Supervisores	Glocal	Elocal	Rlocal	Zlocal
Carga de MTE	1022	9	8686	8686
Carga de MR	2044	2	2555	2555
Carga de MTSx	40	5	192	192
Carga de EDx	10	2	10	10
Roteamento	2044	3	6132	6132

cação ([Su e Wonham, 2004](#)). A complexidade da síntese monolítica é  $O(408800^2 * 5400^2)$ . Para a síntese modular local há 7 supervisores, cuja complexidade é  $O(1022^2 * 9^2)$ . A complexidade envolvida na análise de conflito é a mesma no cálculo da composição síncrona entre os supervisores modulares locais.

### 3.5 Discussão

Neste capítulo apresentou-se a modelagem dos supervisores para um sistema de triagem, proposto por [de Aguiar et al. \(2012\)](#). No decorrer deste capítulo observou-se que a aplicação da síntese modular local apresenta complexidade inferior comparada com a síntese monolítica. Destes dados conclui-se que com 7 supervisores da Tabela 3.4, com especificações de 2 e 3 estados é possível implementar controle sobre o sistema, se comparado aos 648 estados da especificação do supervisor monolítico. O resultado da implementação do controle modular local resultou em considerável economia de memória e processamento na execução do controle supervisório.

A modelagem do sistema de triagem apresentada possui especificações apenas sobre carga e direcionamento de caixas. Pela simulação e análise dos supervisores observou-se erros de modelagem que resultam no mal funcionamento do sistema. Este mal funcionamento está relacionado aos sensores de identificação de caixas S1 e S2 e aos sensores de identificação de presença S7 e S8. A ativação desses sensores ocorre sem a

saída de caixas do tapete alimentador e da mesa rotatória. Dessa forma os supervisores sobre as especificações dessas plantas não descrevem o comportamento correto das plantas locais. Esse comportamento anormal é apresentado na figura 3.18.

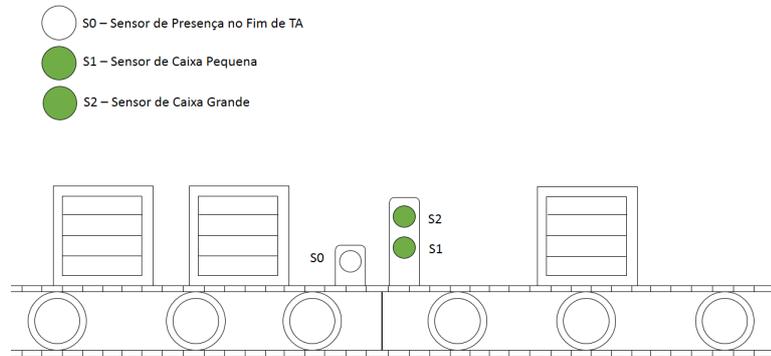


Figura 3.18: Ilustração da Ativação *Fanstasma* dos Sensores.

A modelagem apresentada simplifica os autômatos das plantas, em contrapartida essa simplificação aumenta o número de especificações, e o número de supervisores. Com o objetivo de solucionar esses problemas, o próximo capítulo apresenta a correção do mal funcionamento dos supervisores locais e a otimização dos supervisores. Por meio da remodelagem das plantas e especificações, corrige-se os problemas de mal funcionamento e otimiza-se a complexidade sobre o conjunto de supervisores.



## Capítulo 4

# Otimização do Sistema de Triagem

No capítulo 3 mostrou-se a modelagem sobre o sistema de triagem proposto por [de Aguiar et al. \(2012\)](#), e a síntese modular local de seus supervisores. Entretanto, a modelagem realizada apresenta erros e por essa razão, os supervisores sintetizados não representam o comportamento desejado do sistema. Esses erros serão discutidos posteriormente neste capítulo.

Este capítulo propõe a correção dos supervisores pela adição de restrições sobre o funcionamento de cada  $G_{i\text{local}}$  associada e otimização desses supervisores por meio da remodelagem das plantas e especificações.

### 4.1 Modelagem da Planta

Esta seção apresenta a modelagem otimizada das plantas do sistema de triagem proposto no trabalho de [de Aguiar et al. \(2012\)](#).

### 4.1.1 Tapete Alimentador

A modelagem da planta TA é a mesma apresentada na seção 3.2.1 e o autômato que representa seu comportamento é apresentado na figura 4.1.

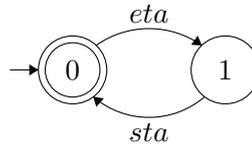


Figura 4.1: Planta de TA.

### 4.1.2 Mesa Transportadora de Entrada

A modelagem da planta MTE é a mesma apresentada na seção 3.2.2 e o autômato que representa seu comportamento é apresentado na figura 4.2.

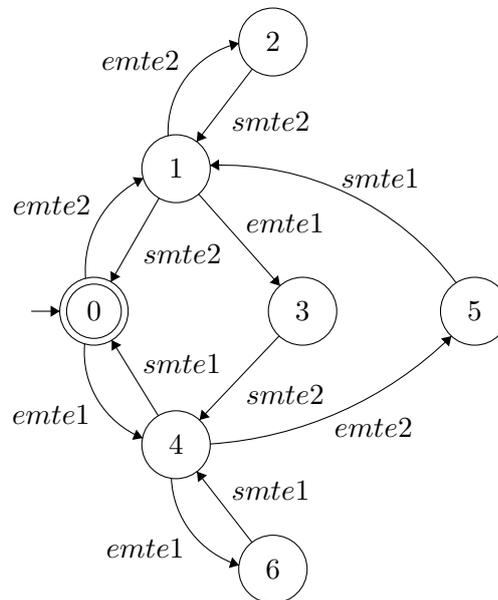


Figura 4.2: Planta de MTE.

### 4.1.3 Mesa Rotatória

A modelagem realizada sobre o autômato que descreve o comportamento de MR, figura 4.3, difere da apresentada na seção 3.2.3 pelos estados “3” e “4”, que representam o direcionamento da caixa para a MTS1 e o direcionamento da caixa para a MTS2, respectivamente, a partir da ocorrência dos eventos *smr1* e *smr2*.

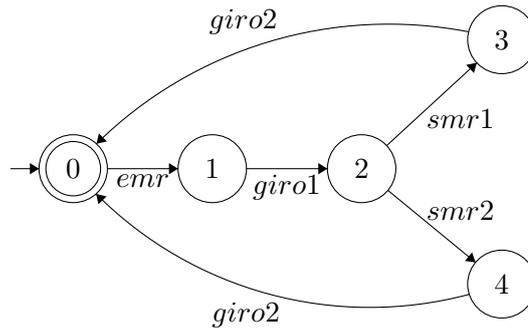


Figura 4.3: Planta da MR.

### 4.1.4 Mesas Transportadoras de Saída

A modelagem das plantas MTS1 e MTS2 é a mesma apresentada na seção 3.2.4 e o autômato que representa seus comportamentos é apresentado na figura 4.4.

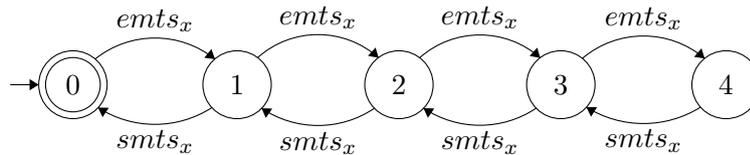


Figura 4.4: Planta de MTS1 e MTS2.

### 4.1.5 Elevadores

A modelagem realizada sobre o autômato que representa o comportamento dos elevadores é apresentada pela figura 4.5, o estado “0” indica que o elevador está em posição

para recebimento de caixas e o estado “1” indica que o elevador está encaminhando uma caixa para o depósito. O evento  $smts_x$ , representa a entrada de uma caixa no elevador e o evento  $tau1$  indica que o elevador entregou uma caixa ao depósito e retornou à posição de recebimento.

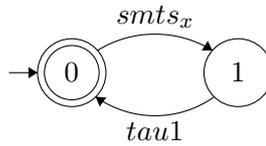


Figura 4.5: Planta de ED1 e ED2.

## 4.2 Modelagem das Especificações

Esta seção apresenta as especificações para correção dos supervisores, assim como a alteração de algumas das especificações adotadas no trabalho de [de Aguiar et al. \(2012\)](#) para a otimização da síntese dos supervisores.

### 4.2.1 Especificação de Carga do Tapete Alimentador

O tapete alimentador não necessita especificação, uma vez que o recebimento de caixas é automático e sua capacidade de carga é considerada infinita.

### 4.2.2 Especificação de Carga sobre a Mesa Transportadora de Entrada

A especificação de carga sobre a mesa transportadora de entrada  $E_{MTE}$  é a mesma apresentada na seção 3.3.2, e é apresentada na figura 4.6.

Entretanto, após a realização da simulação observou-se um mau funcionamento relacionado a ativação dos sensores indentificadores S1 e S2 sem a ocorrência da saída de uma caixa do tapete alimentador, o que gera estados proibidos no supervisor e consequentemente o mesmo não é capaz de representar a máxima linguagem controlável

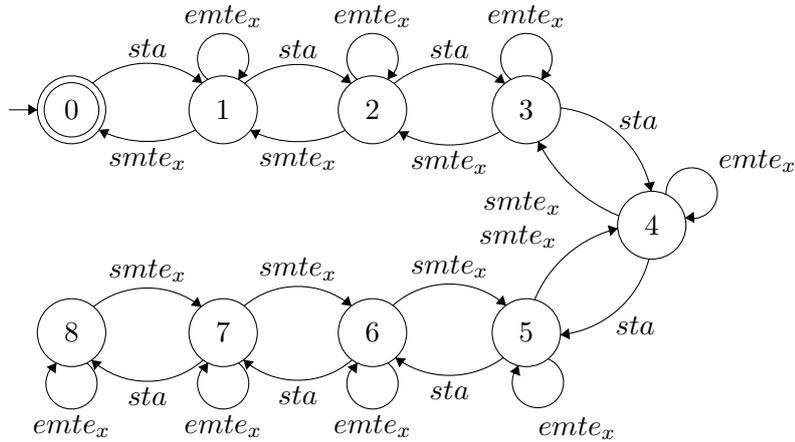


Figura 4.6: Especificação de carga de MTE.

do sistema. Para a correção deste problema implementou-se a especificação de funcionamento  $E_{correção}$  apresentada na figura 4.7, que só permite a ativação dos sensores S1 e S2 (eventos  $emte1$  e  $emte2$ ) após a saída de uma caixa do tapete alimentador (evento  $sta$ ). A composição síncrona  $E_{MTE} || E_{correção}$  representa a especificação corrigida.

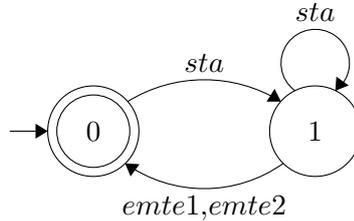


Figura 4.7: Especificação de correção de funcionamento sobre TA e MTE.

### 4.2.3 Especificação de Funcionamento sobre a Mesa Rotatória

A especificação sobre a mesa rotatória  $E_{MR}$ , apresentada na figura 4.8, restringe a saída de caixas da mesa transportadora de entrada, de forma que após a entrega de uma caixa à mesa rotatória (eventos  $smte1$  e  $smte2$ ), outra entrega só poderá ocorrer após a mesa retornar a posição de recebimento de caixas (evento  $giro2$ ). O estado “0” indica que a mesa está vazia e os estados “1” e “2” indicam que a mesa apresenta uma caixa

pequena ou grande, respectivamente.

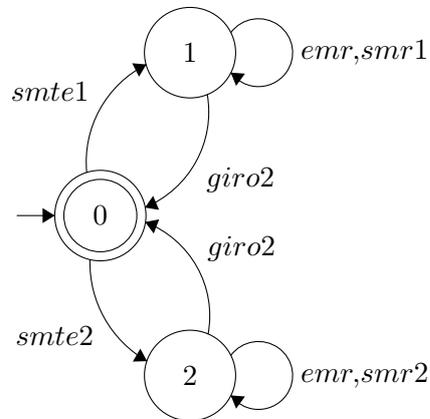


Figura 4.8: Especificação de funcionamento sobre a Mesa Rotatória.

#### 4.2.4 Especificação de Funcionamento sobre as Mesas Transportadoras de Saída

A especificação modelada sobre as mesas transportadoras de saída  $E_{MTS_x}$ , apresentada na figura 4.9, difere da apresentada na seção 3.3.4, pois apenas a especificação de carga das mesas leva ao mesmo mau funcionamento, pontuado na seção 4.2.2, sobre os sensores S7 e S8. Além disso, a modelagem da planta das mesas transportadoras de saída é sua própria especificação de carga, logo a especificação da seção 3.3.4 é redundante para a realização do controle. Diante esses fatos, realizando a análise sobre MTS1, a especificação só permite que o sensor de presença S7 ative (evento *emts1*) após a saída de uma caixa grande da mesa rotatória (evento *smr1*). Os estados “0” e “1” representam uma caixa sobre a mesa rotatória e a saída dessa caixa para a mesa transportadora de saída. A mesma análise é realizada para a especificação de MTS2.

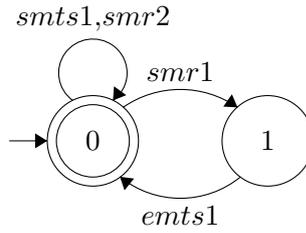


Figura 4.9: Especificação de Funcionamento sobre a Mesa Transportadora de Saída 1.

#### 4.2.5 Especificação de Carga dos Elevadores

Os elevadores não necessitam de uma especificação de carga ou funcionamento, uma vez que a própria modelagem da planta é sua especificação.

### 4.3 Síntese dos Supervisores

O primeiro passo para a obtenção dos supervisores consiste em determinar quais recursos irão compor as plantas locais associadas a cada especificação apresentada na Seção 4.2. A partir do cálculo da RSP essa associação foi obtida e é apresentada na Tabela 4.1.

Tabela 4.1: Especificações e Plantas Otimizadas Associadas.

Especificação	Planta Local
$E_{MTE}    E_{correção}$	$G_{TA}    G_{MTE}$
$E_{MR}$	$G_{MTE}    G_{MR}$
$E_{MTS_x}$	$G_{MR}    G_{MTS_x}    G_{ED_x}$

Para cada especificação e plantas associadas realizou-se a composição síncrona  $E_{local} || G_{local}$  que resulta na obtenção de  $R_{local}$ . A partir de  $R_{local}$ , os supervisores foram obtidos pela extração da máxima linguagem controlável  $supC(R_{local})$  e são apresentados na Tabela 4.2.

Tabela 4.2: Resultados para síntese de Supervisores Modulares Otimizados.

Supervisores	Glocal	E	Rlocal	Zlocal
Funcionamento de TA e MTE	1022	18	16612	16444
Funcionamento de MTE e MR	2555	3	4599	4599
Funcionamento de MR, MTSx e EDx	100	2	180	118

#### 4.4 Discussão

Os resultados apresentados neste capítulo aparentam apresentar maior complexidade em relação ao número de estados apresentados no artigo de [de Aguiar et al. \(2012\)](#), apresentados na Tabela 3.4. Entretanto, o supervisor sobre o funcionamento de TA e MTE não representa o comportamento adequado do sistema e por essa razão é descartado. O supervisor sobre a MTE e MR apresenta mais estados comparado com a modelagem original. Contudo, esse supervisor realiza o controle e direcionamento de caixas na mesa rotatória, eliminando o supervisor de roteamento. Finalmente, o supervisor sobre MR, MTSx e EDx torna desnecessário o uso de um supervisor sobre os elevadores.

## Capítulo 5

# Análise dos Resultados

No capítulo 3 apresentou-se a modelagem proposta por [de Aguiar et al. \(2012\)](#). Entretanto, essa modelagem não descreve corretamente o funcionamento do sistema. Com o objetivo de corrigir esse problema de mau funcionamento e otimizar os supervisores do sistema de triagem, apresentou-se a modelagem contida no capítulo 4.

Neste capítulo realiza-se a comparação entre os resultados obtidos da remodelagem do sistema de triagem no capítulo 4 e os resultados do trabalho de [de Aguiar et al. \(2012\)](#), presentes no capítulo 3.

### 5.1 Discussão dos Resultados

Antes de partir para discussão dos resultados, presente na subseção 5.1.3, uma breve análise sobre os supervisores do trabalho de [de Aguiar et al. \(2012\)](#) e sobre os resultados da otimização dos supervisores é realizada nas subseções 5.1.1 e 5.1.2 respectivamente.

#### 5.1.1 Resultados dos supervisores de [de Aguiar et al. \(2012\)](#)

A síntese dos supervisores resultante da modelagem apresentada no trabalho de [de Aguiar et al. \(2012\)](#) é apresentada na tabela 5.1. Observa-se nessa tabela a associação de cada

especificação sobre cada planta local, associação essa obtida pelo cálculo da RSP, e um total de 7 supervisores, resultado da modelagem aplicada.

Tabela 5.1: Especificações e Plantas Locais Associadas. Adaptado de [de Aguiar et al. \(2012\)](#).

Especificação	Planta Local
Carga de MTE	$G_{TA}  G_{MTE}$
Carga de MR	$G_{MTE}  G_{MR}$
Carga de MTSx	$G_{MR}  G_{MTSx}  G_{EDx}  G_{ROpl}$
Carga de EDx	$G_{MTSx}  G_{EDx}  G_{ROpl}$
Roteamento	$G_{MTE}  G_{MR}$

A complexidade é polinomial e resulta do número de estados da planta local  $m$  vezes o número de estados da especificação  $n$  ( $mn$ ) ([Su e Wonham, 2004](#)). A partir da tabela 5.2, que apresenta o número de estados de cada um dos componentes dos supervisores, calculou-se a complexidade, resultando em  $O(1022^2 * 9^2)$ .

Tabela 5.2: Resultados para síntese de Supervisores Modulares. Adaptado de [de Aguiar et al. \(2012\)](#).

Supervisores	Glocal	Elocal	Rlocal	Zlocal
Carga de MTE	1022	9	8686	8686
Carga de MR	2044	2	2555	2555
Carga de MTSx	40	5	192	192
Carga de EDx	10	2	10	10
Roteamento	2044	3	6132	6132

As especificações foram modeladas considerando o limite de carga de cada equipamento e o direcionamento das caixas pelo tamanho. Contudo, após a realização das simulações observou-se o problema mau funcionamento, apresentado no capítulo 3, sobre os supervisores de carga de MTE e de carga de MTS. Sendo assim, esses supervisores não descrevem corretamente o comportamento dessas plantas locais.

### 5.1.2 Resultados dos Supervisores Otimizados

A síntese dos supervisores apresentados neste trabalho é apresentado na tabela 5.3. Com o intuito de corrigir o problema de mau funcionamento e otimizar os supervisores, realizou-se a modelagem apresentada no capítulo 4.

Tabela 5.3: Especificações e Plantas Otimizadas Associadas.

Especificação	Planta Local
$E_{MTE}    E_{correção}$	$G_{TA}    G_{MTE}$
$E_{MR}$	$G_{MTE}    G_{MR}$
$E_{MTS_x}$	$G_{MR}    G_{MTS_x}    G_{ED_x}$

Observa-se na tabela 5.3 que a primeira especificação consiste da sincronização da especificação de carga da MTE com uma especificação de correção. Essa correção é necessária para que o supervisor descreva corretamente o comportamento da planta local.

A partir dos dados apresentados na tabela 5.4, calculou-se a complexidade dos supervisores, resultando em  $O(1022^2 * 18^2)$ .

Tabela 5.4: Resultados para síntese de Supervisores Modulares Otimizados.

Supervisores	Glocal	E	Rlocal	Zlocal
Funcionamento de TA e MTE	1022	18	16612	16444
Funcionamento de MTE e MR	2555	3	4599	4599
Funcionamento de MR, MTSx e EDx	100	2	180	118

A modelagem das plantas de cada equipamento do sistema de triagem e de cada especificação foram realizadas considerando o limite de carga de cada máquina e o comportamento conjunto de cada planta local no trabalho de [de Aguiar et al. \(2012\)](#). A partir da análise desses comportamentos, observou-se que remodelando a planta da mesa rotatória para definir adequadamente o destino das caixas baseado no seu tamanho, era

possível eliminar a especificação de roteamento, apresentada na Tabela 5.2, responsável pelo direcionamento de caixas na mesa rotatória.

Outro equipamento que fora remodelado para simplificar a síntese dos supervisores foram os elevadores. Como esses possuem capacidade de carga de uma caixa, a modelagem foi feita de forma que a planta fosse a própria especificação de carga. Dessa forma, os supervisores de carga dos elevadores apresentados na Tabela 5.2 se tornam desnecessários.

### 5.1.3 Comparação dos Resultados

A aplicação da especificação de correção aumentou consideravelmente a complexidade do supervisor sobre a planta local composta pelo tapete alimentador e a mesa transportadora de entrada, aumentando o número de estados do supervisor de 8686 para 16444. Consequentemente, a complexidade original do sistema de  $O(1022^2 * 9^2)$  se tornou  $O(1022^2 * 18^2)$ , tudo para que o conjunto de supervisores descrevam o comportamento correto do sistema.

Outro ponto importante a ser destacado é sobre a comparação dos resultados do supervisor da mesa rotatória deste trabalho com o presente no trabalho de [de Aguiar et al. \(2012\)](#). Assim como ocorreu no supervisor sobre a mesa transportadora de entrada, houve o aumento no número de estados de 2555 para 4599. Entretanto, o supervisor original de 2555 estados é responsável apenas pelo limite de carga da mesa rotatória, sendo necessário o auxílio de um supervisor de roteamento de 6132 estados para o direcionamento adequado das caixas que entram no sistema. Sendo assim, com um supervisor de 4599 estados realiza-se o mesmo controle realizado por dois supervisores que em conjunto possuem um total de 8687 estados.

Finalmente, o ultimo ponto a ser observado é referente ao supervisor sobre as mesas transportadoras de saída. A realização da remodelagem dos elevadores e a substituição da especificação de carga sobre as mesas transportadoras de saída pela especificação

de funcionamento resultou na correção do comportamento da planta local. Além dessa correção, houve a redução no número de estados do supervisor de 192 para 118 estados e a eliminação dos supervisores sobre os elevadores, uma vez que a remodelagem da planta dos elevadores tornara esses supervisores desnecessários.

Como resultado final, apesar do aumento da complexidade o número de supervisores fora reduzido de 7 para 4, resultando em economia de memória aplicado a um controlador lógico programável.

## 5.2 Considerações Finais

O objetivo da otimização do sistema de triagem presente no trabalho de [de Aguiar et al. \(2012\)](#) consistia em corrigir o problema de mau funcionamento e otimizar os supervisores sobre o direcionamento das caixas, visando reduzir a complexidade destes, no que se refere ao número de estados. Apesar do número de supervisores serem reduzidos pela remodelagem das plantas dos equipamentos e das especificações, a correção aplicada para corrigir o mau funcionamento do supervisor sobre a mesa transportadora de entrada elevou a complexidade do sistema como um todo. Dessa forma, a otimização aplicada não fora suficiente para corrigir o problema de mau funcionamento e reduzir a complexidade dos supervisores do sistema de triagem.

Para trabalhos futuros, fica a remodelagem da planta da mesa transportadora de entrada, principal responsável pela complexidade do sistema, devido aos seus 511 estados.

# Referências Bibliográficas

- Alves, M. V. S. (2014). Controle supervisorio robusto de sistemas a eventos discretos sujeitos a perdas intermitentes de observação. Master's thesis, Universidade Federal do Rio de Janeiro.
- Berstel, J., Perrin, D., e Reutenauer, C. (2009). *Codes and Automata*, volume 129. Cambridge University Press.
- Cassandras, C. G. e Lafortune, S. (2009). *Introduction to discrete event systems*. Springer Science & Business Media.
- Cunha, A. e Cury, J. (2007). **Hierarchical Supervisory Control Based on Discrete Event Systems With Flexible Marking**. *IEEE Transactions on Automatic Control*, 52(12):2242 –2253.
- Cury, J. E. R. (2001a). Teoria de controle supervisorio de sistemas a eventos discretos. *V Simpósio Brasileiro de Automação Inteligente (Minicurso)*, pg. 8.
- Cury, J. E. R. (2001b). Teoria de controle supervisorio de sistemas a eventos discretos. Em *Simpósio Brasileiro de Automação Inteligente*.
- Curzel e Leal, A. (2008). Implementação de controle supervisorio modular local em uma célula de manufatura didática. *VIII Conferência Internacional de Aplicações Industriais*.

- de Aguiar, R. S. d. S., da Cunha, A. E. C., e Vallim, M. B. R. (2012). Síntese de controle supervisorio modular local para um sistema de triagem. Em *Congresso Brasileiro de Automática*, pgs. 2842–2849.
- de Queiroz, M. H. e Cury, J. E. R. (2000). **Modular Supervisory Control of Large Scale Discrete Event Systems**.
- Eyzell, J. e Cury, J. (2001). **Exploiting Symmetry in the Synthesis of Supervisors for Discrete Event Systems**. *IEEE Transactions on Automatic Control*, 46(9):1500–1505.
- Feng, L. e Wonham, W. (2008). **Supervisory Control Architecture for Discrete-Event Systems**. *IEEE Transactions on Automatic Control*, 53(6):1449–1461.
- Hill, R. e Tilbury, D. (2006). **Modular Supervisory Control of Discrete-Event Systems with Abstraction and Incremental Hierarchical Construction**. Em *8th International Workshop on Discrete Event Systems*, pgs. 399–406.
- Hopcroft, J. E. (2008). *Introduction to automata theory, languages, and computation*. Pearson Education India.
- Hubbard, P. e Caines, P. (2002). **Dynamical Consistency in Hierarchical Supervisory Control**. *IEEE Transactions on Automatic Control*, 47(1):37–52.
- Leduc, R., Brandin, B., Lawford, M., e Wonham, W. (2005). **Hierarchical Interface-Based Supervisory Control-part I: Serial Case**. *IEEE Transactions on Automatic Control*, 50(9):1322–1335.
- Lin, F. e Wonham, W. M. (1988). **Decentralized Supervisory Control of Discrete-Event Systems**. *Information Sciences*, 44(3):199–224.
- Lin e Wonham, Lin F. e Wonham, W. (1991). Verification of nonblocking in decentralized supervision. *Control Theory and Advanced Technology*.

- Ma, C. e Wonham, W. (2006). **Nonblocking Supervisory Control of State Tree Structures**. *IEEE Transactions on Automatic Control*, 51(5):782 – 793.
- Max H. de Queiroz, J. E. R. C. (2002). Controle supervisorio modular de sistemas de manufatura. *SBA, Sociedade Brasileira de Automática, Campinas, 13, 123-133*.
- Max H. de Queiroz, J. E. R. C. (2014). Local modular supervisory control of timed discrete-event systems. *Workshop Discrete Event Syst., 2014, pp. 271–277*.
- Muler, O. P. (2018). Síntese e implementação de controle supervisorio de processos industriais com malha de válvulas. Master's thesis, Universidade Federal de Santa Catarina.
- Pena, P. N. (2007). *Verificação de conflito na supervisão de sistemas concorrentes usando abstrações*. PhD thesis, Universidade Federal de Santa Catarina.
- Queiroz, M. H. e Cury, J. E. R. (2000). Modular supervisory control of large scale discrete event systems. *Proceedings of the Workshop on Discrete Event Systems (WODES)*.
- Ramadge, P. J. e Wonham, W. M. (1987a). Supervisory control of a class of discrete event processes. *SIAM journal on control and optimization*, 25(1):206–230.
- Ramadge, P. J. e Wonham, W. M. (1989a). The control of discrete event systems. *Proceedings of the IEEE*, 77(1):81–98.
- Ramadge, P. J. G. e Wonham, W. M. (1987b). **Modular feedback logic for discrete event systems**. *SIAM Journal of Control Optimization*, 25(5):1202–1218.
- Ramadge, P. J. G. e Wonham, W. M. (1989b). The control of discrete event systems. *Proc. of the IEEE*, 77(1):81–98.
- Rudie, K. e Wonham, W. M. (1992). **Think Globally, Act Locally: Decentralized Supervisory Control**. *IEEE Transactions on Automatic Control*, 37(11):1692–1708.

- Schmidt, K., Marchand, H., e Gaudin, B. (2006). **Modular and Decentralized Supervisory Control of Concurrent Discrete Event Systems Using Reduced System Models**. Em *8th International Workshop on Discrete Event Systems*, pgs. 149–154.
- Sivolella, L., da Cunha, A., e Ades, R. (2006). Redução de supervisores como ferramenta para implementação de controladores discretos. Em *Anais do XVI Congresso Brasileiro de Automática, Salvador, Brasil, CBA'06*, pgs. 2778–2783.
- Su, R. e Wonham, W. M. (2004). Supervisor reduction for discrete event systems. *Discrete Event Dynamic Systems* 14:31-53.
- Torrico, C. R. C. e Cury, J. E. R. (2004). **Controle supervisorório hierárquico modular por agregação de estados**. *Sba Controle & Automação vol.15 no.3 Campinas July/Sept. 2004*, 15:291–299.
- Vaz, A. F. e Wonham, W. M. (1986). On supervisor reduction in discrete event systems. *International Journal of Control*, 44(2):475–491.
- Wong, K. e Wonham, W. M. (1996). **Hierarchical Control of Discrete-Event Systems**. *Discrete event Dynamic Systems: Theory and Applications*, 1:13–30.
- Wong, K. C. e Wonham, W. M. (1998). **Modular Control and Coordination of Discrete-Event Systems**. *Discrete Event Dynamic Systems*, 8(3):247–297.
- Wonham (2018). *Supervisory Control of Discrete-Event Systems*. In: Baillieul J., Samad T. (eds) *Encyclopedia of Systems and Control*. Springer, London.
- Wonham, W. M. (1991). Some current research directions in control of discrete event systems. Em *Proc. of 1991 European Control Conference*, Grenoble, France.
- Wonham, W. M. e Ramadge, P. J. (1988). **Modular Supervisory Control of Discrete Event Systems**. *Mathematics of Control, Signals, and Systems*, 1:13–30.

Zhong, H. e Wonham, W. M. (1990). **On the Consistency of Hierarchical Supervision in Discrete-Event Systems.** *IEEE Transactions on Automatic Control*, 35(10):1125–1134.

# Apêndice A

## Comandos para a Modelagem dos Supervisores

Este apêndice apresenta de forma simplificada os comandos utilizados na parte computacional deste trabalho para modelagem gráfica e cálculo de complexidade dos autômatos. Para exemplificar o processo de construção e análise dos supervisores implementados neste trabalho, utiliza-se neste apêndice o Exemplo 2.4 como base.

### A.1 Criação de Estados e Eventos

Os estados das máquinas  $M1$  e  $M2$  e da especificação do buffer  $B$  são criados utilizando o comando apresentado na Figura A.1. Observa-se que é possível nomear o estado como desejar e definir se este estado é marcado ou não.

```
// COMANDO PARA CRIAR ESTADOS
State s0 = new State("NOME DO EVENTO", Marking.Marked);
State s1 = new State("Empty", Marking.Marked);
State s2 = new State("Full", Marking.Unmarked);
```

Figura A.1: Comando para criação de estados.

Os eventos das máquinas  $M1$  e  $M2$  e da especificação do buffer  $B$  são criados

utilizando o comando apresentado na Figura A.2. Observando a Figura A.2, nota-se que é possível rotular os eventos como se deseja e definir se este evento é controlável ou não.

```
// COMANDO PARA CRIAR EVENTOS
Event e0 = new Event("RÓTULO DO EVENTO", Controllability.Controllable);
Event e1 = new Event("ini_process", Controllability.Controllable);
Event e2 = new Event("end_process", Controllability.Uncontrollable);
Event e3 = new Event("ini_process", Controllability.Controllable);
Event e4 = new Event("end_process", Controllability.Uncontrollable);
```

Figura A.2: Comando para criação de eventos.

## A.2 Criação, Visualização e Total de Estados de Autômatos

Uma vez criados os estados e eventos, utiliza-se o comando da Figura A.3 para criar os autômatos.

```
// COMANDO PARA CRIAR AUTÔMATOS
var M0 = new DeterministicFiniteAutomaton(new[]
{
    new Transition("estado_inicial", "evento", "estado_final"),
    new Transition("estado_inicial", "evento", "estado_final")
}, "estado_inicial_do_automato", "nome do automato");

var M1 = new DeterministicFiniteAutomaton(new[]
{
    new Transition(s1, e1, s2),
    new Transition(s2, e2, s1)
}, s1, "M1");
```

Figura A.3: Comando para criação de eventos.

Para criar o autômato é necessário associar os estados e eventos nas transições adequadamente e especificar seu estado inicial.

A obtenção do número de estados e visualização de autômatos é realizada utilizando-se os comandos apresentados na Figura A.4a. O número de estados é impresso no prompt de comando, como apresentado na Figura A.4b. A visualização gráfica do autômato é realizada no navegador padrão do computador, como apresentado na Figura A.4c.

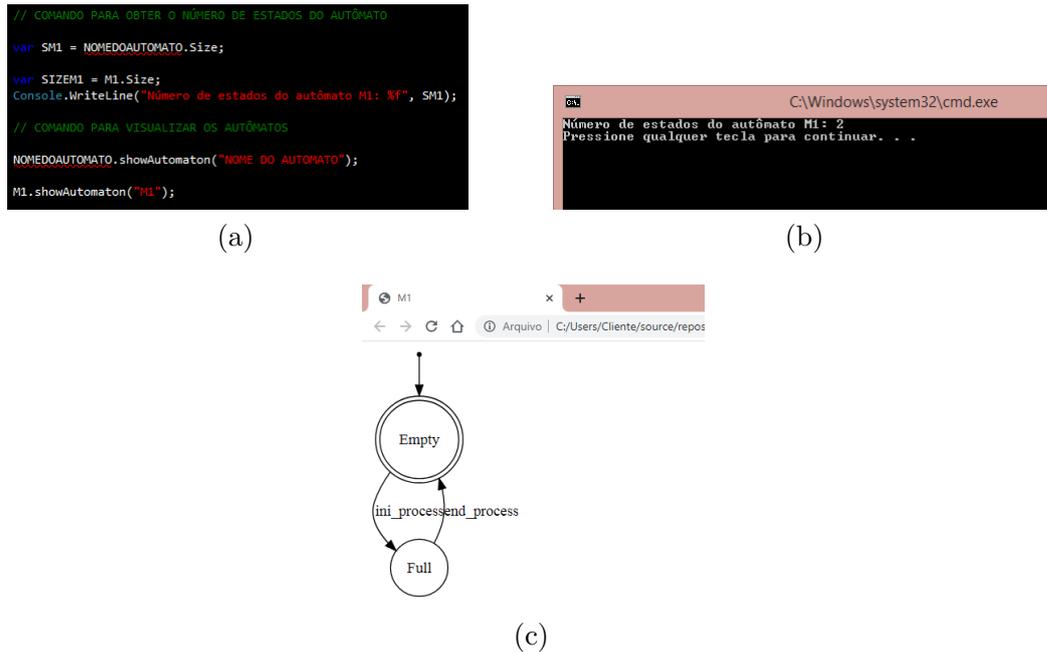


Figura A.4: Exemplificação do uso dos comandos para criação de autômatos: (a) Comandos; (b) Complexidade e (c) Visualização Gráfica.

### A.3 Criação dos Supervisores

Finalmente, após a implementação das plantas de  $M1$  e  $M2$  e da especificação do buffer  $B$ , implementa-se o supervisor utilizando o comando apresentado na Figura A.5.

```
// COMANDO PARA CRIAR SUPERVISORES
var sup0 = DeterministicFiniteAutomaton.MonolithicSupervisor(
    new[] { "plantas a serem sincronizadas" },
    new[] { "especificações a serem sincronizadas" }, true);

var sup1 = DeterministicFiniteAutomaton.MonolithicSupervisor(
    new[] { M1, M2 },
    new[] { ESP1 }, true);
```

Figura A.5: Comando para criação de supervisores.

Os comandos apresentados neste apêndice podem ser aplicados normalmente no supervisor, também considerado como um autômato, logo é possível visualizá-lo graficamente e calcular seu número de estados.