

**UNIVERSIDADE DE BRASÍLIA  
FACULDADE DE TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA MECÂNICA**

**ESTUDO E SIMULAÇÃO DE TÉCNICAS DE CONTROLE  
DE TRÁFEGO DE GRUPO DE ELEVADORES USANDO  
AUTOMAÇÃO INDUSTRIAL**

**ALVARO ANTONIO PATIÑO FORERO**

**ORIENTADOR: GUILHERME CARIBÉ DE CARVALHO.  
DISSERTAÇÃO DE MESTRADO EM SISTEMAS MECATRÔNICOS**

**PUBLICAÇÃO: ENM.DM – 38A/10  
BRASÍLIA/DF: DEZEMBRO – 2010  
UNIVERSIDADE DE BRASÍLIA**

**FACULDADE DE TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA MECÂNICA**

**ESTUDO E SIMULAÇÃO DE TÉCNICAS DE CONTROLE DE  
TRÁFEGO DE GRUPO DE ELEVADORES USANDO  
AUTOMAÇÃO INDUSTRIAL**

**ALVARO ANTONIO PATIÑO FORERO**

**DISSERTAÇÃO SUBMETIDA AO DEPARTAMENTO DE  
ENGENHARIA MECÂNICA DA FACULDADE DE TECNOLOGIA  
DA UNIVERSIDADE DE BRASÍLIA COMO PARTE DOS  
REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE  
MESTRE EM SISTEMAS MECATRÔNICOS.**

**APROVADA POR:**

---

**Prof. GUILHERME CARIBÉ DE CARVALHO, Ph.D. (ENM-UnB)  
(Orientador)**

---

**Prof. CARLOS HUMBERTO LLANOS, Dr. (ENM/UnB)  
(Examinador Interno)**

---

**Prof. Dr. PLÍNIO BENEDICTO DE LAURO CASTRUCCI - (EPUSP)  
(Examinador Externo)**

**BRASÍLIA/DF, 1 DE DEZEMBRO DE 2010**

## FICHA CATALOGRÁFICA

PATIÑO F., ALVARO ANTONIO

Estudo e Simulação de Técnicas de Controle de Tráfego de Grupo de Elevadores Usando Automação Industrial [Distrito Federal] 2010.

xvii, 143p., 210 x 297 mm (ENM/FT/UnB, Mestre, Sistemas Mecatrônicos, 2010).

Dissertação de Mestrado – Universidade de Brasília. Faculdade de Tecnologia.

Departamento de Engenharia Mecânica.

1. Sistema de Elevadores

2. Controladores Lógicos Programáveis

3. Lógica Nebulosa

4. Sistema de Controle

I. ENM/FT/UnB

II. Título (série)

## REFERÊNCIA BIBLIOGRÁFICA

Alvaro A. Patiño-Forero (2010). Estudo e Simulação de Técnicas de Controle de Tráfego de Grupo de Elevadores Usando Automação Industrial. Dissertação de Mestrado em Sistemas Mecatrônicos, Publicação ENM.DM-38A/10, Departamento de Engenharia Mecânica, Universidade de Brasília, Brasília, DF, 143p.

## CESSÃO DE DIREITOS

AUTOR: Álvaro Antonio Patiño Forero.

TÍTULO: Estudo e Simulação de Técnicas de Controle de Tráfego de Grupo de Elevadores Usando Automação Industrial.

**GRAU: Mestre**

**ANO: 2010**

É concedida à Universidade de Brasília permissão para reproduzir cópias desta Dissertação de Mestrado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte dessa dissertação de mestrado pode ser reproduzida sem autorização por escrito do autor.

---

Álvaro Antonio Patiño Forero

CLN 407 Bloco C, Sala 83.

70.855-530 – Brasília-DF-Brasil.

*Ao minha mãe Susana, pela educação e valores concedidos.*

## **AGRADECIMENTOS**

A Deus, por ter me mostrado os melhores caminhos para alcançar este objetivo.

À minha família, em especial à bebê mais linda do mundo minha sobrinha Valeria.

Ao meu orientador, Prof. Dr. Guilherme Caribé de Carvalho, pelos conhecimentos transmitidos, competência, orientação e apoio.

Aos professores que formam o corpo docente do programa de pós-graduação em Sistemas Mecatrônicos, em especial ao prof. Carlos Llanos.

À Empresa Rockwell Atomation do Brasil, pelo apoio financeiro concedido para o desenvolvimento deste trabalho.

Ao Grupo de Automação e Controle (GRACO) e ao Departamento de Engenharia Mecânica da Universidade de Brasília pelos recursos físicos fornecidos;

Aos colegas e amigos do GRACO, em especial Marrocos, Daniel, Ronald, Ana Maria, Diego, Claudia, Hugo, Rodrigo, Chucho, Liliana, Guillermo, Edgar, Magno, Tiago, Eliza, Fillipe.

À Capes pelo apoio financeiro.

A todos os que direta ou indiretamente contribuíram para a realização deste trabalho.

## RESUMO

Grupos de elevadores são normalmente encontrados em edifícios comerciais e seu controle pode ser feito de diversas formas, incluindo desde o simples atendimento da chamada de pavimento pelo elevador que esteja mais próximo até a seleção, por meio da avaliação de múltiplos critérios, daquele que apresente a maior aptidão para atender à chamada. Nesse contexto, este trabalho tem por objetivo principal estudar a aplicação de técnicas de controle nebuloso para definir as prioridades de atribuição de elevadores para o atendimento de chamadas de pavimento, considerando um sistema moderno em que se tem o conhecimento prévio do destino de cada chamada e utilizando uma arquitetura de automação industrial para a implementação do sistema de controle do grupo de elevadores (EGCS – *Elevator Group Control System*). A metodologia adotada neste trabalho consistiu de realizar uma ampla pesquisa sobre os EGCS's atualmente existentes, entender seu funcionamento e extrair de cada sistema analisado as vantagens específicas de cada estratégia e uni-las em um novo modelo baseado em técnicas de controle nebuloso. Desenvolveu-se então um controlador nebuloso com três entradas, relacionadas à distância, à disponibilidade de carga e ao tempo de espera dos passageiros, e uma saída relacionada à prioridade atribuída a cada elevador. Para a realização de testes de desempenho do sistema de controle, foi necessário desenvolver um simulador do comportamento de elevadores, o qual foi implementado por meio de um emulador de CLP, e um sistema gerador de chamadas de pavimento, o qual era responsável por gerar, com base em técnicas probabilísticas, os tempos de ocorrência das chamadas de pavimento. Desenvolveu-se ainda um *software* cliente OPC (*OLE for Process Control*) por meio do qual se fez possível capturar as informações da dinâmica dos elevadores, implementado no emulador de CLP, e realimentá-las. Além disso, incluíram-se no sistema cliente o controlador nebuloso e as rotinas de gerenciamento de dados necessárias à realização da simulação do tráfego de elevadores. Para o desenvolvimento do sistema cliente, utilizou-se a linguagem unificada de modelagem (UML), a qual descreve com diagramas padronizados o desenvolvimento do simulador e a interação com os diferentes componentes do sistema de simulação. Com base no sistema desenvolvido, realizaram-se simulações em um estudo de caso com diversos fluxos de passageiros e os resultados foram comparados com resultados de sistemas semelhantes publicados na literatura. O sistema de controle desenvolvido mostrou um desempenho compatível com o observado na literatura.

## ABSTRACT

Elevator groups are normally used in commercial buildings and their control can be carried out in different ways, for instance by using the nearest elevator for attending the hall calls or by using a selection process, which evaluates multiple criteria in order to compute the most suitable elevator. From this point of view, the main objective of this work is the analysis and application of fuzzy control techniques in order to define the attribution elevator priorities for attending the hall calls. This work considers a DCS (Destination Control System), which provides the control system with an *a-priori* knowledge of the desired floor for each hall call. Also, it makes use of an industrial automation architecture for implementing the elevator group control system (EGCS). The methodology applied in this work considers several stages, as follow: At the first stage, the background and the state of art regarding EGCS's was analyzed in order to extract the main advantages of each control strategy used nowadays for proposing an unified model based on fuzzy logic. Secondly, a fuzzy controller was developed. It uses as inputs the *distance*, the *elevator load availability* and the *passenger waiting time* and outputs the *suitable value of each elevator* for attending the hall call. In order to validate the performance of the elevator system an elevator group simulator was developed. The elevator group dynamics was implemented using a PLC emulator and a hall call generation system was developed. This is responsible for creating a list of times when the hall calls are expected to occur based on probabilistic techniques. Additionally, an OPC (*OLE for Process Control*) client software was developed, which can be used for reading and feeding back all the dynamic information of the elevators. On the other hand, the client system includes the fuzzy controller and the data management routines used for performing the simulation traffic of the elevators. The UML (Unified Modeling Language) was used for implementing the client system. Such a language uses standard diagrams for describing the simulator development and the interaction between the different components of the simulator system. A case study with various passenger flows was used for simulation purposes and the achieved results were compared with the results reported in the literature, demonstrating a compatible performance of the proposed elevator control system.

## SUMÁRIO

<b>RESUMO .....</b>	<b>vi</b>
<b>ABSTRACT.....</b>	<b>vii</b>
<b>1 - INTRODUÇÃO.....</b>	<b>1</b>
<b>1.1 - MOTIVAÇÃO DA PESQUISA.....</b>	<b>2</b>
<b>1.2-OBJETIVOS .....</b>	<b>4</b>
1.2.1-Objetivos Gerais .....	4
1.2.2-Objetivos Específicos .....	4
<b>1.3 – CONTRIBUIÇÕES DO TRABALHO.....</b>	<b>4</b>
<b>1.4 – ESTRUTURA DO DOCUMENTO .....</b>	<b>5</b>
<b>2 - FUNDAMENTAÇÃO TEORICA .....</b>	<b>6</b>
<b>2.1 - HISTÓRIA E ESTADO DA ARTE NA INDÚSTRIA DE ELEVADORES.....</b>	<b>6</b>
<b>2.2 - SISTEMAS DE ELEVADORES.....</b>	<b>8</b>
2.2.1 - Funcionamento do sistema de elevadores .....	8
2.2.2 – Controle do sistema de elevadores computadorizado (Barney, 2003).....	10
2.2.3-Controle de grupo de Elevadores.....	12
2.2.4 - Padrões de Tráfego de Elevadores.....	13
2.2.5 Gerador de tráfego de passageiros .....	15
2.2.6 Métodos para coletar observações estatísticas.....	18
2.2.7 – Tempo de espera médio de passageiros .....	19
2.2.8- Dimensionamento de um sistema de elevadores .....	20
2.2.9 Simulação e CAD .....	24
2.2.10-Sistemas modernos de grupo de elevadores .....	26
<b>2.3- TECNOLOGIAS DE AUTOMAÇÃO INDUSTRIAL.....</b>	<b>28</b>
2.3.1 - Controlador Lógico Programável (CLP) .....	28
2.3.2 – Ethernet.....	28
2.3.3 - DeviceNet® .....	29
2.3.4- A tecnologia OLE/COM.....	30



2.3.5- A tecnologia OPC .....	30
<b>2.4-UTILIZAÇÃO DA MODELAGEM UML .....</b>	<b>35</b>
<b>2.5- LÓGICA NEBULOSA .....</b>	<b>38</b>
2.5.1 - Método de Modelagem Matemática .....	39
2.5.2 - Método Heurístico .....	39
2.5.3 - Conjunto Nebuloso .....	40
2.5.4 - Número nebuloso e funções de pertinência .....	42
2.5.5 - Variáveis lingüísticas.....	43
2.5.6 - Sistemas nebulosos, fuzificação e defuzificação .....	43
2.5.7 –Controle nebuloso de grupo de elevadores .....	47
<b>2.6- CONCLUSÕES DO CAPITULO.....</b>	<b>48</b>
<b>3 - MODELAGEM E SIMULAÇÃO DE SISTEMAS DE GRUPO DE ELEVADORES.....</b>	<b>50</b>
<b>3.1 - Modelagem Monte-Carlo .....</b>	<b>50</b>
<b>3.2 - Técnicas de simulação .....</b>	<b>51</b>
3.2.1 - Construção do modelo .....	51
3.2.2 - Planejamento do simulador .....	53
3.2.3 - Implementação do simulador.....	55
3.2.4 - Execução das simulações.....	71
3.2.5 - Análise dos resultados das simulações .....	74
<b>4 - MODELAGEM UML DO SISTEMA DE GRUPO DE ELEVADORES.....</b>	<b>75</b>
<b>4.1 – CARACTERÍSTICAS PRINCIPAIS DO SISTEMA.....</b>	<b>76</b>
<b>4.2 – MODELO DE CASO DE USODO SISTEMA.....</b>	<b>77</b>
<b>4.3- MODELO DE PLANEJAMENTO.....</b>	<b>84</b>
4.3.1 – Diagramas de seqüência.....	85
4.3.2 – Diagramas de classes .....	91
4.3.3 – Diagrama de implantação .....	95
<b>5 - ESTUDO DE CASO: SISTEMAS DE GRUPO DE ELEVADORES.....</b>	<b>97</b>

<b>5.1 – SISTEMA DE ELEVADORES DESCRIÇÃO E ANALISES.....</b>	<b>97</b>
<b>5.2- ANÁLISE DOS RESULTADOS .....</b>	<b>103</b>
5.2.1 Análise estatística.....	103
5.2.2 Análise comparativa CAO e SAO .....	107
<b>6 - DISCUSSÃO DOS RESULTADOS .....</b>	<b>117</b>
<b>7 - CONCLUSÕES .....</b>	<b>121</b>
<b>7.1- SUGESTÕES PARA TRABALHOS FUTUROS .....</b>	<b>122</b>
<b>REFERENCIAS BIBLIOGRÁFICAS.....</b>	<b>124</b>
<b>ANEXO A .....</b>	<b>131</b>
<b>ANEXO B .....</b>	<b>133</b>
<b>ANEXO C .....</b>	<b>135</b>

## LISTA DE TABELAS

Tabela 2-1- Estrutura do software computadorizado.....	11
Tabela 2-2 - Padrões de tráfego de elevadores (adaptado de Hummet et al, 1978).....	16
Tabela 2-3 - Desempenho recomendado para um bom serviço em diferentes tipos de prédios (adaptado de Barney, 2003) .....	22
Tabela 3-1- Base de regra.....	68
Tabela 4-1 – Fluxos de trabalho do processo unificado .....	75
Tabela 4-2- Características principais do sistema.....	76
Tabela 5-1 - Parâmetros de operação de um elevador (Markon, 2006).....	98
Tabela 5-2 - Parâmetros usados na geração de chamadas na simulação .....	100
Tabela 5-3 -Análise estatística .....	105
Tabela 5-4 – Tempo de espera (s).....	111
Tabela 5-5 – Tempo de espera dentro do carro (s) .....	112
Tabela 5-6- Número de paradas .....	112

## LISTA DE FIGURAS

Figura 1-1 – Arquitetura do sistema de grupo de elevadores com SCD usando tecnologias de automação industrial .....	3
Figura 2-1 - Sistema de controle de grupo de elevadores convencional (adaptado de Bastidas, 1999). .....	9
Figura 2-2- Estrutura software para um sistema de controle de grupo de elevadores (adaptado de (Barney, 2003)) .....	11
Figura 2-3 - Demanda de passageiros em um prédio de escritórios (adotada de Barney, 2003). .....	14
figura 2-4- fluxo de passageiros em um prédio.....	15
Figura 2-5 - Fluxo de tráfego em prédios de escritórios durante o dia (modificado – Sorsa, 2002). .....	24
Figura 2-6- Etapas CAD no sistema de tráfego de elevadores (adaptado de Barney, 2003)...	26
Figura 2-7- Aplicação da rede DeviceNet® - (Rockwell, 2010) .....	29
Figura 2-8- Arquitetura da informação no controle de processos (Amaya, 2008) .....	31
Figura 2-9- Aplicações com vários servidores OPC (Tonaco, 2008) .....	32
Figura 2-10 - Relação entre Clientes e Servidores (Amaya, 2008) .....	33
Figura 2-11- Arquitetura OPC (Amaya, 2008) .....	34
Figura 2-12- Componentes de um diagrama de caso de uso .....	36
Figura 2-13 - Notação para classe em UML .....	37
Figura 2-14- Diagrama de seqüência .....	37
Figura 2-15- Diagrama de implantação de um sistema de informação remota .....	38
Figura 2-17 - Particionamento do universo discurso (adaptada de Shaw e Godoy, 1999).....	41
Figura 2-18 - Funções de pertinência (a) Triangular (b) Trapezoidal (c) Gaussiana (d) Campana .....	43
Figura 2-19 - Sistema nebuloso .....	44
Figura 2-20 - – Método Clássico de agregação Mamdani .....	46
Figura 3-1– Construção do modelo do sistema de elevadores.....	52
Figura 3-2– Diagrama de componentes do sistema simulação e controle de tráfego de elevadores .....	54
Figura 3-3- Interface gráfica para geração de tráfego de chamadas (submódulo parâmetros gerador de tráfego).....	56

Figura 3-4– Gráfico (a) representa a distância percorrida por um elevador. Gráfico (b) representa a curva de velocidade ( $v_{max}$ ) de um elevador entre duas paradas em andares subsequentes .....	59
Figura 3-5 - - Sistema de supervisão do sistema de tráfego de elevadores.....	60
Figura 3-6 – Função de pertinência da variável tempo de espera (tempo de espera normalizado) .....	63
Figura 3-7 – Função de pertinência da variável disponibilidade de carga.....	64
Figura 3-8 – Função de pertinência da variável distância.....	65
Figura 3-9 – Função de pertinência da variável prioridade .....	66
Figura 3-10- Máquina de inferência nebulosa para cálculo de prioridades.....	67
Figura 3-11 – Curva de controle do sistema de lógica nebulosa .....	68
Figura 3-12–Simulação do sistema de lógica nebulosa .....	69
Figura 3-13 – Funcionamento do Algoritmo de ordenamento.....	71
Figura 3-14 – Interface gráfica de configurar comunicação .....	72
Figura 3-15 – Interface gráfica de simulação de chamadas.....	73
Figura 4-1 – Casos de uso do sistema.....	77
Figura 4-2–Interface de configurar comunicação .....	79
Figura 4-3 – Interface para escolher servidor .....	79
Figura 4-4 – Interface de configuração de controle.....	80
Figura 4-5 - Interface simulação de chamadas .....	81
Figura 4-6 - Interface simulação de chamadas .....	82
Figura 4-7– Interface simulação chamadas (resultados).....	83
Figura 4-8 - Interface simulação de chamadas (prioridades).....	83
Figura 4-10 - Diagrama de seqüência de configurar controle.....	87
Figura 4-11 – Diagrama de seqüência de simulação chamada .....	89
Figura 4-13 - Diagrama de seqüência Calcular prioridade .....	90
Figura 4-14 – Diagrama de classes configurar comunicação .....	92
Figura 4-15 – Diagrama de classes de configurar controle.....	93
Figura 4-16 – Diagrama de classes simulação chamadas .....	94
Figura 4-17 – Diagrama de classes agrupar passageiros .....	95
Figura 4-18 – Diagrama de implantação.....	96
Figura 5-1–Tempo de espera e tempo de destino usando o sistema de controle SAO .....	101
Figura 5-2 – Número de parados dos elevadores em função do fluxo de passageiros com o sistema de controle SAO.....	102

Figura 5-3– Tempo de espera e tempo de destino usando um sistema de controle CAO.....	102
Figura 5-4 – Número de paradas dos elevadores em função do fluxo de passageiros com o sistema de controle CAO .....	103
Figura 5-5 - Simulação tempo de espera para valores de PPS de 12% (52 pessoas/5 minutos), 14% (59 pessoas/5 minutos) e 16% (69 pessoas/5 minutos) (chamadas no numero de dados).....	104
Figura 5-6 - Simulação tempo de espera para valores de PPS de 12% (52 pessoas/5 minutos), 14% (59 pessoas/5 minutos) e 16% (69 pessoas/5 minutos) sem o período do transiente .....	105
Figura 5-7 -Gráfico de probabilidade normal baseado nos dados da simulação .....	106
Figura 5-8- Tempo de espera dos sistemas de elevadores propostos.....	108
Figura 5-9-Comparação de intensidade de tráfego PPS = 16% e PPS=14%.....	109
Figura 5-10 -Comparação de intensidade de tráfego PPS=12% e PPS=10%.....	110
Figura 5-11-Comparação de intensidade de tráfego PPS = 6% e PPS= 4%.....	111
Figura 5-12- Tempo de espera do sistema de controle SAO proposto e tempo de espera de Siikonen (2000).....	113
Figura 5-13 - Tempo de espera do sistema de controle CAO proposto e tempo de espera de Siikonen (2000).....	114
Figura 5-14–Tempo de esperas de intensidade de tráfego de pessoas Konepolaris (2009)...	115
Figura 5-15 – Consumo de energia dos dois sistemas propostos neste trabalho .....	116

## LISTA DE SIMBOLOS, NOMENCLATURAS E ABREVIACOES

AI	- <i>Artificial Intelligence</i> – Inteligncia Artificial
CAO	controle Com Algoritmo de Ordenamento
CAD	- <i>Computer Aided Design</i> – Projeto Auxiliado por Computador
CLP	-Controlador Lgico Programvel
COM	- <i>Component Object Model</i>
CORBA	- <i>Commonn Relational Information Schema</i>
DCOM	- <i>Distributed Component Object Schema</i>
DDE	- <i>Dynamic Data Exchange</i>
DLL	- <i>Dynamic Link Library</i>
DS	- Sistema dinmico de setores.
<i>Down-peak</i>	- Trfego de descida.
ERP	- <i>Entreprise Resource Planning</i>
EGCS	- Sistema de Controle de Grupo de Elevadores
FEGCS	- Sistema Nebuloso de Controle de Grupo de Elevadores
HC	-Capacidade de transporte
I	-Intervalo de tempo
IHM	-Interface Homem Mquina
<i>Inter –floor</i>	- Trfego distribudo nos andares.
LCS	-Sistema de Controle Local
LAN	- <i>Local Area Network</i>
N	- Nmero de andares
OLE	- <i>Object Linking and Embedding</i>
OPC	- <i>OLE for Process Control</i>
p	- Nmero de passageiros
PC	- <i>Personal Computer</i> – Computador pessoal
RTT	- Tempo de viagem de ida e volta.
SAO	controle Sem Algoritmo de Ordenamento
<i>Up- peak</i>	- Trfego de subida.
UML	- <i>Unified Modeling Language</i>

## **1 - INTRODUÇÃO**

As demandas do mundo atual tornam a eficiência, a produtividade e a segurança os principais aspectos que devem ser levados em conta ao projetar ou modificar qualquer tipo de sistema. Para conseguir estes objetivos tem-se que, necessariamente, considerar novos conceitos e diferentes técnicas e abordagens que possam ser aplicadas para sua avaliação, além, evidentemente, do aproveitamento eficiente e efetivo dos recursos de alta tecnologia atualmente disponíveis (Finley et al., 1991).

Os sistemas de elevadores têm um importante efeito no uso de prédios modernos para o transporte vertical, utilizando diferentes tipos de instrumentação para a realização de ações de controle. Geralmente, nos prédios modernos existem tecnologias de comunicação distribuídas com o fim de centralizar a informação de toda a instrumentação utilizada no prédio. Tais tecnologias possibilitam a realização de Sistemas de Automação Predial (SAP) e o uso de computadores digitais para seu funcionamento. Nesse contexto enquadram-se os sistemas de controle de elevadores, cujas tecnologias ainda não estão totalmente estabelecidas (Barney, 2003). Esta é uma área ainda com grandes possibilidades de desenvolvimento com o objetivo principal de melhorar o desempenho do sistema relacionado ao aumento da capacidade de transporte e à redução do consumo de energia, assim como relacionado à redução do tempo de espera dos passageiros, de modo a aumentar seu conforto.

O avanço da tecnologia tem proporcionado cada vez mais a pesquisa e a utilização de modernas tecnologias que podem ser empregadas na construção e operação de plantas de elevadores. Para que uma tecnologia seja implementada, é necessário que a mesma atenda a diversas especificações de qualidade. Dentre estas incluem-se a segurança do projeto, a confiabilidade, a redução de custos, o conforto dos passageiros e a eficiência. Desta maneira, propor o uso de tecnologias de automação industrial se torna favorável para o desenvolvimento de arquiteturas modernas de sistema de elevadores, devido à robustez, versatilidade e flexibilidade que fornece este tipo de equipamento.



## 1.1 - MOTIVAÇÃO DA PESQUISA

A principal motivação para a realização deste trabalho surgiu no contexto do projeto ROCKWELL AUTOMATION UNB, que, dentre outros objetivos relacionados a aspectos de automação industrial, propôs um estudo de viabilidade técnica de aplicação de dispositivos, primariamente destinados ao uso em automação industrial, em sistemas de elevadores (Nunes, 2009). A partir desse estudo e da constatação de que há total compatibilidade entre os dispositivos industriais e a aplicação específica nos sistemas foco deste trabalho, decidiu-se por planejar o desenvolvimento de uma arquitetura moderna baseada em tecnologia de automação industrial capaz de propiciar um alto desempenho, aproveitando as vantagens que estas tecnologias fornecem.

A modernização de elevadores traz diversos benefícios, como por exemplo, melhora em segurança e desempenho, eliminação de ruídos e vibrações produzindo viagens mais confortáveis e economia de energia (Tianxiao, 2004, Sachs, 2005). Desta maneira, a aplicação de tecnologia de automação pode ajudar significativamente a melhora dos fatores de segurança e desempenho de um sistema de elevadores.

Dentre as técnicas utilizadas no desenvolvimento dos sistemas de controle de tráfego de elevadores destacam-se as técnicas de Inteligência Artificial (IA), especialmente os sistemas de lógica nebulosa (*Fuzzy Logic*) por sua facilidade de incluir de alguma forma o conhecimento especialista dentro do sistema de controle. Os sistemas de lógica nebulosa podem ser combinados com diversas tecnologias usadas no ambiente industrial, permitindo contribuir com a melhoria da modernização de sistema de elevadores.

Diante do exposto nos parágrafos anteriores, neste trabalho estabeleceu-se como objetivo o estudo do problema de controle de grupo de elevadores, utilizando como estudo de caso a simulação de um sistema moderno implementado por meio das tecnologias de automação industrial atualmente disponíveis no mercado. Para isso realizou-se o planejamento de uma arquitetura moderna de elevadores, a qual é mostrada na Figura 1-1. Nesta, pode-se observar a previsão de um dispositivo de entrada de dados a ser instalada em cada andar, aqui representada por uma interface homem-máquina (IHM) (PanelView®). As IHMs integram o chamado sistema de controle de destino (SCD), o qual possibilita o conhecimento prévio do destino do passageiro que originou a chamada de pavimento. Um controlador lógico programável (CLP) encarrega-se de embarcar vários sistemas de controle local (LCS), sendo um para

cada elevador. Os LCS's são responsáveis pelo controle de velocidade dos carros assim como por operações de portas e por monitoramento de localização no trajeto assim como verificação de capacidade de carga disponível nos elevadores. O sistema de elevadores possui uma rede industrial de campo, (DeviceNet®) para conectar vários instrumentos industriais como: Drives AC (um para cada carro), sensores indutivos, módulos de entrada e saída digital e IHM (Patiño-Forero, 2009). O sistema também possui uma rede Ethernet, para monitoramento do sistema desde qualquer lugar por meio de uma rede local.

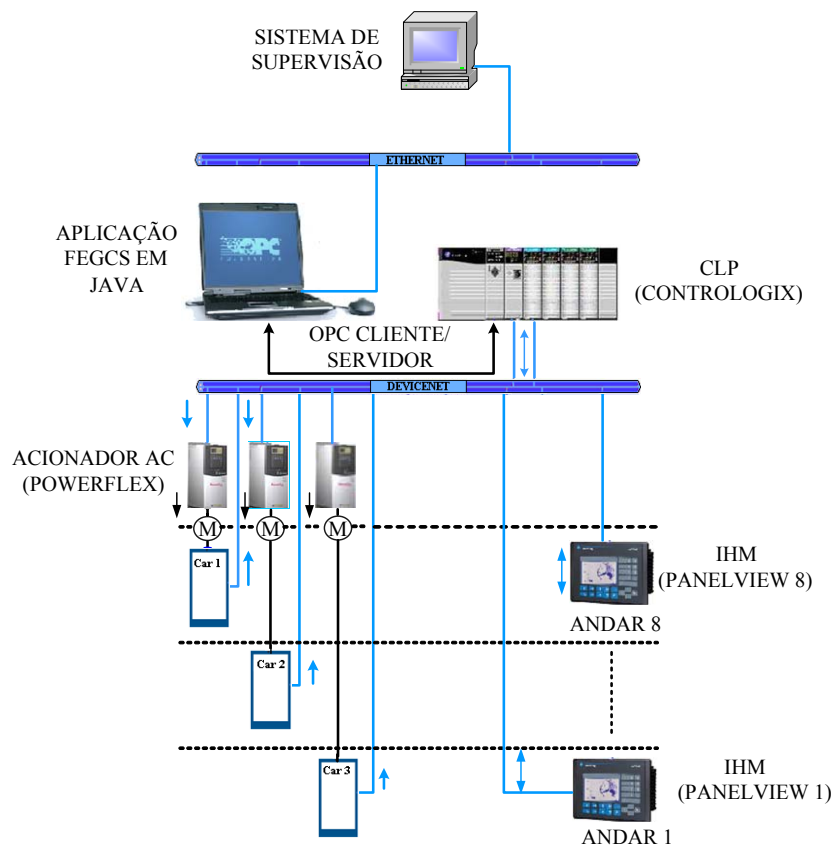


Figura 1-1 – Arquitetura do sistema de grupo de elevadores com SCD usando tecnologias de automação industrial

Para atender às demandas geradas pela proposta de arquitetura propõe-se o desenvolvimento de um simulador de tráfego de elevadores, utilizando as ferramentas fornecidas pela empresa *Rockwell Automation*, e técnicas de controle de tráfego de grupo de elevadores usando lógica nebulosa. O desenvolvimento e o uso do simulador para o estudo comparativo de técnicas de controle de grupo de elevadores constituem a meta principal deste trabalho, a qual é descrita em detalhes no próximo item.

## **1.2-OBJETIVOS**

### **1.2.1-Objetivos Gerais**

O objetivo do trabalho é apresentar a implementação de técnicas de controle de grupo de tráfego elevadores usando tecnologias de automação industrial. Para tanto foi realizado um simulador de tráfego de elevadores usando tecnologias de automação industrial, incluindo o desenvolvimento de um gerador de tráfego baseado em distribuições probabilísticas e adotando padrões de tráfego tipicamente encontrados em prédios comerciais.

### **1.2.2-Objetivos Específicos**

Os objetivos específicos do trabalho são:

- Apresentar as técnicas de controle de tráfego de grupo de elevadores e arquiteturas usadas, que justifiquem a aplicação de ferramentas inteligentes no desenvolvimento do controle de grupo de elevadores baseadas em lógica nebulosa.
- Projetar um controle de tráfego de grupo de elevadores baseado em lógica nebulosa, segundo o tráfego e distribuição de passageiros no prédio.
- Apresentar a modelagem de um simulador de tráfego de grupo de elevadores baseado em lógica nebulosa, usando tecnologias de automação industrial.
- Validar os diferentes algoritmos de controle projetados para a situação de tráfego mais crítica e comparar os resultados.
- Definir uma metodologia que permita estudar e comparar os resultados obtidos das simulações.

## **1.3 – CONTRIBUIÇÕES DO TRABALHO**

No transcurso do desenvolvimento do trabalho foram realizadas três publicações. (a) no “*International Congress of Mechanical Engineering*” (COBEM-

2009) sobre o modelo de grupo de elevadores moderno usando Controlador Lógico Programável (CLP), (b) no “*Congresso Brasileiro de Automática*” (CBA-2010) sobre a modelagem UML para um simulador de tráfego de grupo de elevadores usando um emulador de CLP, e (c) no “*Congresso Nacional de Engenharia Mecânica*” (CONEM-2010) sobre desenvolvimento de um simulador de sistemas para controle nebuloso de grupo de elevadores (Anexo A).

#### **1.4 – ESTRUTURA DO DOCUMENTO**

Esta dissertação está dividida em sete capítulos. O capítulo dois apresenta a fundamentação uma revisão bibliográfica com os principais tópicos relacionados ao tema.

O capítulo três descreve em detalhe a metodologia desenvolvida, especialmente a estrutura proposta para desenvolver simuladores aplicados ao sistema de controle de tráfego de grupo de elevadores.

O capítulo quatro apresenta os detalhes da modelagem UML (*Unified Modeling Language*) desenvolvida, seus requisitos e principais funcionalidades.

O capítulo cinco apresenta os resultados das simulações das técnicas de controle de tráfego implementadas com a finalidade de avaliar o desempenho do sistema e realizar comparações com outros autores. Nesse mesmo capítulo apresentam-se a análise estatística sobre os resultados simulados.

O capítulo seis apresenta a discussões sobre os resultados alcançados comparando-os com resultados apresentados na literatura, e buscando no controlador explicações para diferenças encontradas.

Finalmente, no capítulo sete apresenta-se as conclusões e sugestões para trabalhos futuros.

## **2 - FUNDAMENTAÇÃO TEÓRICA**

Este capítulo apresenta a fundamentação teórica dos temas envolvidos no presente projeto. Inicialmente é introduzido um breve histórico dos sistemas de elevadores, enfatizando a evolução do transporte vertical e o estado da arte. O segundo item descreve algumas considerações de importância na área de sistemas de controle de grupo de elevadores aplicados a edifícios comerciais, visando explicar suas características, modo de funcionamento, classificações de tráfego, técnicas de controle e ferramentas de simulação. O terceiro item descreve algumas definições da modelagem UML. Em seguida é abordado o tema relacionado como as tecnologias de automação industrial adotadas neste trabalho. Finalmente é apresentado o conceito de lógica nebulosa como mecanismo de controle para o despacho de um grupo de elevadores.

### **2.1 - HISTÓRIA E ESTADO DA ARTE NA INDÚSTRIA DE ELEVADORES**

Não é de hoje que a história registra os esforços da humanidade para transportar verticalmente cargas e pessoas. 1500 anos antes do nascimento de Cristo, os egípcios já estavam às voltas com a dura tarefa de elevar as águas do Nilo através de rudimentares elevadores. Com o passar dos séculos, a tração animal, incluindo aí a humana, foi substituída primeiro pela energia do vapor, exclusivamente para o transporte de cargas. Mais tarde, com o surgimento de novos mecanismos de segurança, iniciou-se o transporte de passageiros.

Somente em 1853, ano em que a história confere à americana Elis Graves Otis a invenção do elevador de segurança, o uso de elevadores como meio de transporte de passageiros começou a se popularizar. Os primeiros elevadores movidos a vapor eram muito lentos. Para um passageiro alcançar o oitavo andar de um prédio, levava em média 2 minutos. Atualmente alguns elevadores são capazes de atingir a velocidade de 550 m/min, o que significa dizer que são mais de 45 vezes mais rápidos do que os seus antecessores movidos a vapor.

Os primeiros elevadores brasileiros só começaram a ser fabricados em 1918. Os mesmos não eram movidos nem a vapor, nem a eletricidade. Era o cabineiro, girando uma manivela, que fazia com que o elevador subisse ou descesse. As portas, pantográficas, eram também abertas e fechadas manualmente. Com a explosão demográfica e a construção de edifícios mais altos, o sobe e desce movido à manivela

foi substituído por sistemas elétricos mais complexos que dispensavam o serviço dos cabineiros. Não mais era preciso gritar ou gesticular para o cabineiro. Para chamar o elevador, bastava apenas apertar um botão. Atender às chamadas com o apertar de um botão foi apenas o começo. Para melhorar a eficiência do elevador, relés e circuitos elétricos foram desenvolvidos. O Comando Automático Seletivo foi o próximo passo, permitindo que elevadores trabalhassem isoladamente, o que melhorou significativamente o tráfego nos edifícios.

Com o aporte de tecnologia oriunda da informática, a satisfação dos usuários aumentou sensivelmente. O atendimento aos andares passou a ser controlado de uma forma racional, evitando viagens inúteis. Além dos passageiros, um dos maiores beneficiados foram os responsáveis pela manutenção. A vida útil dos componentes cresceu em razão direta à redução das possibilidades de defeito, economia de energia elétrica e facilidade de conservação.

O projeto e a decoração dos elevadores também foi modificado para melhor. Com linhas mais modernas, valorizando a arquitetura do edifício, os elevadores passaram não apenas a transportar, mas transportar com requinte e sofisticação (Tudosobreimoveis, 2010).

O mercado atual na indústria de elevadores é diversificado. Existem empresas especializadas em fabricação de cabines, de quadros de comando, ou de alguns componentes específicos. Particularmente, as empresas que investem em inovação e tecnologia, possuem a maior parte do mercado mundial, entre elas estão, a *Otis Corporation*, *Kone Corporation*, *ThyssenKrupp* e *Atlas Schindler*. Considerando a alta competitividade existente no mercado de elevadores, novas concepções, novos modelos de controle e melhorias no sistema de elevadores têm sido desenvolvidos e implementados por essas empresas. As publicações científicas encontradas relacionadas ao assunto apresentam em sua maioria novas técnicas de controle.

A maioria das empresas na indústria de elevadores encontradas no Brasil dedicam-se à comercialização dos produtos ou à manutenção dos sistemas de transporte vertical, sendo da *ThyssenKrupp* (ThyssenKrupp, 2010) e *Otis Corporation* (Otis, 2010) as maiores no mercado. Empresas nacionais, como é o caso de *Sectron* (Sectron, 2010) e *Infolev* (Infolev, 2010), possuem uma presença forte e competitiva no mercado nacional e internacional, participando na fabricação dos quadros de comando.

Indústrias locais investem na atualização dos seus produtos aplicando métodos e tecnologias emergentes. O desenvolvimento do elevador moderno tem produzido

mudanças profundas na arquitetura e espera-se uma maior evolução das cidades no sentido de permitir a construção de prédios de maior altura.

## **2.2 - SISTEMAS DE ELEVADORES**

O rápido desenvolvimento da construção das cidades provocou o surgimento de prédios grande altura e a expansão de suas áreas. Portanto, o uso dos elevadores tornou-se mais importante com o objetivo de satisfazer às necessidades de transporte vertical dos usuários fornecendo uma alta qualidade do serviço. O desenvolvimento destes sistemas tem sido afetado pelas exigências de produtividade (que é considerada maior quanto menor o tempo de locomoção, tempo de espera, entre outros) do mundo atual. Muitos avanços têm sido alcançados neste campo com a introdução de melhores sistemas de controle (monitoração e acionamento). A fim de melhorar a eficiência e funcionamento dos sistemas de elevadores o computador começa a ser usado para controlar o funcionamento dos sistemas de mais de dois elevadores, ou seja sistema de grupo de elevadores.

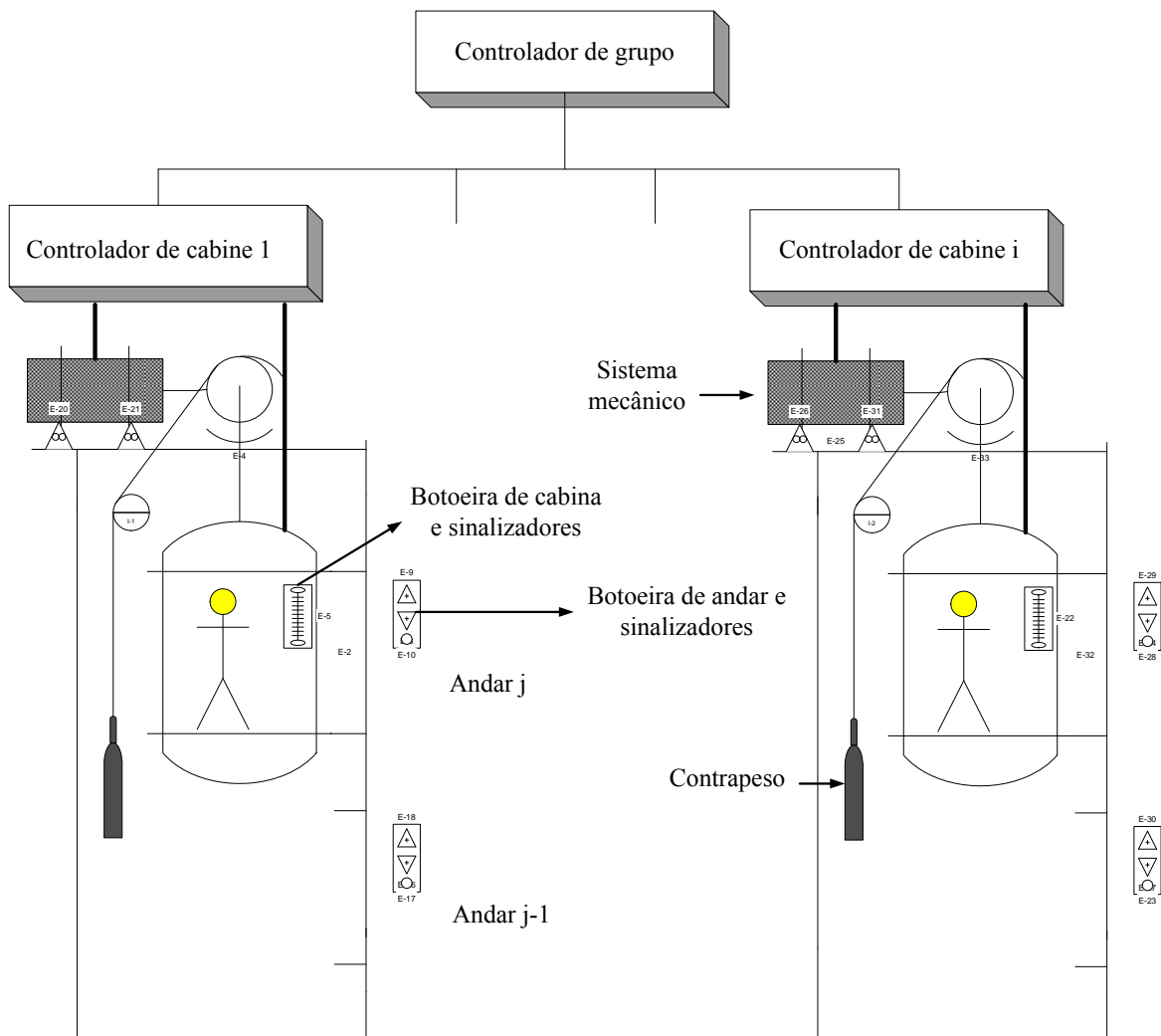
Para a implementação de um sistema de grupo de elevadores, os projetistas devem desenvolver estudos baseados em especificações e simulações de modelos que permitam o exame e análise das várias configurações possíveis e o detalhamento das propostas mais promissoras. O funcionamento especificado destes sistemas é geralmente determinado pela natureza da demanda de transporte do edifício, a qual pode ser atendida utilizando diferentes métodos visando à melhoria do desempenho do sistema tanto quantitativamente como qualitativamente (Bastidas, 1999).

### **2.2.1 - Funcionamento do sistema de elevadores**

A operação de um sistema de elevadores (pode envolver um único elevador ou um grupo com vários elevadores) pode ser explicada ao considerar como um passageiro é transportado do andar A ao andar B: No andar A o passageiro registra uma chamada de andar ao pressionar um botão na botoeira instalada neste (para subir ou descer).

O “controle de grupo” registra a chamada sinalizando isso ao usuário através de um sinal luminoso e seleciona um elevador para servi-lo. O passageiro pode observar a posição e direção de movimento do elevador no edifício por meio de sinalizadores nos andares e um sinalizador de direção cujo estado é atualizado quando o elevador se

movimenta. A sinalização no andar é desligada quando a cabine do elevador selecionado alcança o andar da solicitação e as portas são abertas. O passageiro entra e registra por meio de botões (de comando) dentro da cabine seu destino, isto é o andar B, o qual é devidamente sinalizado para indicar o registro feito pelo “controlador de cabina”. As portas se fecham e o elevador se move até o andar B. Ao chegar próximo ao andar B o elevador reduz a marcha, pára e abre as portas para que o passageiro desembarque. A Figura 2-1 apresenta os elementos envolvidos na operação deste sistema. O “controlador de cabina” comanda o “sistema mecânico” composto pelo variador de velocidade, motor, freios de segurança, sensores de proximidade, sistema de tração com redutor e o sistema de cabina (controle de portas, sistema de sinalização e luminosidade) (Bastidas, 1999).



**Figura 2-1** - Sistema de controle de grupo de elevadores convencional (adaptado de Bastidas, 1999).

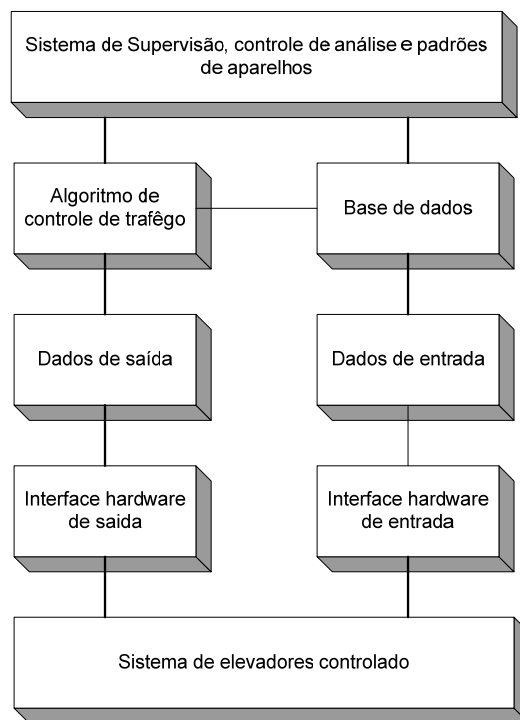


### **2.2.2 – Controle do sistema de elevadores computadorizado (Barney, 2003)**

O controle pelo computador tem sido aplicado desde finais dos anos 1950s, em processos e indústrias geradoras de energia. A aplicação de computadores para controle de grupo de elevadores foi um importante passo nas tecnologias de elevadores. Os computadores oferecem uma alta confiabilidade e permitem uma considerável versatilidade no tipo de controle para ser desenvolvido. As facilidades e funcionalidades contidas na implementação de um algoritmo de controle de elevadores realizado em um computador são só limitadas pela imaginação e criatividade do projetista do algoritmo, a velocidade de processamento do computador e pelas considerações financeiras.

A primeira abordagem na indústria de controle de elevadores computadorizado foi a implementação de algoritmos de controle, baseados em lógica de reles. Essas não apresentavam um bom desempenho, devido a ser implementadas em lógica fixa e portanto limitada.

As técnicas avançadas de controle utilizam matemática complexa, o que faz com que a implementação de um sistema de controle de elevadores seja difícil. A estrutura de software necessária para um controle de elevadores computadorizado é apresentado na Figura 2-2.



**Figura 2-2-** Estrutura software para um sistema de controle de grupo de elevadores  
(adaptado de (Barney, 2003))

Segundo o esquema apresentado na Figura 2-2, as principais partes da estrutura de controle de software são classificadas da seguinte forma (Tabela 2-1).

**Tabela 2-1 -** Estrutura do software computadorizado

<b>Estrutura software do controle computadorizado</b>	<b>Características</b>
Sistema de supervisão, análise do controle e padrão de controladores de dispositivos.	O sistema de supervisão pode permitir selecionar o algoritmo de controle, acessar à base de dados, iniciar a verificação de dados.
Algoritmo de controle de tráfego	O software projetado para controle de tráfego. (escolha de linguagem de programação)
Base de dados	Armazenamento de informação do sistema
Saída/entrada de dados	Controladores de comunicação I/O
Interface de Saída/entrada hardware	Interfaces de comunicação I/O (hardware)
Sistema de elevadores controlado	Planta do sistema de elevadores

O sistema de supervisão pode incluir um sistema de execução que trabalhe com todos os dispositivos periféricos padrões. As linhas de sinal de entrada de dados fornecem informação sobre a posição e estados do elevador. As linhas de saída enviam

os comandos de controle para o sistema de elevadores controlado através da interface hardware de saída.

O monitoramento de tráfego é opcional e sub-programas de análise pode ser incluído para fornecer execução de registros do comportamento do sistema e registrar incidentes sobre o sistema de manutenção e outros fins.

### **2.2.3-Controle de grupo de Elevadores**

Os Sistemas de Controle de Grupo de Elevadores (EGCS) são sistemas que conseguem dirigir sistematicamente três ou mais elevadores com o objetivo de se otimizar alguma função de custo. O EGCS é considerado, computacionalmente, como um problema NP-hard (Nikovsky e Brand, 2003) e envolve processos dinâmicos, estocásticos considerando incertezas sobre os instantes de tempo em que ocorrem novas chamadas de pavimento e sobre o número de passageiros esperando nos andares onde se originaram as chamadas (Beielstein *et al.*, 2003). São,conseqüentemente, sistemas de dinâmica complicada que incluem não linearidades e problemas de controle multi-objetivo (Markon, et al, 2006).

Atualmente, existem muitas técnicas avançadas para controle de grupo de elevadores que podem ser classificadas no campo da inteligência artificial (IA). Por exemplo, no caso de grupos de elevadores que usam o sistema tradicional de botões de chamada para acima e para baixo, várias técnicas têm sido publicadas na literatura especializada, dentre as quais citam-se: controle ótimo (Closs. et al, 1970), lógica nebulosa (Ho e Robertson, 1994), programação dinâmica (Chan e So, 1996), algoritmos genéticos (Miravete. et al, 1999.), sistemas especialistas (Qun et al., 2001) e redes neurais (Barney e Imrak, 2001).

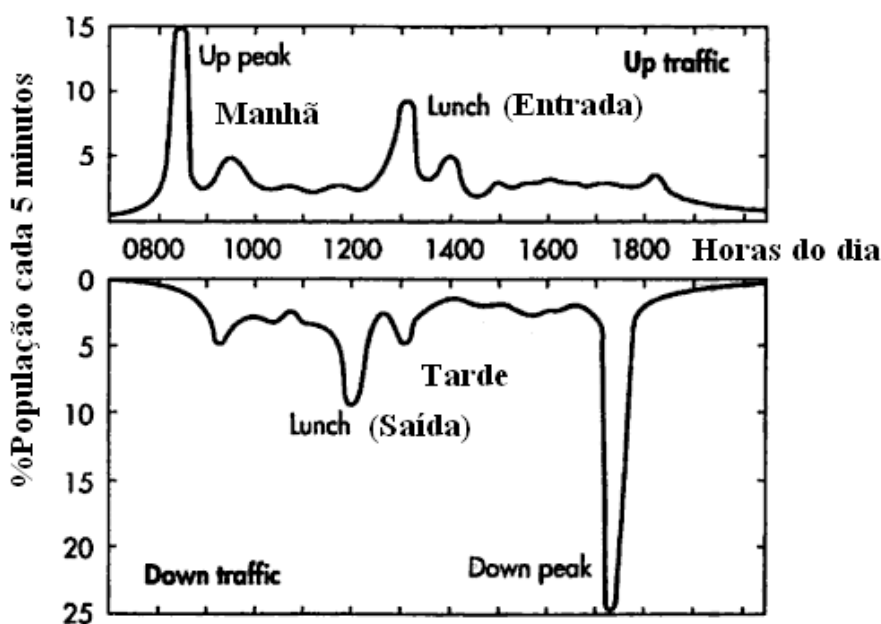
Sistemas de elevadores recentes têm uma estrutura diferente daquelas normalmente utilizadas nos sistemas tradicionais. Uma dessas diferenças é a possibilidade de se conhecer o andar de destino do passageiro antes que ele entre no carro, como nos chamados “Sistemas de Controle de Destino” (SCD) (Markon, 2008). O desempenho de sistema de elevadores que usam o sistema SCD é melhor porque a estratégia de controle pode ser desenvolvida usando informação prévia sobre o andar de destino , assim como o número de passageiros esperando pelo carro no pavimento, supondo que todos os passageiros têm informado seus respectivos destinos. Neste caso, o SCD fornece uma boa aproximação da demanda para o sistema de controle agrupar os

passageiros que irão seguir para um mesmo andar, diminuindo o tempo de espera e o número de paradas enquanto chega ao destino (Sorsa. et al, 2005).

#### **2.2.4 - Padrões de Tráfego de Elevadores**

Os padrões de tráfego são baseados nas características e na distribuição demográfica de um prédio. Atualmente, as empresas de elevadores que operam ao redor do mundo realizam um dimensionamento do sistema de elevadores, por meio de simuladores, de modo a analisar possíveis problemas de congestionamento resultantes das variações nos padrões de tráfego esperados ao longo do dia de trabalho. Assim, pesquisas em torno da finalidade do prédio, estimação de número de andares, número de elevadores e fluxo de pessoas por dia são efetuadas.

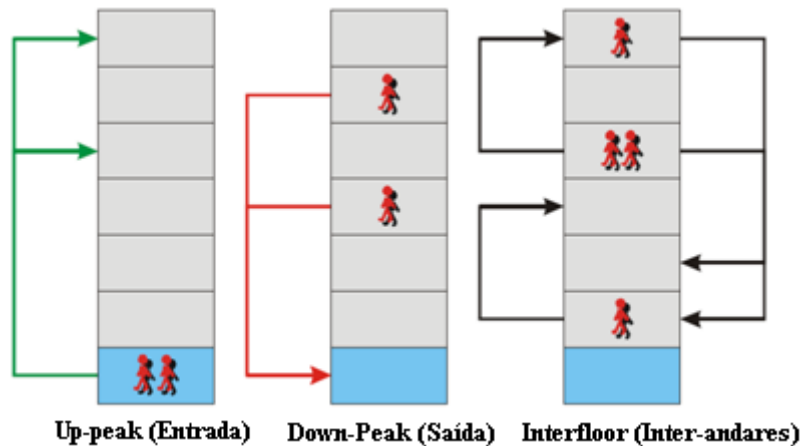
A demanda de passageiros impõe ao sistema de elevadores a necessidade de responder a diferentes padrões de tráfego. A Figura 2-3 apresenta a demanda de passageiros em um edifício de escritórios representando o número individual de chamadas, agrupadas pelo sentido das chamadas para acima ou para baixo. As curvas mostram a demanda de passageiros, medida em valor % em relação à população do prédio, em função do horário durante o dia de trabalho. Claramente observam-se os diferentes padrões de tráfego nas primeiras horas de trabalho pela manhã (*Up-peak*), ao fim do expediente à tarde (*Down-peak*) e no intervalo de almoço próximo ao meio-dia. Observa-se também uma demanda aproximadamente uniforme nos horários intermediários, em que se ocorre tráfego inter-andares (*Interfloor*).



**Figura 2-3** - Demanda de passageiros em um prédio de escritórios (adotada de Barney, 2003).

A Figura 2-3 mostra como é o fluxo de passageiros em um prédio de escritórios de acordo com os padrões identificados na Figura 2-4. Ao início do dia tem um número importante de chamadas de pavimento no térreo do prédio. Isto é devido à entrada de pessoas que trabalham no prédio e que iniciam sua jornada de trabalho. Este padrão de tráfego é chamado *Up-peak*. Ao final da tarde tem um número importante de chamadas de pavimento como sentido para abaixo para o andar principal (térreo). Isto é devido à saída de pessoas que trabalham no prédio e que terminam sua jornada de trabalho. Este padrão de tráfego é chamado *Down-peak*. Ao meio-dia há dois conjuntos característicos de tráfego, *Up-peak* e *Down-peak*. Isto ocorre devido à saída e à entrada do pessoal que trabalha no prédio na hora de almoço (*Lunch*). Na Figura 2-4 especificamente, o padrão *Lunch* é caracterizado por apresentar dois picos de descida afastados por uma hora e dois picos de subida, também afastados por uma hora, indicando que o intervalo de almoço é de 1 hora e que há um revezamento entre os trabalhadores de modo a não interromper o funcionamento da empresa no horário de almoço. Durante resto do dia, as chamadas de pavimento para acima e para abaixo são distribuídas de forma similar e não existe uma concentração de chamadas em um andar ou para um só destino. Este padrão de tráfego é chamado *Interfloor*. Na prática estes padrões podem não ser observados exatamente como é apresentado na Figura 2-4, uma vez que algumas indústrias adotam um regime de atendimento flexível no tempo, em que a chegada e a

saída de trabalhadores ocorrem de maneira distribuída. Contudo, serve como um modelo para discussão (Barney, 2003).



**Figura 2-4-** Fluxo de passageiros em um prédio.

### 2.2.5 Gerador de tráfego de passageiros

Modelar o fluxo de tráfego de passageiros é fundamental para a realização de um simulador de tráfego de elevadores. Apresentar a forma como chegam as pessoas ao sistema de elevadores é de extrema importância para o correto funcionamento de um EGCS. O componente que faz essa tarefa é o gerador de tráfego de passageiros, o qual gera as chamadas de andar de origem e as chamadas de destino, de acordo com o tráfego presente. O controle de tráfego é um típico problema estocástico (Markon, 2002).

Usualmente, admite-se que a de chegada dos passageiros segue uma distribuição Poisson (Dos Santos, 1972) e sua distribuição de probabilidade é mostrada na equação (Zhifengetal, 2007):

$$P(n, T) = \frac{(\lambda T)^n e^{-\lambda T}}{n!}, \quad n = 0, 1, 2, \dots \quad (2-1)$$

Na equação,  $P$  é a probabilidade de chegada de  $n$  passageiros em um tempo  $T$ , onde a taxa de chegada de passageiros é  $\lambda$ . A variável  $n$  é o número de eventos (chegadas de passageiros). Prova-se que então o intervalo de tempo transcorrido entre as chegadas de passageiros segue uma distribuição exponencial (Hummet et al, 1978).

A equação mostra como gerar os tempos em que se esperam chegadas de passageiros com base em intervalos de tempo distribuídos segundo uma função de densidade de probabilidade exponencial.

$$t_0 = 0 \quad t_i = t_{i-1} - \frac{\ln(r)}{\lambda}, \lambda > 0 \quad (2-2)$$

Em que,  $t_0$  é o tempo inicial,  $r$  é um número aleatório com distribuição uniforme entre 0 e 1.

No caso dos grupos de elevadores  $\lambda$  é usualmente a taxa temporal de chegada de passageiros ao sistema como um todo, calculada em termos de um percentual da população em viagem, em um período de 5 minutos. O valor de  $\lambda$  em passageiros por segundo pode ser estimado conforme mostra a equação (Hummet et al, 1978).

$$\lambda = \frac{0.01(TI)}{300} \sum_i POP_i \quad (2-3)$$

em que  $TI$  (%) é a intensidade do tráfego vigente no momento (fornecida como uma porcentagem da população desejando viajar),  $POP_i$  é a população desejando viajar do andar  $i$ . Depois de calcular o tempo de chegada do passageiro seguinte, a próxima tarefa é estabelecer o andar de origem e o andar de destino da chamada. A atribuição de andares é realizada dependendo dos padrões de tráfego (*Up-peak*, *Down-peak*, *Interfloor*, etc.). A este respeito, são considerados três parâmetros, enumerados abaixo:

- **A:** Porcentagem do total da população em viagem no térreo(*Up-peak*).
- **B:** Porcentagem do total da população em viagem que tem como andar destino o térreo (*Down-peak*).
- **C:** Porcentagem do total da população em viagem que tem como origem e destino andares diferentes do térreo (*Interfloor*).

Com base neste tipo de classificação são desenvolvidos outros tipos de padrões modificando os parâmetros básicos. é apresentada uma lista dos padrões mais comuns e os valores de seus respectivos parâmetros (Hummet et al, 1978).

**Tabela 2-2 - Padrões de tráfego de elevadores (adaptado de Hummet et al, 1978)**

<b>Padrão</b>	<b>Parâmetros</b>		
	<b>A</b>	<b>B</b>	<b>C</b>
<i>Up-peak</i>	90	5	5
<i>Interfloor</i>	45	45	10
<i>Lunch</i> (Saída Almoço)	20	60	20
<i>Lunch</i> (Entrada Almoço)	70	10	10
<i>Down-peak</i>	5	90	5

Para a geração do andar de origem e o andar de destino é necessário conhecer a população em viagem do prédio, para o que se constrói um vetor de densidade de ocupação do prédio, cujos elementos correspondem aos percentuais de população, em relação à população em viagem, existentes em cada andar. Conseqüentemente, a dimensão do vetor baseia-se no número de andares existentes. Além disso, é necessário construir-se uma matriz de destino, cujas linhas são compostas por elementos  $ij$  que correspondem aos percentuais da população em viagem do andar  $i$  que desejam ir para o andar  $j$ , considerando que a população em viagem do andar  $i$  deseja sair deste com destino aos demais andares  $j$ .

No cálculo destes parâmetros precisam-se obter os valores proporcionais da população para cada andar, dependendo do tipo de tráfego a simular, para o que é necessário utilizar a equação .

$$\rho(i) = \frac{POP_i}{\sum_{i=2}^N POP(i)} \quad i = 2,3, \dots, N \quad (2-4)$$

em que  $POP_i$  é a população em viagem no andar  $i$ .  $N$  é o número de andares do prédio. Por outro lado, para o cálculo do vetor de densidade são utilizadas as seguintes equações:

$$\left. \begin{array}{l} V(1) = A \\ V(i) = (B + C)\rho(i) \quad i = 2,3, \dots, N \end{array} \right\} \text{Vetor de densidade de população} \quad (2-5)$$

No calculo da matriz de destino se utiliza a equação para determinar os valores proporcionais da população para cada andar.

$$\rho(i,j) = \frac{POP_j}{\sum_{\substack{k=2 \\ k \neq i}}^N POP(k)} \quad (2-6)$$

Para o cálculo da matriz de destino também são utilizadas as seguintes equações (Aggarwal, 2004):

$$\begin{array}{l} OD(1,j) = \begin{cases} 0 & \text{Se } j=1 \\ 100\rho_j & \text{Outro caso} \end{cases} \\ OD(i,j) = \begin{cases} 0 & \text{Se } i=j \\ \frac{c}{B+C} 100\rho_{ij} & \text{Outro caso} \end{cases} \end{array} \quad OD(i,1) = \begin{cases} 0 & \text{Se } i=1 \\ \frac{B}{B+C} 100 & \text{Outro caso} \end{cases} \quad (2-7)$$



Após calcular o vetor de densidade  $V$  e a matriz de destino  $OD$  é necessário gerar os números aleatórios para os diferentes andares, tendo em conta a percentagem fornecida pelo vetor densidade. Isto é realizado, gerando um número randômico entre 0 e 100 e comparando-o com cada intervalo do vetor de densidade, desta maneira é selecionado o andar origem. O cálculo do andar destino é realizado por meio da matriz destino, de forma similar como se realizou o cálculo do andar origem. Finalmente, é gerado um arquivo com a informação necessária para simular o tráfego de passageiros, o arquivo terá os seguintes dados: tempo de chegada do passageiro, andar origem e o andar destino.

### **2.2.6 Métodos para coletar observações estatísticas**

Simulação é um experimento estatístico e seus resultados devem ser interpretados usando ferramentas adequadas de inferência estatística (por exemplo, intervalos de confiança e testes de hipótese). Para executar essa tarefa, as observações do experimento de simulação devem satisfazer três condições:

- 1) Observações devem ser retiradas de distribuições estacionárias (idênticas).
- 2) Observações devem ser amostradas de uma população com distribuição normal.
- 3) Observações devem ser independentes.

Porém, ocorre que, em sentido estrito, o experimento de simulação não satisfaz nenhuma dessas condições. Não obstante, pode-se garantir que essas condições permaneçam estatisticamente viáveis restringindo a maneira como são coletadas as observações de simulação.

Em primeiro lugar, deve-se considerar a questão da estacionariedade. O resultado da simulação é uma função da extensão do período de simulação. O período inicial produz um comportamento errático e usualmente denominado período transiente ou de aquecimento. Quando o sistema se estabiliza, funciona em estado de equilíbrio. Infelizmente não há maneira de prever com antecedência o ponto inicial do estado de equilíbrio. Em geral, uma corrida (*run*) de simulação mais longa tem melhor chance de alcançar o estado de equilíbrio.

Em seguida considera-se o requisito de que as observações de simulação devem ser retiradas de uma população com distribuição normal. Existem muitas provas estatísticas para verificar a distribuição normal de um conjunto de dados. Entre as mais conhecidas têm-se: Shapiro &Wilk, Anderson-Darling, Darling-Pearson, Kolmogorov-Smirnov, Lilliefors e  $X^2$  (Gonçalves, 2002).

A terceira condição trata da independência das observações. A natureza do experimento de simulação não garante independência entre observações de simulações sucessivas. Entretanto, existem vários métodos para coletar observações em simulação, tais como: o método do intervalo, o método regenerativo e o método da replicação. O método da replicação é representado por uma rodada de simulação independente na qual o período transiente é truncado. A vantagem do método da replicação é que cada rodada de simulação é dirigida por uma corrente distinta de números aleatórios (0,1), o que resulta em observações que são verdadeiramente independentes em termos estatísticos (Taha, 2007).

### **2.2.7 – Tempo de espera médio de passageiros**

O tempo de espera médio é o período de tempo médio, em segundos, que um passageiro gasta esperando um elevador, medindo desde o momento que o passageiro realiza a chamada do elevador no andar, até que o passageiro pode entrar no carro do elevador.

Atualmente o tempo de espera poderia ser o melhor indicador de qualidade de serviço que um sistema de elevadores poderia fornecer, menor tempo, melhor serviço. Os passageiros tendem a ficar chateados se eles esperarem por muito tempo a ser atendidos. Entretanto, o tempo de espera de passageiros não pode ser facilmente medido. Alguns projetistas, em consequência usam o *intervalo de tempo (I)* de chegada do elevador ao primeiro andar como indicador de qualidade de serviço. O intervalo de tempo faz parte da avaliação da capacidade de transporte, o qual simplesmente determina a quantidade de serviço. Em forma geral, o intervalo de tempo pode ser usado para indicar a qualidade provável de serviço, quando consideramos prédios de escritórios com um intervalo de (Barney, 2003):

- 20 s. ou menos poderia indicar um excelente serviço.
- 25s. poderia indicar um bom serviço.

- 30s. poderia indicar um serviço satisfatório.
- 40s. poderia indicar um serviço deficiente.
- 50s. ou mais poderia indicar um sistema inaceitável.

## 2.2.8- Dimensionamento de um sistema de elevadores

O sistema de grupo de elevadores abrange diversos conjuntos de variáveis como: o número de elevadores, sua velocidade e capacidade de carga, as quais são projetadas para atender uma alta de demanda de passageiros (*Up-peak*) e procurar uma boa qualidade de serviço. Alguns autores afirmam, que se os sistemas de elevadores podem controlar situações de tráfego altas (*Up-peak*), também pode controlar outras situações de tráfego presente no prédio (Jong e Siikonem, 2001). Segundo Barney (2003), o planejamento de sistema de elevadores compreende dois critérios muito importantes: Capacidade de Transporte (*HC*) e Intervalos de tempo (*I*), os quais são usados para projetar qualquer tipo prédio. Estes critérios estão padronizados no mundo. A Capacidade de Transporte é um critério que ajuda a ajustar o tamanho da cabine do elevador e a área da caixa<sup>1</sup>. O intervalo de tempo permite calcular o número de elevadores em um sistema de grupo de elevadores. Para calcular a Capacidade de carga e o Intervalo do tempo é necessário usar a equação chamada *RTT* (Tempo de viagem de ida e volta) o qual, é calculado de acordo com o número de paradas durante um percurso de ida e volta desde o andar térreo até o último andar do prédio, considerando uma demanda alta de passageiros (*Up-peak*). A equação *RTT* depende de três conjuntos de dados em relação aos dados do prédio, o sistema de elevadores e os passageiros (Barney, 2003).

$$RTT = \underbrace{2Ht_v + (S + 1)t_s}_{\text{Dados do prédio e do sistema de elevadores}} + \underbrace{2Pt_p}_{\text{Dados dos passageiros}} \quad (2-8)$$

Na equação, o valor de *H* pode ser o número de andares do prédio. O valor de *H* também pode ser calculado usando teorias de probabilísticas, as quais dependem do

---

<sup>1</sup>caixa: nome dado ao local no interior do qual a cabina do elevador se desloca.

numero de andares do prédio e o numero de passageiros.  $S$  é o numero de paradas esperadas no prédio, método publicado inicialmente por Jones Basset em 1923, baseado em leis probabilísticas. Os valores de  $S$  e  $H$  podem ser encontrados em uma tabela oferecida pelos fabricantes de elevadores e baseada nas regras de segurança para a construção de instalação de elevadores na BS EN81 e BS ISO4190 (anexo B) e  $P$  é o número médio de passageiros dentro da cabine do elevador. Geralmente, trabalha-se com 80% da capacidade total de carga do elevador. A variável  $t_v$  é o tempo em percorrer um andar com velocidade nominal  $V$ ,  $t_s$  é o tempo que demora parado um elevador incluindo o tempo de abrir e fechar portas e o tempo de aceleração e desaceleração,  $t_p$  é o tempo que demora um passageiro em entrar ou sair do elevador. A Capacidade de Transporte apresenta a quantidade de passageiros que o elevador pode transportar desde o andar térreo até andares superiores em um tempo de 5 minutos ( $HC$  é comumente conhecido como um indicador da qualidade do serviço) (Barney, 2003).

$$HC = \frac{(0,8 \text{ cc } L)}{RTT} \quad (2-9)$$

A equação mostra como calcular  $HC$ , onde  $cc$  é capacidade de carga do elevador em pessoas e o  $L$  é numero de elevadores.

Outro parâmetro também importante é o tempo médio entre as chegadas sucessivas de um elevador ao andar principal com carga em qualquer nível (Barney, 2003). Esse intervalo de tempo, aqui denotado por  $I$ , é calculado por meio da equação ..

$$I = \frac{RTT}{L} \quad (2-10)$$

Para fins de indicar o grau de demanda atuante sobre o grupo de elevadores, define-se outra variável, chamada de *Percentual de população servida*,  $PPS$ . Esta é calculada como um valor percentual entre a demanda efetiva de transporte em um tempo de 5 minutos e a população em viagem do prédio. Na literatura (Barney, 2003), essa variável é denominada de  $\%HC$  e seu cálculo é mostrado na equação .

$$\%HC = \frac{HC}{\text{População em viagem do predio}} \times 100 \quad (2-11)$$

Observa-se nesta que o numerador é a Capacidade de Transporte,  $HC$ , conforme definição explicitada anteriormente neste texto (vide equação ). Entretanto, a Capacidade de Transporte definida anteriormente é um parâmetro fixo, dependente das características do prédio. Portanto, para evitar interpretações equivocadas em relação ao

valor %*HC*, que neste caso teria valor também constante, utilizar-se-á neste texto uma adaptação da equação , em que se substituirá o numerador *HC* pela variável *Demanda Efetiva de Transporte, DET*, medida em passageiros por 5 minutos. Consequentemente, a equação assume a forma explicitada na equação .

$$PPS = \frac{DET}{\text{População em viagem do prédio}} \times 100 \quad (2-12)$$

Algumas recomendações úteis de desempenho típicas para diferentes tipos de construção são apresentadas na seguinte (Barney, 2003).

**Tabela 2-3** - Desempenho recomendado para um bom serviço em diferentes tipos de prédios (adaptado de Barney, 2003)

<b>Tipo de Prédio</b>	<b>% HC (%população/5 minutos)</b>	<b>Intervalo de tempo (s)</b>
Hotel	10-15%	30-50
Residencial	5-7%	40-90
Hospitais	8-10%	30-50
Escolas	15-20%	30-50
Escritórios (Múltiplos arrendamentos)		
Normal	11-15%	25-30
Prestigio	17%	20-25
Escritórios (Únicos arrendamentos)		
Normal	15%	25-30
Prestigio	17-25%	20-25

### **Prédios residenciais**

Nos prédios residenciais a intensidade de tráfego é baixa. Os tempos de espera até duas vezes maior do que em um prédio de escritórios são aceitáveis. Os elevadores para prédios residenciais podem ser escolhidos com normas ISO ou normas locais comparáveis. No Brasil existe a norma NBR10098 que padroniza elevadores elétricos destinados ao transporte de passageiros em edifícios residenciais, fixando dimensões e condições do projeto de construção necessárias à sua instalação (ABNT, 1987).

### **Hospitais e prédios de usos diversos**

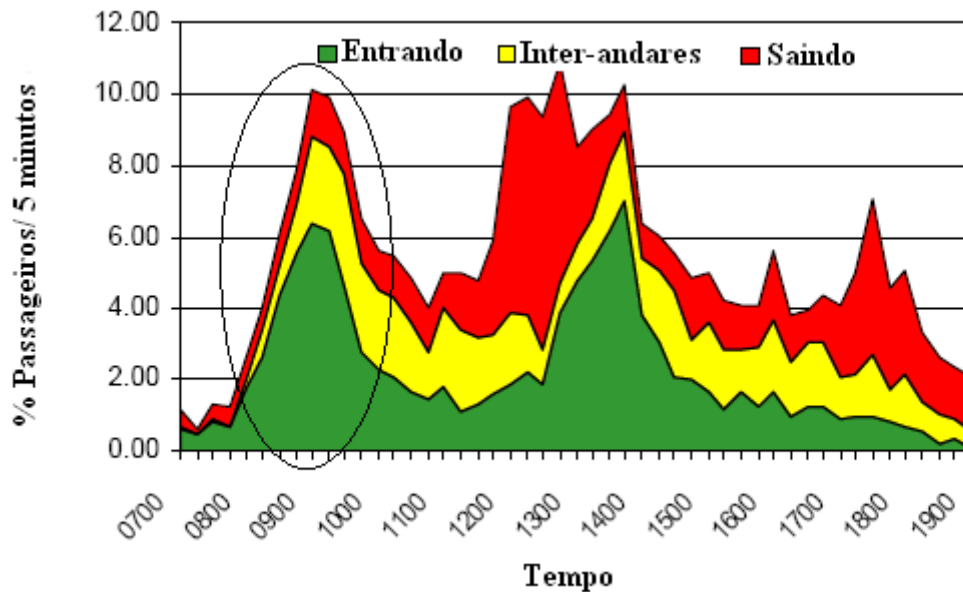
Nos hospitais, o sistema de elevadores precisa de um planejamento detalhado que deve considerar aspectos como a arquitetura e as necessidades especiais do edifício,

por exemplo, transporte de emergências, serviços, camas, pacientes, visitantes e funcionários, além de fatores como a colocação das salas de operações e a unidade de cuidados intensivos que afeta diretamente as disposições do transporte.

### **Prédios de Escritórios**

Os prédios comerciais de altura média são escritórios e hotéis. Nos hotéis a seleção do número de elevadores depende principalmente do número de quartos e camas. Geralmente são necessários elevadores adicionais para serviços diferentes ao transporte de passageiros. Em edifícios de escritórios existem três horas de tráfego intenso: tráfego *Up-peak* de entrada pela manhã, de horas de comida e tráfego *Down-peak* nas horas de saída no final da tarde.

Em prédios de grande altura, só um grupo de elevadores não satisfaz todas as necessidades de tráfego, por tanto distintos grupos de elevadores são divididos em zonas no edifício de forma que possam fornecer um serviço mais eficiente a todos os andares. Em edificações de 50 – 60 andares (50 a 200 metros), ou mais, são utilizados elevadores de cabine dupla ou blocos de elevadores um sobre outro viajando independentemente dentro da mesma caixa (ThyssenKrupp, 2006). A Figura 2-5 apresenta a porcentagem de passageiros presentes que entram, saem e que viajam em inter-andares em um prédio de escritórios, note-se que o cor verde representa a porcentagem de pessoas que entram no prédio, o qual na hora da manhã as 9:00 consegue um valor aproximado de 63% da população do prédio. A cor amarela representa inter-andares na mesma hora da manhã 9:00 consegue um valor aproximado de 22% da população no prédio. A cor vermelha representa os passageiros que saem do prédio, analisando na mesma hora da manhã 9:00 consegue um valor aproximado de 15% da população no prédio.



**Figura 2-5** - Fluxo de tráfego em prédios de escritórios durante o dia (modificado – Sorsa, 2002).

As pesquisas sobre o tipo e uso dos prédios são importantes no projeto de um sistema de elevadores, porém, os sistemas mais eficientes mantêm uma estimativa constante do fluxo de tráfego durante as horas operacionais do prédio com o qual é possível aplicar distintas estratégias de atendimento visando a diminuir o tempo de espera dos usuários (Siikonen, 1997). Estudos sobre a classificação do tráfego em prédios visam melhorar a qualidade do serviço. Diferentes métodos têm sido aplicados, entre eles o uso da inteligência artificial (Kim *et al.*, 1998), entretanto, métodos de predição de tráfego são objeto de estudo nos últimos anos (Luo *et al.*, 2005). Por outro lado, estudos sobre estratégias de atendimento de chamadas são realizados desde começos dos anos 60 (Siikonenb, 1997, Sorsa *et al.*, 2003).

### 2.2.9 Simulação e CAD

A simulação pode ser vista como o estudo do comportamento de sistemas reais por meio da análise do comportamento de modelos. Um modelo incorpora características que permitem aproximar o comportamento do sistema real.

Os benefícios da simulação de sistemas aliados à agilidade oferecida pelos meios computacionais têm sido largamente utilizados como ferramenta auxiliar na solução de problemas diversos. Justifica-se tal afirmação considerando que, com o uso de um computador, uma grande quantidade de eventos pode ser executada em curto espaço de

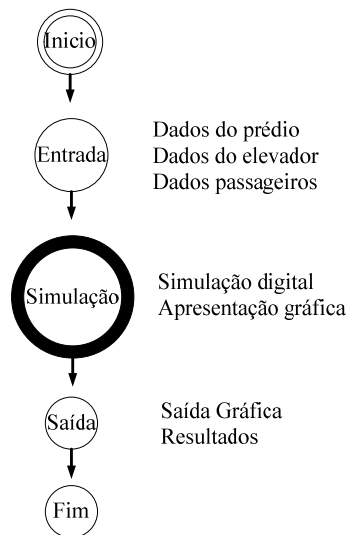
tempo. De modo geral, o uso da simulação é recomendado principalmente em dois casos: (a) quando a solução de problemas é muito cara ou mesmo impossível de ser obtida por meio de experimentos e (b) quando os problemas são muito complexos para tratamento analítico. Com o uso da simulação, principalmente quando se observam características estocásticas, sistemas podem ter seu comportamento representado com fidelidade e realismo. Modelos de simulação podem ser considerados como uma descrição do sistema real. O exercício (execução) de modelos de simulação em computador tem potencial para fornecer resultados precisos sem que seja preciso interferir no sistema real. Tais resultados, quando analisados estatisticamente, produzem informações que podem contribuir grandemente na tomada de decisões que visam à solução de problemas (Mello B., 2001).

Os sistemas CAD (Projeto Auxiliado por Computador) auxiliam em criação, modificação, análise ou otimização de um projeto computadorizado complexo. O software desses sistemas é baseado em interface gráfica orientada ao usuário. Podem ser desenvolvidos projetos mecânicos, elétricos, eletrônicos, de engenharia civil, aeronáutica, naval etc. A necessidade do CAD no projeto de sistema de elevadores foi previsto por Jackson (1970), que escreveu:

*“Uma necessidade real...é um programa de computador para simular o provável desempenho de uma proposta de sistema de elevadores... Diferentes números, velocidades e grupos de elevadores poderiam ser considerados, assim como diferentes sistemas de controle.. Os resultados mostrariam o desempenho de várias propostas.... e permitiria... decisões racionais.”*

O sistema CAD de um sistema de tráfego de elevadores apresenta os seguintes padrões (Barney, 2003):





**Figura 2-6-** Etapas CAD no sistema de tráfego de elevadores (adaptado de Barney, 2003)

Os dados de entrada deste tipo de sistema incluem dados do prédio, dados dos elevadores e dados dos passageiros, como apresentado na Figura 2-6. O número de andares e a distância entre andares são dados do prédio. O número de elevadores, capacidade, velocidade, aceleração, tempos de abrir e fechar portas e o tipo algoritmo de controle de tráfego são dados dos elevadores. O andar de origem, o andar de destino e o tempo de chegada de passageiros são dados dos passageiros. Depois de configurar no sistema os dados necessários para seu funcionamento, a fase de simulação inicia e apresenta de uma forma visual o comportamento do sistema de controle de tráfego. Finalmente, os resultados da simulação (tempos de espera, tempos de vôo, numero de paradas, etc.) podem ser apresentados por meio de uma interface gráfica.

### 2.2.10-Sistemas modernos de grupo de elevadores

O Sistema de Controle de Grupo de elevadores tem um grande efeito na viabilidade e utilidade de prédios, especialmente em prédios modernos. Atualmente, existem muitos trabalhos em que o principal objetivo é o desenvolvimento de melhoras na capacidade de transporte. Portanto, os mais recentes sistemas de elevadores possuem estruturas mais modernas que as usadas nos sistemas tradicionais de botões de chamada para acima e para baixo. Estas estruturas são as seguintes: Sistema de elevadores *Double-deck*, sistema de elevadores com Sistema de Controle de Destino (SCD) e

Sistema de Elevadores *Multi-car* (Markon et al, 2008). As propostas de modernas estruturas para melhorar a capacidade de transporte de sistema de grupo de elevadores são projetadas para não aumentar o espaço nos prédios.

### **Sistema de elevadores *Double-deck***

O sistema está composto por duas cabinas de elevadores que se movimentam em uma mesma caixa, pode-se empilhá-los uns aos outros e conduzi-los por uma mesma unidade de tração. Este moderno sistema propõe muitos desafios para os projetistas de controle, porque atinge varias limitações e objetivos de controle específicos. A forma de operação mais usada é atender o andar par e impar exclusivamente para quando o elevador esta subindo ou esta descendo. Desta maneira, é possível atender dois andares simultaneamente e diminuir o número de paradas (Markon et al, 2008).

### **Sistema de elevadores *Multi-car***

Este sistema é um desenvolvimento revolucionário, totalmente diferente aos encontrados hoje dia em qualquer tipo de prédio, se espera que o futuro dos sistemas de elevadores seja o sistema *Multi-car* (Markon et al, 2008). A proposta deste sistema é operar vários elevadores independentes em uma mesma caixa. O sistema requer uma nova e completa metodologia para controle, devido à potencialidade de trabalhar com vários elevadores e a dificulta de programação de atendimento. Atualmente, o sistema *Multi-car* é um importante problema aberto de pesquisa.

### **Sistema de Controle de Destino (SCD)**

A atual evolução tecnológica permite a implementação de um método de distribuição de chamadas mais sofisticado e eficiente. Isto é possível colocando IHM (Interface Homem Maquina) em cada pavimento. Ao invés de utilizar uma botoeira para realizar a chamada, o usuário interage com a interface IHM, através da qual ocorre uma troca de informações. O usuário informa ao terminal seu pavimento de destino e o terminal mostra o elevador que irá transportá-lo. Já no elevador selecionado, o passageiro é levado ao andar de destino. Não há botoeira dentro da cabine, visto que a chamada de destino foi efetuada antes de entrar no elevador. Sabendo antecipadamente

o destino do usuário, é possível colocar passageiros com o mesmo destino no mesmo elevador, possibilitando agrupar passageiros em um mesmo carro, de modo a diminuir o número de paradas.

## **2.3- TECNOLOGIAS DE AUTOMAÇÃO INDUSTRIAL**

### **2.3.1 - Controlador Lógico Programável (CLP)**

CLP para a IEC (*International Electrotechnical Commission*) é definido como: "*Sistema eletrônico operando digitalmente, projetado para uso em ambiente industrial, que usa uma memória programável para a armazenagem interna de instruções orientadas para o usuário para implementar funções específicas, tais como lógica, seqüenciamento, temporização, contagem e aritmética, para controlar, através de entradas e saídas digitais ou analógicas, vários tipos de máquinas ou processos. O controlador programável e seus periféricos associados são projetados para serem facilmente integráveis em um sistema de controle industrial e facilmente usados em todas as funções previstas.*"

### **2.3.2 – Ethernet**

A rede Ethernet foi desenvolvida pela Xerox no seu Centro de Pesquisa Palo Alto (PARC) nos anos 70. É uma rede mundialmente utilizada para a conexão de computadores pessoais e foi um grande desafio levar a Ethernet para a indústria e torná-la uma das redes de maior crescimento no setor.

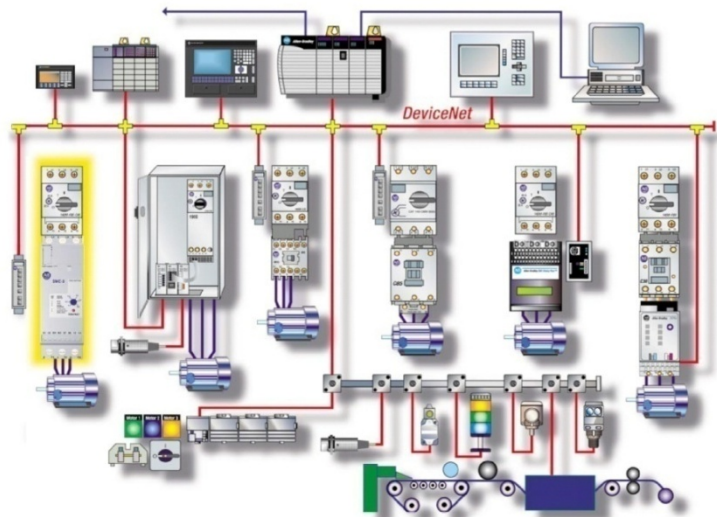
Além disso, o padrão Ethernet é um dos mais populares e difundidos nas redes corporativas instaladas atualmente. Ao contrário dos protocolos industriais como MODBUS e PROFIBUS que são determinísticos, no padrão Ethernet ocorrem colisões de dados na rede, tornando o tempo de resposta não determinístico. Isto, do ponto de vista de automação, não é recomendável, pois a falta de definição do tempo de resposta de uma informação pode comprometer o desempenho do sistema que está sendo controlado (BOARETTO, 2005).

O protocolo Ethernet não foi concebido para aplicações em automação industrial, não apresentando algumas características desejáveis em ambientes de controle em tempo real, como determinismo e segurança na transmissão dos dados. No entanto, Ethernet é provavelmente a tecnologia de rede mais difundida, que permite uma

grande escala de produção e disponibilidade, e, por conseqüência, baixo custo, tornando-se uma alternativa bastante atrativa para interconexão de dispositivos de automação (Seixas Felho, 2003).

### 2.3.3 - DeviceNet®

A rede DeviceNet classifica-se como uma rede de dispositivo, sendo utilizada para interligação de equipamentos de campo, tais como sensores, atuadores, AC/DC drives e CLP's. Esta rede foi desenvolvida pela Allen Bradley sobre o protocolo CAN (*Controller Area Network*) e sua especificação é aberta e gerenciada pela ODVA (*Open DeviceNet® Vendors Association*). É utilizada principalmente em processos de automação industrial e em robótica, sua principal característica é que permite a conexão e desconexão de dispositivos da rede com o sistema em funcionamento. Além disso, é possível operar 64 nodos selecionando-se diferentes velocidades de transmissão de 125 kBd, 250kBd ou 500 kBd permitindo desta maneira atingir diferentes longitudes de rede. O mecanismo de comunicação é *peer to peer* com prioridade. A transferência de dados se dá segundo o modelo produtor consumidor. Para a construção de uma rede simples DiveciNet® é necessário um scanner (um computador ou CLP com um modulo DeviceNet®), Varios cabos para realizar a conexão dos módulos DeviceNet® e uma fonte de alimentação de 24 v DC. Na Figura 2-7 é apresentado uma rede DeviceNet® usada em uma aplicação industrial onde mostra a arquitetura da rede (Rockwell, 2010).



**Figura 2-7-** Aplicação da rede DeviceNet® - (Rockwell, 2010)

### **2.3.4- A tecnologia OLE/COM**

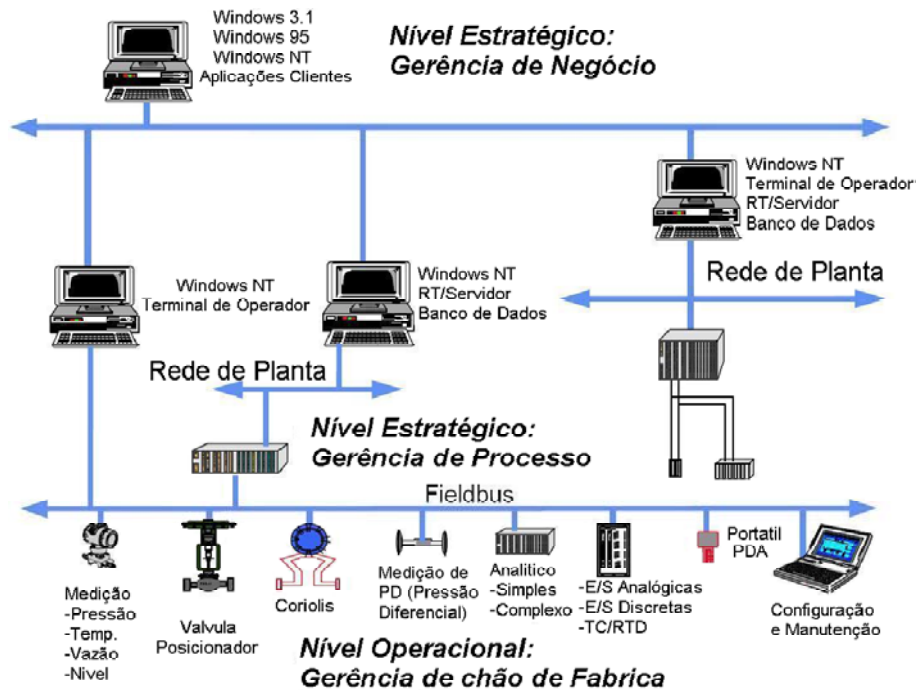
A tecnologia OLE 1.0 foi desenvolvida pela Microsoft em meados de 1990, para suprir a necessidade de se integrar diferentes aplicações dentro da plataforma Windows, de forma a solucionar os problemas de desempenho e confiabilidade do até então utilizado padrão DDE (Fonseca, 2002). Foram introduzidos dois conceitos: *Linking*, o qual cria vínculos ou referências aos objetos, armazenando no documento principal apenas os dados realmente necessários para exibir, imprimir, etc. *Embedding*, o qual incorpora os dados dos objetos ao documento principal. Neste contexto, surgiram os conceitos de objeto vinculado e de objeto incorporado: Objeto vinculado são informações (objetos) criadas em um arquivo (arquivo origem) e inseridas em outro arquivo (arquivo destino).

Embora o objeto vinculado não se torne parte do arquivo de destino, existe um vínculo, uma conexão entre os dois arquivos de forma que o objeto vinculado no arquivo de destino seja automaticamente atualizado quando o arquivo de origem é atualizado. Objeto incorporado consiste de informações inseridas em um arquivo de destino, ao ser incorporado, o objeto se torna parte do arquivo (Duarte *et al.*, 2006). Ao pressionar duas vezes no objeto incorporado, ele é aberto no programa de origem em que foi criado. Qualquer alteração feita no objeto incorporado se refletirá no arquivo de destino. Outro conceito importante na tecnologia OLE é o conceito de Cliente Servidor, cliente é uma aplicação que solicita os dados e servidor é uma aplicação que disponibiliza os dados. Segundo Duarte *et al.* (2006), houve muitas melhorias na tecnologia OLE 2.0, a mais importante é a Automação OLE, pois permite que uma aplicação seja controlada por outra aplicação. A tecnologia OLE é montada sobre a tecnologia COM que define um modo padronizado para a comunicação dos módulos cliente e servidor por meio de uma interface específica. Um módulo indica um aplicativo ou uma biblioteca (uma DLL – *Dynamic Link Libraries*). Os dois módulos podem ser executados no mesmo computador ou em máquinas diferentes conectadas através de uma rede.

### **2.3.5- A tecnologia OPC**

Segundo Anwar *et al.* (2004), a motivação para o desenvolvimento de OPC (*OLE for Process Control*) foi ter um padrão para comunicação de várias fontes de

dados, dispositivos de campo, ou banco de dados na sala de controle. A arquitetura dessas fontes de dados na indústria de processo é mostrada na Figura 2-8 e envolve três níveis:

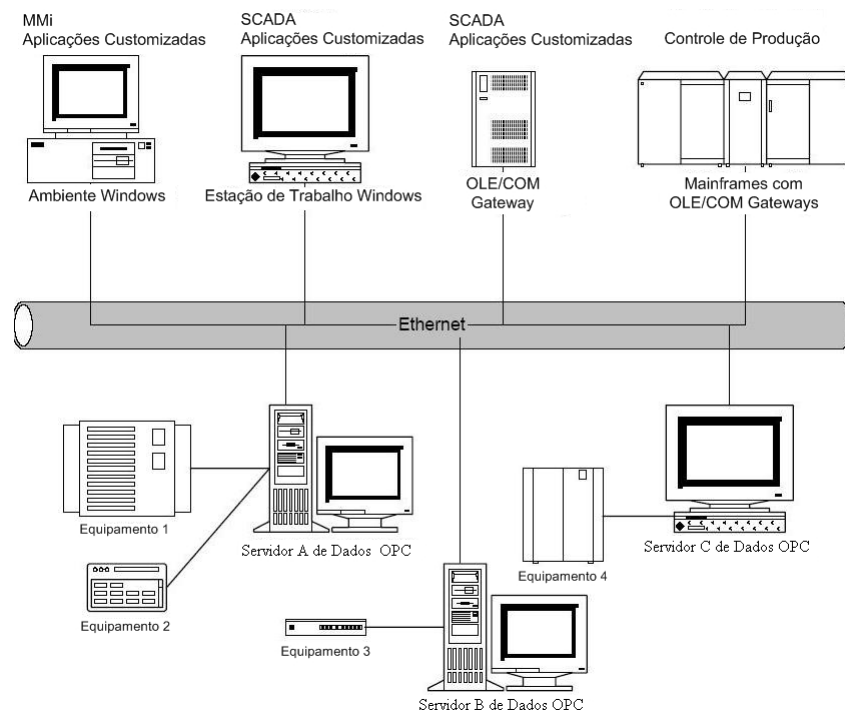


**Figura 2-8-** Arquitetura da informação no controle de processos (Amaya, 2008)

1. Gerência de chão de Fábrica: com a aparição de dispositivos de campo inteligentes, a quantidade de informação pode ser usada para avaliação de saúde dos dispositivos de campo. Estas informações são adquiridas a partir dos dispositivos, parâmetros de configuração, materiais de construção, etc. Toda a informação tem que ser apresentada ao usuário de uma maneira consistente.
2. Gerência de Processo: com a instalação de sistemas de controle distribuído e sistemas SCADA (*Supervisory Control And Data Acquisition*) para monitorar e controlar os processos de manufatura, disponibilizando eletronicamente os dados.
3. Gerência de Negócio: muitos benefícios podem ser ganhos com a instalação de sistemas integrados de informação dentro do sistema empresarial, administrando aspectos financeiros do processo de manufatura.

Segundo Shimanuki (1999), para acessar informações de um equipamento através de uma aplicação deve ser desenvolvida uma interface customizada ou um

*driver* de comunicação. Muitas destas aplicações não conseguem acessar as informações devido à inconsistência entre fabricantes de *drivers* e *hardwares*. Na busca de uma solução para esse problema, foi desenvolvida a tecnologia OPC, que é uma tecnologia para conectar aplicações *Windows* e equipamentos de controle de processos. O OPC é um protocolo de comunicação aberto que permite um método consistente de acesso aos dados de inúmeros equipamentos dos mais diversos fabricantes (Figura 2-9). O método é o mesmo, independente da origem dos dados, o que vem oferecer ao usuário final uma maior liberdade na escolha dos equipamentos independentemente do fabricante.

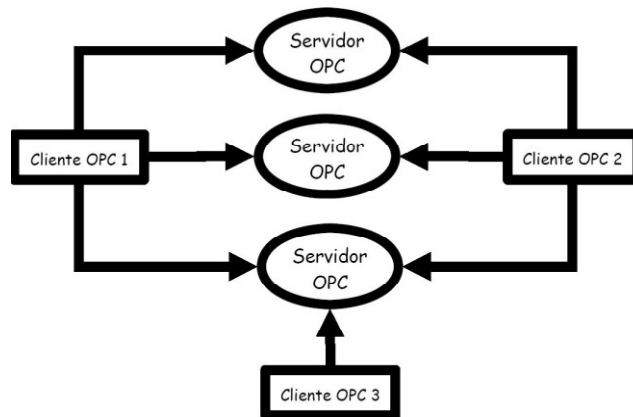


**Figura 2-9-** Aplicações com vários servidores OPC (Tonaco, 2008)

O OPC é construído usando tecnologia *Microsoft* OLE/COM, mas a especificação OPC foi desenvolvida por uma fundação aberta, a *OPC Foundation*, para atender as necessidades gerais da indústria e não as necessidades específicas de alguns fabricantes de *hardware* e *software* (OPC Foundation, 2010). A especificação ainda prevê a evolução das funcionalidades ao longo do tempo e por isso, os componentes OPC podem se manter no topo das necessidades emergentes da indústria.

Os componentes OPC se classificam em duas categorias: clientes OPC e servidores OPC. Um cliente OPC é tipicamente um sistema que gerencia dados tais como uma interface de operação ou um sistema SCADA. Um servidor OPC é uma fonte de dados que coleta ou gera dados a partir de um processo, disponibilizando-os aos

clientes OPC. O cliente OPC interage como servidor OPC usando uma interface definida. Qualquer cliente OPC pode se comunicar com qualquer servidor OPC, independentemente do tipo de dispositivo e do fabricante (Shimanuki, 1999), conforme está esquematicamente representado na Figura 2-10. Essa comunicação é válida somente para o OPC-DA (*Data Access*), uma vez que existem diferentes tecnologias OPC (Duarte *et al.*,2006).

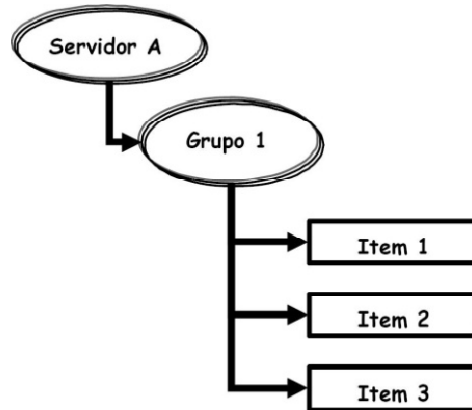


**Figura 2-10 -** Relação entre Clientes e Servidores (Amaya, 2008)

O padrão OPC estabelece as regras para que sejam desenvolvidos sistemas com interfaces padrões para comunicação dos dispositivos de campo (controladores, sensores, atuadores, etc.) com sistemas de monitoração, supervisão e gerenciamento SCADA, MES (*Manufacturing Execution Systems*), ERP (*Enterprise Resource Planning*), etc. Os três componentes básicos da arquitetura OPC apresentados na figura Figura 2-11 são: servidor, grupo e item. Do ponto de vista do cliente, um servidor é essencialmente uma estrutura de armazenagem para grupos que, por sua vez, têm como função básica o armazenamento de itens. Esses itens, elementos mais simples na especificação, representam conexões a pontos de entrada ou saída. Assim, o item OPC não é um valor, mas apenas um meio de acesso a um valor. Desta forma, uma única variável de entrada ou saída pode ser representada por itens diferentes, com propriedades distintas e compartilhada por mais de um cliente. A tarefa dos grupos é juntar o conjunto de itens que interessam a um determinado cliente, assumindo o papel principal na interação cliente-servidor. Os grupos também são responsáveis por satisfazer pedidos de leitura e escrita, bem como por enviar atualizações para seus clientes, periodicamente ou por exceção. Essas transações de atualização podem ser ativadas ou desativadas no grupo ou nos itens individuais. Os grupos presentes em um servidor OPC são normalmente definidos pelos clientes, e somente o cliente criador do



grupo pode acessá-lo; tal tipo de grupo é dito privado. Em alguns casos, porém, pode ser interessante que o servidor ofereça grupos passíveis de serem compartilhados por vários clientes. Quando essa capacidade é desejada, implementa-se a funcionalidade opcional dos grupos públicos.



**Figura 2-11-** Arquitetura OPC (Amaya, 2008)

Do ponto de vista do cliente, a função básica do servidor é prover uma infraestrutura de suporte aos grupos. Além disso, cabe também a ele gerenciar aspectos relacionados à conexão com uma fonte de dados, tais como parâmetros de comunicação ou taxa máxima de amostragem. Outra responsabilidade do servidor é implementar uma estrutura de endereçamento capaz de associar itens com variáveis reais (Souza *et al.*, 1998). O item é uma estrutura a qual estão associadas três propriedades (Fonseca, 2002):

1. *Value*, último valor armazenado pelo servidor no cachê de memória do item e que é atualizado sempre que o servidor faz uma leitura no dispositivo.
2. *Quality*: informação de estado que define a qualidade do dado que pode ser: *Good*, dado válido, *Bad*, perda do *link* de comunicação com o dispositivo de campo, e *Uncertain*, no caso de existir o *link* e o dispositivo de campo estiver fora de comunicação.
3. *Time Stamp*: data e hora em que o item é adquirido.

## 2.4–UTILIZAÇÃO DA MODELAGEM UML

A Linguagem de Modelagem Unificada (UML) é uma linguagem de modelagem voltada para sistemas orientados a objetos. A UML possibilita a visualização de todos os passos de desenvolvimento de um problema de modelagem na forma de diagramas padronizados, envolvendo elementos estruturais (classes, objetos, relacionamentos, etc.), de comportamento (casos de uso, iteração, máquinas de estado), de agrupamento (pacote, modelo, subsistema, etc.), de relacionamento (composição, agregação, generalização). A grande importância da modelagem e simulação é o fato de economizar custos na medida em que erros podem ser eliminados ainda na fase de projeto do sistema. No caso de sistemas de engenharia, a adoção de técnicas modernas de modelagem provenientes da indústria de software tem sido acentuada nos últimos anos.

No caso da Orientação a Objetos (OO) podem-se citar os seguintes fatos que descrevem o aumento da sua importância e impacto: (a) a denominada crise do software acontecida no final dos anos sessenta e sua primeira resposta na indústria e na academia que foi o projeto estruturado de sistemas e a programação estruturada. (b) Tanto o projeto estruturado como a programação estruturada não foram suficientemente apropriadas para resolver o problema porque as mesmas não seguiram uma metodologia unificada, levando a dificuldades para abordar a implementação diretamente a partir da modelagem do projeto; (c) a proposta de orientação a objetos veio a ser uma solução mais apropriada, pois permitiu unificar os processos de modelagem de projeto e a sua implementação em conceitos padronizados e similares (Pressman, 2005).

No contexto de sistemas orientados a objeto, a metodologia proposta fez possível vislumbrar a possibilidade de gerar automaticamente implementações de sistemas (em linguagens de programação adequadas) a partir da especificação dos projetos em alto grau de abstração (Projeto Orientado a Objetos – POO). Adicionalmente, a descrição dos sistemas em um grau de abstração bem perto do mundo real (dos objetos) permitiu a criação de ambientes de simulação que permitem a validação dos sistemas mesmo antes da sua implementação real.

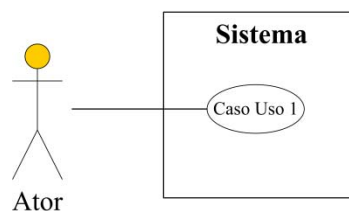
Por outro lado, o formalismo por traz da proposta de modelagem orientada a objetos trouxe a atenção para os engenheiros de projeto, tomando vantagem das propostas feitas no âmbito pelos engenheiros de software. Neste contexto, o POO é um exemplo que mostra como uma metodologia desenvolvida pelos engenheiros de

software e que foi inspirada nas técnicas de projetos de manufatura modernos, vem a ser adotada pelos projetistas de sistemas (incluindo o projeto de hardware, sistemas mecatrônicas, entre outros).

Neste contexto, a linguagem UML representa um padrão para descrever e representar sistemas diversos, tendo em conta suas características estruturais e funcionais (Rumbaugh, et al 1999). Para modelar um sistema de POO geralmente são realizados os seguintes diagramas:

### Diagramas de casos de uso

A modelagem de um diagrama de casos de uso é uma técnica usada para descrever e definir os requisitos funcionais de um sistema. Eles são escritos em termos de atores externos, casos de uso e o sistema modelado. Os atores representam o papel de uma entidade externa ao sistema como um usuário, um hardware, ou outro sistema que interage como o sistema modelado. O caso de uso é uma sequência de cenários de interação mostrado como eventos iniciais. Um cenário é uma narrativa de uma parte do comportamento global do sistema e uma coleção completa de cenário é usada para especificar completamente um sistema. Na Figura 2-12 é apresentada os componentes de um diagrama de casos de uso (Bezerra, 2002).

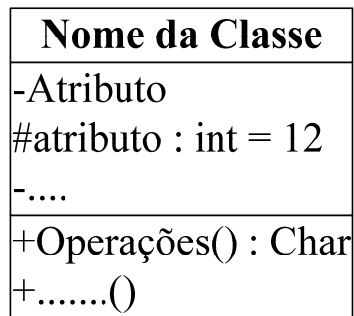


**Figura 2-12-** Componentes de um diagrama de caso de uso

### Diagramas de classe

O diagrama de classes demonstra a estrutura estática das classes de um sistema onde estas representam as “coisas” que são gerenciadas pela aplicação modelada. Uma classe é representada por um retângulo sólido com três partes: Uma parte com o *Nome* da classe, outra para os atributos da classe e a terceira para a declaração das operações

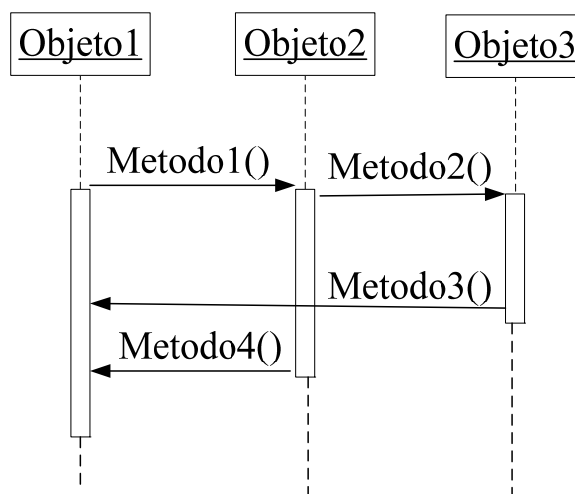
definidas para a classe. A seguinte Figura 2-13 apresenta a notação UML para classes (Booch, 1994).



**Figura 2-13** - Notação para classe em UML

### Diagrama de seqüência

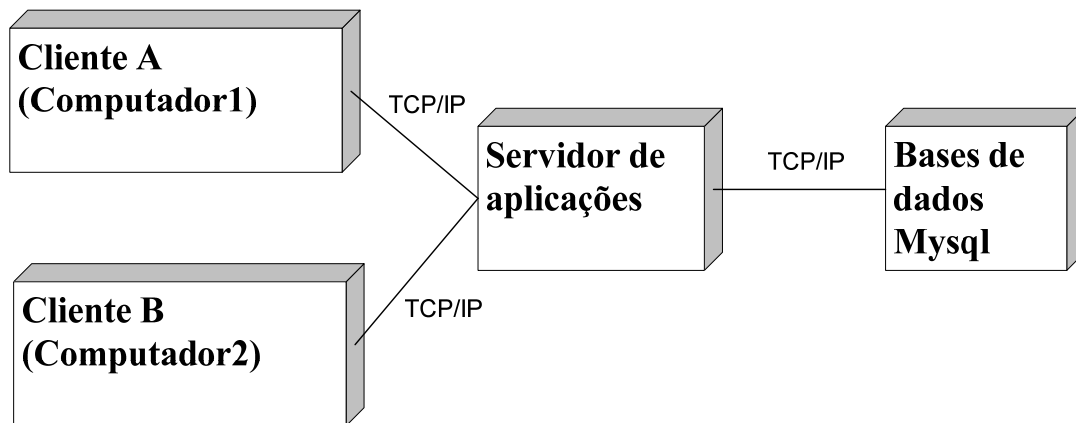
Um diagrama de seqüência mostra a colaboração dinâmica entre os vários objetos de um sistema. Os mais importantes aspectos deste diagrama são que a partir dele percebe-se a seqüência de mensagens enviada entre os objetos. Ele mostra a interação entre objetos, alguma coisa, que acontecerá em um ponto específico da execução do sistema. O diagrama de seqüência consiste em um número de objetos mostrado em linhas verticais. O decorrer do tempo é visualizado observando-se o diagrama no sentido vertical de cima para baixo. O diagrama possui dois eixos: o eixo vertical, que mostra o tempo e o eixo horizontal, que mostra os objetos envolvidos na seqüência de certa atividade. A seguinte Figura 2-14 apresenta a notação UML do diagrama de seqüência.



**Figura 2-14**- Diagrama de seqüência

## Diagrama de implantação

O diagrama de implantação mostra a arquitetura física do hardware e do software no sistema. Pode mostrar os atuais computadores e periféricos, juntamente com as conexões que eles estabelecem entre si e pode mostrar também os tipos de conexões entre esses computadores e periféricos. O diagrama demonstra a arquitetura *run-time* de processadores, componentes físicos, e do software que rodam no ambiente onde o sistema desenvolvido será utilizado. É a última descrição física da topologia do sistema. A Figura 2-15 apresenta um exemplo de um diagrama de implantação “nodes” que fazem parte de um sistema de informação remota, onde tem dois clientes A e B, nos quais roda uma aplicação de forma que através de internet acessa a um servidor para imprimir informação de uma base de dados.



**Figura 2-15-** Diagrama de implantação de um sistema de informação remota

## 2.5- LÓGICA NEBULOSA

O desenvolvimento do sistema de controle de grupo elevadores realizado neste trabalho é baseado em lógica nebulosa. O uso desta possibilita a inclusão de vários objetivos que serão avaliados simultaneamente via regras de decisão, as quais são baseadas nos padrões de tráfego em prédios de escritórios. Desta maneira o sistema nebuloso avalia cada elevador e proporciona um valor de prioridade a cada elevador indicando aquele que se apresente como o mais conveniente para atender às chamadas dos pavimentos.

A lógica nebulosa (*fuzzy logic*) é uma técnica de IA baseada em conhecimento (sistemas especialistas), em que um especialista (humano) gera uma base de conhecimento na forma de um banco de regras e o usuário (humano) consulta, ou

valida, um sistema particular apresentando fatos à base de dados. Os sistemas incorporam a forma humana de pensar em um sistema de controle, porém, não pretendem entender ou explicar como é formado o raciocínio dedutivo do pensamento humano (Shaw e Godoy, 1999).

### **2.5.1 - Método de Modelagem Matemática**

O objetivo principal em engenharia de controle é obter um modelo idealizado do processo a controlar para descrever como ele reage para várias entradas. Tradicionalmente, os modelos expressam-se em forma de equações diferenciais ou equações de diferença e são utilizadas ferramentas matemáticas como a *Transformada de Laplace* ou a *Transformada z*. Algumas restrições são feitas de forma a se obter um modelo matemático simples, entre elas podemos mencionar simplificações de processos *lineares e invariantes no tempo*.

Estas simplificações comumente feitas no controle de processos produzem certas dificuldades no desenvolvimento de uma descrição matemática significativa e realista do sistema (Shaw e Godoy, 1999).

### **2.5.2 - Método Heurístico**

O método heurístico consiste em usar o “senso comum” na solução de problemas. Assim, a heurística considera realizar tarefas de acordo com a experiência prévia ou estratégias já freqüentemente utilizadas. Uma regra heurística associa conclusões (consequências) com condições (antecedentes ou fatos) e é construída como uma implicação lógica:

$$\text{SE } \langle \text{condição} \rangle \text{ ENTÃO } \langle \text{consequência} \rangle$$

A vantagem do método heurístico é que possibilita a construção de uma função, não matemática, entrada-saída descrita ponto a ponto, na qual a restrição de linearidade não é mais necessária. A solução heurística de problemas não garante a solução ótima, no entanto, permite uma grande redução de custo e tempo e pode ser aplicada sempre que um modelo matemático equivalente for muito difícil ou complexo de se obter, considerando a presença de parâmetros incertos, desconhecidos ou variantes no tempo (Shaw e Godoy, 1999).

### 2.5.3 - Conjunto Nebuloso

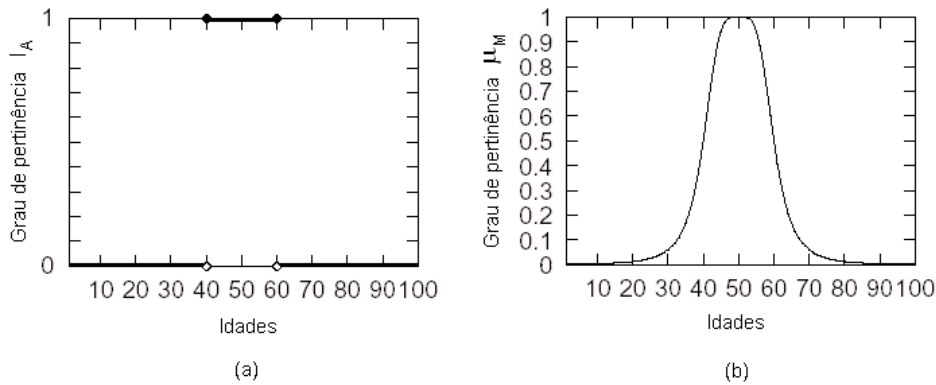
Na teoria clássica de conjuntos um elemento qualquer pode pertencer ou não a um determinado conjunto, havendo só duas opções, sendo definidos os limites nítidos da relação elemento-conjunto. De modo diferente, a teoria dos conjuntos nebulosos (*Fuzzy Sets*) foi introduzida em 1965 por Lotfi Zadeh, que afirmou que a transição desde “pertencer a um conjunto” até “não pertencer ao conjunto” é gradual (Zadeh, 1965 *apud* Shaw e Godoy, 1999).

Um elemento de um conjunto nebuloso não precisa, obrigatoriamente, pertencer ou não pertencer a este conjunto, podendo existir vários graus de pertinência, variando desde a completa exclusão (nível 0), até a pertinência máxima (nível 1). Quanto maior a pertinência de um objeto  $x$  com relação a um conjunto  $A$ , maior a compatibilidade deste objeto com a classe descrita por  $A$ . Assim, a definição de conjunto nebuloso generaliza a concepção de conjunto ordinário (Mamdani, 1975).

Seja  $X$  um conjunto de referência, também chamado de universo de discurso, expressado de forma genérica por  $x$ . Um conjunto nebuloso  $A$  em  $X$  é definido como um conjunto ordenado de pares e por uma *função de pertinência*  $\mu_A(x)$ , que associa, para cada ponto em  $X$ , um número real no intervalo  $[0,1]$  que representa o grau de pertinência de  $x$  em  $A$ .

$$A = \{(x, \mu_A(x)) / X \rightarrow [0,1]\} \quad \text{(2-14)}$$

A equação mostra de modo comparativo o conjunto ordinário ( $A$ ) e nebuloso ( $M$ ) que representam a idade de uma “pessoa de meia idade” (figuras a e b, respectivamente). Seja  $X = R^+$  as idades possíveis para um ser humano. Define-se uma pessoa de meia idade (como entre os 40 e 60 anos).



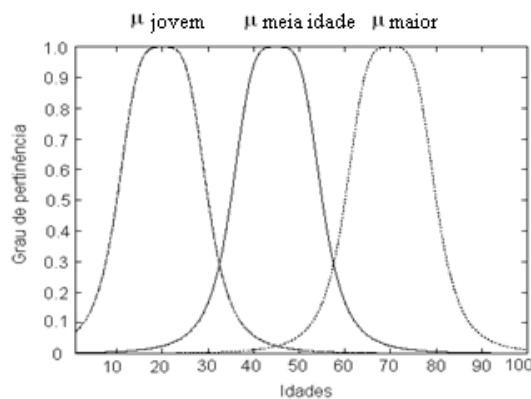
**Figura 2-16-** Conjunto Ordinário e Nebuloso: (a) função do conjunto ordinário A “meia idade” (b) função gaussiana do conjunto nebuloso “pessoa meia idade”.

O conjunto  $A$  não representa completamente o conceito de “pessoa de meia idade”, pois uma pessoa com 39 anos e 11 meses seria considerada completamente incompatível com este conceito. Já o conjunto nebuloso  $M$  permite concluir que qualquer pessoa com idade entre 40 e 60 anos enquadra-se na especificação “meia idade”, acima dos 80 anos e abaixo dos 20 anos não é considerada uma pessoa de meia idade e, no intervalo  $[20,40]$  (respectivamente  $[60,80]$ ), é considerado de meia idade quanto mais próximo de 50 for a sua idade.

$$M = \{(x, \mu_M(x))/x \in X\} \quad (2-15)$$

em que,  $\mu_M(x) = \frac{1}{1 + (\frac{x-50}{10})^4}$

Na prática, sempre que o universo  $X$  seja contínuo é possível particionar  $X$  em vários conjuntos de forma que as novas funções de pertinência  $\mu_M$  abarquem todo  $X$ . A Figura 2-17 mostra o mesmo universo  $X$  particionado em três funções de pertinência, representando três conceitos de idades diferentes: jovem, meia idade e maior (Shaw e Godoy, 1999).



**Figura 2-17 -** Particionamento do universo discurso (adaptada de Shaw e Godoy, 1999)



#### 2.5.4 - Número nebuloso e funções de pertinência

Um número nebuloso é um conjunto nebuloso sobre  $X$ . Embora qualquer função de pertinência seja válida, existem famílias de número com funções de pertinência parametrizáveis, alguns exemplos são relacionados a seguir (Shaw e Godoy, 1999):

- **Funções triangulares. Parâmetros  $\{a,b,c\}$**

$$Triang_{a,b,c}(x) = \max \left[ \min \left( \frac{x-a}{b-a}, \frac{c-x}{c-b} \right), 0 \right] \quad (2-16)$$

- **Funções trapezoidais. Parâmetros  $\{a,b,c,d\}$**

$$Trap_{a,b,c,d}(x) = \max \left[ \min \left( \frac{x-a}{b-a}, 1, \frac{d-x}{d-c} \right), 0 \right] \quad (2-17)$$

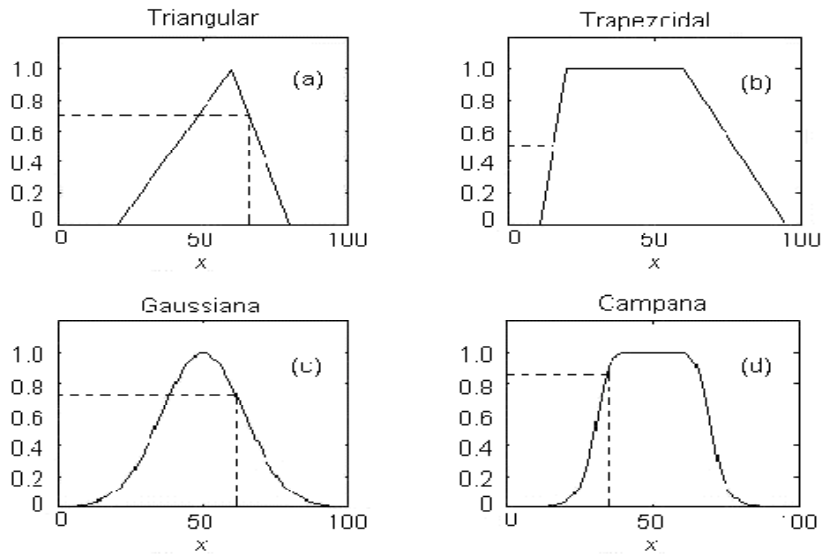
- **Funções gaussianas. Parâmetros  $\{\sigma,c\}$**

$$Guassiana_{\sigma,c}(x) = e^{-\left(\frac{x-c}{\sigma}\right)^2} \quad (2-18)$$

- **Funções Campana. Parâmetros  $\{a,b,c\}$**

$$Campana_{a,b,c}(x) = \frac{1}{1 + \left| \frac{x-c}{a} \right|^{2b}} \quad (2-19)$$

Na Figura 2-18 estão representadas as funções de pertinência: (a) triangular (20,60,80); (b) trapezoidal (10,20,60,95); (c) gaussiana (20,50) e (d) campana (20,4,50).



**Figura 2-18** - Funções de pertinência (a) Triangular (b) Trapezoidal (c) Gaussiana (d) Campana

### 2.5.5 - Variáveis lingüísticas

Como foi mencionado anteriormente, nos controladores convencionais o algoritmo de controle é descrito analiticamente por equações algébricas, diferenciais ou em diferenças, entretanto o controle nebuloso utiliza variáveis lingüísticas ao invés de variáveis numéricas. Uma variável lingüística  $x$  no universo de discurso  $X$  é definida em um conjunto de termos, nomes ou rótulos,  $T(x)$  (Shaw e Godoy, 1999). Assim, as variáveis lingüísticas admitem apenas valores lingüísticos, por exemplo, os termos  $T(a)$  em que  $a$  é altura de pessoa:

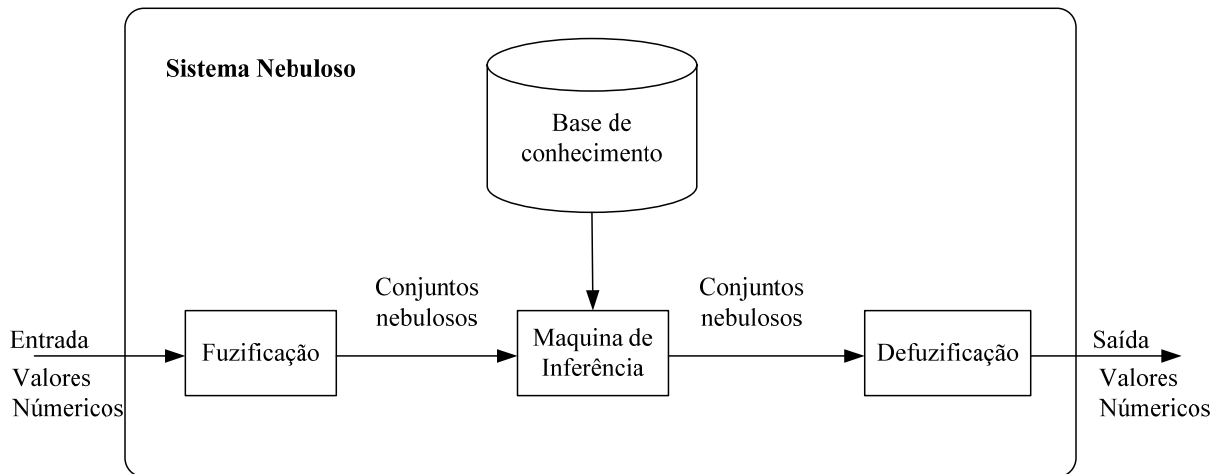
$$T(a) = \{\text{baixo, alto, muito alto, muitíssimo alto}\}$$

Sobre o universo de discurso  $X = [0,100]$ , são termos lingüísticos de tipo altura, deste modo, *altura* é uma variável lingüística. Comumente cada termo lingüístico de uma variável é expresso por uma função de pertinência.

### 2.5.6 - Sistemas nebulosos, fuzificação e defuzificação

A configuração básica de um sistema nebuloso é composta pelos blocos funcionais mostrados na Figura 2-19. Esta estrutura representa a transformação que ocorre no domínio de mundo real, que usa números reais, para o domínio *fuzzy*, que usa

números *fuzzy*. Durante esta transformação um conjunto de inferências nebulosas é utilizado para tomar decisões e, em seguida, é realizada uma transformação inversa do domínio *fuzzy* ao domínio do mundo real, acoplando as saídas *fuzzy* com as variáveis de atuação.



**Figura 2-19 - Sistema nebuloso**

### **Fuzificação**

Os valores discretos (não-*fuzzy*) das variáveis de entrada geralmente são provenientes de sensores das grandezas físicas ou dispositivos de entrada computadorizados. O processo de *fuzificação* é um mapeamento do domínio de números reais para o domínio *fuzzy* considerando uma atribuição de valores lingüísticos, ou valores qualitativos, definidos por funções de pertinência às variáveis de entrada. O processo de *fuzificação* pode ser entendido como um pré-processamento de classes dos sinais de entrada, reduzindo consideravelmente o número de valores a serem processados, significando uma computação mais veloz.

### **Base de conhecimento**

A base de conhecimento representa o modelo do sistema a ser controlado. Consiste em uma *base de dados* e uma *base de regras*. A base de dados contém as funções de pertinência lingüísticas utilizadas para representar os termos das variáveis de entrada e saída. A base de regras *fuzzy* lingüísticas contém o objetivo do controle

armazenando a informação fornecida por *especialistas* na área de atuação do sistema nebuloso.

### **Inferência e tomada de decisões**

O processo de inferência *fuzzy* segue os seguintes passos (Shaw e Godoy, 1999):

1. Comparação dos fatos com as premissas (antecedentes).
2. Cálculo do grau de compatibilidade de cada regra.
3. Cálculo de crença em cada regra.
4. Agregação.

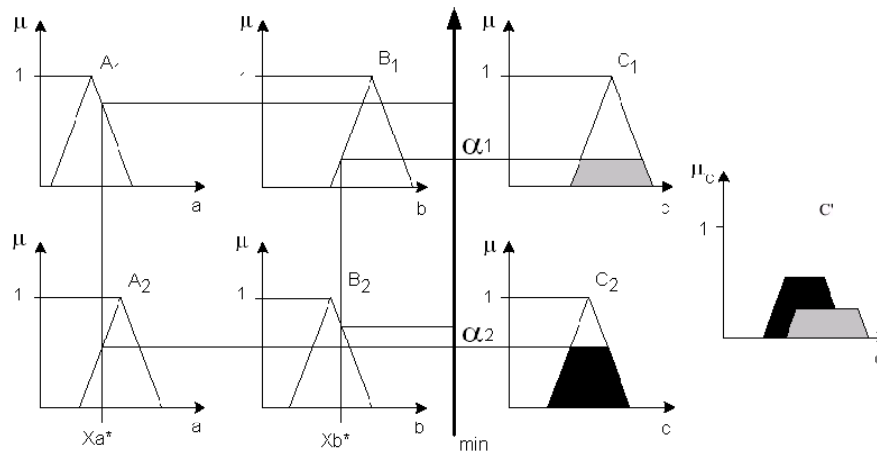
Após o processo de inferência, feito a partir do conhecimento especialista e armazenado na base de regras, é possível utilizar implicações *fuzzy* para simular a tomadas de decisão humanas, gerando ações de controle. Os métodos de agregação mais utilizados são: o método de *Mamdani* (Mamdani, 1975, Mamdani, 1977), método de *Tsukamoto* (Tsukamoto, 1979) e o método de Takagi-Sugeno (Takagi e Sugeno, 1985). O método Mamdani é o modelo de implicação mais importante conhecido na literatura (Shaw e Godoy, 1999). A Figura 2-20 mostra um exemplo do modelo de raciocínio tipo Mamdani para um sistema de duas regras com dois antecedentes A e B cada uma. A implicação representada é:

**Se A x B então C**

Considerando as seguintes regras:

**Se (a é A1) e (b é B1) então c é C1.**

**Se (a é A2) e (b é B2) então c é C2.**



**Figura 2-20** - – Método Clássico de agregação Mamdani

## Defuzificação

Na *defuzificação*, o valor da variável lingüística de saída, inferida pelas regras *fuzzy*, será traduzida num valor discreto, com objetivo de se obter um único valor numérico discreto que executará uma ação de controle. Diferentes métodos de *defuzificação* são conhecidos, alguns deles exigem cálculos computacionais complexos. Abaixo, são descritos alguns desses métodos (Shaw e Godoy, 1999):

### 1. Defuzificação pelo Centro da Área (C-o-A)

Comumente denominado centro de gravidade, pois ele calcula o centróide da área do conjunto nebuloso de saída  $\mu_{out}(\mu_i)$ . O centróide é um ponto que divide a área em duas partes iguais. A equação (2-19) representa o cálculo do centróide.

$$u = \frac{\sum_{i=1}^N u_i \mu_{out}(u_i)}{\sum_{i=1}^N \mu_{out}(u_i)} \quad (2-20)$$

O método do centróide apresenta problemas quando as funções de pertinência não possuem sobreposição e pode acontecer que o centro geométrico não tenha significado físico. Além disso, a necessidade de integração numérica exige esforços computacionais (Shaw e Godoy, 1999).

## 2. Defuzificação pelo Centro do Máximo (C-o-M)

Neste método as áreas das funções de pertinência não desempenham nenhum papel e apenas os máximos (*pertinências singleton*) são usados. A saída discreta é calculada como uma média ponderada dos máximos, cujos pesos são os resultados da inferência conforme indica a equação (2-21), em que  $\mu_{0,k}(\mu_i)$  indicam os pontos dos máximos das funções de pertinência de saída. Este método também é denominado *defuzificação pelas alturas*.

$$u = \frac{\sum_{i=1}^N u_i \cdot \sum_{k=1}^n \mu_{0,k}(u_i)}{\sum_{i=1}^N \sum_{k=1}^n \mu_{0,k}(u_i)} \quad (2-21)$$

## 3. Defuzificação Média do Máximo (M-o-M)

Em casos onde a função de pertinência tenha mais de um máximo a abordagem C-o-M não poderia ser utilizada, devido à necessidade de se escolher qual máximo utilizar, no entanto, pode-se tomar- a média de todos os máximos, equação (2-22).

$$u = \sum_{m=1}^M \frac{u_m}{M} \quad (2-22)$$

Em que, M é o número de máximos encontrados,  $\mu_m$  é o m-ésimo elemento onde a função  $\mu_{out}(\mu_i)$  tenha um máximo. No contexto do sistema nebuloso projetado neste trabalho utilizou-se o método M-o-M, pois é a solução mais plausível por desconsiderar a forma de todas as funções de pertinência de saída além de não requerer cálculos computacionais complexos.

### 2.5.7 –Controle nebuloso de grupo de elevadores

A principal motivação para o desenvolvimento de um controlador baseado em lógica nebulosa veio dos engenheiros, os quais queriam incluir de alguma forma o conhecimento especialista dentro do controle de grupo de elevadores. O conhecimento consistia em dois principais tipos de regras. As regras estratégicas foram criadas para situações de tráfego normais, por exemplo, para impedir um serviço desigual ou agrupado. O conhecimento tático relacionado com detalhes de controle, como por

exemplo, a forma de tratar chamadas especiais para pessoas com deficiência, etc. Uma solução com muito sucesso para o problema da codificação do conhecimento especialista em um computador foi proposta por Zadeh (Markon, 2006), com a introdução da lógica nebulosa. Esta possibilitou a implementação de regras lingüísticas. O nebuloso controle de grupo de elevadores é uma implementação de um método para atribuição de chamadas de pavimento, em que se utiliza conhecimento baseado em regras de especialistas. Em operações normais de prédios e testes de simulação as regras para controle de elevadores consideram várias situações de tráfego. Algumas dessas regras são aplicadas para situações de trafego particulares, como “*Down-peak*” em que os elevadores são dirigidos diretamente para o térreo do prédio, as outras são regras gerais de posicionamento entre eles. As entradas ou antecedentes para as regras difusas são construídas por meio de variáveis difusas (conjuntos nebulosos). Desta maneira, é possível conhecer o estado do sistema de elevadores, como por exemplo, sua posição em termos lingüísticos (muito perto, perto, médio perto,..etc.), pode também identificar os padrão de tráfego presentes no prédio “*Down-peak*”, etc. O sistema de inferência nebuloso envolve as entradas dos conjuntos nebulosos, a base de regras e os valores de pertinência para gerar os valores de decisão nebulosos, os quais finalmente entram no processo de *defuzificação* (Média do máximo).

O sistema de controle nebuloso de grupo consegue um alto desempenho diminuindo o tempo de espera dos passageiros e a taxa de erro no anúncio da atribuição de elevadores. Entretanto, o sistema nebuloso não tem a capacidade de adaptar suas regras para situações específicas de cada prédio, então se o controle projetado não encaixa, ele deve ser ajustado manualmente. O controle de grupo de elevadores nebulosos não pode ser adaptado automaticamente, se as condições do prédio mudam. Essas deficiências foram abordadas no desenvolvimento de controle de grupo de elevadores neural (Markon et al, 2006).

## **2.6- CONCLUSÕES DO CAPITULO**

Neste capítulo foram abordadas algumas definições sobre os sistemas de elevadores, consideradas importantes para o planejamento e desenvolvimento do presente trabalho. O estudo do estado da arte na indústria de elevadores evidenciou uma evolução dos sistemas de transporte vertical, sendo justificado seu uso em prédios comerciais e

residenciais, de modo a diminuir o consumo de energia e aumentar o conforto dos passageiros.

Na atualidade, as empresas no mercado de elevadores fabricam seus próprios sistemas de simulação (ferramentas *software*), para serem executados em computadores pessoais. Desta forma, permite um melhor entendimento de situações complexas que possam acontecer nos sistemas de grupo de elevadores. Por outro lado, hoje em dia existem diversas ferramentas para o desenvolvimento de *softwares*, as quais permitem desenvolver estas aplicações de forma organizada como o objetivo de oferecer uma fácil compreensão. Isto é a modelagem UML, a qual possibilita a visualização de todos os passos de desenvolvimento de um problema de modelagem na forma de diagramas padronizados, envolvendo elementos estruturais (classes, objetos, relacionamentos, etc.), de comportamento (casos de uso, iteração, máquinas de estado), de agrupamento (pacote, modelo, subsistema, etc.), de relacionamento (composição, agregação, generalização).

As técnicas de inteligência artificial baseadas em lógica nebulosa para o desenvolvimento dos sistemas de elevadores foram introduzidas a princípios dos anos 1990's (Aoki et al. 1990; TEKES 1989).

O objetivo principal de uma nova implementação de um sistema de elevadores é aumentar o conforto dos usuários ao mesmo tempo em que o consumo de energia deve ser diminuído. Portanto, neste capítulo também foram apresentadas as tecnologias de automação industrial, examinando a sua potencialidade para serem implementadas em arquiteturas modernas de elevadores. Espera-se, com este estudo, avaliar o desempenho, a flexibilidade e eficiência, de forma que as técnicas de controle de elevadores propostas possam ser uma alternativa inicial de pesquisa nesta área.



### **3 - MODELAGEM E SIMULAÇÃO DE SISTEMAS DE GRUPO DE ELEVADORES**

Na área de sistemas de elevadores, por exemplo, quando os requerimentos no planejamento de um prédio são muito complexos, com vários elevadores a serem controlados de modo a minimizar o tempo de espera dos passageiros e o consumo de energia, o desenvolvimento de algoritmos de controle envolveria o uso exaustivo do sistema real. Isso implicaria, em um primeiro momento, em problemas de atendimento da demanda de tráfego, uma vez que o desenvolvimento do controlador exigiria o conhecimento dos padrões de tráfego atuantes no sistema, além da necessidade de se realizarem muitos testes até se atingir um algoritmo otimizado. Diferentemente, o uso de simuladores de tráfego, baseados em padrões de tráfego já identificados e publicados na literatura (Peter. et al, 1998), aceleraria o desenvolvimento dos algoritmos de controle. É amplamente conhecido que a simulação é uma técnica particularmente adequada em situações em que o processo de estudo é difícil, perigoso e, adicionalmente, de alto custo. Um sistema elevadores cai dentro desta categoria (Barney,2003). Isso tanto é verdade que empresas líderes do setor têm crescentemente desenvolvido e utilizado ferramentas de simulação para a realização de novos projetos de instalação e controle de elevadores (Cortéz, et al, 2006).

#### **3.1 - Modelagem Monte-Carlo**

O método de Monte Carlo é uma ferramenta de simulação numérica de sistemas complexos com componentes estocásticos ou determinísticos, mantendo a entrada e a saída com certo grau de incerteza (Taha, 2007).

Um sistema de transporte vertical tem como elementos principais: elevadores, passageiros, controle de grupo, etc. Para fins de simulação, considera-se que esses elementos possuem comportamentos específicos, os quais são governados por meio de leis de comportamento pré-definido, com componentes determinísticos e estocásticos. Portanto, segundo Markon (2006) o sistema de elevadores permite a aplicação de uma metodologia de simulação para analisar o comportamento do sistema em situações específicas. O sistema é modelado em um programa de computador e, desta maneira, pode-se observar seu comportamento como uma evolução temporal. Esta metodologia é

muito utilizada não só nos sistemas de transporte, mas também no estudo de todo tipo de problemas de pesquisa operacional que envolvem dinâmica, aleatoriedade e logística.

### **3.2 - Técnicas de simulação**

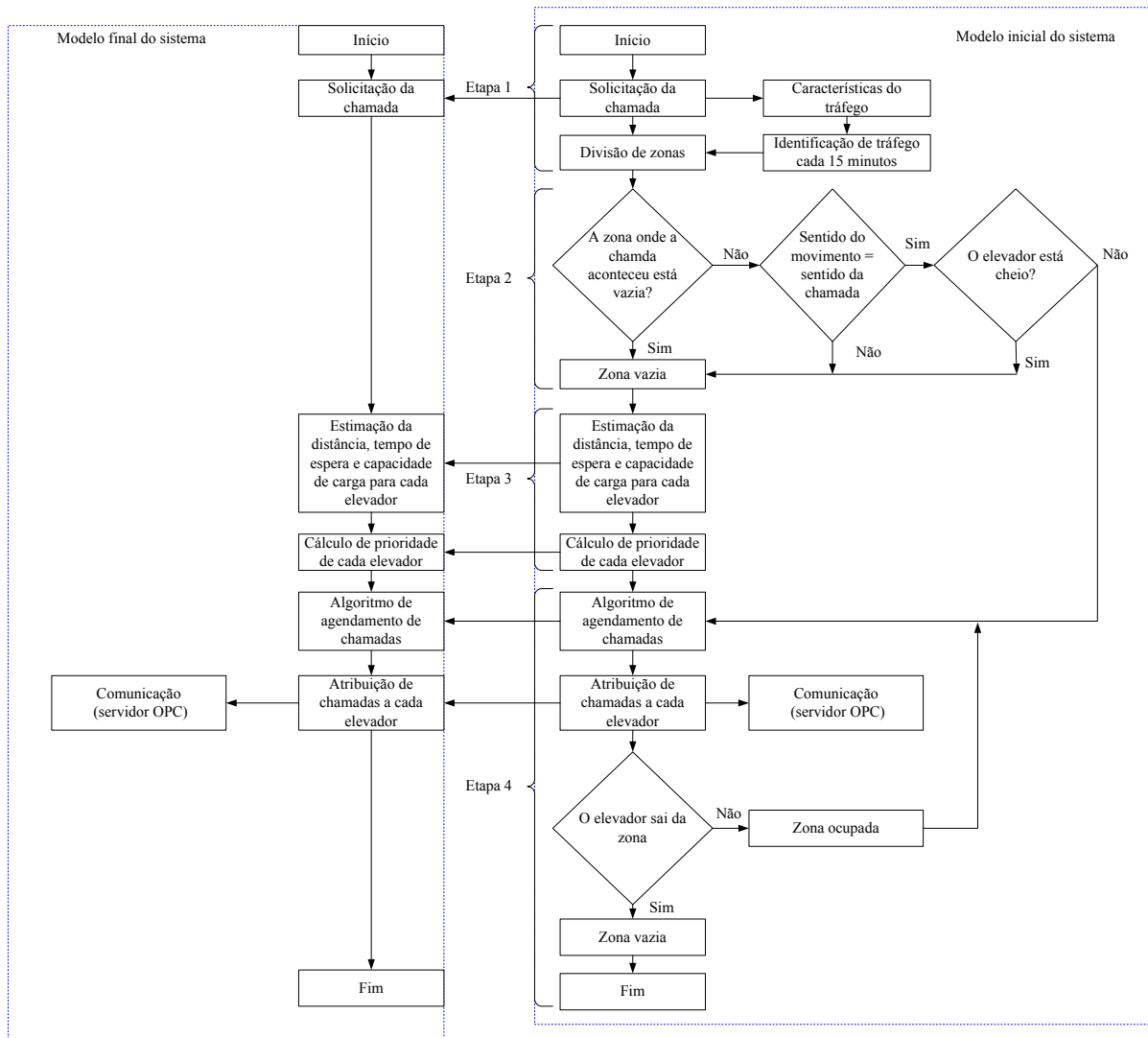
Segundo Mitrani (1982), as etapas mais importantes necessárias em um estudo de simulação são:

1. Construção do modelo
2. Planejamento do simulador
3. Implementação do simulador
4. Execução das simulações
5. Análises dos resultados de simulação

Os subtítulos que se seguem descrevem cada etapa individualmente.

#### **3.2.1 - Construção do modelo**

A Figura 3-1 mostra a etapa de construção do modelo conforme aplicada no problema em estudo neste trabalho. No lado esquerdo da Figura 3.1, apresenta-se a *construção do modelo* do sistema de elevadores, o qual foi realizado por meio de um fluxograma. O modelo computacionalmente planejado inicialmente é apresentado no lado direito da divide-se em quatro etapas.



**Figura 3-1**– Construção do modelo do sistema de elevadores

A primeira etapa lida com as chamadas de pavimento e usa uma técnica de divisão de zonas do prédio de acordo com o número de elevadores (uma zona pode estar composta por um ou vários andares e o número de zonas deve ser igual ao número de elevadores). Esta divisão de zonas foi projetada para que, por meio de um sistema de lógica nebulosa, se realizasse a escolha da divisão mais conveniente de acordo como o tráfego presente no prédio (estas divisões foram preestabelecidas), repetindo este procedimento cada 15 minutos.

A segunda etapa tem vários esquemas de decisão para considerar o estado das zonas vazias ou ocupadas. Estes esquemas contêm perguntas sobre diversos aspectos tais como: (a) o estado de ocupação das zonas; (b) o momento quando uma nova chamada de pavimento é realizada; (c) qual o sentido de viagem de cada elevador e (d)

qual a capacidade de carga disponível dentro do elevador. No momento em que a zona está vazia o algoritmo executa a terceira etapa, em caso contrario executa a quarta etapa.

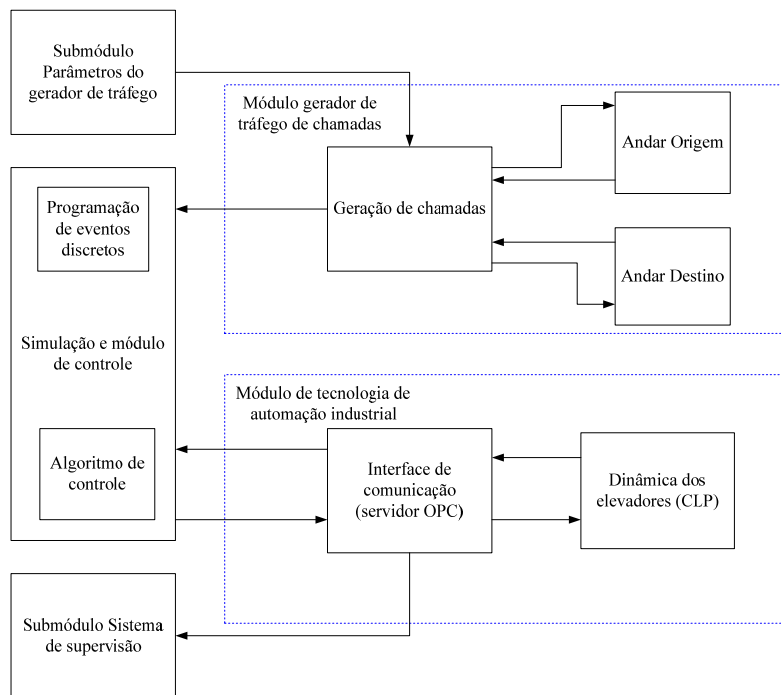
A terceira etapa estima os valores de tempo de espera, distância e disponibilidade de carga do sistema e envia essa informação a um sistema nebuloso para que avalie cada elevador e calcule os valores de prioridade de cada um. Desta maneira é possível escolher o elevador mais conveniente para atender a chamada de pavimento.

Finalmente, a quarta etapa quando a situação do elevador é considerada como ocupada, o algoritmo de agendamento de chamadas realiza um atendimento coletivo (percorrendo andar por andar) até que o elevador saia da zona, deixando a zona em estado vazio. A comunicação entre o sistema de controle e a planta do sistema foi projetada para ser realizada como uma rede cliente servidor. Na etapa quatro foi proposto o desenvolvimento de um modulo de comunicação baseado em OPC (*OLE for Process Control*) (Patiño-Forero, 2009).

O modelo inicialmente construído para o sistema de grupo de elevadores aproveita a utilização de duas técnicas de controle usando dois sistemas de lógica nebulosa. O primeiro para o uso da técnica de divisão de zonas e o segundo usando a estimação parâmetros de tempo de espera, distância e disponibilidade de carga para atribuição do elevador mais conveniente. Entretanto, durante a implementação do *software*, devido à limitação de tempo decidiu-se por considerar uma divisão de zonas estática em que todo o prédio é considerado como uma zona e todos os elevadores do grupo atendem à mesma zona. Em vista disso, somente se implementou a lógica nebulosa para a atribuição de elevadores estimando os parâmetros de tempo de espera, distância e capacidade de carga.

### **3.2.2 - Planejamento do simulador**

Após de realizar a construção do modelo foi necessário desenvolver um diagrama de componentes do sistema, baseado nas ferramentas a serem usadas, ou seja, a linguagem de programação a ser utilizada e as ferramentas de automação disponíveis. A Figura 3-2 apresenta o diagrama de componentes que explica de forma geral os elementos de todo o sistema de simulação de grupo de elevadores.



**Figura 3-2**– Diagrama de componentes do sistema simulação e controle de tráfego de elevadores

O diagrama está composto por três módulos e dois submódulos. O módulo chamado “**gerador de trafego de chamadas**” foi projetado para a obtenção das chamadas de pavimento, baseado no gerador de tráfego de passageiros apresentado no capítulo anterior seção 2.2.5. Esse módulo foi projetado para realizar os algoritmos necessários para obter as chamadas de pavimento dependendo do tipo de tráfego a simular e é composto por três blocos: “Andar Origem”, “Andar Destino” e “geração de chamadas”. O submódulo “**Parâmetros do gerador de tráfego**” foi projetado para servir de interface gráfica e poder usar o módulo “**gerador de tráfego de chamadas**” e, desta forma, gerar as chamadas de pavimento de acordo com as características do prédio (número de andares, número de elevadores, população em viagem no prédio, tipo de tráfego, entre outros).

O módulo chamado “**tecnologia de automação industrial**” foi baseado nas ferramentas disponíveis de automação industrial. Esse módulo é composto por dois blocos (“interface de comunicação” e “dinâmica dos elevadores”), os quais foram projetados para realizar uma comunicação com padrão industrial (OPC) e poder realizar um algoritmo que desempenhe a dinâmica de elevadores (implementado no emulador de CLP) respectivamente.

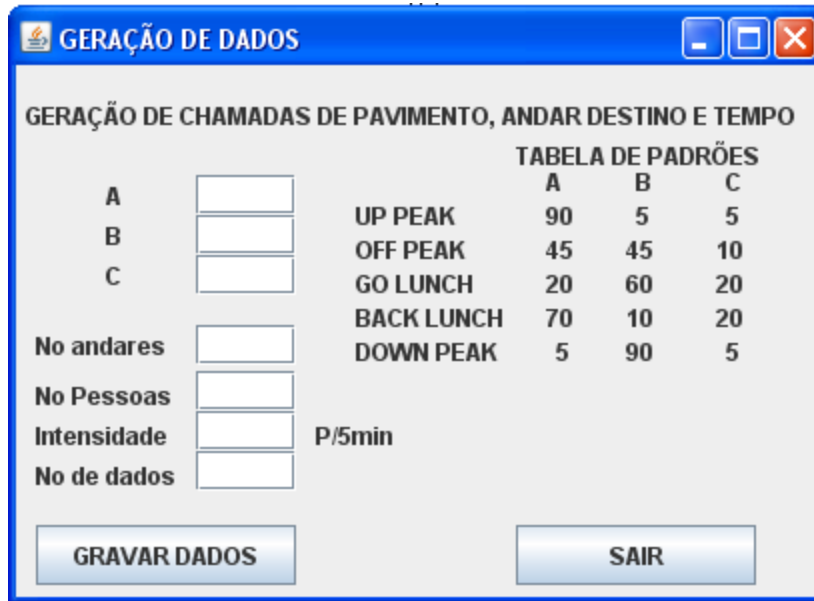
O submódulo “**Sistema de supervisão**” foi projetado para servir de interface gráfica e possibilitar a visualização do comportamento do sistema em outro computador conectado em rede com dados transmitidos por meio da interface de comunicação OPC.

O módulo chamado “**simulação e controle**” foi projetado para controlar todo o sistema. Este módulo toma os dados das chamadas de pavimento e a posições atuais dos elevadores (conforme estado atual do programa implementado no emulador de CLP) para avaliar por meio do algoritmo de controle nebuloso as prioridades de cada elevador. Após avaliar as prioridades atribuídas, determina-se a tarefa de atendimento à chamada ao elevador mais conveniente ou realiza-se um agendamento das chamadas no caso de que os elevadores estejam ocupados. Este módulo inclui uma interface gráfica que permite configurar o sistema para dar início ao processo de simulação. Nesta etapa foi escolhida a linguagem de programação JAVA, devido a esta ser uma ferramenta de uso gratuito.

### **3.2.3 - Implementação do simulador**

#### **Módulo de gerar tráfego de chamadas**

Após o planejamento do simulador se iniciou a etapa de implementação, a qual começou pela realização do módulo gerador de tráfego de chamadas apresentado na Figura 3-2. O desenvolvimento foi baseado no gerador de tráfego de passageiros apresentado na seção 2.2.5. O ambiente de desenvolvimento utilizado foi ECLIPSE em que foi codificado o algoritmo para gerar as chamadas de pavimento (andar origem, andar destino e tempo da chamada). Na Figura 3-3, apresenta-se a interface gráfica de geração de tráfego de chamadas, a qual é composta por uma tabela de dados em que se encontram os padrões de tráfego a serem simulados (). Esses dados são colocados nas caixas de textos A, B e C. Os outros parâmetro (número de andares, número de pessoas, intensidade de tráfego e número de dados) dependem das características do prédio a ser simulado. Finalmente, este módulo permite, por meio do botão gravar dados, realizar um arquivo texto, o qual é salvo no computador com as informações (andar de origem, andar de destino e tempo da chamada) necessárias para executar as simulações.



**Figura 3-3-** Interface gráfica para geração de tráfego de chamadas (submódulo parâmetros gerador de tráfego)

### Módulo de tecnologia de automação industrial

Para a implementação deste módulo, utilizaram-se softwares voltados à automação industrial fornecidos pela *Rockwell Automation*. Por conseguinte, foram escolhidas as seguintes ferramentas de *software*:

- **RSLinx:** Proporciona acesso de controlador programável *Allen-Bradley* a uma ampla variedade de aplicativos da *Rockwell Software* e da *Allen-Bradley*. Isso inclui desde aplicativos de programação e configuração de dispositivos, como o *RSLogix* e o *RSNetWorx*, até aplicativos IHM (Interface Homem-Máquina), como o *RSView32*, e seus próprios aplicativos de aquisição de dados usando *Microsoft Office*, páginas da web ou *Visual Basic®*. O *RSLinx* é um servidor em conformidade com *OPC Data Access* e é um servidor *DDE* (*Rockwell Automation*, 2005).
- **RSLogixEmulate 5000:** É um produto software que emula o comportamento de um processador *Logix 5000*. O emulador permite a realização e depuração de programas em um ambiente seguro e controlado, sem investir em *hardware* (controladores e módulo entrada e saída). Além disso, o emulador permite a

verificação de aplicações de supervisão sem usar controladores reais (Rockwell Automation, 2006).

- **RSLogix 5000:** É o *software* de programação do CLP. Exibe um sistema operacional, multi-tarefas e em conformidade com IEC 61131-3 de um controlador da série Logix como uma árvore gráfica com tarefas, programas e rotinas. Diversos usuários podem monitorar e editar simultaneamente o conteúdo de um controlador Logix5000 em operação. Um projeto do controlador pode incluir rotinas múltiplas usando diversas linguagens tais como diagrama ladder (LD – *Ladder diagram*), diagrama de blocos de funções (FBD – *function block diagram*), texto estruturado (ST – *structured text*) e fluxogramas seqüencial de funções (SFC- *sequential function chart*) (Rockwell Automation, 2005).
- **RSView32:** É um aplicativo de aquisição de dados IHM (Interface Homem-Máquina), para monitoramento e controle de máquinas e processos de automação, desenvolvido para ambiente *Windows*® (Rockwell Automation, 2000).

Após a escolha das ferramentas de *software*, iniciou-se a instalação e configuração destas em um mesmo computador. O software RSView32 foi instalado em outro computador, considerando que esta ferramenta permite realizar o monitoramento dos dados de forma remota, por meio de uma rede Ethernet, através do protocolo OPC. Para apresentar o desenvolvimento deste módulo foi necessário dividir sua explicação em três etapas, descritas a seguir:

1. A primeira etapa foi baseada na programação do CLP (Controlador Lógico Programável), fazendo uso do *software* RSLogix 5000. No RSLogix 5000 foi desenvolvido um algoritmo em texto estruturado, em que é modelado o comportamento de elevadores usando temporizadores, ou seja, este algoritmo calcula o tempo durante o qual o elevador se movimenta de um andar ao outro ( $T$ ), o tempo de abrir ( $T_a$ ) /fechar ( $T_c$ ) portas e o tempo de transferência de passageiros ( $T_{tx}$ ). Na Figura 3-4 (gráfico a) apresenta-se o tempo ( $T$ ) que um elevador demora para percorrer uma distância  $S$ . O tempo  $T$  é calculado por meio de parâmetros dos acionamentos (*drivers*) dos motores, dimensões do prédio e restrições de aceleração máxima relacionadas ao nível de conforto



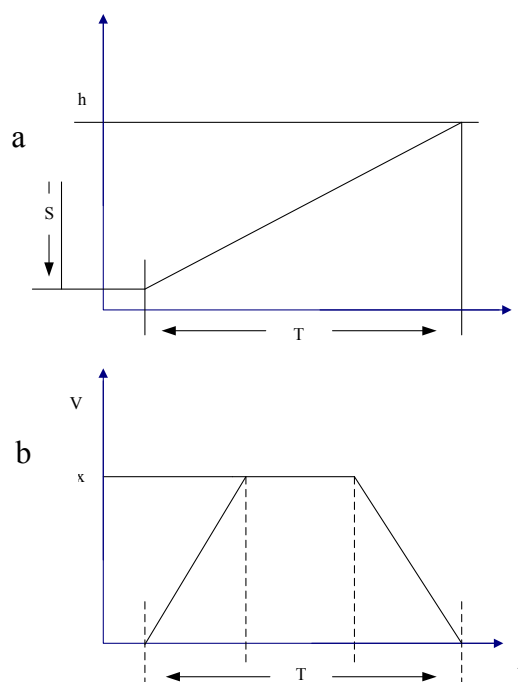
desejado para os passageiros. A curva de velocidade mostrada na Figura 3-4 (gráfico b) apresenta a velocidade máxima,  $V$ , a aceleração máxima,  $a$ , permitida e a altura de um andar,  $x$ . Para o cálculo de  $T$ , foi necessário usar as equações de movimento uniforme variado e de movimento retilíneo uniforme. Portanto, o  $T$  é calculado por meio da equação (3-1):

$$T = 2 \left( \frac{V}{a} \right) + \left( \frac{x - \frac{V^2}{a}}{V} \right) \quad (3-1)$$

Deve-se observar que neste cálculo, as distâncias percorridas pelo carro durante os períodos de aceleração e desaceleração são desprezadas, uma vez que são muito menores que a distância entre andares (Markon, 2006).

Finalmente, para o cálculo do tempo  $T_N$  para um número de andares  $N$  a equação simplificada fica desta maneira:

$$T_N = \left( \frac{V}{a} + \frac{x}{V} \right) + \left( \frac{x}{V} \right) (N - 1) \quad (3-2)$$



**Figura 3-4**— Gráfico (a) representa a distância percorrida por um elevador. Gráfico (b) representa a curva de velocidade ( $v_{\max}$ ) de um elevador entre duas paradas em andares subsequentes

Após de calcular o tempo que demora um elevador em percorrer um andar é calculado o tempo de abrir/fechar portas e o tempo de transferência de passageiros. Para este fim foi usado a seguinte equação:

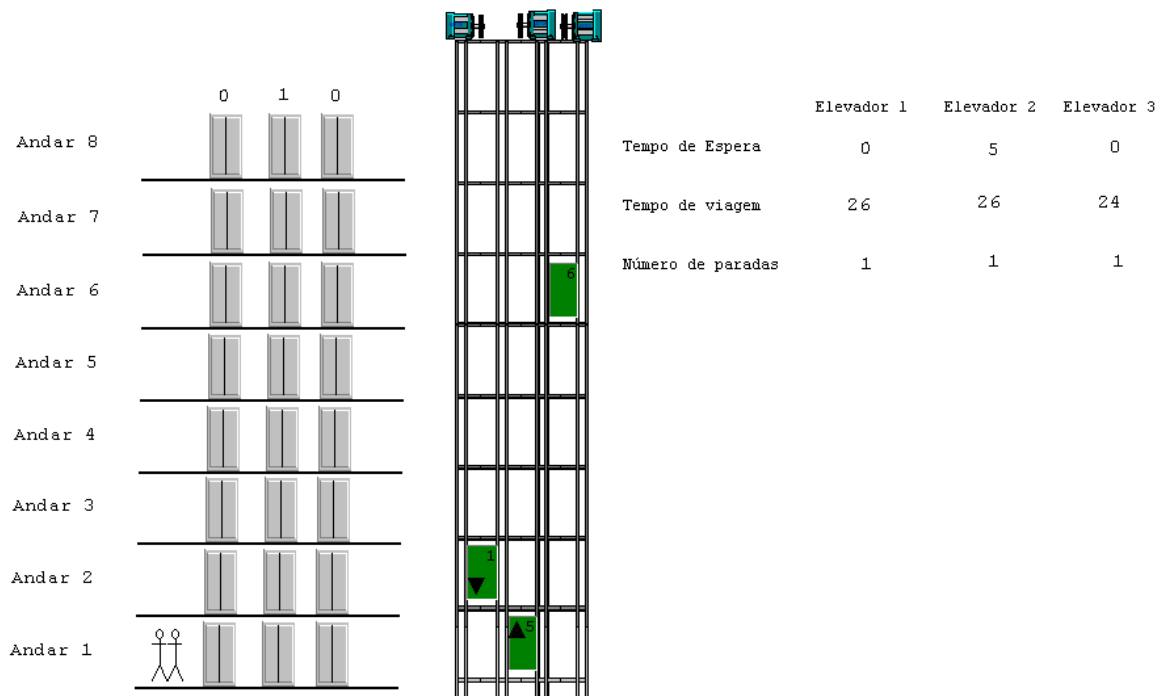
$$T_e = (T_a + T_c) + (T_{tx} * \text{passageiros}) \quad (3-3)$$

em que,  $T_a$  e  $T_c$  são os tempos de abrir e fechar portas respectivamente,  $T_{tx}$  é o tempo que um passageiro demora para entrar e sair do elevador. Desta maneira, por meio das equações (3-2) e (3-3), conseguiu-se representar a dinâmica dos elevadores.

2. A segunda etapa consistiu em utilizar RSLogixEmulate 5000 e o RSLinx. O RSLogixEmulate 5000 serve para emular o comportamento da dinâmica de elevadores, sem necessidade de usar a planta do sistema de elevadores real. O RSLinx é usado para proporcionar e configurar a comunicação entre o software de programação RSLogix 5000 e o RSLogixEmulate 5000. Esse inclui um

servidor OPC, o qual foi usado para acessar os dados do emulador através de uma aplicação cliente realizado em JAVA.

3. Na terceira etapa foi implementado o submódulo “sistema de supervisão” utilizando o *software* RSView32, o qual é destinado ao desenvolvimento de sistemas SCADA (*Supervisor Control And Data Aquisition*). Elaborou-se um sistema de supervisão para acessar os dados gerados pelo sistema de simulação (cliente JAVA) de forma remota usando uma rede Ethernet através do protocolo de comunicação OPC. A interface gráfica do sistema desse sistema de supervisão é apresentada na Figura 3-5:



**Figura 3-5** - - Sistema de supervisão do sistema de tráfego de elevadores

O sistema de supervisão permite o monitoramento das variáveis mais importantes do sistema de controle de grupo de elevadores (EGCS) desenvolvido. Por exemplo, no caso da Figura 3-5 têm-se: os tempos de espera, os tempos de viagem, o número de paradas, o número de passageiros dentro dos elevadores, o andar de destino, o andar de origem e o sentido de viagem de cada elevador. Além disso, é possível acrescentar mais funcionalidades ao sistema (alarmes, curvas de tendência, etc.), tipicamente encontradas em um sistema de supervisão de um processo industrial.

## Módulo de simulação e controle

Neste módulo foi realizado um sistema cliente OPC para possibilitar o acesso aos dados disponibilizados pelo servidor OPC. A função do sistema cliente é ler e escrever no servidor OPC, permitindo desta maneira gerenciar as variáveis do emulador de CLP.

Para o desenvolvimento deste módulo foi necessário fazer uma modelagem UML (*Unified Modeling Language*) com o objetivo de descrever o desenvolvimento de *software* por meio de diagramas padronizados. Esta modelagem permite que o desenvolvimento de software seja realizado de uma forma organizada e possa ser facilmente apresentado no trabalho. Para uma melhor compreensão, a modelagem UML é apresentada no capítulo 4, onde são detalhadas todas as partes do trabalho de desenvolvimento de *software*.

Um método heurístico, baseado em lógica nebulosa, também foi implementado neste módulo de maneira a proporcionar um valor de prioridade a cada elevador buscando determinar aquele que se apresente como o mais conveniente para atender às chamadas de pavimento.

Para o uso do sistema de lógica nebulosa foi necessária a implementação de alguns métodos para ajuste dos dados de entrada ao sistema realizado. Estes métodos utilizam os parâmetros de prédios e os valores da dinâmica dos elevadores. Entre os parâmetros de prédios foram utilizados o número de andares e o número de elevadores. Os valores da dinâmica dos elevadores empregados foram a posição atual dos elevadores, o sentido dos elevadores, o número de chamadas agendadas e o número de pessoas dentro dos elevadores. Esses dados são fornecidos pelo emulador de CLP.

A finalidade destes métodos considera a estimação das seguintes variáveis:

**1) Disponibilidade de carga :** é o numero atual de vagas disponíveis no elevador no instante em que a chamada de pavimento ocorreu.

**2) Distância a ser percorrida pelo elevador:** é o numero de andares que o elevador deve percorrer desde a posição atual até o andar onde a nova chamada de pavimento ocorreu.

**3) Tempo de espera:** é um cálculo estimado do tempo desde o momento em que é realizada uma chamada de pavimento até o elevador chegar ao pavimento indicado, abrindo as portas e permitindo o ingresso dos passageiros. Esta medida depende diretamente dos parâmetros de configuração do sistema de elevadores tais como, velocidade do elevador, relação de aceleração, altura dos andares, tempo de abrir e fechar as portas. O *tempo de espera* depende também do número de paradas que devem ser realizadas pelo elevador antes de chegar ao pavimento solicitado. Este tempo pode ser obtido pela equação (3-4).

$$T_w = (N_s * T_s) + (D * T_f) \quad (3-4)$$

em que,

$T_w$  = Tempo de espera

$N_s$  = Número de paradas

$T_s$  = Tempo de paradas

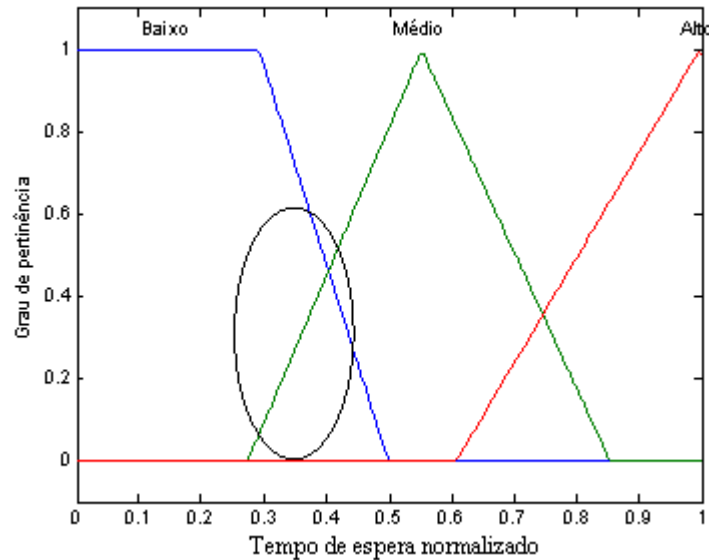
$D$  = Distância percorrida em andares

$T_f$  = Tempo vôo por andar

As três variáveis calculadas formam parte das entradas do sistema de controle de grupo de elevadores nebuloso (FEGCS – *Fuzzy Elevator Group Control System*) implementado e o valor de prioridade é a variável de saída do sistema. Na construção do sistema de lógica nebulosa, foi necessário definir os termos nebulosos de cada variável. Para garantir suavidade e estabilidade permitiu-se em algumas funções de pertinência uma sobreposição parcial entre conjuntos nebulosos vizinhos. As variáveis lingüísticas que descrevem o sistema nebuloso são as seguintes:

- *Tempo de espera* ( $T_w$ ): O tempo de espera é uma medida do conforto dos usuários no edifício. Esta variável representa o tempo estimado desde o momento em que é realizada uma chamada de pavimento até o momento em que o elevador abre completamente a porta no pavimento indicado. Ela é utilizada para medir o grau de pertinência da entrada *Tempo de espera*. O universo de discurso foi estipulado pela equação (3-4), sendo que este valor é normalizado pelo tempo máximo de espera, que corresponde à viagem de maior percurso no prédio (por exemplo do andar térreo ao último andar) considerando que o carro pára em todos os andares

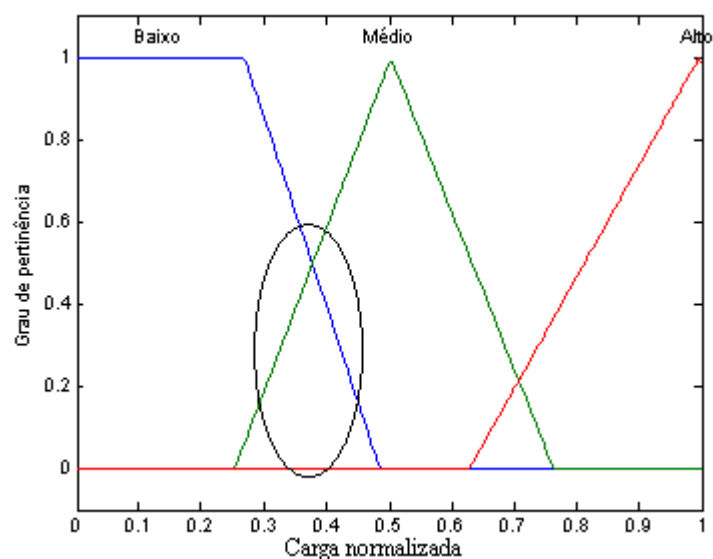
intermediários. A variável tempo de espera possui três funções de pertinência com termos lingüísticos baixo, médio e alto. Os valores adotados para limitar cada função de pertinência assim como suas formas foram adotadas com base em Siikonen (1997).



**Figura 3-6** – Função de pertinência da variável tempo de espera (tempo de espera normalizado)

Pode-se observar na Figura 3-6 que a função de pertinência da variável *tempo de espera* possui uma distribuição não simétrica no universo de discurso. Especificamente os tempos de espera considerados curtos abrangem maior área no seu universo de discurso, tendo em vista a importância desta variável na estratégia de atribuição de carros para o atendimento de chamadas. Uma consequência dessa assimetria é que há maior probabilidade de se considerarem tempos de espera inferiores a valores médios (0,5) como tempos curtos, com algum grau de pertinência. Isso resulta em uma maior sensibilidade nos tempos de espera curtos.

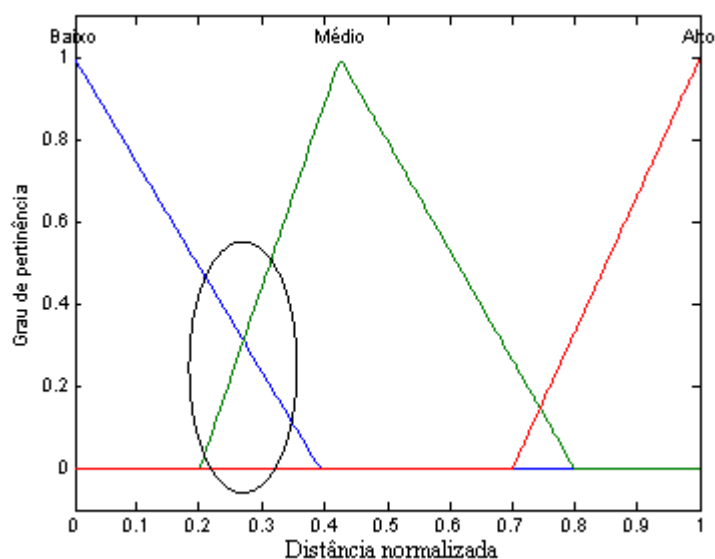
- *Disponibilidade de carga*: é uma medida que depende dos parâmetros do prédio. Esta variável lingüística que representa o número de vagas disponíveis dentro do elevador no momento em que uma chamada de pavimento é realizada. O valor atribuído a esta variável é um valor relativo à capacidade total de passageiros de cada carro e sua utilização se dá para fins de cálculo do grau de pertinência da entrada (*Disponibilidade de carga*). O universo de discurso compreende valores entre 0 e 1 e possui três funções de pertinência com termos lingüísticos baixo, médio e alto. Os valores limites atribuídos a cada função de pertinência também foram escolhidos com base em Siikonen (1997).



**Figura 3-7** – Função de pertinência da variável disponibilidade de carga.

Nota-se que na Figura 3-7 a função de pertinência da variável *Disponibilidade de carga* possui uma distribuição não simétrica no universo de discurso. Especificamente os valores de disponibilidade de carga considerados baixos abrangem maior área no seu universo de discurso. Uma consequência dessa assimetria é que há maior probabilidade de se considerarem como baixos, valores de disponibilidade de carga inferiores a valores médios. Portanto, o sistema de inferência resulta mais sensível quando o tráfego presente no prédio é crítico, ou seja, nesta situação o número de vagas disponíveis nos elevadores tende a diminuir, o que implica que o elevador não poderá atender a novas chamadas de pavimento até que haja liberação de vagas.

- *Distância (D)*: Esta variável lingüística representa a distância a ser percorrida pelo elevador deste a posição atual até a posição onde ocorre a chamada de pavimento. A distância é medida em número de andares e a variável é calculada em termos relativos ao número total de andares do prédio, sendo utilizada para calcular o valor de pertinência da variável de entrada *Distância*. O universo de discurso compreende valores discretos entre 0 e 1 e é mapeado por três funções de pertinência com termos lingüísticos baixo, médio e alto. Os valores limites adotados para cada função de pertinência foram atribuídos com base em Siikonen (1997).



**Figura 3-8** – Função de pertinência da variável distância

Pode-se observar na Figura 3-8 que a função de pertinência da variável *Distância* possui uma distribuição não simétrica no universo de discurso. Especificamente, a região denominada *Médio* foi escolhida de tal forma que abrangesse a maior área em seu universo de discurso e que apresentasse maior sobreposição em relação à região chamada *Baixo*. Uma consequência dessa assimetria é que há maior probabilidade de se considerarem, como valores baixos, valores de distância médios, uma vez que o valor de pertinência para a região de sobreposição entre *Médio* e *Baixo* tende a ser maior que para a região de sobreposição entre *Médio* e *Alto*. Portanto, o sistema de inferência apresenta uma maior sensibilidade na região média-baixa que na região média-alta. Desta forma, é possível atribuir uma alta prioridade ao elevador que tenha a menor distância desde sua posição atual até a posição onde ocorre a chamada de pavimento.

O conjunto de saída, *Prioridade*, compõe a decisão tomada pelo controlador de grupo visando a diminuir o tempo de espera dos passageiros.

- *Valor de Prioridade*: é a variável lingüística de saída da máquina de inferência. Representa o valor de conveniência de cada elevador de atender à chamada quando é realizada uma nova chamada, sendo expressada *valor de prioridade*. Esta variável caracteriza o elevador através de uma menção numérica cujo universo de discurso compreende valores entre 1 e 100, quanto maior a menção, maior a prioridade de um elevador para o atendimento das novas chamadas.



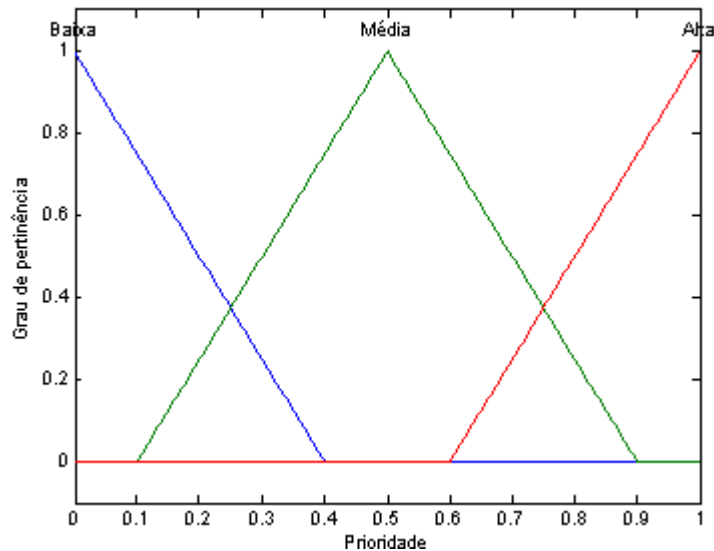


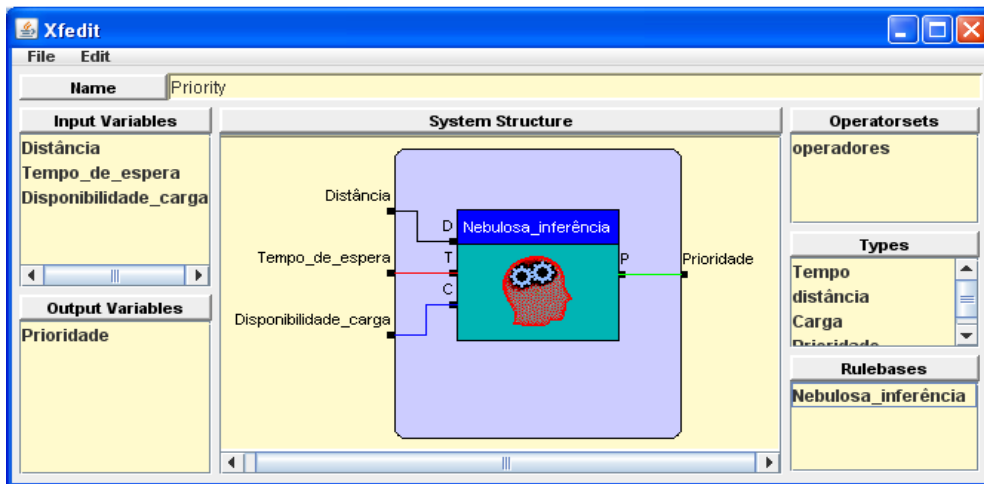
Figura 3-9 – Função de pertinência da variável prioridade

A **Figura 3-9** apresenta a função de pertinência da variável *Prioridade*, a qual tem uma distribuição não simétrica no universo de discurso. Nota-se, que no gráfico é considerado a região média com maior área que nas regiões alta e baixa, indicando uma maior probabilidade de ocorrerem valores médios. Os valores limites adotados para cada função de pertinência foram atribuídos com base em Kim (1998).

Utilizou-se uma ferramenta de CAD para sistemas nebulosos chamada *Xfuzzy 3.0*<sup>3</sup>. Este é um ambiente de desenvolvimento para sistemas de inferência baseados em lógica nebulosa. Além disso, esta ferramenta possibilita a realização de síntese para gerar descrições em linguagens de alto nível para implementações em *software* (*C*, *C++* e *JAVA*) (Cabrera, 2003, Xfuzzy, 2005). A Figura 3-10 representa a máquina de inferência implementada usando a ferramenta *Xfuzzy*. Na mesma figura, podem-se observar as variáveis de entrada consideradas (*Distância*, *Disponibilidade de Carga* e *Tempo de Espera*), as quais são utilizadas simultaneamente nas regras de decisão para fins de atribuição de prioridade aos diversos carros componentes do grupo, de modo a determinar aquele que se apresente como o mais conveniente para atender às chamadas de pavimento. O sistema de controle desenvolvido foi realizado com o objetivo de diminuir o tempo de espera dos passageiros.

---

<sup>3</sup> Criada e fornecida livremente pelo grupo Xfuzzy do Instituto de Microelectrónica de Sevilla (IMSE) do Centro Nacional de Microelectrónica (CNM) da Espanha, localizado na Universidade de Sevilla (Xfuzzy, 2005).



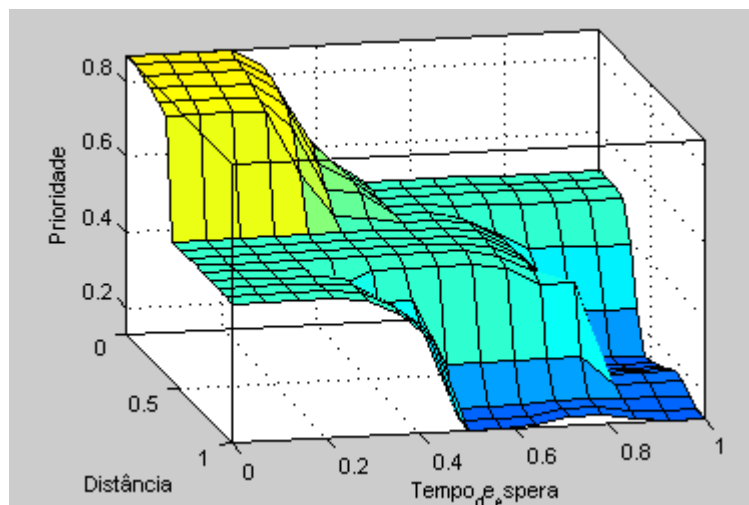
**Figura 3-10-** Máquina de inferência nebulosa para cálculo de prioridades

De modo a possibilitar o teste de desempenho do controlador proposto nas diversas condições de tráfego, normalmente encontradas ao longo do dia em edifícios comerciais, decidiu-se por classificá-las em três grupos: *Up-Peak* (pico de demanda de chegada de trabalhadores); *Business Time* (horário comercial); e *Down Peak* (demanda de saída de trabalhadores). As funções de pertinência e as regras de inferência do controlador nebuloso foram desenvolvidas para definir o comportamento do sistema de controle com base nessa classificação. Na Tabela 3-1, apresentam-se as regras de inferência propostas neste trabalho, as quais têm o objetivo de relacionar as variáveis de entrada às propriedades de saída do sistema.

Tabela 3-1 – Base de regras

	Regras de inferência	Prioridade
R1	SE(Distância ==Alto & Tempo de espera == Alto)	Baixa
R2	SE(Distância == Alto& Tempo de espera == Baixo)	Média
R3	SE(Distância == Alto& Tempo de espera == Médio)	Baixa
R4	SE(Distância == Baixo & Tempo de espera == Alto)	Média
R5	SE(Distância == Baixo & Tempo de espera == Baixo)	Alta
R6	SE(Distância == Baixo & Tempo de espera == Médio)	Média
R7	SE(Distância == Médio& Tempo de espera == Médio)	Média
R8	SE(Distância == Médio & Tempo de espera == Alto)	Baixa
R9	SE(Disponibilidade de Carga == Baixo & Tempo de espera == Alto)	Baixa
R10	SE(Distância == Baixo & Disponibilidade de Carga == Baixo)	Baixa
R11	SE (Disponibilidade de Carga == Médio &Tempo de espera == Baixo)	Alta

As regras R1 até R8 tendem a dar preferência a distâncias e tempos de espera menores, simultaneamente, em qualquer situação de tráfego (fluxos de passageiros baixos e médios). As regras R9 até R11 tendem a considerar valores reduzidos de prioridade a carros que tenham baixa disponibilidade de carga quando ocorrem situações de tráfego crítico (fluxo de passageiros alto). Estas implicações podem ser conferidas nas curvas de controle do sistema nebuloso apresentado na Figura 3-11, na qual se observou comportamento da variável de saída, *valor de prioridade*, de um elevador em função do *tempo de espera* e da *distância*.



**Figura 3-11** – Curva de controle do sistema de lógica nebulosa

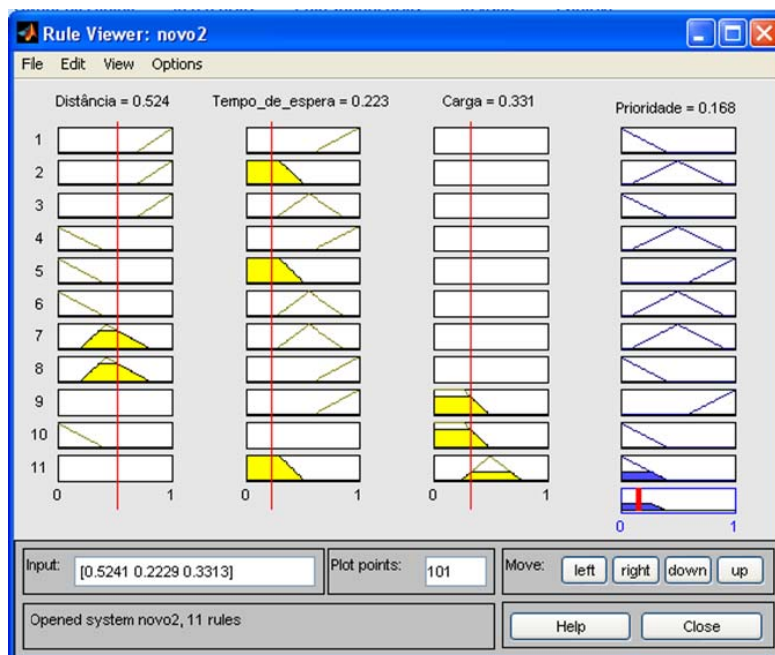
Observa-se na Figura 3-11 que, a partir dos valores baixos e médios das variáveis de entrada (*distância* e do *tempo de espera*), os valores de prioridade começam a aumentar, enfatizando os valores baixos das mesmas variáveis.

### Sintonização do controlador nebuloso

Após planejar os parâmetros estruturais (número de variáveis de entrada, funções de pertinência, variáveis lingüísticas, etc.) do sistema de inferência nebuloso, foram planejados os parâmetros de sintonização. A sintonização do controlador é uma tarefa complexa devido à flexibilidade que decorre da existência de muitos parâmetros. Portanto, o sistema de inferência nebuloso foi testado para um caso particular de tráfego majoritário de subida (*up-peak*), com um grupo de três elevadores e oito andares. Para se ter uma idéia do comportamento do controlador, realizaram-se diversas simulações

em que os parâmetros eram escolhidos de forma aleatória (tentativa e erro). Por meio da análise dos resultados, verificava-se para cada simulação se o comportamento do controlador era melhor o pior e, em caso de discordância, realizavam-se ajustes nas funções de pertinência das variáveis de modo a melhorar o comportamento do controlador. Uma das simulações realizadas com o sistema é mostrada na Figura 3-12.

Neste caso particular, observa-se que para os valores escolhidos para as variáveis de entrada, somente a regra R11 é ativada. Entretanto, a saída para esta regra situa-se na região denominada *Baixo* da função de pertinência de saída (*Prioridade*). Isso resulta na atribuição de um valor de prioridade baixo ao carro correspondente. Conseqüentemente, quando se tem tempo de espera de 0.223 (relativamente baixo) e disponibilidade de carga de 0.331(média- baixa), obtém-se na saída uma prioridade baixa de 0.168. Numa primeira avaliação, pode-se considerar esse resultado como inadequado, uma vez que o tempo de espera é baixa e alguma disponibilidade de carga no carro lhe dão possibilidade de atender a uma chamada. Entretanto, considerando o conjunto dos elevadores, o tipo de tráfego e as condições de demanda impostas ao grupo, esta resposta é considerada adequada. Para obtenção daquele valor, utilizou-se o método de defuzzificação de centro de área (seção 2.5.6). Esse procedimento foi realizado em conjunto com o desenvolvimento de *software*, permitindo o ajuste dos parâmetros do controlador e as funções de pertinência, buscando-se melhorar o desempenho do controlador.

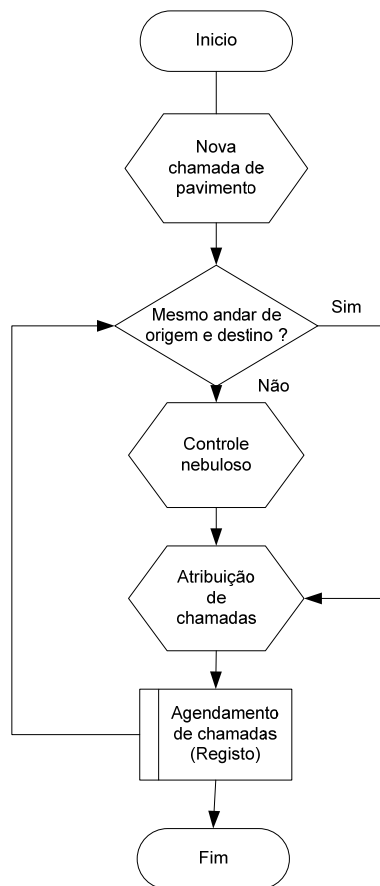


**Figura 3-12**–Simulação do sistema de lógica nebulosa

Após terminar cada simulação do sistema nebuloso foi realizada a síntese para gerar as descrições em linguagem JAVA, com o objetivo de ser incluído o código dentro do desenvolvimento do sistema *software* (cliente) a ser modelado por meio da modelagem UML que será apresentada no próximo capítulo.

### **Algoritmo de ordenamento**

O algoritmo de ordenamento de passageiros é parte da estratégia de controle da arquitetura moderna de elevadores usando o sistema controle de destino (SCD). Este algoritmo examina os registros de agendamento das chamadas de pavimento atribuídas de cada elevador, como o objetivo de realizar uma comparação de estas chamadas, com as novas chamadas a ser avaliadas pelo controlador nebuloso. Portanto, aqueles passageiros com exatamente a mesma trajetória mesmo andar de origem e destino são colocados em um mesmo elevador. Desta maneira, as novas chamadas a ser avaliadas que sejam atribuídas pelo algoritmo de ordenamento não serão avaliadas pelo sistema de controle nebuloso. Na Figura 3-13 apresenta-se o funcionamento do algoritmo de ordenamento, em que chega a nova chamada de pavimento, depois passa ao bloco de decisão (mesmo andar de origem e destino?), de modo a escolher se a nova chamada de pavimento passa ao controle nebuloso ou ao bloco de atribuições de chamadas. No caso de que não ter chamadas agendadas, a nova chamada passa pelo controle nebuloso para realizar a atribuição da chamada.



**Figura 3-13** – Funcionamento do Algoritmo de ordenamento

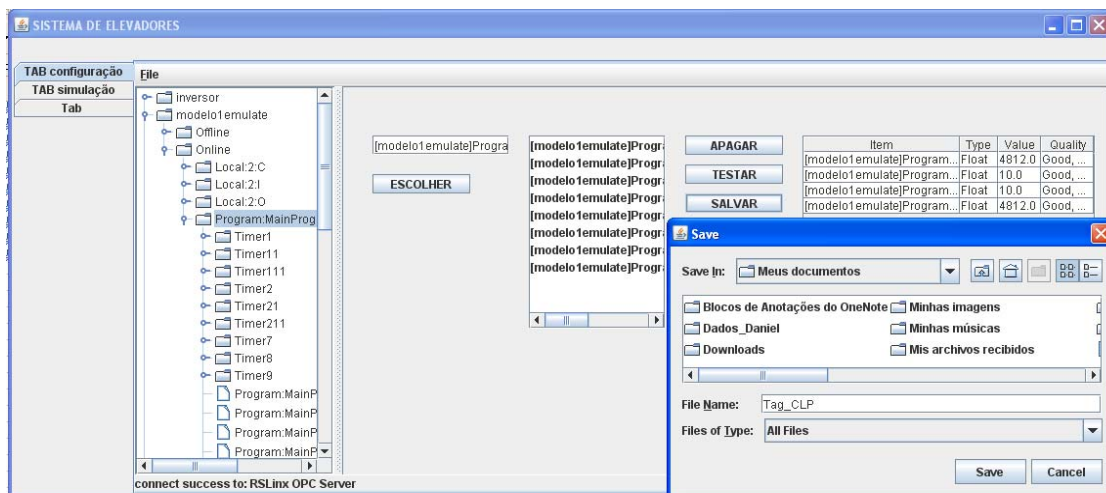
### 3.2.4 - Execução das simulações

No sistema de simulação foram realizados vários testes para avaliação de desempenho do sistema de elevadores. O procedimento de teste era o seguinte:

1) Gerar os arquivos de chama das que estivessem de acordo com os parâmetros de tráfego a simular, neste caso em particular, o tipo de tráfego escolhido foi *up-peak*, por ser a situação mais crítica nos sistema de elevadores. Deste modo, foram gerados nove arquivos com diferentes fluxos de tráfego, segundo os parâmetros de dimensionamento de elevadores descritos no capítulo dois seção 2.2.8. Durante essa primeira etapa de testes, foi utilizada a ferramenta desenvolvida em JAVA para a geração de chamadas (seção 3.2.3).

2) A segunda etapa do teste compreendeu o uso do simulador desenvolvido em JAVA. Para o uso do simulador é necessário configurar cada parte desta ferramenta.

a) A primeira parte relaciona-se à geração de um arquivo com todas as *tag's*<sup>4</sup> a serem usadas no simulador. Essas *tag's* dependem do algoritmo realizado no emulador de CLP, no caso em particular foi realizado um algoritmo para representar a dinâmica de três elevadores (ver Figura 3-2). Assim, por meio da primeira interface visual, *Configuração Comunicação* (Figura 3-14), gerou-se o arquivo de *tag's*.



**Figura 3-14** – Interface gráfica de configurar comunicação

b) A segunda parte relaciona-se à inicialização da configuração do simulador, em que se informa ao mesmo o arquivo realizado na etapa anterior (arquivo de *tag's*). Para iniciar a configuração é necessário estabelecer a comunicação entre o simulador e o emulador, o que é realizado por meio do protocolo OPC, em que o simulador é o cliente e o software RSLinx é o servidor. O arquivo de *tag's* é usado pelo cliente para definir quais variáveis serão acessadas. A ação de leitura do arquivo de *tag's* é ativada pelo botão *configuração leitura*, localizado na interface mostrada na

Figura 3-15. Além da leitura das *tag's*, é também necessário configurar a arquitetura do sistema de elevadores (número de carros, número de andares do prédio capacidade de transporte), as posições iniciais dos carros e disponibilidade de carga inicial de cada elevador. Para realizar o sincronismo é necessário

---

<sup>4</sup>Uma *tag*, ou **etiqueta**, é uma palavra-chave (relevante) ou termo associado com uma informação que descreve e permite uma classificação da informação.

configurar o relógio do computador com a hora de início dos dados de entrada do simulador, que é fornecida pelo arquivo de geração de chamadas.

Call ID	Andar Destino	Andar Origem	No Passageiros	Tempo do voo	Tempo das portas	Sentido do elevador	Movimento do elevador	Distance	Loading	Waiting Time	Prioridade
1	2	2	3	0.0	14418.0	0.0	2	0 %	50 %	19 %	85.1
2	4	5	2	6650.0	1000	0.0	5	2 %	66 %	19 %	80.1
3	8	7	1	6623	2000	0.0	7	2 %	83 %	12 %	78.51

**Figura 3-15** – Interface gráfica de simulação de chamadas

c) A terceira e última parte relaciona-se à leitura do arquivo de chamadas de pavimentos, gerado na primeira etapa do procedimento aqui descrito, e ao disparo do relógio para iniciar a simulação propriamente dita, o que é realizado acionando-se o botão simulação chamadas, mostrado na

Figura 3-15. O sistema de simulação espera até que o relógio do computador seja igual ao tempo que está no arquivo da primeira chamada de pavimento. No momento em que são iguais, o simulador começa a interagir com todos seus componentes (ver Figura 3-2) . O mesmo procedimento foi realizado para cada fluxo, ou seja, foram realizados nove testes. Para possibilitar uma análise estatística dos dados gerados pelo simulador, foi necessário gerar arquivos com os tempos de espera de cada passageiro. O botão coletar dados foi introduzido no simulador para essa finalidade.



### **3.2.5 - Análise dos resultados das simulações**

A análise dos resultados das simulações é uma das principais etapas a serem realizadas na metodologia seguida neste trabalho. Somente, por meio da análise estatística dos dados de saída de um modelo de simulação é que se poderá garantir respaldo científico para compará-los com outros resultados ou ainda para julgar sua adequação a alguma teoria. Realizaram-se várias simulações seguindo o procedimento descrito no item anterior, cujos resultados são mostrados e comentados no capítulo cinco.

#### 4 - MODELAGEM UML DO SISTEMA DE GRUPO DE ELEVADORES

A Linguagem de Modelagem Unificada (UML) é uma linguagem gráfica padrão para a elaboração da estrutura de projetos complexos de software. A UML possibilita a visualização de todos os passos de desenvolvimento de um problema de modelagem na forma de diagramas padronizados, envolvendo elementos estruturais (classes, objetos, relacionamentos, etc.), de comportamento (*casos de uso*, iteração, máquinas de estado), de agrupamento (pacote, modelo, subsistema, etc.) e de relacionamento (composição, agregação, generalização). A grande importância da modelagem é que permite avaliar a aderência e a qualidade da arquitetura, por parte do desenvolvedor, de modo a possibilitar-lhe detectar e corrigir falhas antes que estas possam comprometer o sucesso do projeto. A modelagem visual permite que os detalhes do processo sejam expostos ou escondidos conforme as necessidades, auxiliando o desenvolvimento de projetos complexos e extensos. Além disso, a UML ajuda a manter a consistência entre a especificação e a implementação por meio do desenvolvimento iterativo e do planejamento de testes em cada iteração.

Nesse contexto, este trabalho apresenta a modelagem de um simulador de tráfego de grupo de elevadores nebuloso usando a metodologia de processo unificado de desenvolvimento de software proposta pela Linguagem Unificada de Modelagem (UML). O desenvolvimento da modelagem proposta foi feito de modo a se possibilitar sua realização em uma arquitetura física de um sistema de elevadores moderno, usando instrumentação industrial, como foi apresentado no capítulo um (ver Figura 1-1).

O desenvolvimento de *software* do simulador foi realizado usando a metodologia de processo unificado de desenvolvimento de software (RUP), o qual compreende uma série de fluxos de trabalho (Torrosi, 2010):

Tabela 4-1 – Fluxos de trabalho do processo unificado

<b>Disciplina</b>	<b>Modelos</b>
Requerimentos	Modelo de caso de uso
Análise	Modelo de análise
Planejar	Modelo de planejamento
Implementação	Modelo de implementação
Testes	Modelo de testes

Por simplicidade de apresentação da modelagem UML do simulador, mostra-se a construção final do sistema com seus diagramas de caso de uso e diagramas de planejamento.

#### 4.1 – CARACTERÍSTICAS PRINCIPAIS DO SISTEMA

A metodologia estabelece a base necessária para configurar o processo de desenvolvimento do sistema com as seguintes características:

Tabela 4-2- Características principais do sistema

<b>Identificador</b>	<b>Definição</b>	<b>Conjunto de variáveis</b>
Solicitar Andar	O passageiro pressiona o botão colocando o andar de destino. No sistema de simulação esta informação será fornecida por um arquivo.	<b>Estado:</b> Aprovado <b>Prioridade:</b> Critica <b>Nível de Risco:</b> Ordinário
Escolher Elevador	Deve-se exigir o elevador mais adequado de acordo com a estimação das entradas do sistema de lógica nebulosa e a prioridades para cada elevador.	<b>Estado:</b> Aprovado <b>Prioridade:</b> Critica <b>Nível de Risco:</b> Ordinário
Configurar Comunicação	O operário terá que configurar as variáveis pela comunicação com o CLP	<b>Estado:</b> Aprovado <b>Prioridade:</b> Critica <b>Nível de Risco:</b> Médio
Configurar Controle	O operário configura as variáveis consideradas no controle de elevadores.	<b>Estado:</b> Aprovado <b>Prioridade:</b> Critica <b>Nível de Risco:</b> Ordinário
Enviar dados em tempo real	O emulador do CLP envia dados dos eventos no sistema de simulação de elevadores, tais como: A posição atual do elevador, o tempo de vôo do elevador e o tempo de abrir e fechar portas, informação necessária para operar o sistema de controle.	<b>Estado:</b> Aprovado <b>Prioridade:</b> Critica <b>Nível de Risco:</b> Ordinário

## 4.2 – MODELO DE CASO DE USODO SISTEMA

A modelagem do simulador de controle difuso de tráfego de elevadores é apresentada na Figura 4-1 por meio dos diagramas de *casos de uso*<sup>5</sup>. O diagrama de *casos de uso* propicia uma visão geral do sistema (Rumbaugh et al, 1999).

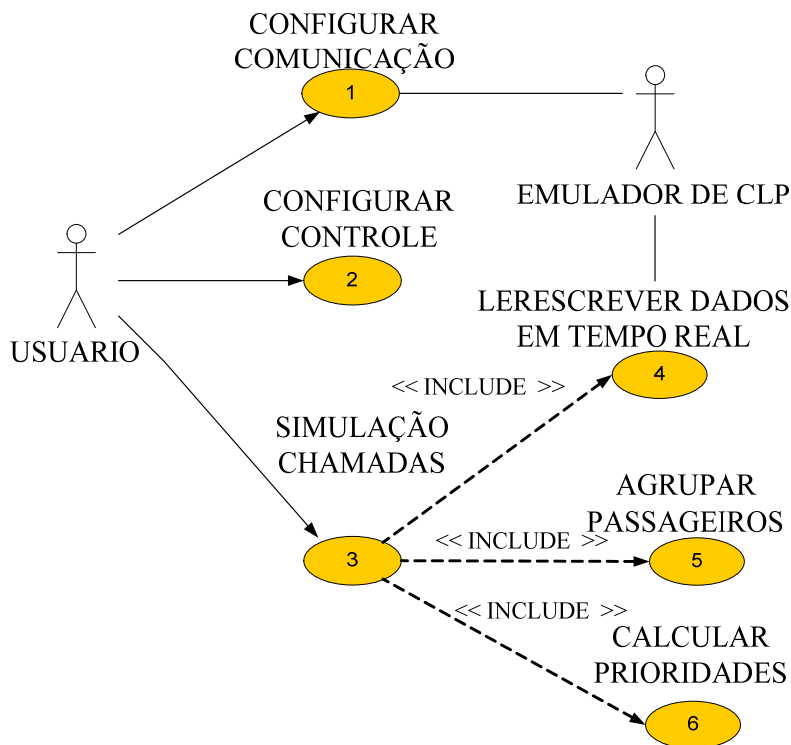


Figura 4-1 – Casos de uso do sistema

Desta maneira, a Figura 4-1 apresenta o sistema composto por um ator usuário que representa a pessoa que inicia a configuração do sistema (pessoa que lida com o sistema para realizar as simulações). “**Configurar comunicação**” é o primeiro *caso de uso* do sistema onde o usuário através do servidor OPC escolhe os parâmetros necessários para o funcionamento do sistema. O *caso de uso* “**Configurar Controle**” é onde se configura: (a) o número de andares, (b) o número de elevadores e (c) a capacidade do sistema. No *caso de uso* “**Simular Chamadas**” o usuário inicia o controle grupal de elevadores e a sua simulação. No *caso de uso* “**Ler-escrever dados em tempo real**” o sistema lê e escreve dados no emulador de CLP por meio do

<sup>5</sup> Na engenharia de software, o termo “caso de uso” é um tipo de classificador representando uma unidade funcional coerente provida pelo sistema, subsistema, ou classe manifestada por seqüência de mensagens entre os sistemas e um ou mais atores.

protocolo OPC. No *caso de uso* “**Agrupar Passageiros**”, o sistema agrupa os passageiros com um mesmo destino. No *caso de uso* “**Calcular Prioridade**” o sistema, por meio do controle de lógica nebulosa, calcula qual elevador é o mais conveniente para atender a chamada de pavimento (Patiño A., 2010).

Cada *caso de uso* representa uma unidade funcional do sistema, a qual pode ser explicada em detalhe por meio de diagramas. Neste caso, cada caso de uso é explicado por meio de cenários<sup>6</sup>, diagramas de seqüências e diagramas de classes (modelo de planejamento). Na continuação serão apresentados os cenários de cada *caso de uso* do sistema:

Caso de uso No.1 :	Configurar Comunicação
Propósito:	Que o usuário possa configurar os parâmetros necessários pela comunicação com o emulador de CLP.
Resumo:	O usuário configura a comunicação escolhendo do servidor OPC as variáveis programadas no emulador de CLP para o funcionamento do sistema.
Requerimentos:	- Servidor OPC instalado no computador.
Precondições:	- Que o usuário conheça os parâmetros do emulador de CLP
<b>CENÁRIO</b>	
<b>Atores</b>	<b>Simulador de tráfego de elevadores</b>
1.O usuário escolhe a primeira aba. (ver Figura 4-2)	1a. O sistema apresenta o formulário para a configuração da comunicação
2. O usuário escolhe a opção file, <i>connect</i>	2a.O sistema apresenta a lista de servidores instalados no computador (Figura 4-3) .
3. O usuário escolhe o servidor a ser conectado. (ver Figura 4-3)	3a. O sistema mostra uma árvore com os endereços das <i>tag's</i> da memória do emulador de CLP.
4. O usuário escolhe os endereços das variáveis a serem usadas que são fornecidas pela árvore de <i>tag's</i> .	4.a. O sistema mostra as opções escolhidas em uma tabela.
5. O usuário testa cada variável ( <i>tag's</i> ) para conferir uma boa conexão, pressionando botão teste.	5.a O sistema lê cada variável ( <i>tag's</i> ) do emulador de CLP.
6. O usuário pressiona o botão salvar, para salvar todos os endereços de memória escolhidos.	6a. O sistema mostra as variáveis ( <i>tag's</i> ) escolhidas em uma lista.
7. O usuário pressiona o botão apagar para eliminar os endereços que não vai usar.	7a. O sistema elimina esse endereço da lista.
8. O usuário pressiona o botão salvar para armazenar os endereços de memória escolhidos.	8a. O sistema salva todos os endereços de memória que estão na lista.

<sup>6</sup>Cenário é uma instância de um caso de uso: um caminho entre os muitos possíveis. Cenários podem ser representados por diagramas de seqüência.

9. O usuário escolhe a opção <i>disconnect</i>	9a. O sistema termina a conexão com o servidor.
Pós-condição:	Sistema de comunicação instalado e configurado com os parâmetros especificados do emulador de CLP.

### Fluxos Alternativos

<b>Atores</b>	<b>Simulador de tráfego de elevadores</b>
1. O usuário pode terminar a configuração.	1a. O sistema fecha o formulário de configuração e não salva as modificações.

### Interfaces Requeridas:

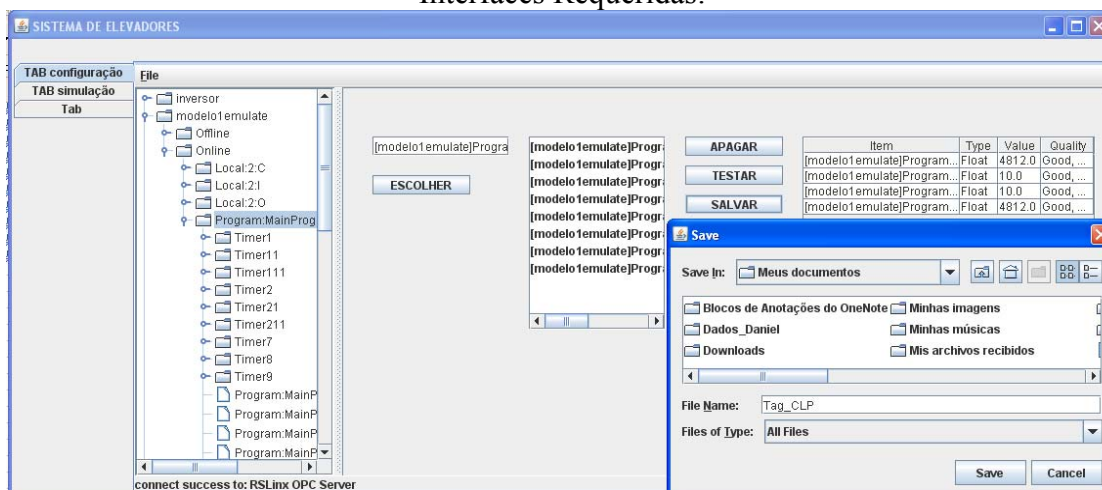


Figura 4-2–Interface de configurar comunicação

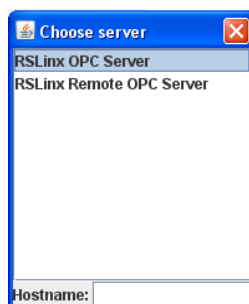


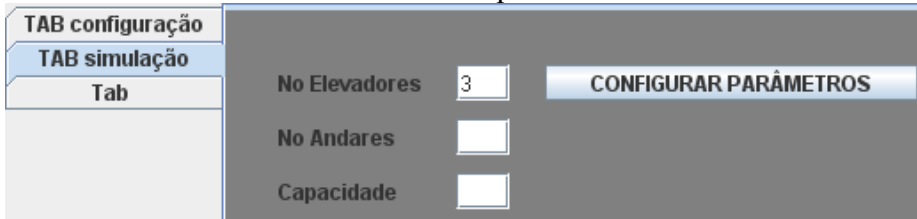
Figura 4-3 – Interface para escolher servidor

Requerimentos não funcionais:	Nenhum até o momento
-------------------------------	----------------------

Caso de uso No.2 :	Configurar Controle
Propósito:	O usuário configura o número de andares e a capacidade do sistema.
Resumo:	O usuário configura os parâmetros do prédio tais como: número de andares e número de pessoas a serem atendidas.
Requerimentos:	- Configuração do sistema de comunicação com o emulador de CLP.
Precondições:	- Conexão com sucesso.

### CENÁRIO

<b>Atores</b>	<b>Simulador de tráfego de elevadores</b>
1. O usuário escolhe a segunda aba.	1a. O sistema mostra o formulário em que

	poderão ingressar os dados
2. O usuário ingressa o número de andares e a capacidade de pessoas que serão atendidas. Pressiona o botão configurar parâmetros.	2a. O sistema armazena os dados ingressados pelo usuário.
Pós-condição:	Configurar comunicação com sucesso
<b>Fluxos Alternativos</b>	
<b>Atores</b>	<b>Simulador de tráfego de elevadores</b>
1. O usuário pode terminar a configuração de controle.	1a. O sistema fecha o formulário de configuração e não salva as modificações.
Interfaces Requeridas:	
	
Figura 4-4 – Interface de configuração de controle	
Requerimentos não funcionais:	Nenhum até o momento

Caso de uso No.3 :	Simulação de chamadas
Propósito:	O usuário inicia o sistema de simulação de tráfego de elevadores
Resumo:	O usuário inicia o sistema de simulação de tráfego com o objetivo de obter os tempos de espera dos passageiros e o número de paradas para avaliar o desempenho do controlador.
Requerimentos:	- Servidor OPC instalado no computador e que “configurar comunicação” seja configurado com sucesso.
Precondições:	-O usuário tenha configurado a comunicação com o emulador de CLP e configurado os parâmetros do configurar controle.

### CENÁRIO

<b>Atores</b>	<b>Simulador de tráfego de elevadores</b>
1. O usuário coloca os valores iniciais para o andar de origem, andar de destino e número de passageiros para cada elevador e logo pressiona o botão configuração leitura.	1a. O sistema mostra uma janela onde se pode buscar o arquivo em que estão armazenadas as variáveis a serem lidas do emulador de CLP (arquivo salvo por configurar comunicação).
2. O usuário escolhe o arquivo onde estão os endereços de memória do emulador CLP (configurado no caso de uso configurar comunicação).	2a. O sistema armazena os endereços de memória dentro de vetores com o objetivo de estabelecer a conexão das variáveis do emulador de CLP via OPC.
3. O usuário pressiona o botão start para iniciar o sistema e colocar os elevadores em uma posição inicial.	3a. O sistema envia os valores iniciais ao emulador de CLP com o objetivo de ler o tempo de voo programado no emulador segundo os valores iniciais fornecidos, como também o tempo de abrir e fechar

	portas e o sentido de viagem do elevador.
5. O usuário pressiona o botão simulação chamadas.	5a. O sistema mostra uma janela onde se escolhe o arquivo em que está o valor do andar de origem, andar de destino e tempo de realização da chamada.
6. O usuário escolhe o arquivo onde se encontram os parâmetros de entrada do sistema (andar de origem, andar de destino e tempo da chamada). (ver Figura 4-6)	6a. O sistema trabalha com o relógio do computador, o qual compara o tempo da chamada do arquivo de entrada com o tempo atual do computador. Desta maneira, quando os tempos são iguais o sistema avalia para qual elevador é atribuída a chamada.
7. O usuário pressiona o botão coletar dados, quando termina a simulação e salva cada tempo de espera dos passageiros.	7a. O sistema cria um vetor com todos os tempos de cada passageiro e salva esta informação em um arquivo.
Pós-condição:	O sistema de tráfego de elevadores é iniciado
<b>Fluxos Alternativos</b>	
<b>Atores</b>	<b>Simulador de tráfego de elevadores</b>
1. O usuário pressiona o botão calcular prioridade.	1a. O sistema calcula a prioridade de forma manual de cada elevador para um destino específico.
Interfaces Requeridas:	
<p>The screenshot shows a simulation interface with the following elements:</p> <ul style="list-style-type: none"> <li><b>Global Controls:</b> <ul style="list-style-type: none"> <li>No Elevadores: 3</li> <li>No Andares: 8</li> <li>Capacidade: 450</li> <li>Buttons: SIMULAÇÃO CHAMADAS, DESCONEXÃO, CONFIGURAR PARÂMETR..., COLETAR DADOS</li> </ul> </li> <li><b>Elevator 1 Data:</b> <ul style="list-style-type: none"> <li>Andar Destino: 2, Andar Origem: 2, No Passageiros: 3</li> <li>Tempo do voo: 0.0, Tempo das portas: 14418.0, Sentido do elevador: 0.0</li> <li>Movimento do elevador: 2</li> <li>Distance: 0 %, Prioridade: 85.1, Loading: 50 %, Wating Time: 19 %</li> </ul> </li> <li><b>Elevator 2 Data:</b> <ul style="list-style-type: none"> <li>Andar Destino: 4, Andar Origem: 5, No Passageiros: 2</li> <li>Tempo do voo: 6650.0, Tempo das portas: 1000, Sentido do elevador: 0.0</li> <li>Movimento do elevador: 5</li> <li>Distance: 2 %, Prioridade: 80.1, Loading: 66 %, Wating Time: 19 %</li> </ul> </li> <li><b>Elevator 3 Data:</b> <ul style="list-style-type: none"> <li>Andar Destino: 8, Andar Origem: 7, No Passageiros: 1</li> <li>Tempo do voo: 6623, Tempo das portas: 2000, Sentido do elevador: 0.0</li> <li>Movimento do elevador: 7</li> <li>Distance: 2 %, Prioridade: 78.51, Loading: 83 %, Wating Time: 12 %</li> </ul> </li> </ul>	
Figura 4-5 - Interface simulação de chamadas	



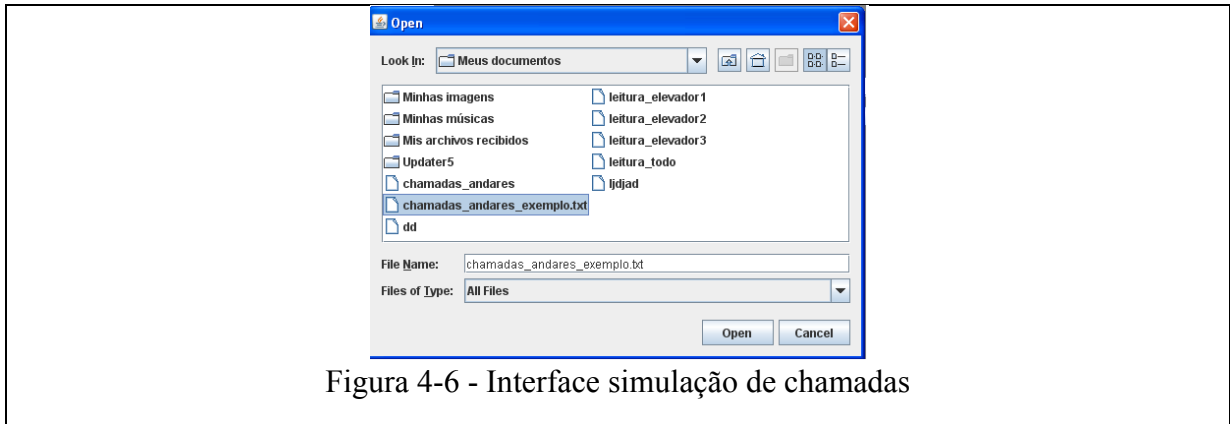


Figura 4-6 - Interface simulação de chamadas

Requerimentos não funcionais:	Nenhum até o momento
-------------------------------	----------------------

Caso de uso No.4 :	Ler-escrever dados em tempo real
Propósito:	O sistema lê do emulador de CLP por meio do servidor OPC, os dados de tempo de viagem de cada elevador, o tempo de abrir e fechar portas e o sentido dos elevadores. O sistema escreve no emulador de CLP por meio do servidor OPC, os dados necessários para simular a dinâmica dos elevadores.
Resumo:	O sistema lê e escreve no emulador de CLP por meio do servidor OPC os dados necessários para iniciar a simulação.
Requerimentos:	- Servidor OPC instalado no computador e que “configurar comunicação” seja configurado com sucesso.
Pré-condições:	-O usuário tenha configurado a comunicação com o emulador de CLP e configurado os parâmetros do configurar controle.

**CENÁRIO**

Atores	Simulador de tráfego de elevadores
	1a. O sistema se conecta ao servidor OPC instalado no computador, enviando os endereços de memória das variáveis para ler e escrever. 1b. O sistema lê os dados das variáveis que foram configuradas para serem mostradas em tela. 1c. O sistema escreve os dados iniciais para colocar os elevadores em uma posição inicial. 1d. Quando o tempo do relógio do computador é igual ao tempo da chamada, o sistema ativa calcular prioridade e agrupar passageiros.
Pós-condição:	Dados necessários lidos, mostrados em tela.
Fluxos Alternativos	
Atores	Simulador de tráfego de elevadores
Não tem	
Interfaces Requeridas:	

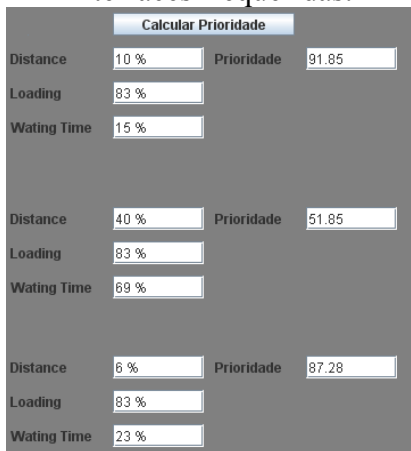
No Elevadores	3	SIMULAÇÃO CHAMADAS		DESCONEXÃO					
No Andares	8	CONFIGURAR PARÂMETR...							
Capacidade	450	COLETAR DADOS							
18:50:25									
Andar Destino	2	Tempo do voo	0.0	Configuração leitura	Calcular Prioridade	Tempo Max.	No. Paradas		
Andar Origem	2	Tempo das portas	14418.0	Start	Distance	0 %	Prioridade 85.1	125000	2
No Passageiros	3	Sentido do elevador	0.0	Loading	50 %	40000			
		Movimento do elevador	2	Waiting Time	19 %				
Andar Destino	4	Tempo do voo	6650.0						
Andar Origem	5	Tempo das portas	1000	Distance	2 %	Prioridade 80.1	200000	2	
No Passageiros	2	Sentido do elevador	0.0	Loading	66 %	56000			
		Movimento do elevador	5	Waiting Time	19 %				
Andar Destino	8	Tempo do voo	6623						
Andar Origem	7	Tempo das portas	2000	Distance	2 %	Prioridade 78.51	200000	3	
No Passageiros	1	Sentido do elevador	0.0	Loading	83 %	300000			
		Movimento do elevador	7	Waiting Time	12 %				

Figura 4-7– Interface simulação chamadas (resultados)

Requerimentos não funcionais:	Nenhum até o momento
-------------------------------	----------------------

Caso de uso No.5 :	Agrupar Passageiros
Propósito:	Agrupar os passageiros com o mesmo andar de destino, usando a informação do andar de destino fornecido pelo passageiro antes de entrar na cabine (Sistema de Controle de Destino).
Resumo:	O sistema agrupa os passageiros com o mesmo destino em um mesmo elevador usando o Sistema de Controle de Destino (SCD)
Requerimentos:	-Servidor OPC instalado no computador, configuração comunicação com sucesso com o emulador de CLP, número de passageiros de cada elevador e chamadas agendadas de cada elevador.
Precondições:	-O usuário ter pressionado a opção simulação de chamada.
<b>CENÁRIO</b>	
<b>Atores</b>	<b>Simulador de tráfego de elevadores</b>
1. O usuário pressiona o botão simulação de chamada.	1a. O sistema agrupa as pessoas com um mesmo andar de destino.
Pós-condição:	O sistema tem que estar informado sobre o andar de destino de cada chamada.
<b>Fluxos Alternativos</b>	
<b>Atores</b>	<b>Simulador de tráfego de elevadores</b>
Interfaces Requeridas: Não tem	
Requerimentos não funcionais:	Nenhum até o momento

Caso de uso No.6 :	Calcular prioridade
Propósito:	O sistema avalia qual elevador é o mais conveniente para atender as chamadas.

Resumo:	O sistema por meio da lógica nebulosa calcula qual dos três elevadores é o mais conveniente para atender as chamadas de cada andar.	
Requerimentos:	-Servidor OPC instalado no computador, configuração comunicação exitosa.	
Pré-condições:	- Servidor OPC instalado no computador, configuração comunicação com sucesso com o emulador de CLP, número de passageiros de cada elevador e chamadas agendadas de cada elevador.	
<b>CENÁRIO</b>		
<b>Atores</b>	<b>Simulador de tráfego de elevadores</b>	
1. O usuário pressiona o botão calcular prioridade	<p>1a.O sistema por meio do andar de destino e a posição atual do elevador, estima a distância e o tempo de espera.</p> <p>1b. O sistema por meio do número de passageiros calcula o número de vagas disponíveis em cada elevador.</p> <p>1c. O sistema com os valores de distância, tempo de espera e disponibilidade de carga calcula a prioridade para cada elevador.</p>	
Pós-condição:	Dados necessários lidos, mostrados em tela.	
<b>Fluxos Alternativos</b>		
<b>Atores</b>	<b>Simulador de tráfego de elevadores</b>	
Não tem		
<p>Interfaces Requeridas:</p>  <p>The screenshot shows a software interface titled 'Calcular Prioridade'. It contains three identical sections, each with three input fields: 'Distance' (with a percentage value), 'Loading' (with a percentage value), and 'Waiting Time' (with a percentage value). To the right of each section is a 'Prioridade' field displaying a numerical value. The first section shows 10% Distance, 83% Loading, 15% Waiting Time, and a Priority of 91.85. The second section shows 40% Distance, 83% Loading, 69% Waiting Time, and a Priority of 51.85. The third section shows 6% Distance, 83% Loading, 23% Waiting Time, and a Priority of 87.28.</p>		
<p>Figura 4-8 - Interface simulação de chamadas (prioridades)</p>		
Requerimentos não funcionais:	Nenhum até o momento	

### 4.3- MODELO DE PLANEJAMENTO

No modelo de planejamento geralmente se utiliza os diagramas de seqüência e diagramas de classes (seção 2.4) para representar as interações detalhadas entre os objetos atuantes no diagrama de casos de uso.

### 4.3.1 – Diagramas de seqüência

#### Caso de uso número 1: configurar comunicação

O *caso de uso* “**configurar comunicação**” tem seu diagrama de seqüência conforme mostrado na Figura 4-9. O usuário inicia o sistema para configurar a comunicação. O software executa o objeto *mygraphics* para apresentar a interface gráfica para que o usuário possa iniciar sua configuração. A primeira coisa que o usuário tem que fazer é ir à barra de menu “file” e selecionar o botão *connect*. Desta maneira, o *software* executa o objeto *myserver* e usa a biblioteca de OPC em JAVA, ativando o objeto *srvList* para que o software encontre todos os servidores OPC instalados no computador e o usuário possa escolher um. Assim, o *software* se conecta ao servidor com o objeto *server* e este envia todos os endereços das variáveis programadas no emulador de CLP, por meio de um *browser* com o objeto *tree*, para facilitar a escolha das variáveis a trabalhar. O usuário pode testar o estado de cada variável para comprovar que é a variável que precisa por meio do objeto *itemsTableModel*. Finalmente, o usuário salva uma lista dos endereços das variáveis a trabalha em um arquivo texto (arquivo de configuração da comunicação) (Patiño-Forero, 2010).

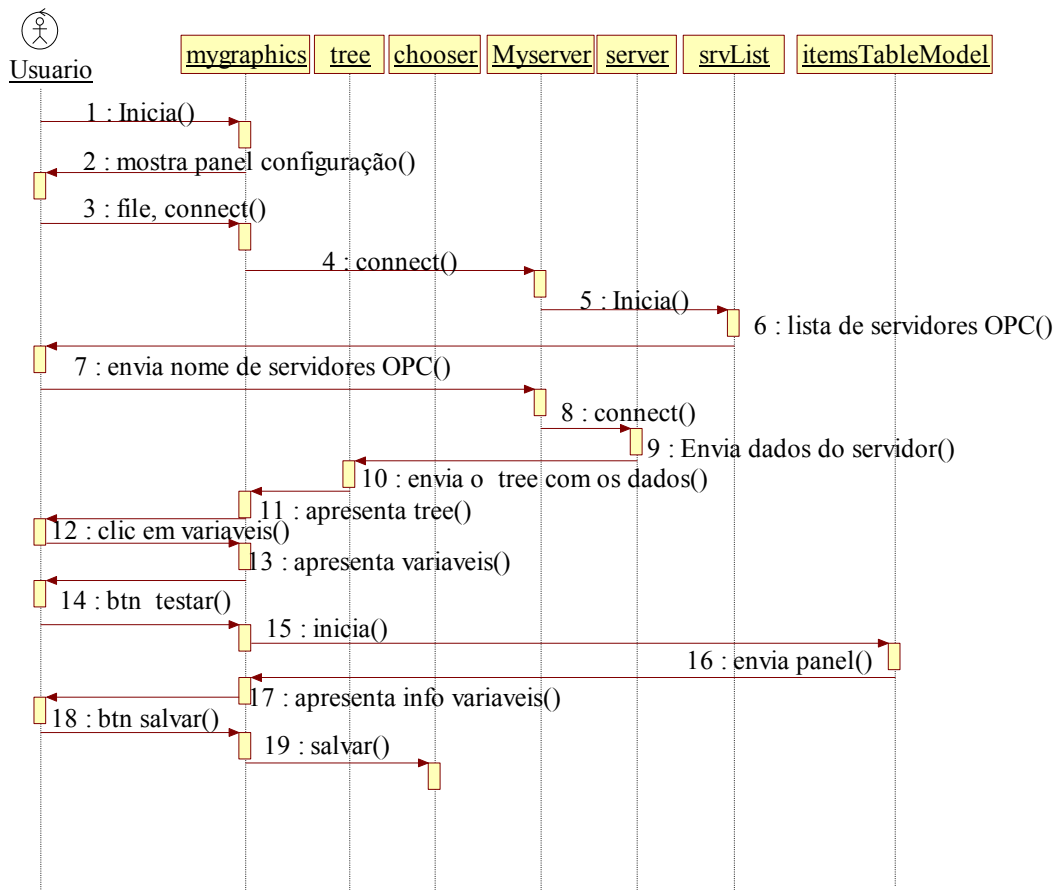


Figura 4-9 - Diagrama de seqüência de configurar comunicação

## Caso de uso número 2: Configurar controle

O caso de uso “**configurar controle**” tem seu diagrama de seqüência conforme mostrado na Figura 4-10. O usuário inicia o sistema para configurar o controle. O software executa o objeto *mygraphics* para apresentar a interface gráfica para que o usuário possa iniciar sua configuração. A primeira coisa que o usuário tem que fazer é escolher a aba simulação e colocar o numero de andares a trabalhar e o numero de população a ser atendida pelo sistema segundo os parâmetros do prédio a simular. O usuário pressiona o botão “configurar parâmetros” para salvar os dados e configurar o sistema por meio do objeto *controle sistema*.

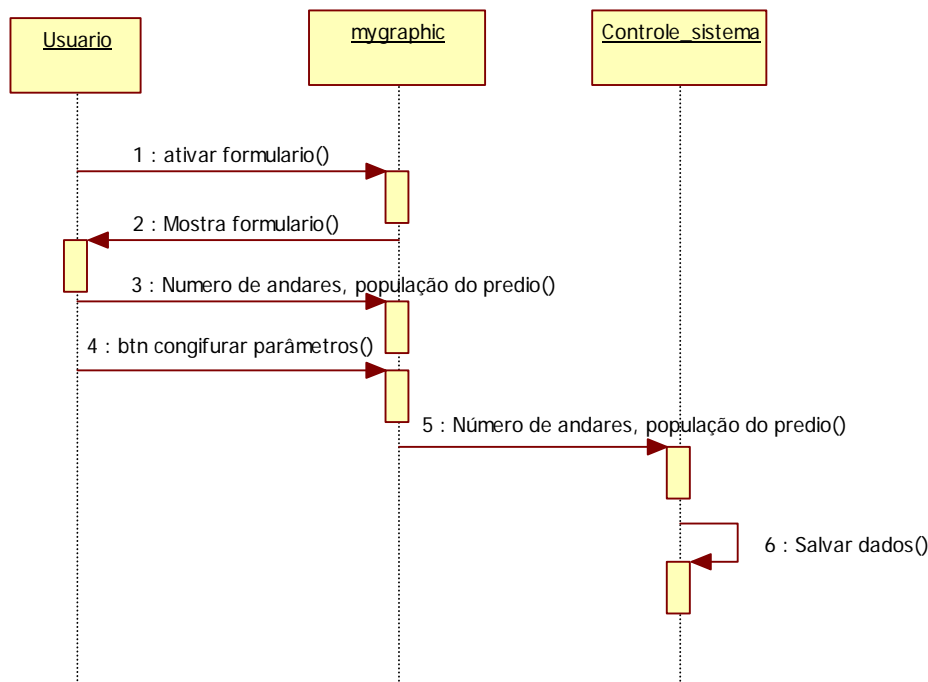


Figura 4-10 - Diagrama de seqüência de configurar controle

### Caso de uso número 3: Simulação chamada

O caso de uso “**Simular Chamadas**” tem seu diagrama de seqüência conforme mostrado na Figura 4-11. O usuário inicia o sistema para iniciar a simulação. O software executa o objeto *mygraphic* para apresentar a interface gráfica para que o usuário possa iniciar a configuração do sistema. A primeira coisa que usuário tem que fazer é colocar o andar origem e andar destino de cada elevador (valores iniciais). Logo, o usuário deve selecionar o botão *Configuração leitura* onde por meio de uma janela se escolhe o arquivo onde estão os endereços das variáveis do emulador para trabalhar dentro do simulador. O usuário depois disso, deve selecionar o botão *start*. Desta maneira, o simulador por meio do objeto *OPCLeitura* envia os valores iniciais da posição atual dos elevadores para o servidor OPC, a mesmo tempo o servidor lê os dados que apresentam mudanças dentro do emulador, de forma concorrente, por meio do método *OnvalueChanged()* da biblioteca OPC. Quando os elevadores chegam às posições iniciais, o usuário seleciona o botão *Simulação chamadas*, o qual dispara a apresentação de uma janela, por meio da qual escolhe-se o arquivo onde se mostram os andares origem, andares destino e tempos (este arquivo é criado pelo gerador de trafego de elevadores). Assim por meio do objeto *Leituradados* o sistema lê os dados de tempo da

simulação e os compara com o relógio do computador, de modo que, a cada instante de tempo em que a comparação de tempo é verdadeira, executa-se o sistema de controle difuso para escolher o elevador mais conveniente para cada chamada.

O *caso de uso* “**Lerescrever dados em tempo real**” seu diagrama de seqüência está incluído dentro do diagrama de seqüência de simular chamadas, devido ao modelo inicial de planejamento.

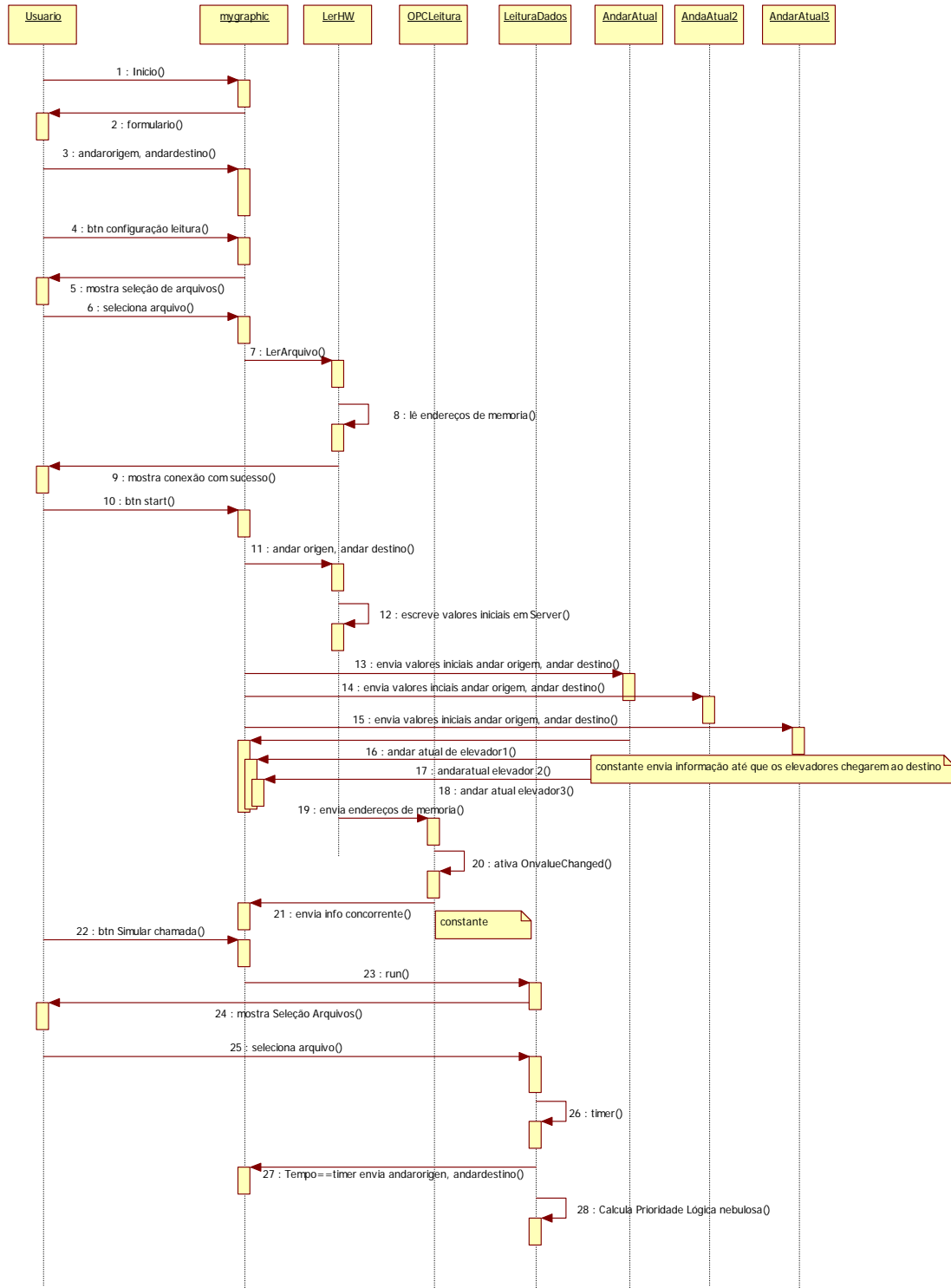


Figura 4-11 – Diagrama de seqüência de simulação chamada

### Caso de uso número 5: Agrupar passageiros

O caso de uso “agrupar passageiros” tem seu diagrama de seqüência conforme mostrado na Figura 4-12. Por meio do diagrama de seqüência anterior é executado o



objeto *mygrupo*, com o objetivo de enviar a informação do número de chamadas agendadas de cada elevador. Desta maneira, é comparada cada chamada que entra com as chamadas agendadas de cada elevador, assim por meio do objeto *AgruparPassageiros* é agrupada a chamadas que tem o mesmo andar origem e andar destino.

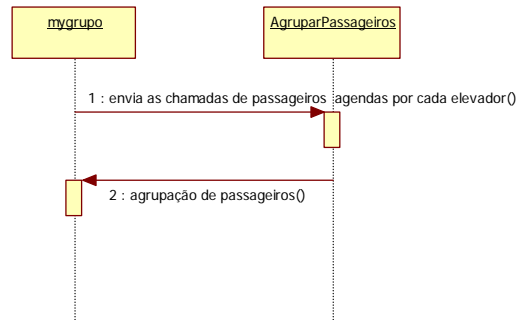


Figura 4-12 – Diagrama de seqüência agrupar passageiros

### Caso de uso número 6: Calcular prioridades

O caso de uso “**Calcular prioridade**” tem seu diagrama de seqüência conforme mostrado na Figura 4-13. Por meio do diagrama de seqüência simulação chamadas é executado o objeto *Controle*, o qual envia um vetor com a informação da distância, disponibilidade de carga, tempo de espera e número do elevador parâmetros estimados segundo a dinâmica dos elevadores. Assim, por meio do objeto *Prioridade* é enviado o resultado da prioridade calculado realizado pela lógica nebulosa (código sintetizado por Xfuzzy módulo de controle) para cada elevador individualmente.

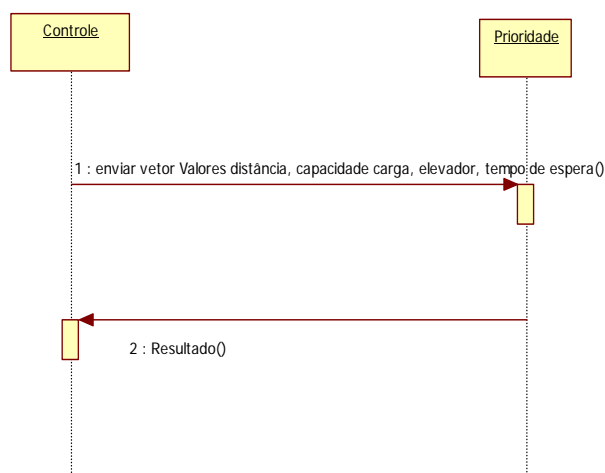


Figura 4-13 - Diagrama de seqüência Calcular prioridade

### 4.3.2 – Diagramas de classes

#### Caso de uso número 1: configurar comunicação

O caso de uso “**Configurar configuração**” (ver Figura 4-14), apresentam-se 6 classes. A classe *interfazcontrole* que é onde está a parte visual e onde se criam os objetos para interagir com as outras classes, como o caso da classe *ControladorComunicação*. Esta tem vários objetos criados por meio da biblioteca OPC (*OPC-Gateway*) para poder usar os métodos da biblioteca e desta maneira configurar a comunicação. A classe *controlador comunicação* executa a classe *Showconnectdialog*, a qual apresenta uma janela onde mostra os diferentes servidores OPC instalados dentro do computador. A classe *OpcConnection* realiza a conexão do sistema com o servidor escolhido. As classes *DinamicIOtemView* e *AbstractTableModel* apresentam os valores das variáveis selecionadas do emulador dentro de uma tabela, de modo que se atualizam os valores de forma dinâmica dentro da tabela a medida que os valores do emulador variam.

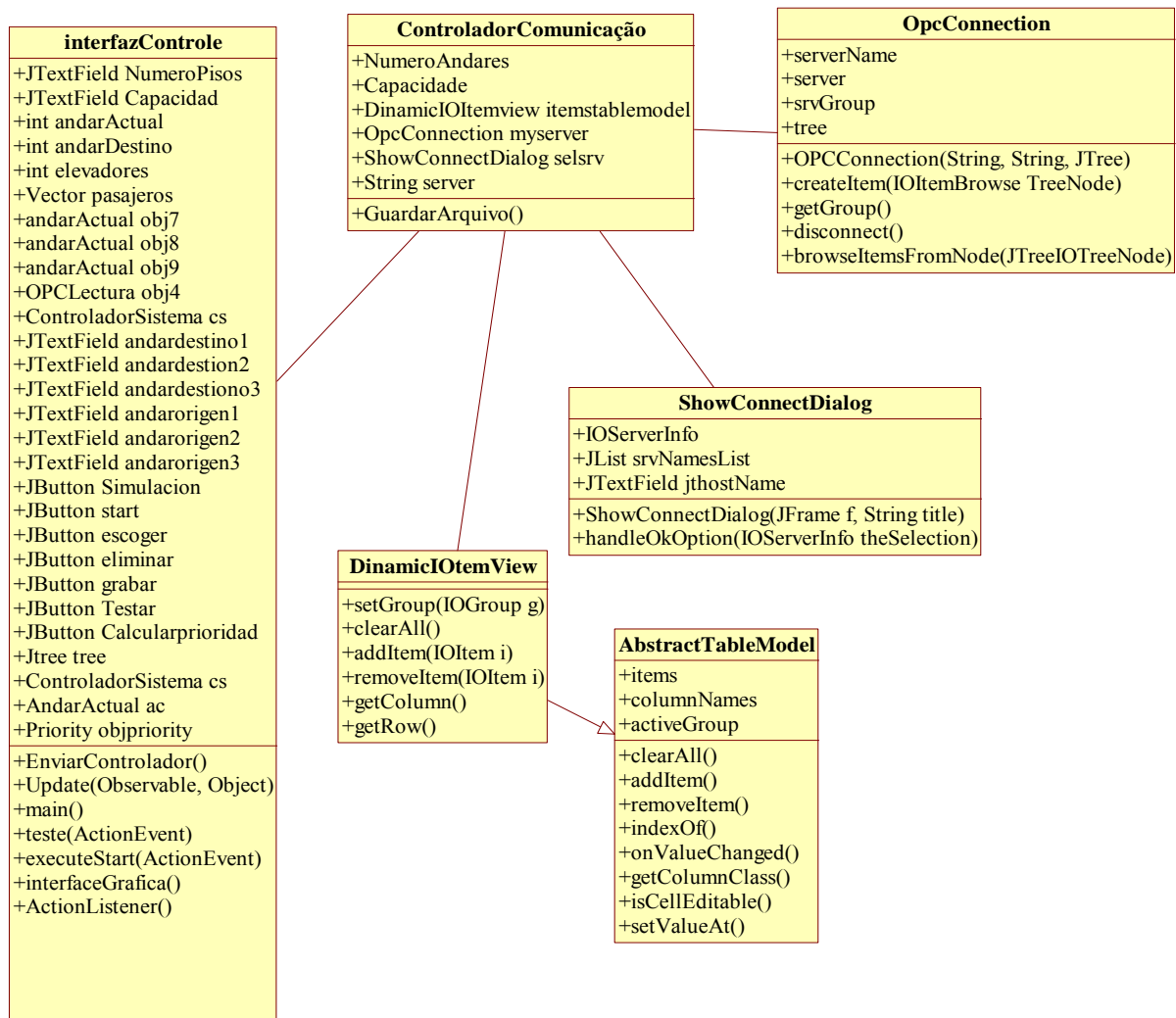


Figura 4-14 – Diagrama de classes configurar comunicação

### Caso de uso número 2: configurar controle

O caso de uso “**Configurar controle**” (ver Figura 4-15), apresentam-se 3 classes. A classe *interfazcontrole* utiliza a classe *controlador sistema* para configurar o número de andares do sistema e a capacidade de pessoas a ser atendidas. O método *Controlador()* é quem chama a classe *controlador sistema* para configurar o sistema.

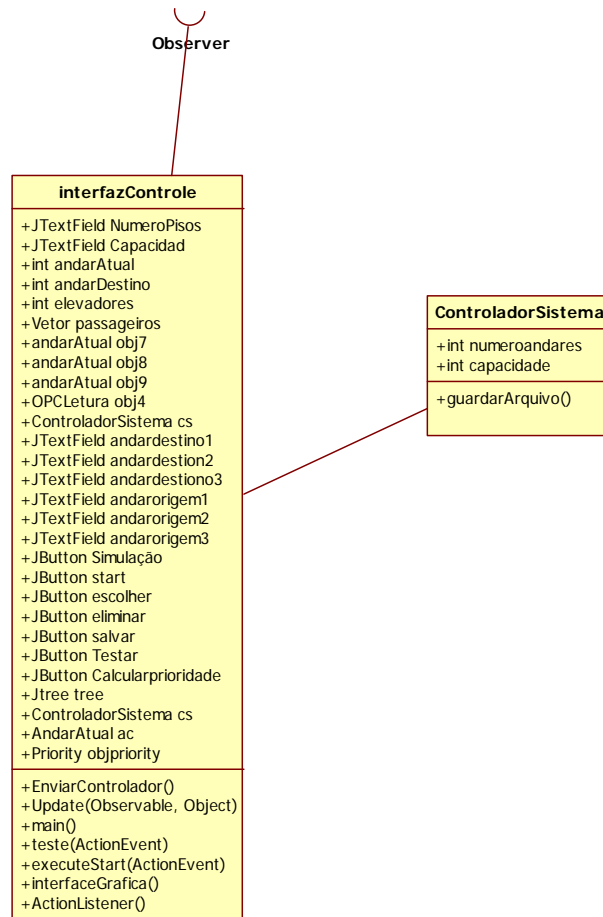


Figura 4-15 – Diagrama de classes de configurar controle

### Caso de uso número 3: Simulação chamadas

O caso de uso “**simulação chamadas**” (ver Figura 4-16), apresentam-se 8 classes. A classe *interfazcontrole* é onde está a parte visual e onde se criam os objetos para interagir com as outras classes, como *OPCLeitura*. Este último tem vários objetos criados por meio da biblioteca OPC para estabelecer a comunicação e executar um *thread* para que, de forma concorrente, o *software* leia as mudanças das variáveis do emulador de CLP. A classe *OPCLeitura* também interage com a classe *LerHW* a qual apresenta uma janela para que o usuário procure o arquivo das chamadas realizadas pelo gerador de trafego e coloque a informação em vetores. A classe *andaratual* foi desenvolvida com o objetivo de calcular o andar atual de cada elevador por meio do calculo de tempo de viagem.

O caso de uso “Lerescrever dados em tempo real” seu diagrama de classes está incluído dentro do diagrama de classes de simular chamadas, devido ao modelo inicial de planejamento.

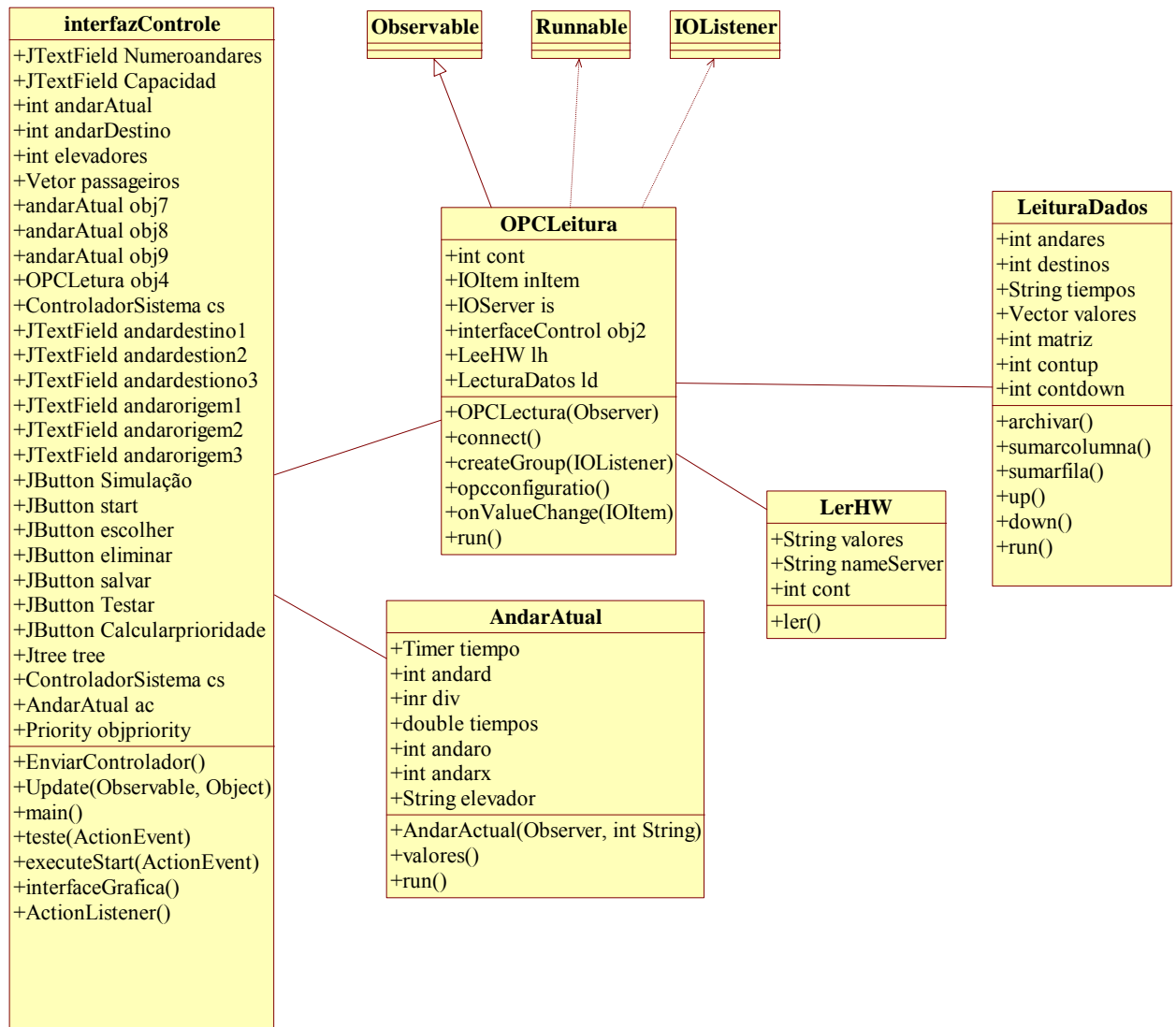


Figura 4-16 – Diagrama de classes simulação chamadas

### Caso de uso número 5: Agrupar passageiros

O caso de uso “Configurar controle” Figura 4-17, apresentam-se 2 classes. A classe *interfazcontrole* utiliza a classe *AgruparPassageiros* realiza as operações de agrupamento de chamadas por meio do método *agrupar passageiros*.

O diagrama de classes do caso de uso “**calcular prioridades**” não foi planejado, devido a que foi utilizada a ferramenta CAD (Xfuzzy) que gerou as classes para ser usadas dentro do sistema.

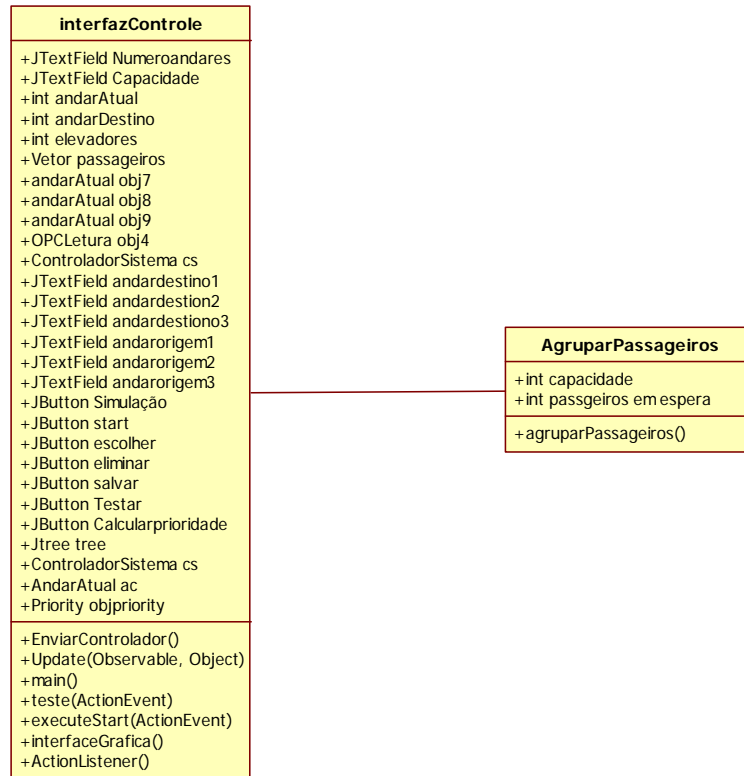


Figura 4-17 – Diagrama de classes agrupar passageiros

### 4.3.3 – Diagrama de implantação

O diagrama de implantação (ver Figura 4-18) apresenta de forma geral o sistema como um todo. O cliente é o simulador de trafego desenvolvido em Java, o qual se comunica com o servidor por meio do protocolo OPC. O servidor OPC está em um *software* que funciona como uma interface de comunicação, cujo objetivo é possibilitar a comunicação entre diferentes aplicativos com instrumentos físicos. O servidor se encontra nos programas de comunicação de cada fabricante de CLP (para o desenvolvimento deste trabalho utilizou-se o software RSLinx® da Rockwell Automation). O Emulador de CLP é uma ferramenta de *software* que substitui o CLP, em que é possível programar como se estivesse utilizando um controlador real. No caso deste trabalho, utilizou-se o emulador RSLogixEmulate 5000®, da empresa Rockwell Automation. O sistema de supervisão foi desenvolvido com o software RSVIEW32

fornecido pela mesma empresa Rockwell Automation. A comunicação do sistema de supervisão foi realizada por meio de uma rede local Ethernet usando o protocolo de comunicação OPC.

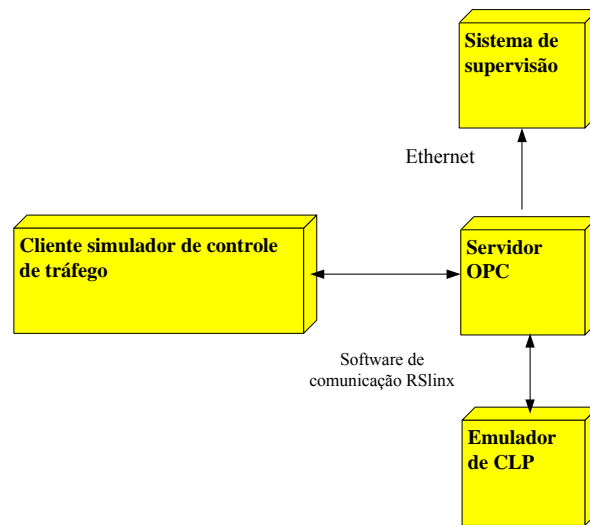


Figura 4-18 – Diagrama de implantação

## **5 - ESTUDO DE CASO: SISTEMAS DE GRUPO DE ELEVADORES**

Neste capítulo é apresentado um estudo de caso para demonstrar as técnicas e os algoritmos desenvolvidos na metodologia. Além disso, é realizada uma análise estatística de resultados de simulação.

Considerou-se como estudo de caso um sistema de grupo de elevadores usando o sistema de controle de destino (SCD) (seção 2.2.10), de modo a se poderem comparar os resultados das simulações com os resultados obtidos por outros autores. Para isso, realizaram-se simulações com o sistema de controle nebuloso de grupo utilizando duas abordagens:

- (a) Controle baseado no conhecimento do destino dos passageiros em que se aplica um algoritmo de ordenamento dos passageiros com origens e destinos em comum nos mesmos carros (chamado de controle CAO - Com Algoritmo de Ordenamento);
- (b) Controle também baseado no conhecimento de origem e destino dos passageiros, entretanto sem utilizar o algoritmo de ordenamento (chamado controle SAO – Sem Algoritmo de Ordenamento).

Com os dados de simulação obtidos, realizou-se uma análise de desempenho do sistema de grupo de elevadores, com o fim de estabelecer comparações entre as duas abordagens. Realizaram-se também comparações qualitativas de desempenho em relação a sistemas de controle de grupo de elevadores apresentados na literatura.

Nos itens que se seguem, apresentam-se os resultados das simulações bem como suas respectivas análises.

### **5.1 – SISTEMA DE ELEVADORES DESCRIÇÃO E ANALISES**

Para a execução das simulações foi considerado como cenário do estudo de caso um prédio comercial de oito andares, três elevadores e uma população em viagem de 430 pessoas. Para realizar o dimensionamento foi necessário usar as equações apresentadas na, seção 2.2.8.

A primeira etapa no dimensionamento foi definir os parâmetros dos elevadores, ou seja, a capacidade de carga, a velocidade nominal de viagem, a aceleração, a distância entre andares e os tempos de transferência (incluindo tempos de abrir e fechar portas, os quais dependem da largura das portas). Segundo Markon (2006), alguns



valores típicos dos parâmetros listados a cima são apresentados na Tabela 5-1. Esses valores foram adotados para fins das simulações realizadas neste trabalho.

**Tabela 5-1** - Parâmetros de operação de um elevador (Markon, 2006)

Item	Valores	Unidades
Capacidade nominal	10	peessoas
Velocidade nominal	0,8	m/s
Aceleração/desaceleração	0,7	m/s <sup>2</sup>
Abrir/fechar Portas	2,0	s
Tempo de transferência de passageiros	0,8	s
Altura de um andar	3,5	m

A segunda etapa de planejamento do grupo de elevadores envolveu o cálculo do valor de *RTT* (Tempo de viagem de ida e volta). Esse parâmetro é talvez o mais importante para o planejamento de prédios. Lembrando a equação 2.8, tem-se:

$$RTT = 2Ht_v + (S+1)t_s + 2Pt_p$$

Com o número de andares e a capacidade nominal de carga, os valores de *S* e *H* podem ser encontrados em uma tabela oferecida pelos fabricantes de elevadores (vide Anexo B). O tempo  $t_v$  é o período de tempo para percorrer um andar com velocidade nominal. Para o calculo de  $t_v$  foi necessário conhecer a altura ( $x$ ) de um andar do prédio e posteriormente usar a equação (5-1) de movimento retilíneo uniforme. O tempo  $t_s$  é composto pelo conjunto dos tempos relacionados aos diversos eventos que ocorrem durante a movimentação do elevador entre dois andares consecutivos, excluindo o tempo de viagem à velocidade nominal,  $t_v$ . Então, para o cálculo de  $t_s$ , utilizou-se a equação (5-2) em que  $t_f(1)$  é o tempo de vôo de um andar, ou seja, levando em conta o movimento uniforme variado (incluindo tempos de aceleração e desaceleração) e movimento retilíneo uniforme (tempo à velocidade nominal),  $t_c$  é o tempo de fechar portas e  $t_a$  é o tempo de abrir portas. A variável *P* é o número médio de passageiros dentro da cabine do elevador, de acordo com a literatura (Barney, 2003), seu valor é considerado como  $P=80\%$  da capacidade total (cc) de carga do elevador e  $t_p$  é o tempo que demora o passageiro em entrar no elevador (ver seção , conforme mostra a equação (5-3).

$$t_v = \frac{x}{v} \tag{5-1}$$

$$t_s = t_f(1) + t_c + t_a - t_v \tag{5-2}$$

$$P = (0.8)CC \quad (5-3)$$

Com base nos parâmetros da Tabela 5-1e nas equações anteriores, foram calculadas as variáveis para a estimação do *RRT*. A primeira operação foi calcular  $t_v$ , como é apresentado na equação (5-4). Posteriormente, tendo o valor de  $t_v$ , foi possível calcular  $t_s$  como é mostrado na equação (5-5). Finalmente, a equação (5-6) apresenta o cálculo do valor  $P$ .

$$t_v = \frac{3.5m}{0.8 \text{ m/s}} = 4.375 \text{ s} \quad (5-4)$$

$$t_s = \left( \underbrace{\left( \frac{v}{a} + \frac{x}{v} \right)}_{t_f} + 2 + 2 - \underbrace{\frac{x}{v}}_{t_v} \right) = 5.142 \text{ s} \quad (5-5)$$

$$P = (0.8)(10) = 8 \quad (5-6)$$

Desta forma, encontrando os valores para o cálculo de *RTT* e substituindo na equação ) resulta a equação .

$$(5-7)$$

$$RTT = 2(7.5)(2.69s) + (5.3 + 1)5.85s + 2(8)0.8s = 110 \text{ s}$$

Continuando com as equações de planejamento do sistema de grupo de elevadores, o passo seguinte foi calcular o tempo médio ( $I$ ) entre as chegadas sucessivas de um elevador ao andar principal com carga em qualquer nível, por meio da equação ) conhecendo os valores de *RTT* e  $L$  (número de elevadores), conforme explicitado na equação :

$$I = \frac{RTT}{L} = \frac{110}{3} = 36 \text{ s} \quad (5-8)$$

Com o valor do intervalo de tempo foi possível calcular a capacidade do transporte (*HC*) do grupo de elevadores por meio da equação ), conforme explicitado na equação .

$$HC = \frac{300(0.8)(10)}{I} = 64 \text{ pessoas /5 minutos} \quad (5-9)$$

Obtendo a capacidade de transporte do sistema, define-se a porcentagem de população total que pode ser atendida em uma situação de tráfego *up-peak* em um período de 5 minutos. Para o cálculo desta porcentagem foi usada a equação ), conforme explicitado na equação .

$$\begin{aligned} \%HC &= \frac{HC}{\text{População em viagem do prédio}} \times 100 && (5-10) \\ &= \frac{64 \text{ pessoas}/5 \text{ minutos}}{430 \text{ pessoas}} \times 100 = 14\% \end{aligned}$$

Após finalizados os cálculos do planejamento do sistema de grupo de elevadores, escolheu-se o tipo de tráfego a simular. Foi escolhido o tipo de tráfego *up-peak* por ser uma das situações mais críticas presentes nos sistemas de grupo de elevadores, além de permitir uma avaliação mais completa do desempenho do sistema de controle.

Para simulação, utilizou-se o método de replicações para a coleta de observações (Taha, 2007), devido a que esse método é muito usado na literatura para representar o desempenho (tempo de espera) do sistema de elevadores. O método é realizar simulações com fluxos de passageiros variáveis em um intervalo de tempo de cinco minutos. Os fluxos estão relacionados com a variável que se chama *Percentual de População Servida (PPS)*, segundo cálculos de planejamento, o sistema teria a capacidade nominal de transportar 64 passageiros em cinco minutos ( $\%HC = 14\%$ ) em um intervalo de tempo de espera aceitável (ver seção 2.2.7). Para avaliar o desempenho do sistema, decidiu-se iniciar com um fluxo baixo,  $PPS = 2\%$ , e chegar a um fluxo alto, por exemplo  $PPS = 18\%$ . Desta forma, é possível observar o comportamento que o controlador de grupo apresenta sob condições variáveis de tráfego.

Levando-se em conta os requisitos dos fluxos de passageiros e o tipo de tráfego a simular, geraram-se os arquivos de chamadas por meio do módulo gerador de tráfego, utilizando os parâmetros apresentados na Tabela 5-2.

**Tabela 5-2** - Parâmetros usados na geração de chamadas na simulação

Tráfego	<i>up-peak</i>
A (passageiros subindo)	95%
B (passageiros descendo)	1%
C (passageiros Inter-andares)	4%
Número de andares	8
Numero de pessoas desejando viajar	430
<i>PPS</i>	2% até 18%

Conseqüentemente, geraram-se os arquivos para cada fluxo, começando a variável *PPS* em 2% com passos de 2 até chegar a 18%. Esses arquivos incluem o andar de origem, o andar de destino e o tempo de chamada, parâmetros necessários para

realizar as simulações. Desta maneira, obtendo os arquivos de chamadas, só foi necessário configurar o módulo de simulação (seção 3.2.4), para iniciar as simulações para os diferentes fluxos. A Figura 5-1 apresenta os resultados de desempenho do sistema de controle de elevadores sem o algoritmo de ordenamento (SAO). No gráfico, podem-se observar as duas curvas dos tempos de espera e dos tempos de destino (tempo de espera no pavimento somado com o tempo de espera dentro do carro) durante um tráfego *up-peak*.

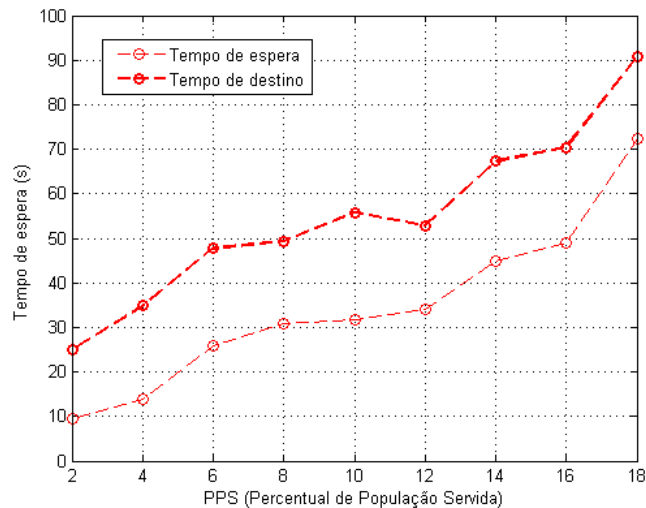


Figura 5-1–Tempo de espera e tempo de destino usando o sistema de controle SAO

A Figura 5-2 apresenta o resultado do número de paradas dos elevadores em função do fluxo de passageiros para o caso do sistema de elevadores sem o algoritmo de ordenamento. Conseqüentemente, para efeitos de avaliação do consumo de energia foram consideradas apenas as paradas de destino dos passageiros.

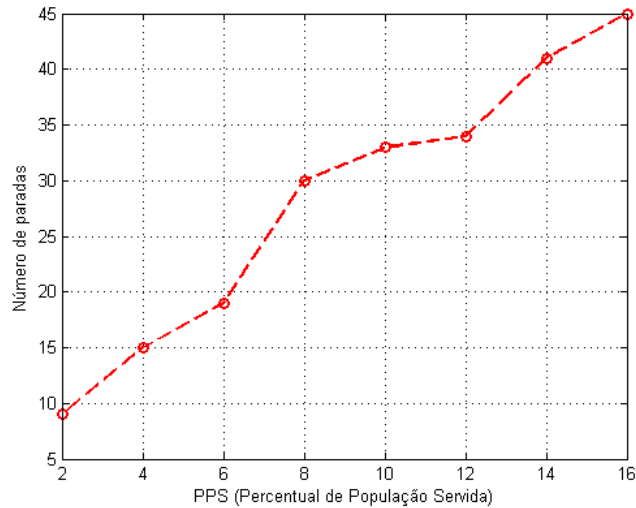


Figura 5-2 – Número de paradas dos elevadores em função do fluxo de passageiros com o sistema de controle SAO

Da mesma forma, foram realizadas simulações do sistema de controle de grupo de elevadores com o algoritmo de ordenamento (CAO), usando os mesmo arquivos gerados pelo sistema de geração de chamadas (andar de origem, andar de destino tempo de espera). A Figura 5-3 apresenta os resultados de desempenho do sistema de controle de elevadores no caso CAO. No gráfico, podem-se observar as duas curvas dos tempos de espera e dos tempos de destino (tempo de espera no pavimento somado com o tempo de espera dentro do carro) durante um tráfego *up-peak*.

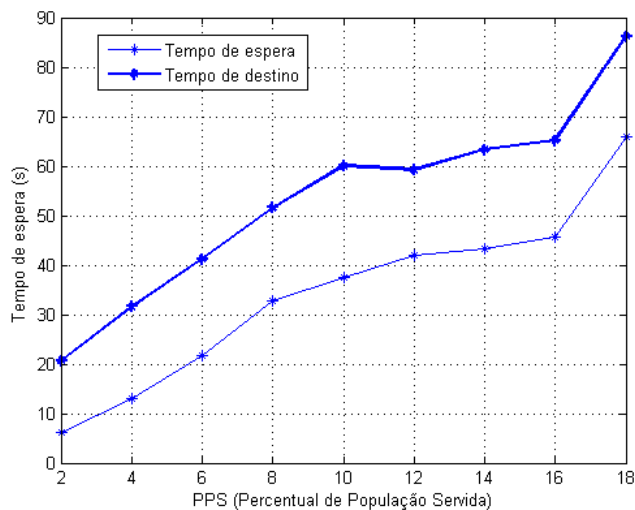


Figura 5-3– Tempo de espera e tempo de destino usando um sistema de controle CAO

Na Figura 5-4 apresenta o resultado do número de paradas dos elevadores em função do fluxo de passageiros no caso de um sistema de elevadores CAO. Para efeitos

de avaliação do consumo de energia foram consideradas apenas as paradas de destino dos passageiros.

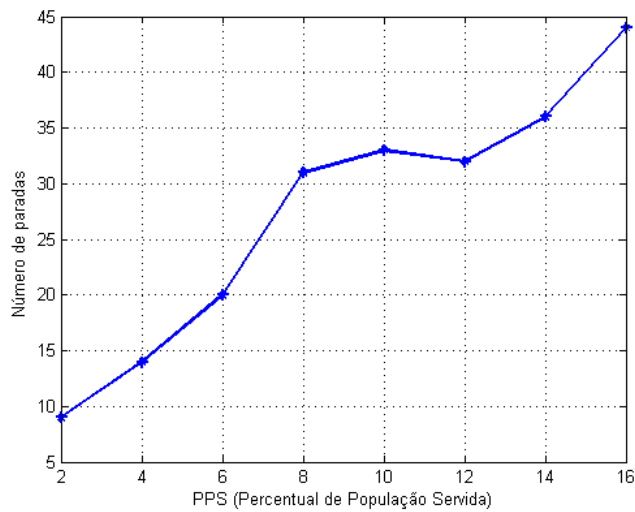


Figura 5-4 – Número de paradas dos elevadores em função do fluxo de passageiros com o sistema de controle CAO

## 5.2- ANÁLISE DOS RESULTADOS

### 5.2.1 Análise estatística

Os resultados das simulações devem ser interpretados usando as ferramentas adequadas de inferência estatística como, por exemplo, determinação do intervalo de confiança. Esta ferramenta possibilita analisar estatisticamente dados com conteúdo aleatório de modo a se determinar intervalos em que a variável real pode estar, com alguma probabilidade. No caso em estudo, os valores de tempos de espera são gerados a partir de sinais estocásticos, devendo ser expressos como um valor médio acrescido de um intervalo de confiança. Para executar essa tarefa, as observações do experimento de simulação devem satisfazer algumas condições (vide seção 2.2.6). Na realização desta análise foram escolhidas amostras do sistema sem o algoritmo de ordenamento e com intensidades de fluxo, *PPS*, iguais a 12%, 14% e 16%. Estas intensidades correspondem a um número de dados proporcionalmente considerável para possibilitar uma análise estatística, além de resultar em um tempo de espera médio ainda aceitável (abaixo de 50 segundos). As condições mencionadas acima são explicitadas nos itens que se seguem:

- 1) Observações devem ser retiradas de distribuições estacionárias (idênticas).

Para esta análise, realizaram-se gráficos dos resultados da simulação (tempos de espera) em função do número de dados, de maneira, a identificar o comportamento dos dados e sua tendência à estacionariedade (seção 2.2.6). Na Figura 5-5 apresentam-se os comportamentos dos dados em uma simulação de: (a) 69 pessoas/5 minutos, (b) 59 pessoas/5 minutos e (c) 52 pessoas/5 minutos. As linhas horizontais das curvas apresentam o valor da média estatística encontrada.

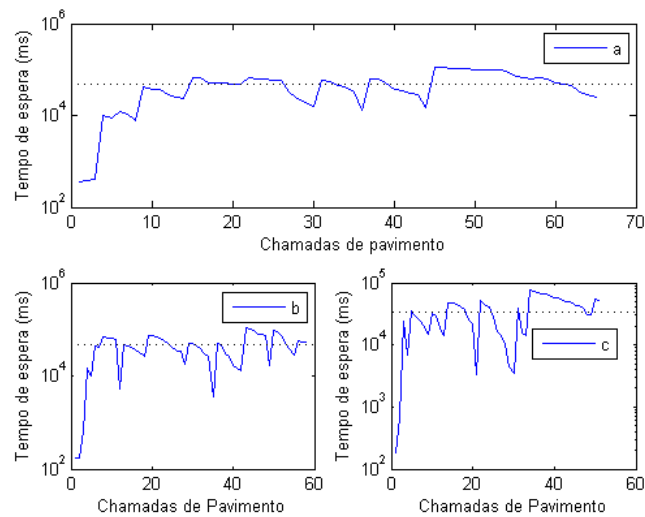


Figura 5-5 - Simulação tempo de espera para valores de PPS de 12% (52 pessoas/5 minutos), 14% (59 pessoas/5 minutos) e 16% (69 pessoas/5 minutos)

Tendo em vista o baixo número de amostras (tempo de espera por pessoa atendida), observa-se uma flutuação muito intensa (da ordem do valor médio encontrado) o que dificulta a verificação de alguma tendência. Devido a este fato, decidiu-se por representar os tempos de espera em uma escala logarítmica em que os valores dos tempos de espera têm unidades de milissegundos. Desta maneira, possibilitou-se o estudo das diferentes regiões da curva. Nota-se, que as curvas na parte inicial da simulação têm um período errático, denominado período transiente ou de aquecimento (Taha, 2007), e depois do dado 10 a média dos tempos de espera tende a se estabilizar, ou seja, pode-se dizer que o sistema começa a trabalhar em um regime estacionário ou de equilíbrio. O interesse da análise é no período estacionário, em que é necessário conhecer o intervalo de confiança do tempo de espera. O desempenho do controlador pode ser avaliado com base no valor médio do tempo de espera e no seu respectivo intervalo de confiança, isto é, que flutuação é esperada em relação ao valor médio do tempo de espera.

A Figura 5-6 apresenta as curvas para os valores de PPS de 12%, 14% e 16%, em que os 10 primeiros dados de simulação são retirados da amostra, devido a que estes representam o período transiente das curvas de simulação. As linhas horizontais das curvas apresentam o valor da média estatística encontrada.

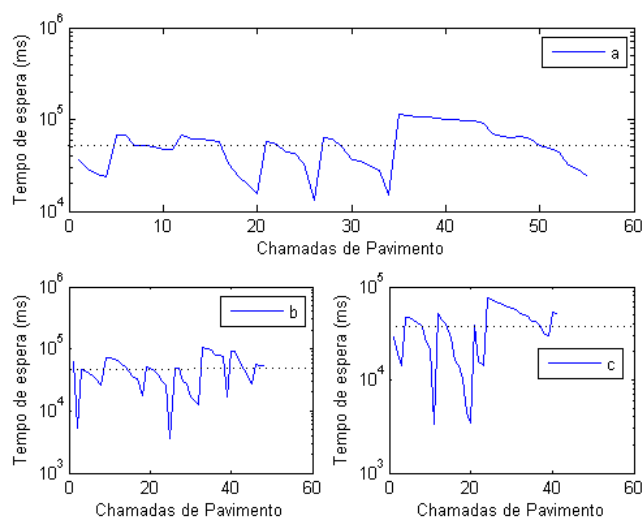


Figura 5-6 - Simulação tempo de espera para valores de PPS de 12% (52 pessoas/5 minutos), 14% (59 pessoas/5 minutos) e 16% (69 pessoas/5 minutos) sem o período do transiente

Na Tabela 5-3 são apresentados os valores da média estatística, do desvio padrão e do intervalo de confiança correspondente às simulações realizadas para os valores de PPS= 12%, 14% e 16% com o período transiente e sem o período transiente, com o objetivo de estudar o comportamento dos dados.

**Tabela 5-3 -Análise estatística**

	Com período transiente			Sem período transiente		
	12%	14%	16%	12%	14%	16%
PPS						
Média	34,5 s	45,5 s	48.7 s	38,4 s	48,5 s	55,2 s
Desvio padrão	20,2	26,2	29.8	19,9	25,5	27,2
Intervalo de confiança de 95%*	4,9	6,3	7,2	3,8	6,2	6,6

\* Para o cálculo do intervalo de confiança assumiu-se que os dados são normalmente distribuídos, isto é, seguem uma distribuição normal. Essa hipótese será provada no item que se segue.

Observa-se que a amplitude do desvio padrão e o intervalo de confiança sem o período transiente diminuem, o que indica um maior grau de precisão no tempo de espera. Nota-se que o valor médio encontrado nos diferentes valores de fluxos aumenta



quando é eliminado o período transiente. Portanto, mostra-se possível melhorar a precisão (intervalo menor) das simulações eliminando o período transiente na coleta de dados. Segundo a literatura (Taha, 2007), essa opção é válida quando o número de dados é relativamente pequeno, que é o caso na análise aqui realizada. Já quando o número de dados é suficientemente grande, o efeito do período transiente nos valores da média e da variância torna-se desprezível, não sendo necessário eliminá-lo.

2) Observações devem ser amostradas de uma população com distribuição normal.

Existem vários métodos para verificar que os dados amostrados têm comportamento de uma distribuição normal. O método mais utilizado consiste em traçar um gráfico de distribuição normal acumulada (Doebelin, 2004) com os dados de cada simulação ( $PPS = 12\%$ ,  $14\%$  e  $16\%$ ). A Figura 5-7 apresenta os três gráficos correspondentes aos dados em análise. Observe-se que os pontos nos gráficos estão distribuídos aproximadamente em uma reta com uma inclinação de 45 graus, o que indica uma grande aderência dos dados a uma distribuição normal nas regiões centrais dos gráficos.

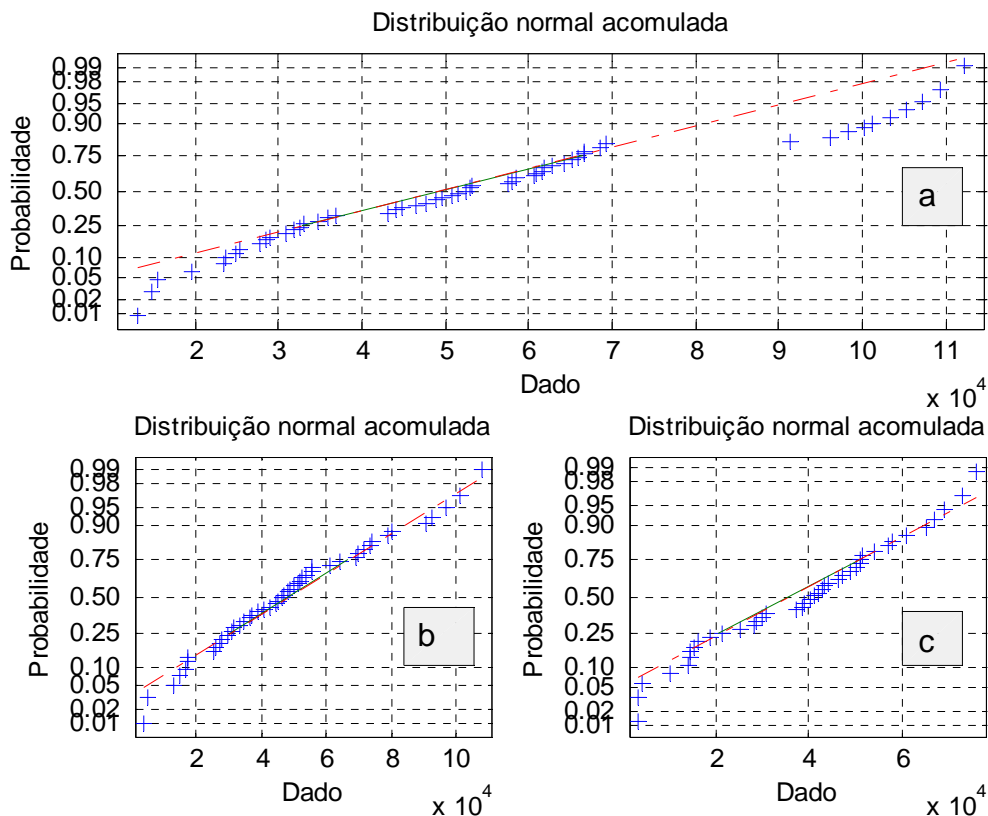


Figura 5-7 -Gráfico de probabilidade normal baseado nos dados da simulação

3) Observações devem ser independentes.

O método usado para coletar observações em simulação é o método de replicação (seção 2.2.6). Portanto, cada rodada de simulação é dirigida por uma corrente distinta de números aleatórios (entre 0 e 1), o que resulta em observações que podem ser consideradas independentes em termos estatísticos. A desvantagem é que cada observação pode ser influenciada pelo viés do efeito das condições transitórias.

Considerando que os dados satisfazem as três condições anteriores, pode-se assumir que as amostras submetidas à análise seguem distribuições normais. Entretanto, como em qualquer experimento estatístico, a “regra de ouro” diz (Taha, 2007): “*valores mais altos de número de amostras significam resultados de simulação mais confiáveis*”. No simulador desenvolvido neste trabalho, devido a limitações do software, só foi possível trabalhar com uma população em viagem de 430 pessoas, ou seja, conseguiu-se realizar simulações com fluxos máximos de até 77 pessoas por 5 minutos (PPS=18%), o que resultou em poucos dados para uma análise estatística mais aprimorada.

### **5.2.2 Análise comparativa CAO e SAO**

As curvas de tempo de espera nos pavimentos e de tempo de espera nos carros para diferentes fluxos de passageiros são usadas para avaliar o desempenho do sistema de elevadores (Siikonen, 2000). A partir destas curvas é possível conferir a eficiência do sistema de controle de grupo e seu funcionamento. A Figura 5-8 apresenta as duas curvas de tempo de espera a serem comparadas. Nota-se, que o gráfico está composto por nove valores de fluxos de passageiros, os quais podem ser divididos em três grupos: (a) Um primeiro grupo chamado baixo (2, 4, 6), (b) um segundo grupo chamado médio (8, 10, 12) e (c) um terceiro grupo chamado alto (14, 16, 18). Observe-se, que a curva do sistema de controle CAO, nas regiões de fluxo baixo e alto, tem melhor desempenho (valores de tempo de espera mais baixos) no tocante ao tempo de espera nos pavimentos. Entretanto, na região de fluxo médio, o sistema de controle SAO apresenta um melhor desempenho nos tempos de espera nos pavimentos.

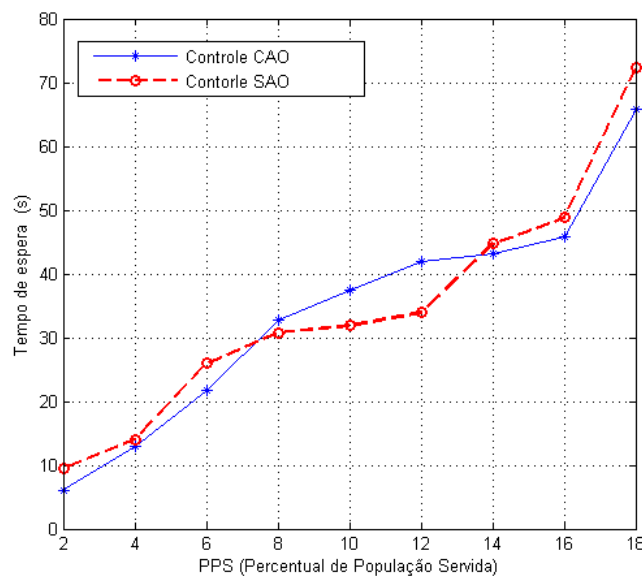


Figura 5-8- Tempo de espera dos sistemas de elevadores propostos

Visando a realizar uma melhor comparação do desempenho dos sistemas de controle nebuloso propostos, foram considerados os três grupos de fluxos referidos anteriormente. Assim, detalham-se nas figuras 5-8, 5-9 e 5-10 as diferenças nos valores de tempo de espera dos passageiros nos pavimentos (em verde) e as diferenças nos tempos de espera dentro dos carros (em amarelo) em função de *Porcentagem de População Servida (PPS)*.

A Figura 5-9 apresenta os tempos de espera nos pavimentos (TEPV) e os tempos de espera dentro dos carros (TEDC) de dois fluxos de passageiros PPS = 14% e PPS = 16%, de modo a comparar os sistemas de controle de grupo de elevadores. Dado que foram realizadas as simulações no caso de tráfego majoritário de subida (*up-peak*) é esperado um melhor desempenho no sistema de controle CAO em relação ao sistema de controle SAO, devido ao fato de o primeiro possibilitar o ordenamento dos passageiros. A explicita os desempenhos do sistema CAO para os fluxos de 16% e de 14%, os quais apresentam uma diminuição no TEPV e no TEDC em relação ao sistema SAO.

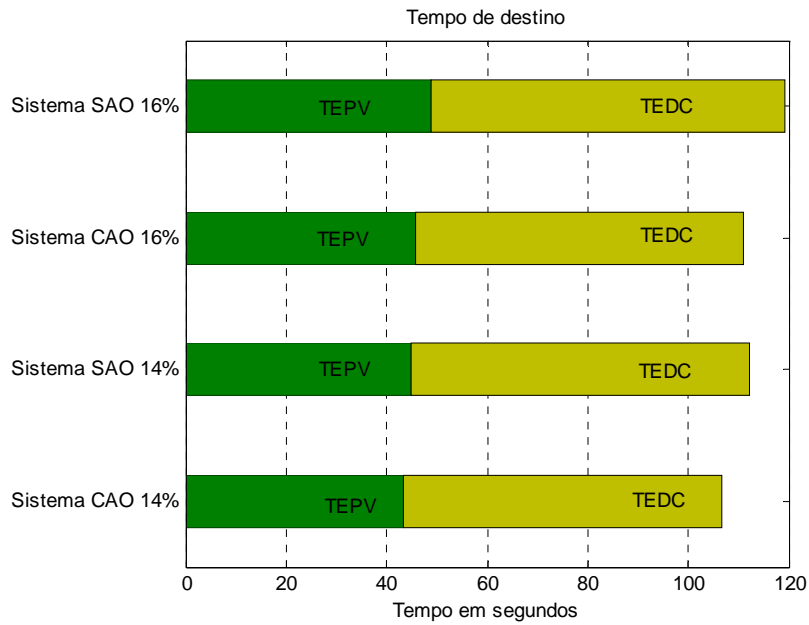


Figura 5-9-Comparação de intensidade de tráfego PPS = 16% e PPS=14%

Com a Figura 5-10 é possível comparar o desempenho dos controladores propostos para fluxos médios de passageiros de  $PPS = 10\%$  e  $PSS = 12\%$ . Observando-se no gráfico um aumento nos tempos de espera para o sistema de controle CAO. Este comportamento pode ser explicado, porque o sistema de controle CAO não permite que o elevador mais conveniente seja atribuído para uma nova chamada de pavimento em situações de tráfego médias, uma vez que o sistema de controle CAO atribui a nova chamada de pavimento ao elevador com o mesmo andar origem e andar destino. Entretanto, observa-se que em geral os tempos de TEDC no sistema de controle CAO são mais baixos, devido à vantagem que oferece esse tipo de arquitetura de diminuir o número de paradas (Markon, 2008).

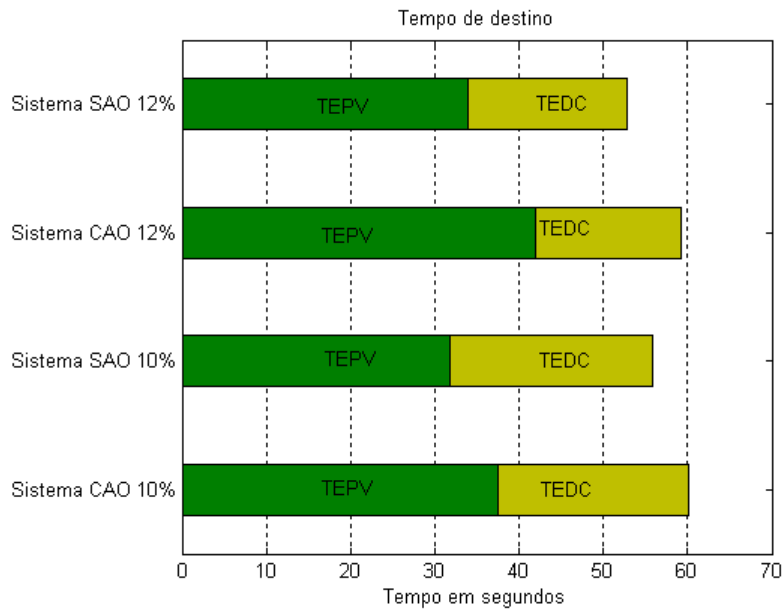


Figura 5-10 -Comparação de intensidade de tráfego PPS=12% e PPS=10%

A Figura 5-11 apresenta os valores de TEPV e de TEDC para fluxo de passageiros baixo. Observe-se, que os TEPV's e os TEDC's são bastante próximos entre os dois sistemas em estudo, para cada fluxo. Apesar de apresentarem pouca diferença entre os tempos, o sistema CAO ainda mostrou-se melhor que o sistema SAO. Além disso, considerando os baixos valores de fluxo, o número de chamadas de pavimento também é baixo, resultando em um baixo número de amostras para análise. Portanto, os resultados da análise podem estar bastante influenciados pelos tempos decorrente do período transiente dos controladores, motivo pelo qual considera-se essa comparação pouco representativa.

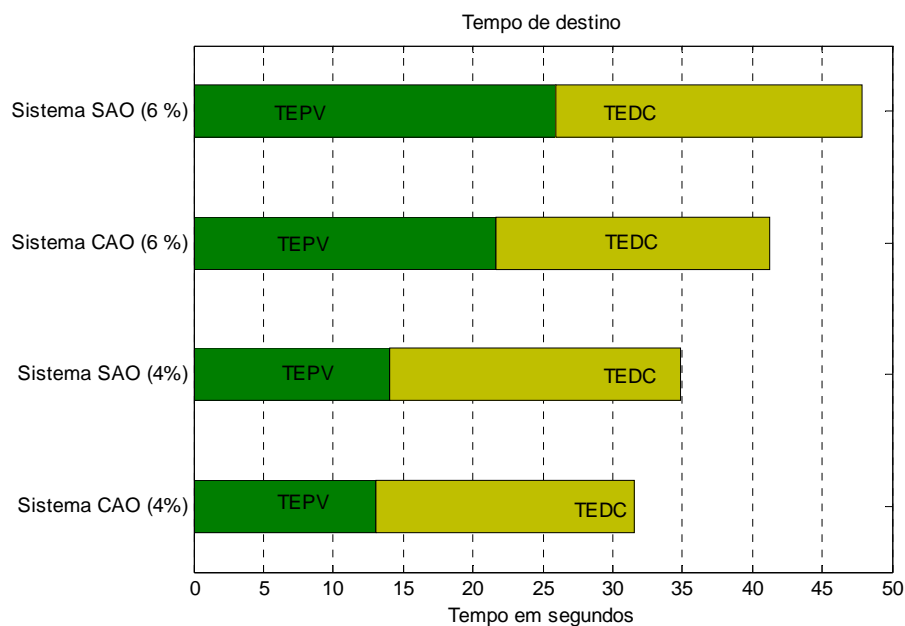


Figura 5-11-Comparação de intensidade de tráfego PPS = 6% e PPS= 4%

Na Tabela 5-4, na Tabela 5-4 e na Tabela 5-5, apresentam-se os dados de tempos de espera nos pavimentos (TEPV), tempos de espera dentro do carro (TEDC) e de número de paradas do elevador (paradas de destino). Nestas são relacionados os tempos obtidos das simulações dos sistemas propostos assim como os tempos de espera obtidos por Siikonen (2000), com o objetivo de realizar comparações de desempenho do controlador desenvolvido com outro controlador apresentado na literatura.

Tabela 5-4 – Tempo de espera (s)

Sistema	PPS								
	2	4	6	8	10	12	14	16	18
Sistema de controle SAO(proposto)	9,4	13,9	25,9	30,7	31,8	33,9	44,7	48,8	72,3
Sistema de controle CAO (proposto)	6,1	13	21,6	32,7	37,5	41,9	43,2	45,7	65,9
Sistema tradicional (Siikonen, 2000)	4	4	4	5	7	10	75		
Sistema com controle de destino (Siikonen, 2000)	10	11	12	18	23	26	30	33	37

Tabela 5-5 – Tempo de espera dentro do carro (s)

Sistema	PPS								
	2	4	6	8	10	12	14	16	18
Sistema de controle SAO(proposto)	15,5	20,9	21,9	18,6	24	18,9	22,5	21,6	18,4
Sistema de controle CAO (proposto)	14,5	18,6	19,6	18,9	22,5	17,3	20,1	19,4	20,3

Tabela 5-6- Número de paradas

Sistema	PPS							
	2	4	6	8	10	12	14	16
Sistema de controle SAO(proposto)	9	15	19	30	33	34	41	45
Sistema de controle CAO (proposto)	9	14	20	31	33	32	36	44

O sistema de controle tradicional realizado por Siikonen (2000) utiliza um tipo de controle coletivo convencional (*fullcollective*). Este sistema de controle considera um atendimento de todos os andares, por todos elevadores, em condições de tráfego (*up-peak*) em que os elevadores sempre chegam no térreo com disponibilidade de carga, e o passageiro pode entrar em qualquer elevador. Em consequência, os tempos de espera diminuem como também sua capacidade de transporte (HC). Deve-se notar que o sistema de controle SAO proposto neste trabalho utiliza um controlador nebuloso. Desta maneira, as simulações realizadas para avaliar o desempenho do controlador não consideram o atendimento para todos os andares por todos os elevadores, de modo que os passageiros não podem entrar em qualquer elevador. Portanto, pode-se observar na Figura 5-12 que os tempos de espera resultantes do sistema de controle SAO proposto são muito mais altos.

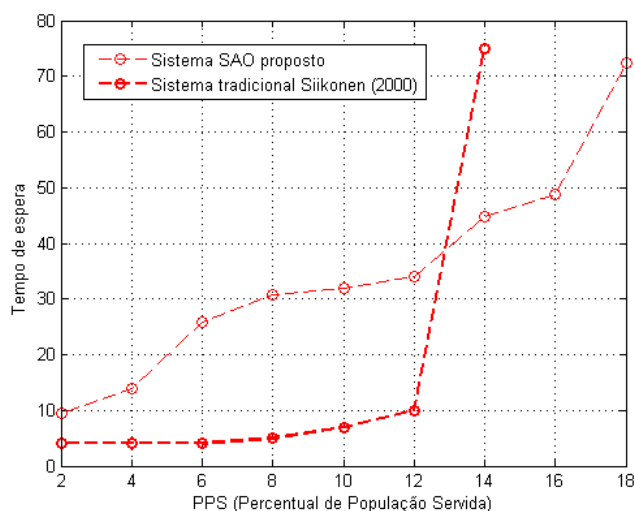


Figura 5-12- Tempo de espera do sistema de controle SAO proposto e tempo de espera de Siikonen (2000)

Na Figura 5-13 mostram-se as curvas do sistema de controle CAO proposto neste trabalho e o sistema moderno com Sistemas de Controle de Destino (SCD) proposto por Siikonen (2000). Os sistemas de controle de elevadores modernos (SCD) apresentam uma capacidade de transporte geralmente alta comparada com a capacidade de transporte dos sistemas tradicionais. Pelas curvas na Figura 5-13, observa-se um comportamento monotônico crescente com taxa de crescimento tendendo para uma constante. Obviamente, os fluxos de passageiros mostrados não ultrapassam a capacidade de transporte do sistema, após a qual espera-se um crescimento acelerado dos tempos de espera. No sistema simulado neste trabalho, isso é evidenciado para o fluxo  $PPS=18\%$ . O trabalho de Siikonen (2000), o sistema de elevadores apresentava uma capacidade de transporte maior, possibilitando a coleta de um maior número de amostras e, em consequência, resultando em uma curva mais bem comportada. No mesmo trabalho, não há uma descrição precisa sobre as regras de controle utilizadas, o que também deve contribuir para a discrepância entre os desempenhos dos controladores. Na região de baixo fluxo ( $2\% < PPS < 6\%$ ), observam-se valores tempos de espera relativamente baixos com uma taxa de crescimento relativamente alta (em relação aos tempos de espera obtidos neste trabalho para fluxos maiores). Esse resultado pode ter sido influenciado pela imprecisão no cálculo dos valores médios dos tempos de espera devido ao baixo número de amostras coletadas. Amostras pequenas são mais influenciadas pelo período transiente da simulação, como foi mostrado anteriormente (vide item 5.2.1), fazendo com que o mesmo tenha causado uma redução dos valores



médios calculados em relação ao valor esperado. Observa-se também na Figura 5-13 que a curva obtida a partir do controlador CAO implementado neste trabalho apresenta uma inclinação semelhante à curva obtida por Siikonen (2000) entre valores de PPS iguais a 6% e 16%. Para o valor de  $PPS = 18\%$ , observa-se um início de saturação do sistema de controle, evidenciado pelo salto observado no respectivo tempo de espera resultante.

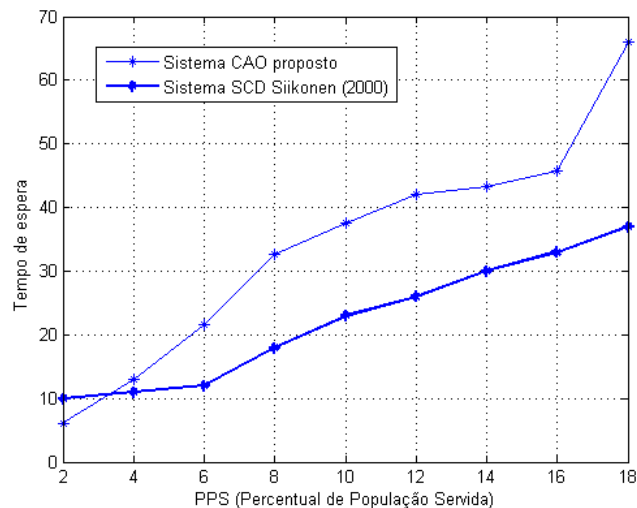
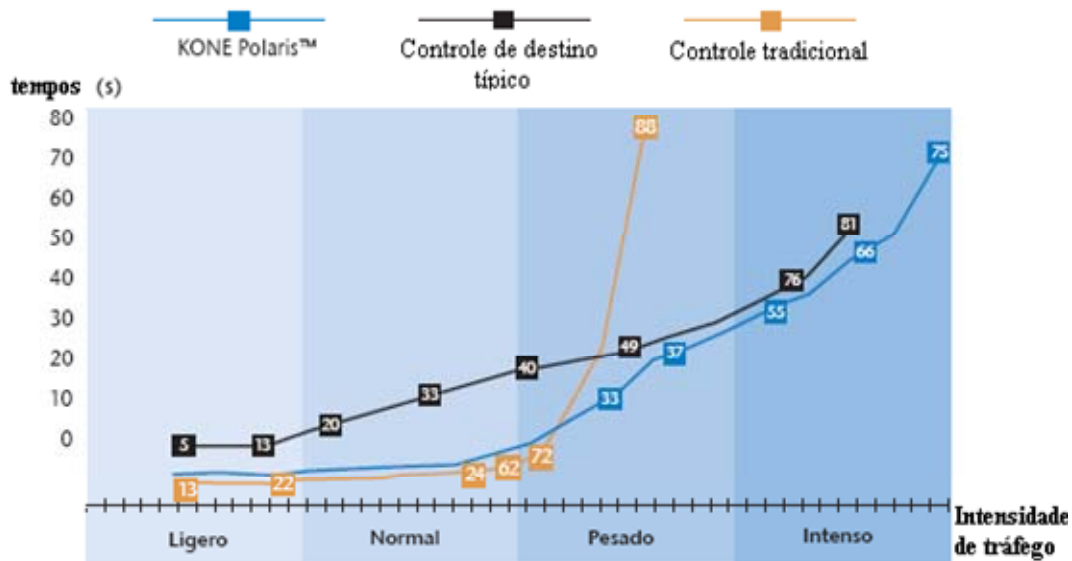


Figura 5-13 - Tempo de espera do sistema de controle CAO proposto e tempo de espera de Siikonen (2000)

Para fins de comparação, em termos qualitativos, apresentam-se na os desempenhos de três tipos de controladores de grupo de elevadores da empresa Kone Polares™ (2009). O primeiro controlador é um controle tradicional com tempos de espera baixo, mas com capacidade de transporte baixa. Um segundo controlador chamado controle de destino típico, em que os tempos de espera não são muito baixos, mas sua capacidade de transporte é alta. Finalmente, o terceiro controlador chamado KonePolaris™, o qual tem o comportamento de um controle de destino típico, mas otimiza dinamicamente os tempos de espera nos fluxos de passageiros baixos. Desta maneira, esse sistema inclui vantagens do sistema tradicional e do sistema de controle de destino típico tais como: tempo de espera baixos para fluxos de passageiros baixos e capacidade de transporte alta para fluxos de passageiros alta. Não foi possível adicionar as curvas obtidas neste trabalho à Figura 5-14, uma vez que esta não detalha a escala de intensidade de tráfego em termos numéricos. Entretanto, podem-se observar as formas das curvas, principalmente a chamada “controle de destino típico”, a qual se assemelha bastante com as curvas obtidas neste trabalho (vide Figura 5-8).



**Figura 5-14**—Tempo de esperas de intensidade de tráfego de pessoas Konepolaris (2009)

Na Figura 5-15 apresenta os resultados de número de paradas de cada elevador em função de fluxo de passageiros para os dois sistemas de controle propostos. O número de paradas de cada elevador está relacionado com o número de vezes que o motor deve iniciar e frear a marcha. Isto é proporcional ao consumo de energia (Kim et al., 1998). Neste gráfico é possível conferir a diminuição do número de paradas no sistema de controle CAO proposto a partir do fluxo 12 até o fluxo 16. Nota-se que em todas as simulações são considerados apenas as paradas do andar destino (o andar origem geralmente é o mesmo). Desta forma, pode-se mostrar as vantagens de se ter um sistema de controle CAO de elevadores, o qual permite colocar em um mesmo elevador os passageiros com um mesmo andar origem e destino, possibilitando assim a diminuição do número de paradas dos elevadores.

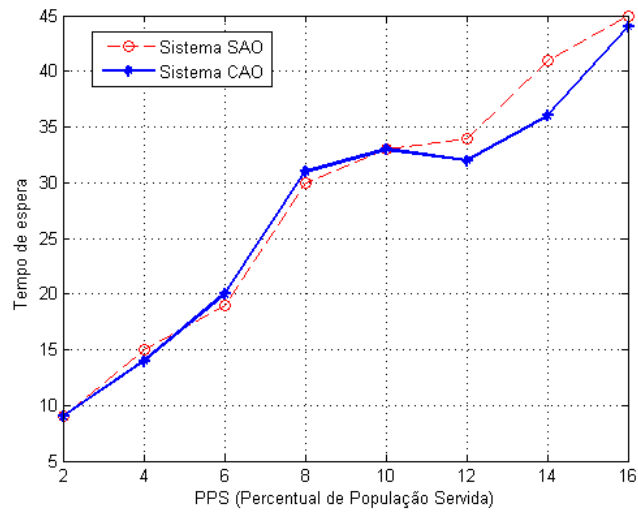


Figura 5-15 – Consumo de energia dos dois sistemas propostos neste trabalho

## 6 - DISCUSSÃO DOS RESULTADOS

Este trabalho propõe a implementação de técnicas de controle de grupo de elevadores baseado em lógica nebulosa usando, tecnologias de automação industrial. Estas técnicas foram projetadas para ser implementadas em uma arquitetura moderna usando sistemas de controle de destino (SCD), tendo como objetivo melhorar o desempenho do sistema de controle e o conforto dos passageiros. O desenvolvimento das técnicas de controle baseadas em lógica nebulosa permitiu o estudo das vantagens das arquiteturas modernas usando (SCD), possibilitando a realização de novos algoritmos que melhoram o desempenho dos sistemas de elevadores. Os parágrafos abaixo apresentam uma discussão do trabalho desenvolvido, e sugerem possíveis melhorias a serem implementadas no sistema.

O primeiro passo para consecução dos objetivos propostos envolveu a construção de um modelo de controle de grupo de elevadores, baseada nas técnicas de controle de elevadores recentes. A implementação desse modelo constituiu-se em um trabalho que demandava muito tempo, devido à grande quantidade de elementos necessários a serem ajustados, além da sincronização entre os diversos componentes do sistema de simulação implementado, de modo a realizar a atribuição de elevadores.

O sistema de elevadores proposto incorpora modelos heurísticos e lógica nebulosa como para definir a atribuição de elevadores. Para o estudo das técnicas propostas realizou-se um simulador, o qual foi projetado usando as tecnologias de automação industrial. O simulador integrou vários elementos e constituiu-se em um trabalho complexo, devido à grande quantidade de componentes necessários para serem sincronizados. Os diversos componentes de *software* tornam-se potenciais pontos de falha: um simples software mal configurado ou incompatibilidade com o sistema operacional, pode causar um comportamento do simulador fora do esperado.

Para desenvolver o *software* que integra todos os componentes do simulador, necessitou-se analisar o funcionamento de cada ferramenta de automação compatível com a arquitetura de elevadores proposta. Portanto foi necessário adquirir experiência com todas as atividades envolvidas com o uso dos sistemas: RSLinx, RSLogix 5000 e RSLogixEmulate 5000. Desta forma, desenvolveu-se, a partir da experiência e do conhecimento adquirido, o software que gerencia as ferramentas que compõem o sistema de simulação, baseada na modelagem UML (*Unified Modeling Language*). No entanto, hoje em dia existem pesquisadores que não usam a modelagem UML, devido a

que implica a aprendizagem de uma ferramenta de desenvolvimento de *software*, o que requer tempo no domínio desta modelagem. Entretanto, o uso da modelagem UML permite que a ferramenta de *software* possa facilmente ser adaptada a diferentes ambientes caso necessário, de modo a possibilitar estudos mais aprofundados na área de controle de grupo de elevadores. Demonstrou-se de forma didática a potencialidade de utilização da UML na especificação de tempo real a partir da perspectiva da Orientação a Objetos (OO). Além disso, a modelagem UML apresentada neste trabalho aspira deixar uma metodologia de desenvolvimento aberta, de modo a mostrar a aplicação da modelagem aos sistemas de elevadores com SCD.

O *software* desenvolvido apresentou desempenho adequado, quando no computador em que rodava o *software* de simulação (cliente) só executava esta única tarefa. Uma das limitações do *software* é uso de ferramentas fornecidas pela empresa privada Rockwell Automation. Entretanto, os resultados obtidos (tempo de espera) nas simulações mostraram que o sistema de simulação realizado satisfaz os requerimentos desta pesquisa. Embora, tenha sido realizada uma análise estatística, de modo a estudar os resultados das simulações, obteve-se um resultado com baixa precisão. Uma possível solução para este problema seria trabalhar com um número de dados maior, ou seja, uma população que deseje viajar mais alta. Isso implicaria em tempos de simulação maiores, em que o período transiente afetaria em uma menor proporção os resultados obtidos. Também poderiam ser melhorados os resultados das simulações, eliminando os valores do período transiente em cada simulação, de modo só trabalhar com os resultados do período estacionário. Para facilitar a reprodução das simulações apresenta-se no anexo C os arquivos necessários para a execução das simulações realizadas neste trabalho.

Após a realização das simulações, iniciou-se um estudo sobre as curvas de desempenho de controladores. Realizou-se uma análise comparativa entre os tempos de espera dos sistemas propostos. Para a realização da análise comparativa os sistemas de controle propostos foram classificados em dois tipos: (a) um sistema de controle com algoritmo de ordenamento (CAO) e (b) um sistema de controle sem algoritmo de ordenamento (SAO). Esta abordagem, permitiu estudar o desempenho dos controladores propostos, possibilitando avaliar as vantagens da arquitetura usada. Desta maneira, para facilitar a análise dividiu-se o comportamento do desempenho de controle em três regiões chamadas de fluxo baixo, médio e alto. Inicialmente, a análise do fluxo alto permitiu observar o comportamento do sistema de controle nos limites, em que se

buscou aproveitar as vantagens da arquitetura moderna, por meio do algoritmo de ordenamento. Portanto, evidenciaram-se algumas melhorias relacionadas à capacidade de transporte, mas também deficiências em relação à atribuição dos passageiros aos elevadores, devido à possibilidade de se ter dois ou mais elevadores com a mesma prioridade e não se poder otimizar qual elevador poderia ser o mais conveniente. Conseqüentemente, o fluxo na região de fluxo médio apresentou certa desvantagem no sistema de controle CAO em relação ao sistema SAO. Isso ocorreu devido ao fato de, no sistema CAO, as chamadas de pavimento de cada elevador com mesmos andares de origem e de destino serem sempre agendadas diretamente, sem passar por uma avaliação prévia do controlador nebuloso. Entretanto, os tempos de espera dentro do carro obtidos com o sistema de controle CAO apresentaram-se menores, devido principalmente pelo menor número de paradas. Finalmente, os controladores SAO e CAO apresentaram desempenhos aproximadamente iguais para fluxos classificados dentro da região baixa. Considerando que, para fluxos baixos o número de amostras também era baixo, a semelhança observada não pode ser considerada como totalmente verdadeira, uma vez que o período transiente das simulações pode ter afetado os resultados. Entretanto, fazendo uma análise qualitativa, para fluxos baixos, o sistema de elevadores trabalha com folga, o que lhe poderia possibilitar ter tempos reduzidos de espera em ambos tipos de controladores. Ainda, considerando que no caso do controle CAO há uma tentativa de se agruparem os passageiros com mesmos andares de origem e de destino, seria de se esperar que o sistema CAO apresentasse tempos de espera superiores àqueles obtidos pelo sistema SAO. Deve-se observar que o baixo número de dados obtidos deveu-se a limitações do *software* desenvolvido, cuja solução demandaria tempo adicional de programação, tempo este não disponível durante a realização desta pesquisa.

Para se verificar a compatibilidade dos resultados obtidos neste trabalho com os resultados apresentados por outros autores, de modo a demonstrar a validade do sistema de controle desenvolvido, incluíram-se na análise dos resultados gráficos de desempenho de sistemas semelhantes: (a) gráfico de comparação entre o sistema de controle SAO proposto e o sistema de controle tradicional (Siikonen, 2000), sem o sistema de controle de destino (vide Figura 5-12); (b) gráfico de comparação entre o sistema de controle CAO e o sistema de controle de destino de Siikonen (2000) (vide Figura 5-13); e (c) gráfico de comparação entre os sistemas de controle utilizados pela empresa KonePolaris™, incluindo controle tradicional, controle de destino típico e

controle KonePolaris™ (Konepolaris, 2009) (vide Figura 5-14). Por meio de observação das curvas apresentadas, pode-se observar que os controladores desenvolvidos neste trabalho apresentam comportamento compatível com aqueles observados na literatura. Entretanto, os desempenhos resultantes dos controladores CAO e SAO mostraram-se aquém dos desempenhos apresentados na literatura, provavelmente pelo fato de não ter havido tempo hábil para a realização de uma sintonia fina do controlador nebuloso, que por sua natureza, resultaria em um trabalho intenso de tentativa e erro. Para a melhoria do controlador, poder-se-ia sugerir que em trabalhos futuros se realizasse uma busca por melhores limites nos conjuntos das funções de pertinência ou mesmo se incluíssem outras regras na máquina de inferência. Poder-se-ia também buscar uma otimização do controlador nebuloso por meio de algoritmos genéticos ou outras técnicas de inteligência artificial. Além disso, poder-se-ia buscar uma otimização dinâmica do desempenho em função do fluxo de passageiros, por exemplo mesclando no controlador as virtudes dos diversos tipos de controladores encontrados na literatura.

No final da discussão, espera-se que este trabalho represente uma base inicial para estudos futuros na área de sistemas de elevadores, de modo a, contribuir no desenvolvimento de novos projetos nesta área. Desta maneira, proporcionar ao laboratório GRACO (Grupo de Automação e controle) da UNB o início de novas abordagens nos sistemas de controle de grupo de elevadores.

## 7 - CONCLUSÕES

O objetivo deste trabalho foi apresentar a implementação de técnicas de controle de grupo de elevadores, com vistas a sua utilização em uma arquitetura moderna dotada de um sistema de controle de destino (SCD). Esta arquitetura foi planejada para usar tecnologias de automação industrial. Desta maneira, desenvolveram-se ferramentas que possibilitam avaliar o desempenho das técnicas de controle propostas seguindo uma metodologia para a realização de cada etapa do trabalho.

Os resultados do trabalho permitiram concluir:

- É possível implementar técnicas de controle nebuloso para grupo de elevadores modernos (SCD) usando tecnologias de automação industrial.
- O sistema de simulação implementado possibilitou a avaliação qualitativa das técnicas de controle de interesse, permitindo estudar o comportamento do controlador
- A modelagem UML mostrou-se como uma ferramenta útil na implementação do simulador de sistema de controle de tráfego de elevadores, além documentá-lo detalhadamente, o que facilita a continuação de seu desenvolvimento no futuro.
- A linguagem UML pode ser utilizada na especificação em tempo real da perspectiva da Orientação a Objeto (OO), o que foi demonstrado de forma didática ao longo do texto deste trabalho.
- O sistema de controle com o algoritmo de ordenamento tem um melhor desempenho em relação ao sistema sem o algoritmo de ordenamento, possibilitando aproveitar a informação prévia sobre o destino dos passageiros, disponível em sistemas dotados de SCD, para aumentar a ocupação dos elevadores e reduzir o número de paradas.
- A sequência de tarefas seguidas neste trabalho para realizar o estudo, a implementação e a comparação das técnicas de controle de grupo de elevadores propostas pode constituir uma metodologia de abordagem de problemas deste tipo.



- Os sistemas de controle com e sem o algoritmo de ordenamento (CAO e SAO) desenvolvidos neste trabalho apresentaram comportamentos semelhantes aos encontrados na literatura.
- Este trabalho representa uma base inicial para pesquisas futuras na área de sistema de grupo de elevadores.

## 7.1- SUGESTÕES PARA TRABALHOS FUTUROS

Este projeto contribui no desenvolvimento de sistema de controle de grupo de elevadores no grupo de pesquisa GRACO da Universidade de Brasília - UnB. A idéia principal é criar uma metodologia que permita utilizar as vantagens das tecnologias de automação industrial tais como versatilidade, flexibilidade e escalabilidade, permitindo a exploração de novas abordagens para controlar sistemas integrados de automação em edifícios modernos.

As experiências descritas e análises dos dados foram realizadas a partir de técnicas de controle baseadas em lógica nebulosa. Uma sugestão de trabalhos futuros, para dar continuidade à pesquisa desenvolvida neste trabalho, poderia envolver o desenvolvimento de novos controladores, também baseados em técnicas de inteligência artificial, incluindo outros tipos de parâmetros de entrada. Por exemplo, poder-se-iam utilizar redes neurais ou lógica nebulosa, incrementando, por exemplo, o número de máquinas de inferência que permitam estimar outros tipos de parâmetros, de modo a melhorar o desempenho do controlador proposto.

Outro trabalho a ser realizado seria a aplicação de técnicas de otimização, buscando melhorar pontos específicos do controlador proposto, otimizando os conjuntos das funções de pertinência ou realizando otimização dinâmica nas regiões em que o sistema moderno de controle de destino apresenta um baixo desempenho.

A modelagem UML permite a reutilização de código, possibilitando continuar com o desenvolvimento do modelo inicialmente construído. Conseqüentemente, outra sugestão de trabalho futuro poderia ser continuar o desenvolvimento do *software* de modo a dotá-lo de uma maior modularidade na execução de testes de novos algoritmos de controle e possibilitá-lo trabalhar com uma população em viagem maior, de maneira a melhorar a análise estatística das simulações e permitir a obtenção de resultados mais precisos.

Nos sistemas atuais de automação predial, o sistema de controle de grupo integra varias funcionalidades, como o caso dos controles de acesso. Portanto, a criação de um sistema de base de dados que armazene a informação dos usuários possibilitaria vislumbrar outros tipos de soluções para melhorar o desempenho do sistema de controle de grupo de elevadores.

## REFERENCIAS BIBLIOGRÁFICAS

- ABNT (1987). Associação Brasileira de Normas Técnicas. “*Normas Técnicas Para Elevadores em Brasil.*” In: NBR10098.
- Aggarwal Tanuj, (2004), “Intelligent Control of Group Elevators”, Department of Aerospace Engineering, Mumbai, India.
- AMAYA, E. J. (2008). Aplicação de Técnicas de Inteligência Artificial no Desenvolvimento de um Sistema de Manutenção Baseada em Condição. Dissertação de Mestrado em Sistemas Mecatrônicos, Publicação ENM.DM-21A/08, Departamento de Engenharia Mecânica, Universidade de Brasília, Brasília, DF, 172p.
- Anwar, M.R., Anwar, O., Shamim, S.F. e Zahid, A.A. (2004), *Human Machine Interface Using OPC (OLE for Process Control)*, Engineering, Sciences and Technology, Student Conference On, pp. 35- 40.
- Barney G.C, (2003), “Elevator Traffic Handbook Theory And Practice”, Spon Press, London and New York.
- Bastidas, G. (1999). “Aplicação de Redes de Petri Interpretadas na Modelagem de Sistemas de Elevadores em Edifícios Inteligentes.” In: Dissertação (Mestrado), Escola Politécnica, Universidade de São Paulo, São Paulo, Brasil.
- Barney, G.C. and Imrak, E. (2001): applications of neural networks to lift traffic control, Elevator World, May. -11.4.4.
- Boaretto, N., (2005), “Tecnologia de comunicação em sistema SCADA– enfoque em Comunicação Wireless com espalhamento espectral.”. Monografia (Mestrado em Engenharia de Produção) CEFET-PR, Ponta Grossa.
- Beielstein Thomas, Markon S., MikePreuss, MIC (2003) “A Parallel Approach to Elevator Optimization Based on Soft Computing” in The Fifth Metaheuristics International Conference.
- Bezerra, E.(2002)“Princípios de Análise e Projeto de Sistemas com UML”. São Paulo: Campus.
- Booch, G. (1994) “Object-Oriented Analysis and Design with Applications”, Benjamim-Cummings PublishingCo.

- Bussab Wilton de O., Morettin Pedro A. (2003) “Estatística Básica”, Saraiva 5ª Edição, Inferência estatística, pag.255-283.
- Closs, G.D, (1972): The computer control of passenger traffic in large lift system, Phd thesis, UMIST.-10.2.3.
- Chan, W.L. and So, A.T.P. (1996): Dynamic zoning for intelligent supervisory control, Int J. Elev. Eng., 1.-11.4.4.
- Cortéz P., Delgado M., Ibañez N., Muñuzuri J. (2006), “Herramienta de planificación y simulación de sistemas de transporte vertical”, X congreso de Ingeniería de organización, Valencia.
- Conover, W. J., (1980), “Practical Nonparametric Statistics”. Wiley.
- Dos Santos, S. M. (1972), Lift simulation, Msc dissertation, UMIST.16.3.4
- Doebelin E. O.(2004) “Measurement Systems: application and design”, McGrawHill, Boston.
- Duarte, C., Figueiredo, L. e Corrêa, M. (2006), *Utilização do Matlab no Ensino da Tecnologia OPC Aplicada a Controle De Processos*, XVI Congresso Brasileiro de Automática, CBA 2006, Salvador, Bahia, Brasil.
- Fonseca, M. O. (2002), *Comunicação OPC, uma abordagem prática*. In VI Seminário De Automação de Processos, Associação Brasileira de Metalurgia e Materiais, Vitória,ES.
- Gonçalves C., F. (2002). Estatística. Universidade Estadual de Londrina. Brasil 304p. ISBN 85-7216-328-X.
- Ho, M. and Robertson, B. (1994): Elevator group supervisory control using fuzzy logic, Canadian Conference on Elevator and Computer Engineering, 2.-11.4.4.
- Hummet George T., Moser. D., Powell A. (1978), “Real time simulation of elevators”, IEEE computerSociety Press, Los Alamitos, CA, USA.
- Infolev. (acessado em 06/2010).  
Disponível em: <http://www.infolev.com.br/home/pagina/17>.
- Jackson, C. (1970), Analytical techniques: simulation case study, Architects Journal information Library, 8.-16.1.
- Jong Johannes, M. L. Siikonen, (2001) “New Trends in Elevating Solutions for medium

- to medium-high buildings to improve flexibility ”. CTBUH (Council on Tall Buildings and Urban Habitat) Conference, London, December.
- Kone Corporation, (2009), “KonePolaris™ Sistema de control de destino para lograr un flujo optimizado de pasajeros”, Disponível em: <http://www.kone.com>.
- Luo, F., Xu, Y., Cao, J. (2005). “*Elevator Traffic Flow Prediction With Least Squares Support Vector Machines.*”In: IEEE Proceedings of the Fourth International Conference on Machine Learning and Cibernetics, pp. 4266-4270, Guangzhou, China.
- Mamdani, E. H., Assilian, S. (1975). “*An Experiment in Linguistics Synthesis With a Fuzzy Logic Controller.*”In: International Journal Man-Machine Studies, Vol. 7(1), pp. 1-13.
- Mamdani, E. H. (1977). “*Application of Fuzzy Logic to Approximate Reasoning Using Linguistic Synthesis.*”In: IEEE Transaction on Computers, Vol. C(26), pp. 1182-1191.
- Markon S., Aoki ichi Ken, Masami Nakagawa, and Takeshi Sudo, (2008 )“Recent Trends Group Control System”, Conference System computers and communication.
- Markon S., Kita Hajime, Hiroshi Kise, Beielstein, (2006) “ Control of Traffic Systems inBuidings”, Advances in Industrial Control, Springer, pages. 69-78.
- Markon S. and Y. Nishikawa (2002). On the analysis and optimization of dynamic cellular automata with application to elevator control. In The 10th Japanese-German Seminar, Nonlinear Problems in Dynamical Systems, Theory and Applications. Noto Royal Hotel, Hakui, Ishikawa, Japan.
- Mello B. A. (2001), “Modelagem e simulação de sistemas”. Universidade Regional Integrada do Alto Uruguai e das Missões, Última atualização 10/10/2007.
- Miravete, A. (1999): Genetics and intense vertical traffic, Elevator World, July- 11.4.4.
- Muñoz, D. M. (2006). Implementação e Simulação de Algoritmos de Escalonamento para Sistemas de Elevadores Usando Arquiteturas Reconfiguráveis.  
Dissertação de Mestrado em Sistemas Mecatrônicos, Publicação ENM.DM-11A/06,  
Departamento de Engenharia Mecânica, Universidade de Brasília, Brasília, DF, 128p.

- Nikovsky, D., Brand, M. (2003). “*Decision-Theoretic Group Elevator Scheduling.*”  
In:AAAI Proc. 13th Int. Conference on Automated Planning and Scheduling, pp.  
133-142,Trento, Italy.
- Nunes, T. C. O., Rodrigues, V. H. (2009). Estudo da Viabilidade Técnica e  
financeira da Aplicação da Rede Industrial DeviceNet na Construção e Operação  
de Elevadores de Passageiros. Trabalho de Graduação em Engenharia de  
Controle e Automação, Publicação FT TG-016/2009Faculdade de Tecnologia,  
Universidade de Brasília, Brasília, DF, 124p.
- OPC Foundation – Dedicated to Interoperability in Automation. Disponível em:  
<http://www.opcfoundation.org/>. Acesso: 19 de julho de 2010.
- Otis Corporation. (acessado em 06/2010).  
Disponível em: <http://www.otisworldwide.com/>.
- Patiño-Forero. Alvaro A., (2009). “Modeling of elevator group control system using  
programmable logic control and destination control system”. 20th International  
congress of Mechanical Engineering (COBEM).November 15-20,  
Gramado(Brazil).
- Patiño-Forero. Alvaro A. (2010). “ Modelagem uml de um simulador de trafego  
elevadores e sua  
implementação por meio de um emulador de controlador lógico programável ”.  
Congresso Brasileiro de Automática (CBA). Setembro 12-16, Bonito (Brasil).
- Patiño-Forero Alvaro A. (2010). “ Modelagem uml e desenvolvimento de um simulador  
para  
controle difuso de grupo de elevadores” . Congresso Nacional de Mecânica  
(CONEM). Agosto 18-21, Campina Grande (Brasil).
- Pressman, Roger S. (2005) “Ingenieria del software” Mcgraw-Hill 6a edição.
- Peters, R.D. (1998). Simulation for control system design and traffic analysis, Elevator  
Technology 9, proceedings of ELEVCON'98.
- Qun, Z. Ding, S., Yu, C. and Xiaofeng, L. (2001): Elevator group control system  
modelling  
based on object orientated Petri Net, Elevator world, August.-11.4.4.
- Rockwell Automation, DeviceNet – Visão Geral do Sistema Publicação DN-2.5PT –  
Setembro 1999.
- Rockwell Automation, (acessado em 07/2010). Disponível em:

<http://www.ab.com/industrialcontrols/products/solid->

[tate\\_motor\\_control/soft\\_starters/smc-3\\_smc\\_devicenet.html](http://www.ab.com/industrialcontrols/products/solid-tate_motor_control/soft_starters/smc-3_smc_devicenet.html)

Rockwell Automation, (2005) “Programming Manual of Programming Languages”, Allen-

Bradley, August, 2005.

Rockwell Automation, (2000) “Sampler & tutorial of RSView32”, Rockwell software.

Rockwell Automation, (2006) “Programming Manual of Emulate 5000”, Allen-Bradley,

August, 2006.

Rumbaugh J., G. Booch, I. Jacobson, (1999) "The Unified Modeling Language

Reference

Manual", Addison Wesley.

Shaw, I.S., Godoy, M. (1999).“*Controle e Modelagem Fuzzy.*” In: Ed. Edgar

BlücherLtda,

FAPESP, São Paulo, Brasil.

Sectron Tecnologia vertical. (acessado em 06/2010).

Disponível em: <http://www.sectron.com.br/2/>.

Seixas Filho, C. Industrial Ethernet. (2003). Disponível

em:<<http://www.cpdee.ufmg.br/~seixas/PaginaSDA/Download/DownloadFiles/Ethernet.PDF>> Acesso em 16 Agosto. 2010.

Siikonen L. M., (2000) “On traffic planning methodology”. International Congress on Vertical Transportation. Berlin.

Siikonen, M.L. (1997), “*Elevator Group Control with Artificial Intelligence.*” In: Research Report A67. System Analysis of Technology, Helsinki University of Technology, pp. 1-32, Helsinki, Finland.

Shimanuki, Y. (1999), OLE for process control (OPC) for new industrial automation systems, In: Systems, Man, and Cybernetics, Vol. 6, pp. 1048-1050 vol.6, Tokyo, Japan.

Sorsa J., Hakonen and M. L. Siikonen, (2005) “Elevators selection with Destination Control System”. International Congress on Vertical Transportation. Peking.

Sorsa, J., Siikonen, M.L., Ehtamo, H. (2003). “*Optimal Control of Double-deck Elevator Group Using Genetic Algorithms.*” In: International Transactions in Operational Research, Vol. 10, pp. 103-114.

- Souza, L. C. A., Filho, C. S. e Pena, R. T. (1998), *Padrão de Acesso a Dados OPC e sua Implementação em um Driver OPC-MODBUS*, In: V Simpósio Regional de Instrumentação/ II Congresso Mineiro de Automação, ISA /GRINST – IBP, Belo Horizonte, p. 157-164.
- Taha Hamdy A. (2007), “Pesquisa Operacional”. Pearson 8ª Edição, Modelagem por simulação, pag. 272-287.
- Takagi, T., Sugeno, M. (1985), “*Fuzzy Identification of Systems and its Applications To Modeling and Control.*” In: IEEE Transactions Systems, Man, and Cybernetics, Vol.SMA-15(1), pp. 116-132.
- Tsukamoto, Y. (1979). “*An Approach to Fuzzy Reasoning Method.*” In: Advances in Fuzzy Set Theory and Applications, pp. 137-149, The Netherlands.
- Tudosobreimoveis. (acessado em 06/2010).  
Disponível em:  
<http://www.tudosobreimoveis.com.br/conteudo.asp?t=1&id=501&sid=9&subid>
- ThyssenKrupp Corporation (acessado em 06/2010).  
Disponível em:  
<http://www.thyssenkruppelevadores.com.br/sitenovo/site/Default.aspx>.
- ThyssenKrupp Corporation (acessado em 07/2010).  
Disponível em: <http://www.thyssenkruppelevadores.com/>
- Tonaco, R. P. (2008). Metodologia para Desenvolvimento de Base de Conhecimento Aplicada à Manutenção Baseada em Condição de Usinas Hidrelétricas, Publicação ENM.DM 22/08, Departamento de Engenharia Mecânica e Mecatrônica, Universidade de Brasília, Brasília, DF, 167p.
- Xfuzzy 3.0. *Ferramenta de CAD para Lógica Difusa*. (acessado em 08/2010).  
Disponível em: [http://www.imse.cnm.es/Xfuzzy/Xfuzzy\\_3.0/Xfuzzy3.0\\_sp.pdf](http://www.imse.cnm.es/Xfuzzy/Xfuzzy_3.0/Xfuzzy3.0_sp.pdf)
- Zadeh, L.A. (1965). “*Fuzzy Sets.*” In: Information and Control 8, University of California, pp. 338-353, USA.
- Zhifeng Pan, FeiLuo, YugeXu. (2007): “Elevator Traffic flow model based on dynamic



passanger distribution ” IEEE International Conference Automation Guangzhou,  
CHINA - May 30.

## **ANEXO A**

## **TRABALHOS PUBLICADOS**

### **Modelo de controle de grupo de elevadores**

- Patiño-Forero. Alvaro A., (2009). “Modeling of elevator group control system using programmable logic control and destination control system”. 20th International congress of Mechanical Engineering (COBEM).November 15-20, Gramado(Brazil).

### **Modelagem do simulador de tráfego de controle de grupo de elevadores**

- Patiño- Forero. Alvaro A. (2010). “ Modelagem uml de um simulador de trafego elevadores e sua implementação por meio de um emulador de controlador lógico programável ”. Congresso Brasileiro de Automática (CBA). Setembro 12-16, Bonito (Brasil).
- Patiño- Forero. Alvaro A. (2010). “ Modelagem uml e desenvolvimento de um simulador para controle difuso de grupo de elevadores” . Congresso Nacional de Mecânica (CONEM). Agosto 18-21, Campina Grande (Brasil).

## **ANEXO B**

**TABELADE VALORES DE H E S POR EN81 TAXA DE CAPACIDADE**

**(Adaptada de Barney , 2003)**

8 (6,4)		10 (8,0)		13(10,4)		16(12,8)		21(16,8)		26(20,8)	
H	S	H	S	H	S	H	S	H	S	H	S
4,7	3,8	4,8	4,2	4,9	4,5	4,9	4,7	5	4,9	5	5
5,6	4,1	5,7	4,6	5,8	5,1	5,9	5,4	6	5,7	6	5,9
6,5	4,4	6,6	5	6,8	5,6	6,8	6	6,9	6,5	7	6,7
7,4	4,6	7,5	5,3	7,7	6	7,8	6,6	7,9	7,2	7,9	7,5
8,2	4,8	8,4	5,5	8,6	6,4	8,7	7	8,8	7,8	8,9	8,2
9,1	4,9	9,3	5,7	9,5	6,7	9,7	7,4	9,8	8,3	9,9	8,9
10	5	10,2	5,9	10,5	6,9	10,6	7,8	10,8	8,8	10,8	9,5
10,8	5,1	11,1	6	11,4	7,1	11,5	8,1	11,7	9,2	11,8	10
11,7	5,2	12	6,1	12,3	7,3	12,5	8,3	12,7	9,6	12,8	10,5
12,6	5,3	12,9	6,3	13,2	7,5	13,4	8,6	13,6	10	13,7	11
13,4	5,4	13,8	6,4	14,1	7,7	14,3	8,8	14,6	10,3	14,7	11,4
14,3	5,4	14,7	6,5	15	7,8	15,3	9	15,5	10,6	15,7	11,8
15,3	5,5	15,6	6,5	16	8	16,2	9,2	16,5	10,9	16,6	12,2
16	5,5	16,6	6,6	16,9	8,1	17,1	9,3	17,4	11,1	17,6	12,5
16,9	5,6	17,4	6,7	17,8	8,2	18,1	9,5	18,4	11,3	18,5	12,8
17,8	5,6	18,2	6,7	18,7	8,3	19	9,6	19,3	11,6	19,5	13,1
18,6	5,6	19,1	6,8	19,6	8,4	19,9	9,8	20,3	11,7	20,5	13,4
19,5	5,7	20	6,8	20,5	8,4	20,9	9,9	21,2	11,9	21,4	13,6
20,4	5,7	20,9	6,9	21,4	8,5	21,8	10	22,1	12,1	22,2	13,9
21,2	5,7	21,8	6,9	22,4	8,6	22,7	10,1	22,1	12,3	23,3	14,1

## **ANEXO C**

## DADOS DE ENTRADA DO SIMULADOR

<i>PPS = 2% (passageiros/5minutos)</i>		
Andar de destino	Andar de origem	Tempo de chamada
1	3	09:22:40
1	8	09:23:17
1	5	09:23:20
1	2	09:24:07
1	2	09:24:32
1	2	09:24:47
1	7	09:25:03
1	3	09:25:39
1	3	09:25:42
1	8	09:26:51

<i>PPS = 4% (passageiros/5minutos)</i>		
Andar de destino	Andar de origem	Tempo de chamada
1	3	00:35:02
1	4	00:35:43
1	7	00:36:25
1	2	00:36:31
1	6	00:36:39
1	5	00:36:50
1	5	00:37:22
1	5	00:37:25
1	5	00:37:28
1	2	00:37:50
1	8	00:38:35
1	4	00:38:56
1	5	00:38:59
1	4	00:39:06
1	3	00:39:27
1	8	00:39:45
1	3	00:39:48
1	7	00:40:02

<i>PPS = 6% (passageiros/5minutos)</i>		
Andar de destino	Andar de origem	Tempo de chamada
1	6	00:57:08
1	7	00:57:14
1	8	00:57:20
1	3	00:57:21
1	3	00:57:27
1	5	00:57:37
1	5	00:58:36
1	7	00:58:43
1	6	00:59:01
1	8	00:59:04

1	6	00:59:07
1	5	00:59:12
1	5	00:59:21
1	2	00:59:34
1	5	00:59:38
1	7	00:59:49
1	5	01:00:03
1	7	01:00:05
1	4	01:00:08
1	5	01:00:28
1	3	01:00:41
1	5	01:01:10
1	8	01:01:15
1	5	01:01:23
1	4	01:01:37
1	8	01:01:51
1	6	00:57:08

<i>PPS = 8% (passageiros/5minutos)</i>		
Andar de destino	Andar de origem	Tempo de chamada
1	2	16:41:36
1	7	16:41:40
1	3	16:41:52
1	8	16:41:54
1	8	16:42:12
1	4	16:42:22
1	2	16:42:36
1	7	16:42:38
1	3	16:42:47
1	4	16:42:53
1	6	16:42:54
1	2	16:43:03
1	7	16:43:04
1	5	16:43:20
1	4	16:43:26
1	8	16:43:27
1	4	16:43:33
1	2	16:43:49
1	8	16:43:53
1	3	16:44:05
1	5	16:44:42
1	6	16:45:04
1	6	16:45:05
1	2	16:45:18
1	3	16:45:26
1	6	16:45:34
1	5	16:45:42
1	2	16:45:50



1	8	16:45:54
1	3	16:45:56
1	3	16:45:59
1	2	16:46:00
1	3	16:46:01
1	8	16:46:21
1	2	16:41:36

<i>PPS = 10% (passageiros/5minutos)</i>		
Andar de destino	Andar de origem	Tempo de chamada
1	8	22:26:19
1	4	22:26:25
1	7	22:26:31
1	7	22:26:40
1	3	22:26:42
1	4	22:26:54
1	8	22:26:57
1	4	22:27:12
1	5	22:27:13
1	8	22:27:22
7	1	22:27:30
1	6	22:27:40
1	4	22:27:44
1	8	22:27:49
1	6	22:27:56
1	3	22:28:01
1	5	22:28:03
1	7	22:28:23
1	3	22:28:29
1	2	22:28:39
1	8	22:28:48
1	3	22:28:49
1	2	22:28:53
1	3	22:28:58
1	3	22:29:03
1	7	22:29:14
1	8	22:29:15
1	8	22:29:20
1	4	22:29:24
1	7	22:29:26
1	8	22:29:43
1	6	22:29:44
1	8	22:29:47
1	5	22:29:50
1	7	22:29:53
1	8	22:29:59
1	6	22:30:07
1	3	22:30:14

1	6	22:30:37
1	2	22:30:42
1	3	22:30:44
1	5	22:30:47
1	8	22:26:19

<i>PPS = 12% (passageiros/5minutos)</i>		
Andar de destino	Andar de origem	Tempo de chamada
1	8	23:06:10
1	3	23:06:18
1	3	23:06:20
1	3	23:06:37
1	5	23:06:50
1	3	23:06:56
1	7	23:07:00
1	6	23:07:05
1	8	23:07:10
1	3	23:07:15
1	4	23:07:18
1	7	23:07:28
1	2	23:07:31
1	5	23:07:34
1	2	23:07:49
1	3	23:07:50
1	5	23:07:52
1	3	23:07:55
1	7	23:07:59
1	2	23:08:11
1	7	23:08:15
1	2	23:08:19
1	5	23:08:27
1	2	23:08:30
1	8	23:08:33
1	8	23:08:42
1	3	23:08:47
1	2	23:08:54
1	3	23:08:56
1	4	23:09:00
1	5	23:09:06
1	3	23:09:07
1	6	23:09:10
1	2	23:09:11
1	5	23:09:30
1	6	23:09:33
1	8	23:09:37
1	2	23:09:39
1	7	23:09:41
1	4	23:09:45

1	2	23:09:48
1	8	23:09:49
1	2	23:09:55
1	4	23:09:56
1	4	23:09:57
1	5	23:10:03
1	3	23:10:04
1	2	23:10:07
1	4	23:10:15
1	5	23:10:16
1	5	23:10:22
1	5	23:10:25
1	8	23:06:10

<i>PPS = 14% (passageiros/5minutos)</i>		
Andar de destino	Andar de origem	Tempo de chamada
1	4	23:39:05
1	8	23:39:08
1	8	23:39:12
1	5	23:39:14
1	6	23:39:15
1	5	23:39:16
1	5	23:39:20
1	3	23:39:26
1	4	23:39:30
1	7	23:39:33
1	3	23:39:36
1	4	23:39:42
1	3	23:39:44
1	7	23:39:48
1	5	23:39:52
1	4	23:39:57
1	7	23:40:03
1	3	23:40:16
1	5	23:40:21
1	4	23:40:23
1	4	23:40:25
1	3	23:40:30
1	4	23:40:39
1	4	23:40:43
1	8	23:40:52
1	6	23:41:00
1	5	23:41:01
1	3	23:41:09
1	6	23:41:12
1	3	23:41:16
1	2	23:41:17
1	7	23:41:18

1	8	23:41:20
1	8	23:41:22
1	4	23:41:30
1	3	23:41:35
1	5	23:41:38
1	7	23:41:41
1	7	23:41:51
1	3	23:41:53
1	4	23:41:55
1	3	23:41:57
1	7	23:42:00
1	7	23:42:16
1	5	23:42:23
1	8	23:42:27
1	4	23:42:44
1	6	23:42:45
1	7	23:42:50
1	7	23:42:52
1	2	23:42:54
1	3	23:43:15
1	8	23:43:34
1	8	23:43:36
1	3	23:43:37
1	3	23:43:38
1	4	23:43:47
1	7	23:43:48
1	4	23:43:57

<i>PPS = 16% (passageiros/5minutos)</i>		
Andar de destino	Andar de origem	Tempo de chamada
1	3	00:02:28
1	2	00:02:48
1	2	00:02:52
1	2	00:02:54
1	4	00:02:55
1	5	00:02:58
1	8	00:02:59
1	4	00:03:04
1	7	00:03:05
1	7	00:03:16
1	2	00:03:19
1	6	00:03:24
1	8	00:03:27
1	4	00:03:29
1	3	00:03:35
1	2	00:03:37
1	4	00:03:51
1	4	00:03:52

1	5	00:03:54
1	6	00:03:56
1	3	00:03:57
1	3	00:04:11
1	2	00:04:16
1	3	00:04:17
1	4	00:04:19
1	6	00:04:20
1	3	00:04:25
1	7	00:04:30
1	5	00:04:39
1	8	00:04:40
1	4	00:04:42
1	2	00:04:45
1	3	00:04:50
1	5	00:04:54
1	3	00:04:58
1	4	00:05:02
1	7	00:05:10
1	6	00:05:21
1	7	00:05:27
1	4	00:05:30
1	8	00:05:32
1	8	00:05:34
1	4	00:05:36
1	6	00:05:38
1	6	00:05:39
1	5	00:05:41
1	8	00:05:43
1	6	00:05:48
1	4	00:05:52
1	8	00:05:54
1	5	00:06:04
1	5	00:06:10
1	8	00:06:14
1	5	00:06:15
1	3	00:06:18
1	5	00:06:20
1	2	00:06:24
1	5	00:06:27
1	8	00:06:32
1	8	00:06:36
1	4	00:06:40
1	7	00:06:45
1	7	00:06:49
1	2	00:06:53
1	7	00:07:06
1	5	00:07:09

1	3	00:07:13
---	---	----------

<i>PPS = 18% (passageiros/5minutos)</i>		
Andar de destino	Andar de origem	Tempo de chamada
1	8	22:12:21
1	7	22:12:23
1	3	22:12:24
1	6	22:12:25
1	6	22:12:27
1	3	22:12:28
1	7	22:12:29
1	6	22:12:31
1	4	22:12:33
1	2	22:12:37
1	3	22:12:39
1	7	22:12:43
1	5	22:12:53
1	3	22:12:55
1	4	22:12:57
1	4	22:12:59
1	8	22:13:03
1	2	22:13:04
1	2	22:13:07
1	8	22:13:11
1	2	22:13:21
1	3	22:13:23
1	8	22:13:26
1	3	22:13:30
1	8	22:13:32
1	8	22:13:34
1	7	22:13:39
1	3	22:13:43
1	2	22:13:48
1	4	22:13:51
1	5	22:14:00
1	2	22:14:01
1	2	22:14:03
1	8	22:14:12
1	4	22:14:15
1	7	22:14:18
1	6	22:14:20
1	4	22:14:22
1	2	22:14:24
1	7	22:14:26
1	6	22:14:28
1	2	22:14:30
1	3	22:14:31
1	2	22:14:32

1	8	22:14:36
1	8	22:14:39
1	2	22:14:24
1	7	22:14:26
1	6	22:14:28
1	2	22:14:30
1	3	22:14:31
1	2	22:14:32
1	8	22:14:36
1	8	22:14:39
1	7	22:14:40
1	3	22:14:41
1	4	22:14:43
1	3	22:14:45
1	5	22:14:47
1	7	22:14:48
1	4	22:14:52
1	7	22:14:55
1	6	22:14:59
1	6	22:15:00
1	7	22:15:01
1	8	22:15:05
1	2	22:15:07
1	6	22:15:10
1	5	22:15:12
1	2	22:15:13
1	2	22:15:14
1	3	22:15:20
1	2	22:15:25
1	3	22:15:32
1	2	22:15:36
1	5	22:15:39