

UNIVERSIDADE FEDERAL DE SANTA CATARINA  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO

CAPÍTULO 1  
1.1

1.1.1

1.1.1.1

1.1.1.1.1

## Distâncias em Redes de Transporte - Um Enfoque Usando Sistemas Especialistas

Dissertação submetida à Universidade Federal de Santa Catarina para  
obtenção do grau de Mestre em Engenharia.



UFSC-BU

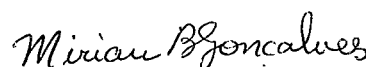
Karin Cristina Siqueira


Abril de 1998.

# DISTÂNCIAS EM REDES DE TRANSPORTE - UM ENFOQUE USANDO SISTEMAS ESPECIALISTAS

Karin Cristina Siqueira

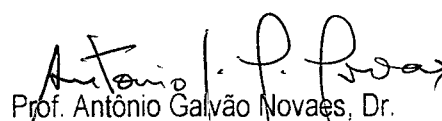
Esta dissertação foi julgada para a obtenção do título de Mestre em Engenharia, especialidade Engenharia de Produção, área de concentração Transporte e Logística, e aprovada em sua forma final pelo curso de Pós-Graduação.

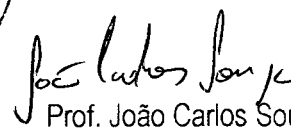
  
Prof.<sup>a</sup> Mirian Buss Gonçalves, Dr.<sup>a</sup>  
Orientadora

  
Prof. Ricardo Miranda Barcia, Ph.D.  
Coordenador do Curso de Pós-Graduação em Engenharia de Produção

Banca Examinadora:

  
Prof.<sup>a</sup> Mirian Buss Gonçalves, Dr.<sup>a</sup>

  
Prof. Antônio Galvão Novaes, Dr.

  
Prof. João Carlos Souza, Dr.

  
Prof.<sup>a</sup> Cléide Regina Lentz Paladini, Msc.

## ***Resumo***

No planejamento de grande parte dos serviços urbanos, as medidas de distância têm papel importante. Numa fase inicial do planejamento de um sistema urbano, não é usual medir as distâncias reais, mas sim aproximar estas por medidas geométricas simples na busca de soluções mais abrangentes. Usa-se para tanto uma métrica espacial, sendo esta ponderada posteriormente por um coeficiente de correção.

Os Sistemas Especialistas constituem atualmente uma das áreas mais interessantes de aplicações da Inteligência Artificial. Aplicam técnicas de inferência e conhecimento humano em problemas específicos de um dado domínio para simular a atuação de peritos humanos, servindo principalmente como consultores para usuários não especialistas.

O presente trabalho tem como objetivo inicial analisar os critérios que devem ser considerados essenciais para a obtenção de uma estimativa para a distância média esperada e para o coeficiente de correção, critérios esses que dependem de certas características da rede de transporte, como a forma da região de estudos, a localização da unidade de serviço e a existência ou não de barreiras de viagem.

Como objetivo principal, propõe-se a elaboração e construção de um protótipo de Sistema Especialista que atue como consultor para planejadores em formação e que, com a entrada de alguns dados feita por este, venha a informar estimativas para a distância esperada de viagem e para o coeficiente de correção em situações específicas.

## ***Abstract***

Measures of distance play an important role in the planning of most urban services. During the initial planning phase of an urban system, approximated values of distance obtained with simple geometric measures are used instead of the real values in the search for more general solutions. Later, these measures will be weighted with a corrective coefficient.

Nowadays, Expert Systems are one of the most interesting fields for application of Artificial Intelligence. These systems apply inference techniques and human knowledge to an specific problem simulating the performance of human specialists, working mainly as advisers to unskilled users.

This work presents the essential criteria that have to be considered for obtaining an estimated value of the expected mean distance and the corrective coefficient. These criteria depend on some specific characteristics of the transportation network, such as the district design, the response unit's position and whether barriers to travel exist or not.

As its main purpose, this work proposes the design and development of a prototype of Expert System that works as an adviser to unskilled planners. This prototype is able to estimate the expected mean distance and the corrective coefficient for specific situations based on information provided by the user.

# Índice

<b>RESUMO</b>	<b>I</b>
<b>ABSTRACT</b>	<b>II</b>
<b>INTRODUÇÃO</b>	<b>1</b>
1.1. APRESENTAÇÃO DO PROBLEMA	1
1.2. OBJETIVOS DO TRABALHO	2
1.3. METODOLOGIA	2
1.4. ESTRUTURA DO TRABALHO	3
<b>O DOMÍNIO DE CONHECIMENTO</b>	<b>4</b>
2.1. MÉTRICAS ESPACIAIS	4
2.1.1. MÉTRICA EUCLIDEANA	4
2.1.2. MÉTRICA RETANGULAR	5
2.2. DISTÂNCIA ESPERADA DE VIAGEM NO ESPAÇO CONTÍNUO	6
2.2.1. CASO SIMPLES: REGIÃO REGULAR E DISTRIBUIÇÃO UNIFORME	7
2.2.2. CASO GERAL	9
2.2.3. CASOS ESPECIAIS	10
2.3. DISTÂNCIA ESPERADA DE VIAGEM NA GRADE VIÁRIA	11
2.3.1. ESTIMATIVA DO COEFICIENTE DE CORREÇÃO	14
2.4. PERTURBAÇÕES ADICIONAIS: IMPEDIMENTOS E BARREIRAS DE VIAGEM	15
2.4.1. BARREIRAS DE VIAGEM VERTICAIS	19
2.4.2. BARREIRAS DE VIAGEM HORIZONTAIS	20
2.4.3. UM CASO A PARTE: BARREIRA DE VIAGEM INCLINADA	21
2.5. ESTIMATIVA DA DISTÂNCIA ESPERADA: COEFICIENTES DE CORREÇÃO	27
<b>INTELIGÊNCIA ARTIFICIAL E SISTEMAS ESPECIALISTAS</b>	<b>30</b>
3.1. A INTELIGÊNCIA ARTIFICIAL	30
3.2. A INTELIGÊNCIA ARTIFICIAL NA HISTÓRIA	32
3.3. PRINCIPAIS ÁREAS DE APLICAÇÃO DA INTELIGÊNCIA ARTIFICIAL	34
3.3.1. SISTEMAS ESPECIALISTAS	35
3.3.2. PROCESSAMENTO DE LINGUAGEM NATURAL	35
3.3.3. RECONHECIMENTO DE PADRÕES E PERCEPÇÃO	35
3.3.4. ROBÓTICA	36
3.3.5. PROVA DE TEOREMAS	36
3.3.6. JOGOS	37
3.4. OS SISTEMAS ESPECIALISTAS	37
3.5. ARQUITETURA DE UM SISTEMA ESPECIALISTA	40
3.5.1. BASE DE CONHECIMENTOS	40
3.5.2. MÁQUINA DE INFERÊNCIA	41

3.5.3. A INTERFACE	42
<b>3.6. ALGUNS SISTEMAS ESPECIALISTAS</b>	<b>42</b>
<b>3.7. FASES DE DESENVOLVIMENTO DE UM SISTEMA ESPECIALISTA</b>	<b>44</b>
3.6.1. FASE DE IDENTIFICAÇÃO	45
3.6.2. FASE DE CONCEITUAÇÃO	45
3.6.3. FASE DE FORMALIZAÇÃO	46
3.6.4. FASE DE IMPLEMENTAÇÃO	46
3.6.5. FASE DE TESTE E AVALIAÇÃO	47
3.6.6. FASE DE REVISÃO	47

---

**A SHELL KAPPA** **49**

<b>4.1. INFORMAÇÕES GERAIS</b>	<b>49</b>
<b>4.2. DIFERENÇAS ENTRE KAPPA E PROCESSADORES CONVENCIONAIS</b>	<b>50</b>
<b>4.3. DEFINIÇÃO DOS OBJETOS, CLASSES E SLOTS</b>	<b>51</b>
<b>4.4. A CARACTERIZAÇÃO ATRAVÉS DE SLOTS</b>	<b>52</b>
<b>4.5. REGRAS, FUNÇÕES E MÉTODOS</b>	<b>53</b>
<b>4.6. FORMAS DE RACIOCÍNIO</b>	<b>55</b>
<b>4.7. A LINGUAGEM KAL</b>	<b>56</b>
<b>4.8. INTERFACE</b>	<b>58</b>
4.8.1. <i>OBJECT BROWSER</i>	59
4.8.2. <i>SESSION</i>	59
4.8.3. <i>EDIT TOOLS</i>	59
4.8.4. <i>KAL INTERPRETER</i>	60
4.8.5. <i>KAL VIEW DEBUGGER</i>	60

---

**O PROTÓTIPO PROPOSTO** **62**

<b>5.1. DEFINIÇÃO DO PROBLEMA</b>	<b>62</b>
<b>5.2. OBJETIVOS DO PROTÓTIPO</b>	<b>63</b>
<b>5.3. RECURSOS NECESSÁRIOS</b>	<b>63</b>
<b>5.4. ESTRUTURA DO PROTÓTIPO</b>	<b>63</b>
<b>5.5. REPRESENTAÇÃO DO CONHECIMENTO</b>	<b>65</b>
5.5.1. A BASE DE CONHECIMENTOS	65
5.5.2. A INTERFACE COM O USUÁRIO	68

---

**CONCLUSÕES E PERSPECTIVAS** **76**

<b>6.1. RESUMO E CONTRIBUIÇÕES DO TRABALHO</b>	<b>76</b>
<b>6.2. PROPOSTAS DE CONTINUIDADE DE TRABALHO E PERSPECTIVAS FUTURAS</b>	<b>78</b>

---

**REFERÊNCIAS BIBLIOGRÁFICAS** **79**

---

**BIBLIOGRAFIA COMPLEMENTAR** **81**

---

**ANEXO** **83**

## Figuras

Figura 2.1 - Distância euclideana entre os pontos A e B.	5
Figura 2.2 - Distância retangular entre os pontos A e B.	6
Figura 2.3 - Pontos situados numa grade retangular.	12
Figura 2.4 - Barreira de viagem posicionada em $x = b$ e de altura $y = a$ .	16
Figura 2.5 - Distância extra percorrida devido à barreira.	16
Figura 2.6 - Local onde deve-se construir a ponte.	19
Figura 2.7 - Tipos de barreiras de viagem verticais.	20
Figura 2.8 - Tipos de barreiras de viagem horizontais.	21
Figura 2.9 – Barreira de viagem inclinada.	22
Figura 2.10 - Distância percorrida quando $(x_2, y_2)$ está no triângulo.	23
Figura 2.11 - Função distribuição acumulada por probabilidade geométrica.	23
Figura 2.12 - Distância percorrida quando $(x_2, y_2)$ está no retângulo.	25
Figura 2.13 – Distância esperada de viagem em uma região unitária, dada a existência de uma barreira de viagem inclinada	26
Figura 2.14 - Representação de regiões compactas e convexas e outras não.	28
Figura 4.1 - Estrutura hierárquica do sistema Poli 1.0.	52
Figura 4.2 - <i>Slots</i> da classe Triângulos.	53
Figura 4.3 – Janela <i>Edit Tools</i> .	60
Figura 4.4 – Mensagem de alerta do KAL View Debugger.	61
Figura 5.1 - Estrutura do módulo Aproximação de Distâncias	64
Figura 5.2 - Tela inicial do ForPlan.	69
Figura 5.3 - Apresentação do ForPlan.	70
Figura 5.4 - Ajuda sobre barreiras de viagem.	71
Figura 5.5 - Escolha sobre o espaço.	71
Figura 5.6 - Ilustração da rede viária.	72
Figura 5.7 - Entrada de dados referente à rede viária.	72
Figura 5.8 - Escolha quanto à configuração da barreira de viagem	73

Figura 5.9 - Resposta dada pelo sistema. _____	74
Figura 5.10 - Fórmula de onde foi derivado o resultado. _____	74

## **Quadros**

Quadro 2.1 - Distâncias esperadas de viagem para a métrica retangular. _____	10
Quadro 2.2 - Distâncias esperadas de viagem para a métrica euclideana. _____	11
Quadro 2.3 - Distância esperada de viagem dada uma barreira de viagem vertical. _____	20
Quadro 2.4 - Distância esperada de viagem dada uma barreira de viagem horizontal. _____	21
Quadro 2.5 - Distância extra percorrida considerando a existência de uma barreira de viagem inclinada e o serviço localizado no extremo _____	27
Quadro 2.6 - Coeficientes de correção para alguns casos específicos. _____	29
Quadro 2.7 - Coeficientes de correção para regiões compactas e convexas _____	29
Quadro 3.2 - Sistemas Especialistas nas diversas áreas de conhecimento. _____	44
Quadro 5.1 - Slots do ForPlan. _____	66



## *Introdução*

### **1.1. Apresentação do Problema**

Os Sistemas Especialistas (SE) constituem atualmente uma das áreas mais interessantes e com considerável variedade de aplicações bem sucedidas dentro da área de Inteligência Artificial (IA). Um Sistema Especialista é um sistema de computador (*software*) que usa o conhecimento nele armazenado e procedimentos de inferência para resolver problemas que exigem o conhecimento especializado em um dado domínio, simulando assim a atuação de um perito humano. Autores como Hayes-Roth, Waterman e Lenat (1983), dedicaram-se ao estudo detalhado destes sistemas.

Uma das destacadas aplicações destes sistemas ocorre quando tem-se muitas informações sobre um dado domínio de conhecimento e não se sabe como utilizá-las de forma dinâmica para encontrar a resposta que se deseja. De acordo com Rich (1988), o Sistema Especialista, devido a sua arquitetura, encadeia todo um raciocínio lógico a partir do conhecimento nele armazenado e de fatos inferidos sobre este conhecimento, até que alguma solução para o problema seja encontrada.

No planejamento de grande parte dos serviços urbanos, a distância desempenha um papel de destaque. Problemas como a localização de um serviço de atendimento emergencial ou a roteirização de veículos de entrega de mercadorias em domicílio, dependem essencialmente de medidas de distância. Problemas específicos onde a medida de distância é um fator essencial são abordados por Novaes em seus trabalhos (ver, por exemplo, Novaes (1986)).

Numa fase inicial do planejamento de um sistema de transporte, não é usual medir as distâncias reais da rede viária em questão, mas sim aproximar estas por outras mais simples e de fácil obtenção, na busca de soluções mais abrangentes. Usa-se, nestes casos, uma medida geométrica da distância entre os pontos, sendo esta ponderada por um coeficiente de correção adequado. As medidas geométricas mais comumente utilizadas são dadas pela Métrica Euclidiana e pela Métrica Retangular.

Para o planejador de tal serviço seria de grande interesse que a partir de algumas características conhecidas por ele quanto ao espaço em que ele está trabalhando, e mais especificamente quanto à rede de transporte, ele pudesse obter uma estimativa satisfatória tanto para distância média esperada entre pontos deste espaço, quanto para o coeficiente de correção entre a distância real e a distância calculada a partir das métricas. Estudos feitos por Larson (1971) indicam que as características mais relevantes nestes casos são as que se referem a topologia da região, a localização dos pontos nesta região e a existência ou não de barreiras de viagem, sendo que em caso afirmativo, sua localização e forma devem ser conhecidas.

O grande número de combinações destas e de outras características que determinam uma rede de transporte conduz a várias possibilidades de valores para o coeficiente de correção que se deseja estimar, bem como alteram significativamente o cálculo da distância média esperada.

Para solucionar tais problemas que dependem das especificações da rede de transporte, propõe-se com o presente trabalho um protótipo de Sistema Especialista de apoio à decisão que, através de uma interação amigável com o usuário feita por perguntas e respostas ou seleção de alternativas, indique ao planejador qual o coeficiente mais adequado a ser adotado e apresente uma estimativa da distância média entre os pontos para as diversas possibilidades.

## **1.2. Objetivos do Trabalho**

O objetivo principal do presente trabalho é desenvolver uma ferramenta computacional de apoio à formação de planejadores de serviços urbanos que auxilie na tomada de certas decisões estratégicas, construindo para tanto um protótipo de Sistema Especialista.

Especificamente dentro deste protótipo, objetiva-se estimar um valor para a distância média esperada de viagem entre pontos da região, bem como estimar um coeficiente de correção entre a distância real e a distância calculada através de alguma métrica. Os valores apresentados pelo sistema serão aqueles que melhor se adaptarem às características do problema que o planejador analisa e deseja solucionar.

## **1.3. Metodologia**

Para alcançar tais objetivos, primeiramente apresentar-se-á a teoria sobre aproximações de distâncias já desenvolvida, principalmente as métricas a serem adotadas e os modelos construídos para a estimativa da distância média esperada e do coeficiente de correção.

Apresentar-se-á também uma análise quanto a influência de barreiras de viagem sobre a distância média esperada de viagem. Tal análise será feita para algumas configurações específicas da rede de transporte, visto a complexidade dos cálculos que devem ser mais cuidadosamente processados.

Além disso, para tornar possível a construção do referido protótipo, serão revistos os conceitos principais de Inteligência Artificial e Sistemas Especialistas, as técnicas que serão utilizadas durante o desenvolvimento do mesmo, como também apresentar-se-á a *shell* dentro da qual o sistema será programado.

#### **1.4. Estrutura do Trabalho**

O texto descrevendo o protótipo de Sistema Especialista proposto nesta dissertação bem como a ferramenta utilizada na construção do mesmo, será organizado da seguinte forma:

- No capítulo 2 será apresentado o embasamento teórico do trabalho, visando descrever como usualmente são feitas as estimativas de valores importantes durante o planejamento de serviços urbanos, valores como a distância esperada de viagem e os coeficientes de correção. Neste mesmo capítulo também serão analisadas, em situações específicas, as influências de barreiras de viagem sobre estes valores.
- No terceiro capítulo serão introduzidos os conceitos básicos sobre Inteligência Artificial e Sistemas Especialistas. Também será descrita a arquitetura de um Sistema Especialista e ainda serão apresentadas algumas das principais áreas de aplicação da Inteligência Artificial.
- A ferramenta utilizada na construção do protótipo proposto neste trabalho será apresentada em detalhes no quarto capítulo.
- O quinto capítulo descreverá como foi estruturado o conhecimento durante a elaboração do protótipo e como o mesmo será apresentado, ainda que preliminarmente, aos usuários.
- No capítulo 6 serão apresentadas as conclusões e perspectivas de continuidade do presente trabalho.

## *O Domínio de Conhecimento*

Neste capítulo será apresentado o embasamento teórico que sustenta o presente trabalho. Falar-se-á, de um modo geral, sobre como é possível, de acordo com autores especialistas neste assunto, estimar a distância esperada de viagem e o coeficiente de correção entre a distância real e a distância calculada. Para que o estudo de tais estimativas seja possível, inicialmente deve-se conhecer as medidas geométricas de distância que serão utilizadas, quais sejam, as métricas espaciais.

### **2.1. Métricas Espaciais**

Um elemento básico do planejamento de serviços urbanos é a noção de distância. Problemas como a localização de um serviço emergencial e a roteirização de um veículo de entrega de mercadorias, dependem essencialmente de medidas de distância.

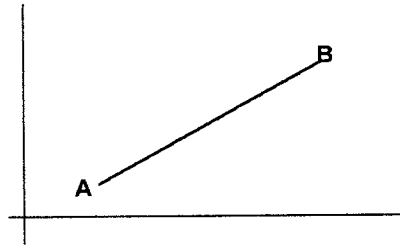
Cada sistema de transporte, seja urbano, rodoviário ou mesmo aéreo, possui vias de tráfego bem definidas, sendo que o conjunto destas, interligadas, forma uma rede. Os meios de transportes então, somente poderão se locomover através de sua respectiva rede. Cada uma destas redes é caracterizada por dois elementos: os nós, que são os pontos de origem e destino de viagens; e os arcos, que são as ligações existentes entre os diversos nós.

Porém, nas fases iniciais do planejamento, onde o planejador não preza tanto pela exatidão das respostas e sim por uma boa aproximação desta, é comum fazer uma simplificação da rede de transporte, considerando assim todas as possíveis ligações entre os pontos, ou seja, trabalhar sobre o espaço contínuo. A distância então será aproximada por medidas geométricas simples, medidas estas apresentadas a seguir.

#### **2.1.1. Métrica Euclideana**

Uma aproximação bastante útil no planejamento de serviços urbanos é aquela que faz uso da métrica euclideana para estimar a distância. A distância euclideana é medida como sendo a distância em linha reta entre os pontos, e assim sendo, ela será sempre menor ou igual

à distância real entre estes pontos. Na Figura 2.1 pode-se ver representada tal medida de distância.



**Figura 2.1 - Distância euclidiana entre os pontos A e B.**

As principais vantagens de se estimar a distância através desta métrica são quanto a simplicidade de representação analítica, facilitando assim os cálculos matemáticos, e quanto a sua característica de unicidade de percurso para o deslocamento de A até B, já que existe somente uma reta que une cada par de pontos.

Sejam  $A = (x_A, y_A)$  e  $B = (x_B, y_B)$  pontos quaisquer no plano. A distância euclidiana ( $D_E$ ) entre estes, é dada por:

$$D_E = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2} \quad (2.1)$$

A distância efetiva  $D$  correspondente ao percurso da rede viária, obedecendo as mãos de direção, restrições de conversão, etc, é estimada multiplicando-se o resultado acima por um coeficiente de correção médio  $r_E$ , como segue:

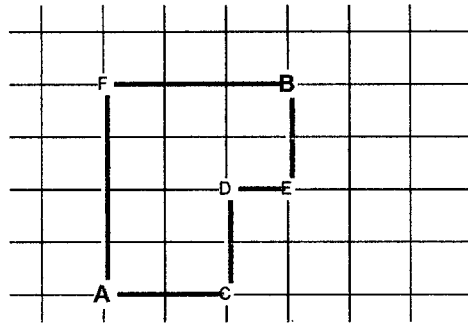
$$D = r_E D_E \quad (2.2)$$

Nos estudos referentes a malhas urbanas, comumente adota-se um acréscimo médio de trinta por cento sobre a distância euclidiana para a obtenção de uma aproximação inicial para a distância efetiva.

### 2.1.2. Métrica Retangular

Supondo uma malha viária na forma de uma grade retangular ou aproximada, com os eixos cartesianos orientados paralelamente nas duas direções predominantes das vias ortogonais; pode-se definir uma outra medida de distância, esta através da Métrica Retangular.

A distância retangular corresponde a qualquer uma das possíveis ligações ao longo da rede, sem retorno, entre os pontos  $A$  e  $B$ . Na Figura 2.2 observa-se que há diferentes caminhos a seguir para ir do ponto  $A$  ao ponto  $B$ , como, por exemplo, aquele que passa pelo ponto  $F$ , ou ainda aquele que passa pelos pontos  $C$ ,  $D$  e  $E$ . Logo, diferentemente da métrica euclidiana que garantia a unicidade, a métrica retangular possibilita vários caminhos possíveis a seguir, sendo que a escolha do melhor caminho vai depender dos interesses individuais de cada pessoa que deseja se deslocar de um ponto a outro.



**Figura 2.2 - Distância retangular entre os pontos A e B.**

A distância retangular ( $D_R$ ) entre os pontos  $A = (x_A, y_A)$  e  $B = (x_B, y_B)$  é dada por:

$$D_R = |x_B - x_A| + |y_B - y_A| \quad (2.3)$$

De acordo com a Regra da Marcha Sem Retrocesso (Novaes, 1989), que diz que o avanço ao longo dos dois eixos deve se processar sempre no mesmo sentido e sem retornos, pode-se afirmar que a distância percorrida entre dois pontos será sempre a mesma, não importando qual o caminho escolhido para se seguir.

Tal como foi afirmado para a distância euclideana, pode-se relacionar matematicamente a distância efetiva com a distância retangular através de um coeficiente de correção médio  $r_R$ , ou seja, pode-se estimar a distância efetiva  $D$  como sendo:

$$D = r_R D_R \quad (2.4)$$

Tanto o valor de  $r_R$  como o de  $r_E$  dependem da estrutura da malha viária local e da área da região em que se localizam os pontos. Maiores detalhes sobre como estimar tais valores serão apresentados mais adiante neste capítulo.

## 2.2. Distância Esperada de Viagem no Espaço Contínuo

Os sistemas que serão aqui estudados são, em sua maioria, espacialmente distribuídos; logo, a análise que será feita sobre os mesmos deve incorporar tais componentes espaciais. Para tanto, utilizar-se-á de propriedades de probabilidade geométrica.

Cada experimento de modelagem probabilística pode ser resolvido seguindo os quatro estágios descritos abaixo:

- 1º.) Definir as variáveis aleatórias de interesse;
- 2º.) Identificar o espaço de estados conjunto;
- 3º.) Determinar a distribuição de probabilidade conjunta sobre o espaço de estados;

4º.) Trabalhar sobre o espaço de estados para determinar as respostas de quaisquer questão sobre o experimento.

As respostas a serem encontradas neste último estágio dependerão essencialmente de características específicas do problema, tais como poderá se ver nos casos a seguir.

### 2.2.1. Caso Simples: Região Regular e Distribuição Uniforme

Supõe-se, em nível de simplificação, que a região  $R$  de estudo é regular ou aproximadamente regular, isto é, esta tem geometria conhecida e é fácil de se trabalhar sobre a mesma, regiões de forma como, por exemplo, quadrada, retangular ou circular. Supõe-se também que os pontos de demanda desta região são uniformemente distribuídos sobre a mesma.

De acordo com tais suposições, no último estágio apresentado anteriormente serão determinadas, na ordem apresentada, as seguintes funções:

$$(i) \text{ função de probabilidade acumulada: } F_Y(y) = P(Y \leq y) \quad (2.5)$$

$$(ii) \text{ função densidade de probabilidade: } f_Y(y) = \frac{dF_Y(y)}{dy} \quad (2.6)$$

$$(iii) \text{ distância esperada de viagem: } E[D] = \int y f_Y(y) dy \quad (2.7)$$

Este é dito o Método das Distribuições Derivadas, ou como costuma chamar Larson (1981), o Método que Nunca Falha. Generalizando, o mesmo consiste em:

1. Identificar o conjunto de pontos no espaço amostral original  $(X_1, X_2, \dots, X_N)$  correspondente ao evento conjunto:

$$\{Y_1 = g_1(X_1, \dots, X_N) \leq y_1; Y_2 = g_2(X_1, \dots, X_N) \leq y_2; \dots; Y_M = g_M(X_1, \dots, X_N) \leq y_M\} \quad (2.8)$$

2. Para cada conjunto de valores de  $y_i$  ( $i = 1, 2, \dots, M$ ), determinar a probabilidade, no espaço amostral  $(X_1, X_2, \dots, X_N)$ , deste evento conjunto, obtendo então:

$$F_{Y_1, Y_2, \dots, Y_M}(y_1, y_2, \dots, y_M) \quad \text{para} \quad -\infty < y_1, y_2, \dots, y_M < +\infty \quad (2.9)$$

3. Se a função (2.9) for “bem comportada”, então tem-se o seguinte:

$$f_{Y_1, Y_2, \dots, Y_M}(y_1, y_2, \dots, y_M) = \frac{\partial^M}{\partial y_1 \partial y_2 \dots \partial y_M} F_{Y_1, Y_2, \dots, Y_M}(y_1, y_2, \dots, y_M) \quad (2.10)$$

Um exemplo simples é aquele onde se deseja determinar a distância de resposta de um serviço emergencial aleatoriamente posicionado em  $(x_1, y_1)$  dentro de uma região quadrada de lados unitários, até um outro ponto também aleatório  $(x_2, y_2)$  dentro desta mesma região. Este tipo de situação pode ocorrer para um serviço de patrulhamento que se desloca em uma área,

exercendo assim um patrulhamento preventivo e ao mesmo tempo recebendo chamadas de incidentes que ocorrem nesta área.

Utilizando a métrica retangular, faz-se então a separação em dois sub-casos unidimensionais, isto é, representa-se a distância  $D_R$  como sendo a soma de duas funções:

$$D_x = |x_2 - x_1| \text{ e } D_y = |y_2 - y_1| \quad (2.11)$$

Seguindo os quatro estágios para a resolução do experimento e considerando a priori a distância  $D_x$ , tem-se:

1º.) Variáveis aleatórias:

$x_1 =$  local da viatura;  $0 \leq x_1 \leq 1$ .

$x_2 =$  local do incidente;  $0 \leq x_2 \leq 1$ .

2º.) Espaço de estados conjunto:

Quadrado unitário  $[0,1] \times [0,1]$ .

3º.) Distribuição de probabilidade conjunta:

$f_{X_1}(x_1) = f_{X_2}(x_2) = 1$ , logo,  $f_{X_1 X_2}(x_1, x_2) = 1$ .

4º.) (i) função de probabilidade acumulada:

$F_{D_x}(y) = P(D_x \leq y) = P(|x_2 - x_1| \leq y) = -y^2 + 2y$  para  $0 \leq y \leq 1$ .

(ii) função densidade de probabilidade:

$$f_{D_x}(y) = \frac{dF_{D_x}(y)}{dy} = 2(1 - y) \text{ para } 0 \leq y \leq 1.$$

(iii) distância esperada de viagem:

$$E[D_x] = \int_0^1 y f_{D_x}(y) dy = \frac{1}{3}.$$

A generalização deste caso se dá quando tem-se um retângulo de lados  $X_0$  e  $Y_0$  e os pontos de serviço e demanda aleatoriamente posicionados dentro deste. Como as variáveis aleatórias são independentes, então o valor estimado da distância média será dado pela soma do valor esperado da distância nos dois casos unidimensionais da seguinte maneira:

$$E[D] = E[D_{x_0} + D_{y_0}] = E[D_{x_0}] + E[D_{y_0}] = \frac{X_0}{3} + \frac{Y_0}{3} = \frac{X_0 + Y_0}{3} \quad (2.12)$$



### 2.2.2. Caso Geral

Até o presente momento a distância esperada de viagem somente foi estimada para regiões de forma relativamente regular e para pontos de demanda (incidentes) distribuídos uniformemente nesta região. Seria bastante útil poder fazer uso destes resultados como aproximações para situações mais gerais, ou seja, aquelas em que o distrito não tem uma forma muito regular e que os pontos de demanda não sejam uniformemente distribuídos.

Seja então  $(x_1, y_1)$  a representação cartesiana do local em que se encontra a unidade de serviço e  $(x_2, y_2)$  a representação do local onde o serviço é requisitado, ambos internos à região  $R$ . A função densidade de probabilidade (fdp) e a distância entre estes pontos serão denotadas respectivamente por:

$$f_{X_1, X_2, Y_1, Y_2}(x_1, x_2, y_1, y_2) \quad (2.13)$$

$$D = d[(x_2, y_2), (x_1, y_1)] \quad (2.14)$$

Neste caso, o valor esperado da distância viajada nesta região é estimado através da seguinte integração:

$$E[D] = \iiint_R d[(x_2, y_2), (x_1, y_1)] \cdot f_{X_1, X_2, Y_1, Y_2}(x_1, x_2, y_1, y_2) dx_1 dy_1 dx_2 dy_2 \quad (2.15)$$

podendo a mesma ser generalizada para o caso onde existam mais unidades de serviço que atendam a região  $R$ .

É fácil ver que na prática o valor de  $E[D]$  não é tão simples de computar. Tal cálculo torna-se inviável a medida que aumenta o número de variáveis aleatórias ou quando a forma da região ou a fdp conjunta tornam-se muito mais complexas.

Como o que interessa no momento é tão somente uma aproximação, então não importará nem o número de unidades de serviço que atendam a região, nem tão pouco a forma do distrito ou a fdp (Larson e Odoni, 1981).

Fazendo então uso destas hipóteses, ter-se-á que o valor esperado da distância viajada numa certa região  $R$  será estimado através dos seguintes passos:

1º.) Dividir o distrito  $R$  em várias partes não sobrepostas, chamando-as de zonas. Cada uma destas zonas deve ter:

- a) uma forma aproximadamente regular que seja simples de trabalhar, de preferência formas como triângulos e quadriláteros;
- b) uma fdp, única em cada zona, com distribuição de incidentes uniforme ou que possa ser aproximada por uma expressão mais simples.

2º.) Computar todos os valores esperados para a distância esperada de viagem, isto é, de todas as unidades de serviço aos pontos de demanda (incidentes).

3º.) Multiplicar cada valor esperado da distância pelas respectivas probabilidades aproximadas, para assim obter o valor total da distância esperada de viagem dentro da região  $R$ .

Uma das situações que podem ser resolvidas desta maneira é aquela onde se encontram barreiras de viagem inclinadas. Tal situação será apresentada como um caso especial ainda neste capítulo.

### 2.2.3. Casos Especiais

Restringir-se-á aqui, em nível de simplificação dos cálculos, a região de estudo como sendo da forma retangular. Através do Método das Distribuições Derivadas, o qual foi descrito na sessão 2.2.1, podem ser determinados os valores das distâncias esperadas de viagem para alguns casos em particular.

Considerar-se-á o ponto de serviço como sendo ou fixo, podendo este estar no centro ou num extremo da região, ou aleatoriamente posicionado dentro da região. Já o ponto de demanda, onde o serviço é requisitado, terá sempre uma posição aleatória. A métrica a ser utilizada poderá ser tanto a euclideana quanto a retangular, esta escolhida de acordo com as características do problema que se deseja solucionar.

Sejam então  $X_0$  e  $Y_0$  as dimensões da região de estudo. Sejam  $(x_1, y_1)$  a representação cartesiana do local em que se encontra a unidade de serviço e  $(x_2, y_2)$  a representação do local onde o serviço é requisitado, ambos internos à região de área  $X_0Y_0$ . De acordo com resultados obtidos por Larson e Odoni (1981) e Eilon *et al* (1971), pode-se estimar valores para a distância esperada de viagem em algumas situações particulares, estas subdivididas de acordo com a métrica a ser adotada, como segue:

i) Se a métrica for a retangular, o valor da distância esperada de viagem fica bastante simplificado e dependendo somente das dimensões da região. Como dito anteriormente, tais resultados foram obtidos como no exemplo da sessão 2.2.1, e estes são então apresentados no Quadro 2.1 abaixo.

	<b>Serviço Central</b>	<b>Serviço no Extremo</b>	<b>Serviço Aleatório</b>
<b>Distância Esperada de Viagem</b>	$\frac{(X_0 + Y_0)}{3}$	$\frac{(X_0 + Y_0)}{2}$	$\frac{(X_0 + Y_0)}{4}$

**Quadro 2.1 - Distâncias esperadas de viagem para a métrica retangular.**

ii) Se a métrica for a euclideana, os resultados tornam-se mais complexos e dependerão, além das dimensões da região, da localização do serviço, caso este esteja fixo. A expressão

genérica da distância esperada de viagem para os casos onde o ponto de serviço está fixo é a seguinte:

$$E[D] = \frac{1}{6X_0Y_0} (2ABr_1 + 2Ay_1r_2 + 2Bx_1r_3 + 2x_1y_1r_4 + A^3 \ln H_1 + B^3 \ln H_2 + x_1^3 \ln H_3 + y_1^3 \ln H_4) \quad (2.16)$$

sendo que  $r_i$  são os valores da distância entre o ponto fixo de serviço e os extremos do retângulo; então  $A$ ,  $B$ ,  $r_i$  e  $H_i$  são dados por:

$$A = X_0 - x_1 \text{ e } B = Y_0 - y_1 \quad (2.17)$$

$$r_1 = \sqrt{A^2 + B^2}, r_2 = \sqrt{A^2 + y_1^2}, r_3 = \sqrt{x_1^2 + B^2}, r_4 = \sqrt{x_1^2 + y_1^2} \quad (2.18)$$

$$H_1 = \frac{B + r_1}{r_2 - y_1}, H_2 = \frac{A + r_1}{r_3 - x_1}, H_3 = \frac{B + r_3}{r_4 - y_1}, H_4 = \frac{A + r_2}{r_4 - x_1} \quad (2.19)$$

Então, os valores esperados da distância média de viagem, dado que o ponto onde se localiza o serviço é fixo, são apresentados no Quadro 2.2, como segue:

	<b>Distância Esperada de Viagem</b>
<b>Serviço Central</b>	$\frac{r_1}{3} + \frac{X_0}{24} \left( \frac{X_0}{Y_0} \ln H_1 + \frac{Y_0^2}{X_0^2} \ln H_2 \right)$
<b>Serviço no Extremo</b>	$\frac{1}{6X_0Y_0} \left( 2X_0Y_0r_1 + X_0^3 \ln \frac{Y_0 + r_1}{X_0} + Y_0^3 \ln \frac{X_0 + r_1}{Y_0} \right)$

**Quadro 2.2 - Distâncias esperadas de viagem para a métrica euclideana.**

Quando a posição do serviço é aleatória, o cálculo da distância esperada de viagem torna-se por demais complexo. Logo, sugere-se que este valor seja encontrado através de simulação, atribuindo vários valores para  $X_0$  e  $Y_0$ .

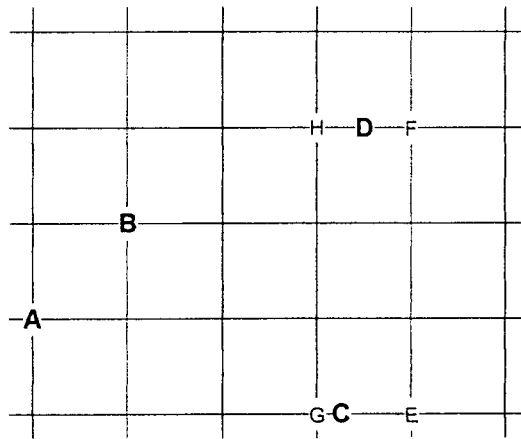
### 2.3. Distância Esperada de Viagem na Grade Viária

Considera-se agora uma grade viária composta por ruas equi-espaciaadas agrupadas em feixes ortogonais entre si. Devido a esta mudança, o caminho a ser percorrido para se deslocar de um ponto a outro da grade ficará restrito às vias disponíveis, e portanto a distância esperada de viagem não poderá ser estimada da maneira como se estava fazendo até então. Far-se-á neste momento um estudo de como é possível estimar tal valor para esta situação específica.

Inicialmente, supõe-se que a região de estudos seja retangular, com o módulo da grade, ou seja, a extensão do quarteirão, igual a  $a$ , tendo com isto as seguintes expressões para as dimensões da região:

$$\begin{aligned} X_0 &= na \\ &e \\ Y_0 &= ma \end{aligned} \tag{2.20}$$

para  $n$  e  $m$  números inteiros. Através da Figura 2.3 pode-se visualizar melhor tal configuração.



**Figura 2.3 - Pontos situados numa grade retangular.**

Pode-se perceber na figura acima que irão existir duas situações a serem consideradas: a primeira é quando deseja-se estimar a distância entre pontos como  $A$  e  $B$ , sendo isto feito de acordo com a equação (2.3); já a segunda é quando os pontos estão situados na mesma faixa vertical, como estão os pontos  $C$  e  $D$ .

Para esta última situação descrita, deve-se analisar os dois caminhos possíveis a serem percorridos. O primeiro passa pelos pontos  $E$  e  $F$ , logo este percurso, chamado de  $P_1$ , terá a seguinte extensão:

$$LP_1 = \overline{CE} + \overline{FD} + \overline{FE} \tag{2.21}$$

O segundo caminho possível para chegar de  $A$  em  $B$  é aquele que passa pelos pontos  $G$  e  $H$ , tendo este percurso  $P_2$  a seguinte extensão:

$$LP_2 = \overline{CG} + \overline{HD} + \overline{GH} \tag{2.22}$$

sendo que  $\overline{FE} = \overline{GH}$ .

Como o intuito neste momento é minimizar as distâncias, deve-se optar por adotar o percurso que tenha a menor extensão. Define-se para tanto a seguinte grandeza:

$$\Delta d_R = \min\{\overline{CE} + \overline{FD}; \overline{CG} + \overline{HD}\} \quad (2.23)$$

e assim, a distância retangular na grade será dada por:

$$d_R = |y_E - y_F| + \Delta d_R \quad (2.24)$$

Analogamente, pode-se encontrar a distância retangular na grade entre dois pontos situados numa mesma faixa horizontal.

Com a suposição de que o ponto onde se localiza um certo serviço e o local que demanda este serviço estejam aleatoriamente posicionados dentro de uma região retangular, sabe-se que a distância média esperada entre estes pontos, de acordo com a métrica retangular, é dada pela equação (2.12), que é a seguinte:

$$E[d_R] = \frac{X_0 + Y_0}{3} \quad (2.25)$$

e substituindo as equações de (2.20) nesta, tem-se então:

$$E[d_R] = \frac{an + am}{3} = \frac{a(n + m)}{3} \quad (2.26)$$

Já a estimativa da distância média de viagem, de acordo com a métrica euclideana é dada por (Novaes, 1993):

$$E[d_E] = \frac{\pi a}{12}(n + m) \quad (2.27)$$

Mas o que se deseja de fato é encontrar uma estimativa da distância considerando que o espaço agora não é mais contínuo e sim uma grade viária. Nesta situação a distância média entre dois pontos, respeitando-se as vias disponíveis e considerando os pontos aleatoriamente posicionados, é dada por (Larson e Odoni, 1981):

$$E[d_G] = \frac{a(n + m)}{3} + \frac{4anm(n + 1)(m + 1)}{3[(n + 1)m + (m + 1)n]^2} \quad (2.28)$$

Para melhor compreender o quanto a distância de viagem cresce quando se considera uma grade viária, supõe-se que esta seja quadrada, isto é, que  $m = n$ . Com isso, a equação (2.28) resume-se a:

$$E[d_G] = \frac{2a(n + m)}{3} + \frac{a}{3} \quad (2.29)$$

ou seja, quando se restringe os caminhos possíveis somente àquelas vias possíveis de se trafegar, a distância esperada de viagem entre dois pontos aleatoriamente posicionados

sobre uma grade viária quadrada fica acrescida de um valor igual a  $\frac{a}{3}$ , ou seja, um terço da extensão do quarteirão.

### 2.3.1. Estimativa do Coeficiente de Correção

Para se estimar o valor da distância média esperada entre dois pontos aleatoriamente posicionados numa grade retangular, calcula-se o valor da distância média através de uma das métricas espaciais, adotando posteriormente um coeficiente de correção adequado.

Supondo que a métrica adotada seja a euclideana, ter-se-á o seguinte:

$$E[d_G] = r_E E[d_E] \quad (2.30)$$

ou seja:

$$r_E = \frac{E[d_G]}{E[d_E]} \quad (2.31)$$

Como visto na sessão anterior,  $E[d_G]$  e  $E[d_E]$  são dados respectivamente pelas equações (2.28) e (2.27), logo a expressão do coeficiente de correção  $r_E$  entre a distância esperada de viagem na grade e a distância euclideana estimada é a seguinte:

$$r_E = \frac{4}{\pi} + \frac{16}{\pi} \left[ \frac{nm(n+1)(m+1)}{(n+m)[(n+1)m + (m+1)n]^2} \right] \quad (2.32)$$

O mesmo acontece quando a métrica adotada é a retangular, sendo que:

$$r_R = \frac{E[d_G]}{E[d_R]} \quad (2.33)$$

e através das equações (2.28) e (2.26) tem-se que:

$$r_R = 1 + \frac{4nm(n+1)(m+1)}{(n+m)[(n+1)m + (m+1)n]^2} \quad (2.34)$$

Cabe aqui ressaltar que os resultados obtidos através destas equações para os coeficientes de correção são apenas aproximações, já que se fez uso de estimativas da distância euclideana e da distância retangular considerando o espaço como sendo o plano contínuo. Além disso, as relações que expressam  $r_E$  e  $r_R$  são satisfatórias somente para valores de  $m$  e  $n$  relativamente próximos, ou seja, quando o coeficiente de esbeltez  $\beta$  se aproxima da unidade, sendo:

$$\beta = \frac{m}{n} \quad (2.35)$$

Novaes (1993) fez uso de simulações, considerando uma grade viária quadrada, para mostrar que o valor teórico de  $r_E$  é uma boa estimativa quando o quociente  $\frac{a}{X_0}$  é pequeno, ou seja, quando o tamanho do quarteirão é pequeno comparado com o comprimento da grade.

Logo, sob estas condições, é possível estimar a distância média esperada de viagem na grade viária através da métrica euclideana ou da métrica retangular, multiplicando esta posteriormente por um coeficiente de correção adequado, tal como aqueles determinados pelas equações (2.32) e (2.34).

Em pesquisa mais recente, Novaes (1996) utiliza uma extensão destes resultados com intuito de definir uma aproximação satisfatória para estimar a mínima distância viajada em problemas similares ao do Caixeiro Viajante. O autor faz uso de SIG (Sistema de Informações Geográficas) para obter a distância real entre os pontos consecutivos do roteiro a ser percorrido e, através de técnicas de regressão que utilizam a métrica euclideana, obtém o coeficiente de correção desejado. Novaes contempla, através de exemplos práticos sobre a cidade de São Paulo, que para regiões relativamente grandes, o valor encontrado na literatura para este coeficiente é bastante satisfatório como aproximação, podendo portanto ser usado para estimar a distância média de viagem numa grade viária.

## 2.4. Perturbações Adicionais: Impedimentos e Barreiras de Viagem

Muitas vezes, no meio urbano, aparecem elementos que dificultam a passagem de veículos ou pedestres, influenciando assim a distância média de viagem dentro da região onde este se encontra. Tais elementos podem ser barreiras de viagem, como parques, praças e rios, ou complicações de trânsito, como retornos, rotatórias e vias de mão única.

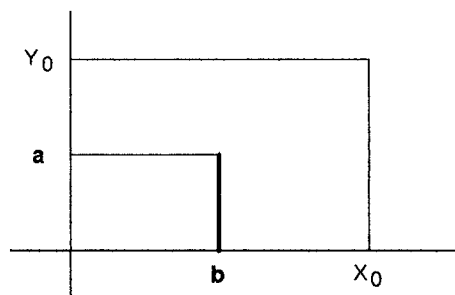
Quando da presença de algum destes elementos, é possível estimar o valor da distância média de viagem primeiro resolvendo o problema simplificado, isto é, ignorando tal impedimento, e posteriormente adicionando o mesmo como sendo uma perturbação do problema original.

Uma aproximação feita pelo método das perturbações adicionais pode ser descrita como segue:

- 1º.) Expressar a variável aleatória de interesse  $D'$  como a soma de uma variável aleatória mais simples  $D$  com uma ou mais variáveis aleatórias de perturbação  $De'$ ,  $De''$ ,  $De'''$ , etc.
- 2º.) Computar o valor esperado de  $D'$  somando o valor esperado da variável original  $D$  com o valor esperado de cada uma das variáveis de perturbação, dado que ela é positiva, multiplicado pela probabilidade de que ela seja positiva:

$$E[D'] = E[D] + E[De' / De' > 0] \cdot P(De' > 0) + E[De'' / De'' > 0] \cdot P(De'' > 0) + \dots \quad (2.36)$$

Para que o entendimento de tal método seja o melhor possível, continuar-se-á com o exemplo apresentado no item 2.2.1, supondo que a tal configuração seja adicionada uma barreira vertical, posicionada em  $x = b$  e de altura  $y = a$ , como mostra a Figura 2.4.

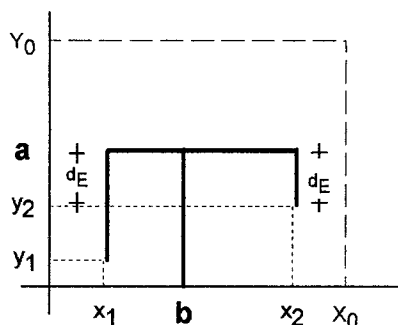


**Figura 2.4 - Barreira de viagem posicionada em  $x = b$  e de altura  $y = a$ .**

Sendo  $(x_1, y_1)$  e  $(x_2, y_2)$  dois pontos aleatoriamente posicionados na região retangular de área  $X_0Y_0$  e fazendo uso da métrica retangular para medir a distância entre ambos, viu-se anteriormente que o valor esperado para a distância viajada no caso simplificado, como dito, sem a barreira de viagem, é dado pela equação (2.12), qual seja:

$$E[D] = \frac{X_0 + Y_0}{3}$$

Resta agora estimar, de acordo com o método das perturbações adicionais, o quanto a barreira de viagem aumenta a distância média de viagem (Figura 2.5). Para tanto serão calculados o valor esperado da distância extra percorrida e a probabilidade desta ser positiva.



**Figura 2.5 - Distância extra percorrida devido à barreira.**

Inicialmente devem ser convenientemente definidos eventos  $A_i$  exaustivos e exclusivos. Com isso, o valor esperado da distância extra  $De'$  percorrida devido a barreira será determinado por:

$$E[De'] = \sum_i E[De' / A_i] P(A_i) \quad (2.37)$$

No exemplo considerado, define-se então os seguintes eventos:



$$\begin{aligned} A_1 &= De > 0 \\ &e \\ A_2 &= De = 0 \end{aligned} \quad (2.38)$$

De fato estes são exaustivos e exclusivos. Como  $E[De/A_2] = 0$ , então a equação (2.37) se resume a:

$$E[De'] = E[De'/A_1]P(A_1) \quad (2.39)$$

Como pode-se notar através da Figura 2.5, os pontos  $(x_1, y_1)$  e  $(x_2, y_2)$  devem estar em lados opostos a barreira e ambas as ordenadas devem ser menores que a altura da barreira. Isto gera duas possibilidades:

$$\begin{aligned} x_1 < b, x_2 > b, y_1 < a \text{ e } y_2 < a \\ \text{ou} \\ x_1 > b, x_2 < b, y_1 < a \text{ e } y_2 < a \end{aligned} \quad (2.40)$$

O evento  $A_1$  pode então ser representado pelas seguintes condições:

$$\begin{aligned} \min(x_1, x_2) < b \\ \max(x_1, x_2) > b \\ y_1 < a \text{ e } y_2 < a \end{aligned} \quad (2.41)$$

Sendo assim, a probabilidade do evento  $A_1$  ocorrer é a seguinte:

$$P(A_1) = P(x_1 \leq b, y_1 \leq a)P(x_2 \geq b, y_2 \leq a) + P(x_2 \leq b, y_2 \leq a)P(x_1 \geq b, y_1 \leq a) \quad (2.42)$$

Através de integração, tem-se então o valor:

$$P(A_1) = \frac{2a^2b(X_0 - b)}{X_0^2 Y_0^2} \quad (2.43)$$

Para o cálculo de  $E[De'/A_1]$ , utiliza-se o Método das Distribuições Derivadas, sendo as variáveis aleatórias definidas por:

$$z_1 = a - y_1 \text{ e } z_2 = a - y_2 \quad (2.44)$$

já que o evento condicional  $De'/A_1$ , como pode-se verificar através da Figura 2.5, é determinado, por:

$$De'/A_1 = 2 \min\{z_1, z_2\} \quad (2.45)$$

Tem-se assim que:

$$E[De' / A_1] = \frac{2a}{3} \quad (2.46)$$

Logo, de acordo com a equação (2.39):

$$E[De'] = \frac{4a^3 b(X_0 - b)}{X_0^2 Y_0^2} \quad (2.47)$$

e portanto o valor esperado da distância média viajada dada a existência de tal barreira, como visto na equação (2.36), é determinado pela soma das equações (2.12) e (2.47), qual seja:

$$E[D'] = \frac{X_0 + Y_0}{3} + \frac{4a^3 b(X_0 - b)}{3X_0^2 Y_0^2} \quad (2.48)$$

Pode-se utilizar o mesmo método para situações distintas. Uma extensão do exemplo anterior seria no caso de ter-se uma barreira estendida completamente de  $y = 0$  até  $y = Y_0$  com um único ponto de passagem em  $y = a$ . Na realidade esta poderia ser a situação onde existe um rio com uma única ponte.

A maneira mais simples de resolver tal problema é adicionar seqüencialmente uma segunda perturbação, dado que o problema com uma única perturbação já esteja resolvido. A distância esperada de viagem será dada por:

$$D'' = D' + De' \quad (2.49)$$

onde  $De'$  denota a distância extra percorrida comparada com  $D'$ . Da mesma maneira como foi feito acima, os resultados obtidos serão então os seguintes:

$$E[De'] = \frac{4(Y_0 - a)^3 b(X_0 - b)}{3X_0^2 Y_0^2} \quad (2.50)$$

logo:

$$E[D''] = \frac{X_0 + Y_0}{3} + \frac{4b(X_0 - b)}{3X_0^2 Y_0^2} [a^3 + (Y_0 - a)^3] \quad (2.51)$$

Seria interessante para o projeto de colocação de uma ponte que servirá como a única passagem sobre um rio, saber em qual ponto dever-se-ia posicionar esta para que a distância viajada fosse a mínima. Para tanto, em geral, deriva-se a equação (2.51) em função de  $a$  e iguala-se tal expressão a zero. Ter-se-á o seguinte resultado:

$$a = \frac{Y_0}{2} \quad (2.52)$$

Logo, o ponto ideal para se posicionar a ponte é exatamente no ponto médio da extensão do rio, como ilustra a Figura 2.6 abaixo:

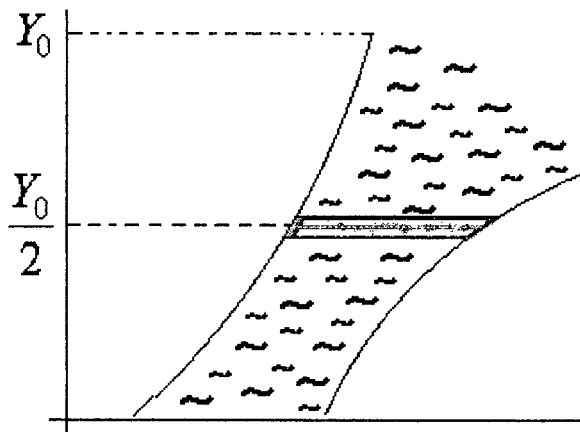


Figura 2.6 - Local onde deve-se construir a ponte.

Para analisar mais detalhadamente os efeitos exercidos pelas barreiras de viagem sobre a estimativa da distância viajada, serão apresentadas neste momento algumas situações específicas onde as mesmas aparecem. Resultados referentes ao estudo destas situações foram apresentados no XI ANPET e no CNMAC 97, como pode ser visto em Siqueira e Gonçalves (1997a e 1997b).

Por se tratarem de situações mais complexas, restringir-se-á a forma da região, devendo esta ser retangular de dimensões  $X_0$  e  $Y_0$ . Da mesma forma, restringir-se-á a métrica a ser utilizada, que será a métrica retangular. Ambas restrições foram adotadas para facilitar o desenvolvimento matemático do modelo a ser utilizado na resolução de tais problemas, qual seja, o método das perturbações adicionais, este descrito no início desta sessão.

O ponto  $(x_1, y_1)$  onde se localiza o serviço urbano que se está planejando poderá estar fixo, no centro ou em algum extremo da região, ou ainda estar em uma posição aleatória, enquanto que o ponto  $(x_2, y_2)$  que demanda o serviço estará sempre aleatoriamente posicionado dentro da região de estudo.

Serão apresentadas, separadamente, situações onde encontram-se barreiras de viagem verticais e horizontais. Após, apresentar-se-á um caso específico onde exista uma barreira de viagem inclinada.

#### 2.4.1. Barreiras de Viagem Verticais

Ter-se-á dois tipos de barreiras de viagem verticais a se considerar. A primeira delas caracteriza-se por estar localizada em  $x = b$  e ter extensão  $y = a$ . A segunda caracteriza-se também por estar localizada em  $x = b$ , mas tem extensão  $Y_0$  e um único ponto de passagem em  $y = a$ . Estas estão representadas por (a) e (b) respectivamente na Figura 2.7.

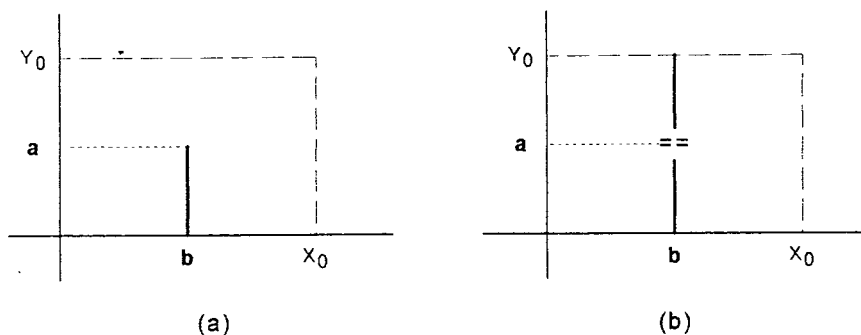


Figura 2.7 - Tipos de barreiras de viagem verticais.

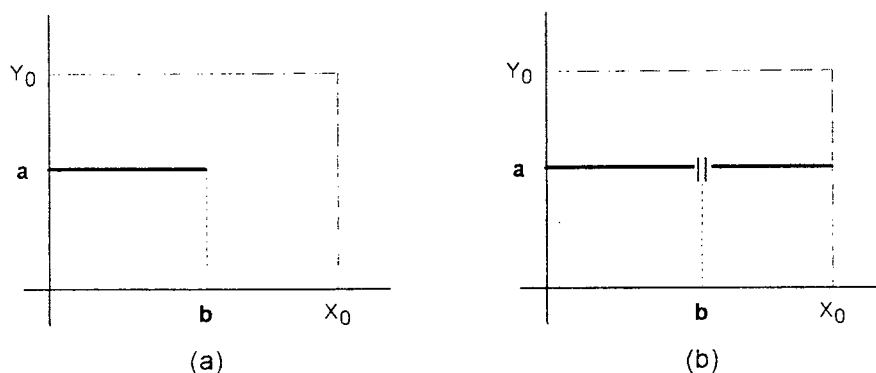
Como dito, o valor da distância esperada de viagem para cada um dos casos em particular foi estimado tal como feito no exemplo da sessão 2.4. Estes valores dependerão somente das dimensões da região e da localização e extensão da barreira de viagem; salvo o caso em que o serviço, representado pelo ponto  $(x_l, y_l)$ , estiver localizado no centro da região, em que ainda deve-se informar se a barreira está localizada antes ( $b < x_l$ ) ou depois ( $b > x_l$ ) deste. Os resultados desta estimativa podem ser verificados no quadro abaixo.

	Serviço Central	Serviço no Extremo	Serviço Aleatório
Barreira tipo (a)	$\underline{b < x_l}: \frac{2a^2 b}{3X_0 Y_0}$ $\underline{b > x_l}: \frac{2a^2 (X_0 - b)}{3X_0 Y_0}$	$\frac{a^2 (X_0 - b)}{X_0 Y_0}$	$\frac{4a^3 b (X_0 - b)}{3X_0^2 Y_0^2}$
Barreira tipo (b)	$\underline{b < x_l}: \frac{2b}{3X_0 Y_0} [a^2 + (Y_0 - a)^2]$ $\underline{b > x_l}: \frac{2(X_0 - b)}{3X_0 Y_0} [a^2 + (Y_0 - a)^2]$	$\frac{a^2 (X_0 - b)}{X_0 Y_0}$	$\frac{4b(X_0 - b)}{3X_0^2 Y_0^2} [a^3 + (Y_0 - a)^3]$

Quadro 2.3 - Distância esperada de viagem dada uma barreira de viagem vertical.

#### 2.4.2. Barreiras de Viagem Horizontais

Analogamente à situação anterior, haverá dois tipos de barreiras de viagem horizontais, estes representados por (a) e (b) na Figura 2.8. No primeiro a barreira se localiza em  $y = a$  e tem extensão  $x = b$ . No segundo a barreira está localizada em  $y = a$ , tem extensão  $X_0$  e um único ponto de passagem em  $x = b$ .



**Figura 2.8 - Tipos de barreiras de viagem horizontais.**

Os valores da distância esperada de viagem nestes casos serão praticamente os mesmos apresentados para barreiras de viagem verticais, salvo uma rotação nos eixos cartesianos, invertendo então os valores de  $a$  para  $b$ , de  $X_0$  para  $Y_0$  e vice versa. Tais valores são apresentados no quadro seguinte.

	<b>Serviço Central</b>	<b>Serviço no Extremo</b>	<b>Serviço Aleatório</b>
<b>Barreira tipo (a)</b>	$\underline{a < y_L}: \frac{2b^2a}{3X_0Y_0}$ $\underline{a > y_L}: \frac{2b^2(Y_0 - a)}{3X_0Y_0}$	$\frac{b^2(Y_0 - a)}{X_0Y_0}$	$\frac{4b^3a(Y_0 - a)}{3X_0^2Y_0^2}$
<b>Barreira tipo (b)</b>	$\underline{a < y_L}: \frac{2a}{3X_0Y_0} [b^2 + (X_0 - b)^2]$ $\underline{a > y_L}: \frac{2(Y_0 - a)}{3X_0Y_0} [b^2 - (X_0 - b)^2]$	$\frac{b^2(Y_0 - a)}{X_0Y_0}$	$\frac{4a(Y_0 - a)}{3X_0^2Y_0^2} [b^3 + (X_0 - b)^3]$

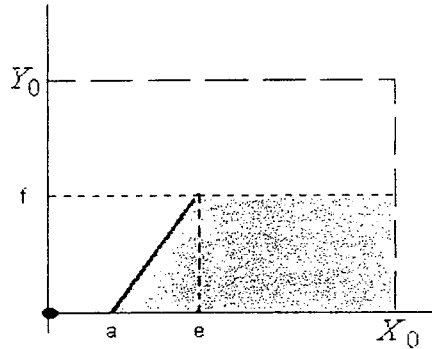
**Quadro 2.4 - Distância esperada de viagem dada uma barreira de viagem horizontal.**

### 2.4.3. Um caso a parte: Barreira de Viagem Inclinada

Considerar-se-á, neste caso específico, a unidade de serviço localizada no extremo  $(0,0)$  da região de estudo. Tal como nas situações analisadas anteriormente, o ponto de demanda, representado por  $(x_2, y_2)$  está aleatoriamente distribuído na região e a métrica adotada para medir distâncias é a métrica retangular. A barreira de viagem será caracterizada por seu ponto inicial e ponto final, estes representados respectivamente por  $(a, b)$  e  $(e, f)$ .

Como a unidade de serviço está localizada no extremo, para que a barreira de viagem exerça algum efeito sobre a distância percorrida entre o ponto de demanda e a unidade de serviço, deve-se ter a coordenada  $a$  ou a coordenada  $b$  igual a zero.

Supondo que  $b$  seja igual a zero, uma configuração possível neste caso está representada na Figura 2.9:



**Figura 2.9 – Barreira de viagem inclinada.**

onde  $a < e$  e  $b < f$ .

A área sombreada representa os pontos que serão afetados pela barreira, isto é, aqueles que terão aumentada a distância para chegar até a unidade de serviço.

Para aplicar o Método das Perturbações Adicionais como descrito na sessão 2.4, subdivide-se a região sombreada em duas sub-regiões: o triângulo de vértices  $(a, 0)$ ,  $(e, f)$  e  $(e, 0)$  e o retângulo de vértices  $(e, 0)$ ,  $(e, f)$ ,  $(X_0, f)$  e  $(X_0, 0)$ . Feita esta divisão, devem ser calculados, para cada sub-região em separado, o valor esperado da distância extra percorrida e a probabilidade desta distância extra ser positiva.

Para a aplicação do referido método, devem ser definidos os seguintes eventos:

$$\begin{aligned} A_1 &= D_e' > 0 \text{ e } (x_2, y_2) \text{ está no triângulo} \\ &e \\ A_2 &= D_e' > 0 \text{ e } (x_2, y_2) \text{ está no retângulo.} \end{aligned} \quad (2.53)$$

Por probabilidade geométrica, verifica-se facilmente que as probabilidades destes eventos ocorrerem, isto é, das distâncias extras percorridas serem positivas para cada sub-região em particular, são dadas por:

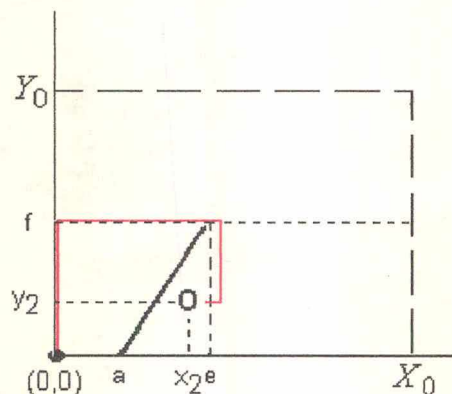
$$P(A_1) = \frac{(e-a)f}{2X_0Y_0} \text{ e } P(A_2) = \frac{(X_0-e)f}{X_0Y_0} \quad (2.54)$$

Para o cálculo de  $E[D_e' | A_1]$  e de  $E[D_e' | A_2]$ , utiliza-se o Método das Distribuições Derivadas, descrito na sessão 2.2.1. Através da Figura 2.10 pode-se verificar que o evento condicional  $D_e' | A_1$  é determinado por:

$$D_e' | A_1 = 2(v+w) \quad (2.55)$$

onde  $v$  e  $w$  são variáveis aleatórias que satisfazem às seguintes condições:

$$v = f - y_2 \quad \text{e} \quad w = e - x_2 \quad (2.56)$$

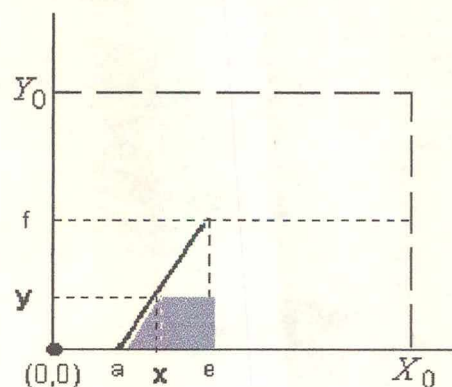


**Figura 2.10 - Distância percorrida quando  $(x_2, y_2)$  está no triângulo.**

A variável aleatória  $w$  está uniformemente distribuída no intervalo  $[0, e-a]$  e com isso:

$$E[w] = \int_0^{e-a} wf(w)dw = \frac{1}{e-a} \int_0^{e-a} wdw = \frac{e-a}{2} \quad (2.57)$$

Já a variável aleatória  $v$  pertence ao intervalo  $[0, f]$  mas não está uniformemente distribuída neste. Através da Figura 2.11 pode-se constatar que se o ponto de demanda  $(x_2, y_2)$  está no triângulo, a probabilidade deste ponto estar na base do triângulo é bem maior.



**Figura 2.11 - Função distribuição acumulada por probabilidade geométrica.**

Então, por probabilidade geométrica, tem-se que:

$$F(y) = P(y_2 \leq y) = \frac{\text{Área do trapézio sombreado}}{\text{Área do triângulo}}, \quad \text{para } 0 \leq y \leq f.$$

$$F(y) = \frac{\left\{ \frac{(x-a)y}{2} + (e-x)y \right\}}{\frac{(e-a)f}{2}} \quad \text{para } 0 \leq y \leq f \quad (2.58)$$

e por semelhança de triângulos, tem-se o seguinte:

$$x = \frac{(e-a)y}{f} + a \quad (2.59)$$

Substituindo (2.59) em (2.58) e feitas as devidas simplificações, a função distribuição acumulada fica a seguinte:

$$F(y) = \frac{2y}{f} - \frac{y^2}{f^2} \quad \text{para } 0 \leq y \leq f \quad (2.60)$$

Derivando a expressão acima em relação à  $y$ , tem-se que:

$$f(y) = \frac{2}{f} - \frac{2y}{f^2} \quad \text{para } 0 \leq y \leq f. \quad (2.61)$$

e integrando esta no intervalo  $[0, f]$ , tem-se a seguinte expressão para a distância esperada:

$$E[y_2] = \int_0^f y \left( \frac{2}{f} - \frac{2y}{f^2} \right) dy = \frac{f}{3} \quad (2.62)$$

e com isso:

$$E[v] = f - E[y_2] = f - \frac{f}{3} = \frac{2f}{3} \quad (2.63)$$

Assim:

$$E[D_e' | A_1] = 2\{E[w] + e[v]\} = (e-a) + \frac{4f}{3} \quad (2.64)$$

Da mesma forma, verifica-se através da Figura 2.12 que o evento condicional  $D_e' | A_2$  é determinado por:

$$D_e' | A_2 = 2z \quad (2.65)$$

onde  $z$  é uma variável aleatória uniformemente distribuída em  $[0, f]$  e satisfaz a seguinte condição:



$$z = f - y_2 \quad (2.66)$$

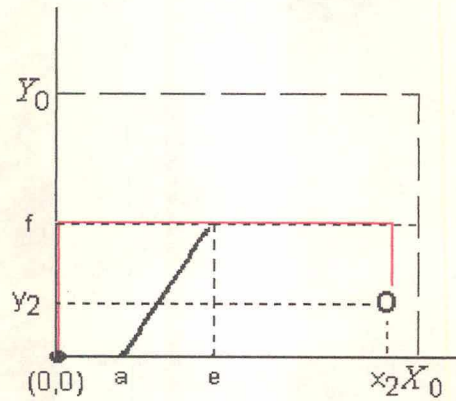


Figura 2.12 - Distância percorrida quando  $(x_2, y_2)$  está no retângulo.

Deste modo:

$$E[z] = \frac{1}{f} \int_0^f z dz = \frac{f}{2} \quad (2.67)$$

e com isto:

$$E[D'_e | A_2] = f \quad (2.68)$$

De acordo com a equação (2.37) :

$$\begin{aligned} E[D'_e] &= (e - a + \frac{4f}{3}) \frac{(e - a)f}{2X_0Y_0} + f \frac{(X_0 - e)f}{X_0Y_0} \\ &= \frac{f}{6X_0Y_0} \{3(e - a)^2 + 2f(3X_0 - e - 2a)\} \end{aligned} \quad (2.69)$$

e com isso:

$$E[D'] = \frac{X_0 + Y_0}{2} + \frac{f}{6X_0Y_0} \{3(e - a)^2 + 2f(3X_0 - e - 2a)\} \quad (2.70)$$

De acordo com este resultado, pode-se concluir que a barreira de viagem vertical é um caso específico de barreira de viagem inclinada. Para isto basta fazer  $a = e$ , e com isso a equação (2.69) ficará assim reduzida :

$$E[D'_e] = \frac{f^2(X_0 - a)}{X_0 Y_0} \quad (2.71)$$

que é de fato o valor esperado da distância extra percorrida quando da existência de uma barreira vertical de comprimento  $f$  localizada em  $x = a$ , como foi visto na sessão 2.4.1.

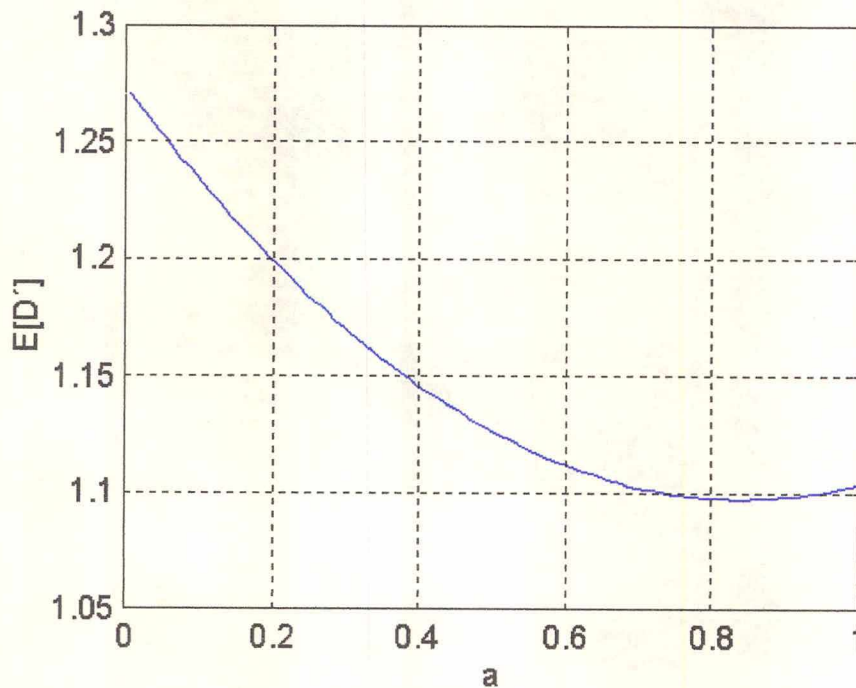
Para melhor avaliar o efeito da barreira de viagem inclinada sobre a distância a ser percorrida entre a unidade de serviço e o ponto de demanda, supõe-se que a região de estudo seja um quadrado unitário, onde o ponto final da barreira está localizado exatamente no centro da região, isto é :

$$X_0 = Y_0 = 1 \quad \text{e} \quad e = f = \frac{1}{2} \quad (2.72)$$

Deste modo a equação (2.70) reduzir-se-á ao seguinte:

$$E[D'] = \frac{a^2}{4} - \frac{5a}{12} + \frac{61}{48} \quad (2.73)$$

A Figura 2.13 mostra esta função plotada para valores da coordenada  $a$  variando no intervalo  $[0, 1]$ . Isso vai poder indicar em que local a barreira exerce o maior acréscimo na distância esperada de viagem.



**Figura 2.13 – Distância esperada de viagem em uma região unitária, dada a existência de uma barreira de viagem inclinada**

Pode-se verificar que quando  $a = \frac{1}{2}$ , tem-se uma barreira vertical onde  $E[D'] = 1,125$ , ou seja, há um acréscimo de 12,5% na distância média percorrida sem considerar nenhuma barreira de viagem.

O ponto onde o aumento na distância média percorrida é mínimo se dá quando  $a = \frac{5}{6}$ , sendo este acréscimo algo em torno de 10%. Já o acréscimo máximo a esta distância ocorre quando  $a \rightarrow 0$  ( $a$  muito próximo de zero), sendo este acréscimo algo em torno de 27%.

As demais situações onde encontra-se barreiras de viagem inclinadas, dado que a unidade de serviço está no extremo da região, são resolvidas analogamente. Os resultados obtidos para a distância extra percorrida nestes casos são apresentados no quadro abaixo:

Posição das coordenadas da barreira	Valor de $E[De]$
$a < e, b < f$ com $b = 0$	$\frac{f}{6X_0Y_0} \{3(e-a)^2 + 2f(3X_0 - e - 2a)\}$
$a > e, b < f$ com $b = 0$	$\frac{f^2}{2X_0Y_0} (2X_0 - a - e)$
$a < e, b < f$ com $a = 0$	$\frac{e}{6X_0Y_0} \{3(f-b)^2 + 2e(3Y_0 - b - 2f)\}$
$a < e, b > f$ com $a = 0$	$\frac{e^2}{2X_0Y_0} (2Y_0 - f - b)$

**Quadro 2.5 - Distância extra percorrida considerando a existência de uma barreira de viagem inclinada e o serviço localizado no extremo**

## 2.5. Estimativa da Distância Esperada: Coeficientes de Correção

Para encontrar as dimensões da região  $R$  que levam à mínima distância esperada de viagem, basta minimizar o valor esperado da distância média de viagem,  $E[D]$ , mantendo a área do distrito constante. Através do exemplo que se está considerando (região retangular de área  $A_0 = X_0Y_0$  com ambos os pontos aleatórios), pode-se ver que este problema seria o seguinte:

$$\text{Minimizar } E[D] = \frac{(X_0 + Y_0)}{3} \tag{2.74}$$

$$\text{Sujeito a } Y_0 = \frac{A_0}{X_0}$$

Utilizando o Método de Lagrange (Larson e Odoni, 1981), facilmente se descobre que  $E[D]$  é mínima quando a forma da zona se resume a um quadrado, ou seja, quando:

$$X_0 = Y_0 = \sqrt{A_0} \quad (2.75)$$

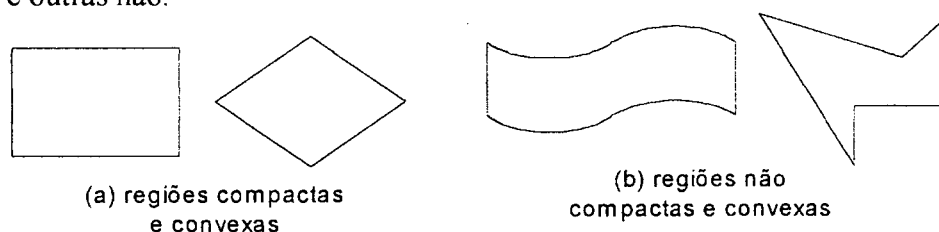
e conseqüentemente:

$$E[D] = \frac{2\sqrt{A_0}}{3} \quad (2.76)$$

Uma análise de sensibilidade mostra que este valor é insensível a pequenas mudanças quanto à geometria da área. Portanto, poder-se-á encontrar estimativas bastante satisfatórias para regiões de qualquer forma, tais que:

- (i) uma das dimensões não seja muito maior do que a outra;
- (ii) não existam barreiras ou impedimentos muito significativos na região.

Regiões que satisfazem ambas as condições acima são ditas aproximadamente compactas e convexas (Larson e Odoni, 1981). A Figura 2.14 apresenta regiões compactas e convexas e outras não.



**Figura 2.14 - Representação de regiões compactas e convexas e outras não.**

Para regiões aproximadamente compactas e convexas com pontos de demanda independentes e uniformemente distribuídos no espaço, tem-se:

$$E[D] = K \cdot \sqrt{A_0} \quad (2.77)$$

onde  $K$  é dito ser um coeficiente de correção. Alguns valores de  $K$  são apresentados a seguir de acordo com a localização da unidade de serviço, a métrica adotada e a forma da região:

		Quadrado	Losango	Círculo
Unidade de Serviço Aleatória	Métrica euclideana	$0.52 = \pi/6$	0.52	$\cong 0.511$
	Métrica retangular	$2/3 \cong 0.667$	$\cong 0.660$	$\cong 0.650$
Unidade de Serviço Central	Métrica euclideana	$\cong 0.383$	$\cong 0.383$	$\cong 0.376$
	Métrica retangular	0.50	$\cong 0.471$	$\cong 0.479$

**Quadro 2.6 - Coeficientes de correção para alguns casos específicos.**

Larson e Odoni (1981) argumentam que, nas condições descritas acima, tal coeficiente depende somente da métrica em uso e do local onde está a unidade de serviço. Neste caso, os autores optam por adotar o maior valor de  $K$  listado em cada linha das três colunas do Quadro 2.5, como segue:

		Regiões Compactas e Convexas
Unidade de Serviço Aleatória	Métrica euclideana	0.52
	Métrica retangular	0.67
Unidade de Serviço Central	Métrica euclideana	0.38
	Métrica retangular	0.50

**Quadro 2.7 - Coeficientes de correção para regiões compactas e convexas**

Uma estimativa mais precisa para os valores da distância esperada de viagem pode ser obtida considerando dois coeficientes de correção adicionais, tendo com isso uma expressão do seguinte tipo:

$$E[D] = k_1 k_2 \sqrt{A_0} \quad (2.78)$$

onde  $k_1$  é o coeficiente de correção obtido através do Quadro 2.6 e  $k_2$  é uma constante que incorpora correções viárias, ou seja, é o coeficiente de correção, este descrito no item 2.3.1 do presente capítulo.

Como pôde-se verificar neste capítulo, há diversas considerações a serem feitas pelo planejador antes de estimar certos valores fundamentais para o planejamento inicial de um serviço urbano. Para auxiliá-lo então, propõe-se o desenvolvimento de uma ferramenta computacional, no caso um protótipo de Sistema Especialista, que, a partir de algumas informações fornecidas pelo planejador, indique boas estimativas de tais valores fundamentais. Para tanto, apresentar-se-á os conceitos básicos de Inteligência Artificial e Sistemas Especialistas, bem como o ambiente no qual este será desenvolvido, qual seja, a shell KAPPA.

## *Inteligência Artificial e Sistemas Especialistas*

Neste capítulo serão apresentados os conceitos básicos de Inteligência Artificial e de Sistemas Especialistas. O texto a seguir é baseado principalmente nos seguintes autores: Rabuske (1995), Rich (1988) e Waterman (1987).

### **3.1. A Inteligência Artificial**

Por ser um ramo recente da ciência da computação, a Inteligência Artificial (IA) não tem uma definição universalmente aceita. Ela tem sido definida como um “Ramo da ciência que se preocupa com a teoria e o desenvolvimento de programas de computador ou sistemas que podem realizar tarefas que são normalmente associadas com a inteligência humana”, definição esta retirada da Nova Enciclopédia Ilustrada Folha (1996). Pesquisadores desta área vêm tentando durante anos conceituar tal ciência. Marvin Minsky (ver Rabuske (1995)), por exemplo, apresentou uma definição de Inteligência Artificial tão abrangente que engloba todas as atividades computacionais, qual seja, “É a ciência de fazer com que as máquinas façam coisas que requereriam inteligência se feitas pelos homens”. Outros como Barr e Feigebaum (1981) e Rich (1988) definem a IA respectivamente por “A parte da ciência da computação relacionada com o projeto de sistemas que exibem características que associamos com a inteligência humana” e “O estudo de como fazer os computadores realizarem tarefas que, no momento, as pessoas são melhores”.

O problema do emprego do termo Inteligência Artificial se dá quanto ao significado da palavra inteligência. O dicionário Folha Aurélio (1995) define esta como sendo a “Faculdade de aprender, apreender ou compreender; qualidade ou capacidade de compreender e adaptar-se facilmente; maneira de entender ou interpretar”. Já a definição tradicional dada por psicólogos - “Inteligência é o que é medido pelos testes de inteligência” - é totalmente inútil na área de inteligência computacional. Se interpretada dessa maneira, toda a pesquisa de IA se resumiria em encontrar maneiras de responder às perguntas feitas por tais testes de inteligência. Simons (1984) apresenta definições de inteligência dadas por vários outros autores, a maioria fazendo referência ao raciocínio. Alguns exemplos são os que seguem:

“Inteligência é julgar bem, compreender bem, raciocinar bem” - Binet;

“A atividade inteligente consiste na compreensão do essencial de uma situação e numa resposta reflexa apropriada” - Helm;

“Adaptação ao ambiente físico e social” - Piaget.

Outro problema quanto a nomenclatura Inteligência Artificial foi abordado por Simon (1981) em seu livro “As Ciências do Artificial”. Neste o autor disserta sobre o significado do termo artificial, que, de acordo com o mesmo, deve ser entendido como algo ‘fabricado pelo homem’. Quanto a Inteligência Artificial o autor faz o seguinte comentário: “O meu próprio grupo de pesquisa ... tem preferido frases como ‘processamento complexo de informação’ e ‘simulação de processos cognitivos’. Mas então entramos em novas dificuldades terminológicas, porque o dicionário também diz que ‘simular’ significa ‘assumir ou ter mera aparência de; imitar’. De qualquer forma, ‘inteligência artificial’ parece ter vindo para ficar, e pode-se revelar mais fácil depurar a frase do que passar sem ela.”. De acordo com o autor, deve-se então adotar o termo sem alguma preocupação adicional com a morfologia do mesmo.

Um programa inteligente é aquele que, no decorrer do processo para alcançar seu objetivo, toma decisões de maneira semelhante às da inteligência humana. A Inteligência Artificial realmente permite que o computador “pense”. Simplificando como são feitas estas programações, a IA imita o processo básico do aprendizado humano, onde novas informações são absorvidas e ficam assim disponíveis para uso futuro. A mente humana é capaz de incorporar novos conhecimentos sem alterar outros fatos já armazenados no cérebro. Um programa de IA funciona quase do mesmo modo, ou seja, ao contrário das programações convencionais, as mudanças feitas em tais programas são mais simples de implementar e podem ser feitas a qualquer momento sem alterar o funcionamento dos mesmos.

A Inteligência Artificial tem como objetivo principal construir sistemas mais úteis, que auxiliem os homens a tomar decisões mais rapidamente e com entendimento inteligente, onde a comunicação homem - máquina possa ser feita em linguagem natural, com o computador aprendendo e utilizando a linguagem do homem e não o inverso. Estes sistemas podem ser classificados como sendo cooperativos, ou seja, que necessitam da interação humana para encontrar alguma solução; ou autônomos, quais sejam, aqueles em que o computador irá operar sozinho e apenas fornecerá o resultado final ao usuário.

Pode-se dizer que uma das características mais marcantes da IA é que ela trata de processos simbólicos e não numéricos. Por tratar de coisas abstratas, estes processos podem envolver questões muito complexas e ambíguas, onde o computador deve “parar e pensar”. Para tais processos simbólicos geralmente não encontram-se soluções algorítmicas, sendo então necessária a utilização de um processo de busca por tentativa e erro, conhecido por heurística. Heurísticas são simplificações que efetivamente limitam a busca pelas soluções. Um algoritmo garante uma solução correta para o problema, enquanto um método heurístico produz, na maioria das vezes, uma solução aceitável num espaço de tempo razoável.

Outra característica importante da Inteligência Artificial é a utilização de inferência e deduções a partir das informações disponíveis e a aplicação destes conhecimentos, armazenados e induzidos, na resolução de problemas. Além disso, a utilização destes

conhecimentos em forma de regras associadas, a fim de limitar o crescimento exponencial de possibilidades de caminhos a seguir, que acontece em situações reais, é também um fato que visivelmente diferencia a IA da programação convencional.

A Inteligência Artificial tem tudo para ser a tecnologia mais promissora a ser utilizada no próximo século, já que diversos países estão investindo grandes somas em pesquisas nesta área. Em nível de Brasil, somente há alguns anos é que foram organizados encontros, seminários e congressos para que fossem conhecidas as pesquisas aqui desenvolvidas (ou em desenvolvimento) na área de Inteligência Artificial. Estes encontros tem tudo para estimular as pesquisas na área e com isso levar o país a ter uma evolução computacional no nível da dos países de primeiro mundo.

### **3.2. A Inteligência Artificial na História**

A Inteligência Artificial pode parecer um ramo recente da ciência dos computadores, mas na verdade, durante toda a história da humanidade, aparecem vários exemplos de antecessores da IA. Em Esparta, por volta de 200 a.C., acreditava-se que o ditador Nobis dispunha de um robô para obrigar os cidadãos devedores a pagarem seus impostos. Já no século XIII, há relatos de que Alberto Magno havia fabricado um escravo animado de tamanho natural capaz de falar e abrir a porta a visitantes. Estes entes com certeza seriam considerados sistemas inteligentes se na época em que foram desenvolvidos a idéia de IA já tivesse sido proliferada.

A história da IA está em muito correlacionada com a história da ciência dos computadores, que iniciou com Charles Babbage, nascido em 1792 e frequentemente considerado o pai da computação. Durante o século XIX alguns projetos foram sendo criados, como o tabulador concebido em 1890 para auxiliar na análise do censo americano e a primeira máquina de somar a ser comercializada apresentada em 1892.

Mas foi somente na década de quarenta, com o advento das válvulas termoiônicas, que o desenvolvimento da ciência dos computadores pôde ser acelerado. Nesta época o mundo estava em guerra, e complexos cálculos tinham que ser feitos sob pressão de tempo e com máxima precisão possível, para que fossem criadas poderosas armas ou para que fossem descobertos códigos secretos do lado inimigo. Nesta época foi desenvolvido o ENIAC (*Electronic Numerical Integrator and Calculator*), o primeiro computador que utilizava válvulas (dezoito mil destas) e que ocupava uma sala com dezoito por sete metros e sessenta centímetros. Este tinha por objetivo computar tabelas balísticas para armas de fogo e mísseis.

Até este momento nada se conhecia por Inteligência Artificial. Em 1947 Alan Turing, um dos nomes mais expressivos referindo-se ao surgimento desta ciência, publicou um artigo profético sobre a IA, onde afirmava que uma ferramenta para a construção de sistemas computacionais, a ACE (*Automatic Computing Engine*) seria capaz de modelar as ações executadas pelo cérebro humano. Ele desenvolveu o que hoje é conhecido como o Teste de Turing, um teste para avaliar se uma máquina pode ou não ser considerada inteligente.



Os anos cinquenta são geralmente considerados os anos em que a IA se concretizou. Oficialmente este termo foi cunhado em 56 em uma conferência de verão no Dartmouth College em Hanover, reunindo pesquisadores como John McCarthy, Marvin Minsky, Nataniel Rochester, Allen Newell, Herbert Simon e Claude Shannon. Estes elaboraram um documento que foi submetido à fundação Rockefeller, onde consta a intenção dos participantes de realizar “um estudo durante dois meses, por dez homens, sobre o tópico de Inteligência Artificial”. Tudo indica que esta foi a primeira menção oficial ao termo IA. Newell e Simon apresentaram nesta conferência o *Logic Theorist*, uma das manifestações mais antigas de sistema inteligente e considerado o primeiro programa efetivo de IA. Este programa, com o objetivo de provar teoremas, foi o primeiro capaz de processar símbolos ao invés de quantidades numéricas. Em 1958, McCarthy define as funções básicas do LISP (*list processing*), uma das linguagens preferidas de programação em I.A.

Um dos pensamentos fundamentais na década de sessenta era a crença de que um único programa poderia solucionar todo o tipo de problemas, desde que este possuísse as capacidades de inferências suficientemente poderosas para tal. O GPS (*General Problem Solver*) é um exemplo destes programas que foi utilizado para resolver problemas de lógica elementar e álgebra, bem como para responder perguntas e jogar xadrez. Sistemas gerais como este tiveram certo êxito, mas apenas em questões muito específicas, pois o objetivo nestes casos era o de demonstrar a técnica que foi utilizada em tal sistema e não o de demonstrar a solução encontrada para um certo problema real. No final desta década os pesquisadores de IA concluíram que a abordagem generalizada não estava levando a resultados satisfatórios e com isso os problemas passaram a ser encarados separadamente, o que se revelou uma estratégia correta.

Durante a década de setenta, com o advento de uma minúscula partícula de sílica com milhares de transistores operacionais chamada de *chip*, houve uma importante mudança no advento dos computadores. Estes diminuíram de tamanho, tornaram-se mais potentes e também ficaram mais viáveis em termos de utilização.

Nos meados desta década deu-se ênfase na engenharia do conhecimento, pois ficou claro para a comunidade científica que as soluções heurísticas, as estratégias de busca e as representações do conhecimento limitados eram inadequadas quando da solução de problemas reais. Os projetos que visavam a estruturação de sistemas baseados em conhecimento foram largamente desenvolvidos nesta época. Tais sistemas, conhecidos como Sistemas Especialistas, têm por objetivo resolver problemas cuja solução dependa de certos conhecimentos especializados, estes frequentemente simulados de acordo com o saber de um especialista humano. Feigenbaum, em Stanford, desenvolveu o primeiro destes sistemas, o DENDRAL, utilizado para análise de informações espectrográficas. Em Arseille, Colmerauer implementou o Sistema Q, posteriormente denominado PROLOG, uma linguagem de programação em IA das mais utilizadas até hoje.

As expectativas quanto aos Sistemas Especialistas foram enormes e, como geralmente ocorre, estas não foram atendidas. Isso porque os empresários, entusiasmados com o sucesso destes, estavam interessados em comercializar tais produtos, mas estes sistemas não eram de fato um produto, visto que, na visão do empresário, um produto não deve servir somente para situações específicas. Para solucionar o problema de construção de Sistemas Especialistas

foram lançadas no mercado ferramentas para tal, as chamadas ASE (Arcabouços de Sistemas Especialistas). Mesmo estas não serviram totalmente para o fim planejado, pois muitos dos usuários continuavam sem saber o que programar para que fosse possível a construção de tais sistemas, e esse, sabe-se, é o ponto mais importante a ser considerado. Atualmente uma variedade enorme de Sistemas Especialistas está sendo desenvolvida e muitos outros já estão sendo utilizados, e nos mais diversos campos de conhecimento. Outras especificidades quanto a estes sistemas serão abordadas ainda neste texto.

A questão na qual foi dada maior ênfase a partir dos anos oitenta foi a da aprendizagem do computador, ou seja, como fazer com que a máquina adquira conhecimento de acordo com as inferências feitas sobre a base de conhecimentos interna e as respostas dadas pelo usuário do sistema.

Com relação ao futuro da Inteligência Artificial nada pode-se afirmar, mas, como dito anteriormente, este tem tudo para ser dos mais promissores, pois deverão surgir então os resultados tecnológicos de todas as pesquisas feitas no passado.

### **3.3. Principais Áreas de Aplicação da Inteligência Artificial**

Neste texto serão destacadas, não desmerecendo outras, as áreas de aplicação da Inteligência Artificial que estão sendo exaustivamente pesquisadas, que conseqüentemente são as que apresentam maiores aplicações em situações reais. Estas são:

- Sistemas Especialistas;
- Processamento de Linguagem Natural;
- Reconhecimento de Padrões e Percepção;
- Robótica;
- Prova de Teoremas;
- Jogos.

Cada uma destas apresenta aplicações específicas, mas muitas vezes podem ocorrer relações entre estas como, por exemplo, no desenvolvimento de um sistema especialista capaz de reconhecer figuras geométricas (padrões).

### **3.3.1. Sistemas Especialistas**

Esta é a área de maior pesquisa atualmente da Inteligência Artificial. Esta tem aplicações em muitas áreas de conhecimento. Sendo um dos objetivos do presente trabalho a construção de um protótipo de Sistema Especialista, este tópico será a seguir devidamente detalhado para que o entendimento do mesmo seja o melhor possível.

### **3.3.2. Processamento de Linguagem Natural**

Os computadores não serão capazes de realizar certas tarefas do cotidiano das pessoas enquanto não forem capacitados para a utilização da linguagem normal de comunicação entre as pessoas. Para tanto então, deve-se desenvolver meios de fazer com que a interação homem - máquina se dê através de linguagem natural ao invés de linguagem computacional, visando uma interação mais natural possível entre ambos e permitindo assim o acesso de mais pessoas às ferramentas computacionais.

O processamento computacional da linguagem natural é, sem dúvida alguma, o campo mais árduo de pesquisas da Inteligência Artificial, pois para que a compreensão de língua seja possível, deve-se ter armazenado tanto o conhecimento lingüístico desta quanto o conhecimento associado ao assunto que estiver sendo tratado. Para a elaboração de um programa eficaz de compreensão da linguagem, é preciso ainda estudar como ambos os conhecimentos, o lingüístico e o do mundo, se combinam, evitando assim expressões mal entendidas.

Rich sugere a divisão do problema da compreensão da linguagem em duas tarefas, quais sejam:

- Compreensão da redação, utilizando o conhecimento léxico, sintático e semântico da linguagem, e informações necessárias sobre o mundo;
- Compreensão da fala, utilizando todo o conhecimento descrito acima, mais o conhecimento sobre fonoaudiologia e sobre ambigüidades da fala.

O principal ponto que torna dificultosa tal aplicação é quanto à linguagem figurada que é muito utilizada numa conversação, ou quanto à dupla interpretação que pode ser dada a uma frase específica. Outra questão crítica é quanto à forma que se fala, fazendo uso de diferentes sotaques e emendando as palavras umas às outras. Em ambos os casos o computador não possui informações suficientes, logo não consegue interpretar tal expressão, muitas vezes tornando a frase sem sentido e conseqüentemente comprometendo todo o trabalho feito até então.

### **3.3.3. Reconhecimento de Padrões e Percepção**

Os seres humanos fazem uso dos cinco sentidos para perceberem o mundo em sua volta. A Inteligência Artificial estuda dois desses, quais sejam, a visão e audição, esta última de suma importância na compreensão da linguagem natural. A maneira de como o

computador irá 'ver' o mundo em sua volta dependerá dos padrões que ele tem armazenado e com os quais ele poderá associar as coisas que está vendo para que então possa reconhecê-las.

Portanto, esta é a área da Inteligência Artificial responsável por tornar o computador capaz de perceber e interpretar inteligentemente padrões pelos quais ele esteja procurando. Além disso, são estudados aqui os meios de o computador armazenar o conhecimento que por ele foi reconhecido de maneira que futuramente possa este ser utilizado.

Exemplos desta aplicação que já estão sendo adotados são o reconhecimento do dono de uma impressão digital e validação de uma assinatura num cheque bancário. Outros que requerem maior cuidado são quanto a leitura e posterior digitalização de um texto escrito, pois isto dependerá de como o mesmo foi escrito e quais os padrões que o mesmo aceita como sendo semelhantes.

Esta área da Inteligência Artificial, tem fortes relações com outras aqui apresentadas, especialmente com o processamento de linguagem natural, como já dito aqui através da percepção auditiva, e com robótica.

#### **3.3.4. Robótica**

Os robôs elaborados nesta área da Inteligência Artificial não são puramente mecânicos, ou seja, que apenas realizam tarefas mecanizadas e pré definidas, mas são adicionados de um dispositivo eletrônico que funciona como se fosse o cérebro do computador, armazenando conhecimentos que poderão vir a ser úteis e dar maior autonomia a estes.

Tais robôs têm sido usados preferencialmente em indústrias, como a eletrônica e automobilística, mas já existem casos onde os mesmos são utilizados no lugar do ser humano por se tratar de atividades de perigo ou de extrema dificuldade para este último. Exemplos destes casos específicos podem ser o envio destes robôs em viagens espaciais, a prospecção de petróleo no fundo do oceano e também a utilização dos mesmos como assistentes de operações médicas de grande precisão.

De acordo com Ribeiro (1987), no ano de 1982 já existiam em todo o mundo cerca de cinquenta mil robôs, onde destes, trinta mil estavam concentrados no Japão (não incluindo manipuladores manuais nem robôs de seqüência fixa). Passados quinze anos, com a entrada de novas tecnologias e com a expansão das ferramentas computacionais, este número deve ter aumentado consideravelmente.

#### **3.3.5. Prova de Teoremas**

Esta é uma aplicação tipicamente matemática que, quando realizada por seres humanos, é considerada uma atividade puramente intelectual. Esta não requer apenas a habilidade de levantar hipóteses, como também requer uma perícia intuitiva na busca de resultados já provados e armazenados anteriormente, acrescentando assim informações relevantes para a prova do teorema principal. Problemas como a recuperação de informações e o diagnóstico médico podem ser formulados como provas de teoremas.

### 3.3.6. Jogos

Os jogos têm o poder de atrair as pessoas, tanto crianças como os adultos, ainda mais se os mesmos apresentam movimento, cor e uma interação fascinante para o usuário. Os computadores têm esse poder, e então nada mais natural do que fazer uso desta ferramenta não somente para o trabalho, mas também como um instrumento de diversão.

A idéia de que os computadores poderiam competir em jogos com os humanos não é recente, pelo contrário, é tão antiga quanto o próprio computador. Babbage, o pai da computação, pensou em programar seu computador (*Analytical Engine*) para jogar xadrez e jogo-da-velha. Shannon e Turing, separadamente, também pensaram sobre o xadrez; o primeiro escreveu um trabalho descrevendo os mecanismos que poderiam ser usados para programar o computador para participar de tal jogo, enquanto o segundo, alguns anos depois, chegou a conceber tal programa, mas sem nunca o ter implementado. No início da década de 60, Samuel construiu o primeiro programa de jogos. Tal programa, que jogava damas, não somente participava do jogo, mas procurava tirar proveito dos erros para ‘aprender’ novas jogadas.

Os jogos geralmente não necessitam de muitas regras para funcionar, mas o seu espaço de estados (configurações possíveis do sistema) é geralmente muito grande, gerando assim um vasto número de combinações, tornando assim o sistema bastante complexo. Como há sempre um tempo finito de processamento de informações, os jogos devem ser acoplados de técnicas capazes de reduzir o número de combinações, sendo esta técnica geralmente um procedimento heurístico.

Como dito, já que o objetivo do presente trabalho é a construção de um protótipo de Sistema Especialista de apoio ao planejamento de serviços urbanos, deve-se ter um certo conhecimento mais aprofundado sobre esta área de aplicação da Inteligência Artificial. Portanto, apresentar-se-á aqui as vantagens da utilização destes sistemas, bem como a estrutura e interface dos mesmos, além de exemplos de sistemas especialistas bastante conhecidos e consagrados.

## 3.4. Os Sistemas Especialistas

Como para o termo Inteligência Artificial, várias definições são sugeridas para Sistemas Especialistas. Pedersen (1987) apresenta duas correntes distintas; aquela que descreve tais sistemas em função das técnicas e ferramentas usadas para construí-los - “Sistemas Especialistas são programas computacionais que fazem uso de técnicas de inferência”; e outra que enfatiza o conhecimento humano - “Sistemas Especialistas são programas computacionais que empregam o conhecimento humano para resolver problemas que normalmente requerem inteligência humana”.

Para melhor descrever tais sistemas, deve ficar clara inicialmente a noção da palavra “especialista”. Jonhson e Keravnou (ver Rabuske (1995)) definem o especialista como sendo “uma pessoa que, devido ao treino e experiência, é capaz de executar coisas que os outros não

conseguem: especialistas não são apenas proficientes, mas também exímios e eficientes no que fazem (...) Atrás do comportamento do especialista está o corpo de conhecimento operativo que denominamos perito. É razoável supor, então, que os especialistas são aqueles que devemos consultar quando queremos representar a perícia que torna seus comportamentos possíveis.”. Com isto pode-se dizer que os Sistemas Especialistas são aqueles que têm por característica um vasto conhecimento armazenado e organizado sobre o qual o computador deverá aplicar os métodos de inferência para alcançar o objetivo proposto pelo mesmo.

Como estes sistemas são aplicados em áreas onde é marcante a atuação de peritos humanos, os mesmos são de grande valia servindo como consultores para usuários não especialistas. Podem ser usados em problemas rotineiros ou em atividades estressantes, liberando assim a atuação direta de peritos. Podem também levar conhecimento especializado a certos lugares de difícil acesso de pessoal ou onde certos serviços sejam por demais dispendiosos. Ainda podem ser úteis para padronizar as atividades de diferentes operadores ou para treinar operadores menos experientes.

Uma outra vantagem bastante significativa para que se escolha utilizar um Sistema Especialista para a realização de um certo serviço é quanto ao custo final de utilização e reprodução do mesmo, que vem a ser muito inferior ao custo de contratação de um perito.

Mesmo com essas vantagens, nem sempre os Sistemas Especialistas poderão substituir a presença de um perito, já que os mesmos apresentam certas limitações. Eles por si só não são capazes de reconhecer seus próprios limites, isto é, mesmo quando o conhecimento armazenado é insatisfatório para a resolução de um certo problema, ainda assim o sistema realiza longas buscas dentro da base de conhecimentos e acaba por fornecer uma resposta absurda. A tabela a seguir mostra algumas diferenças significantes entre o conhecimento do perito humano e o conhecimento armazenado pelo computador.

Conhecimento HUMANO	Conhecimento ARTIFICIAL
Perecível	Permanente
Imprevisível	Consistente
Dispendioso	Acessível
Individualizado	Genérico
Criativo	Sem Inspiração
Adaptável	Inflexível
Senso Comum	Senso Técnico

Quadro 3.1 - Comparação entre o conhecimento humano e o artificial.

Quanto às limitações dos Sistemas Especialistas, estas ocorrem principalmente pelo fato de as ferramentas disponíveis em IA não serem de todo satisfatórias, apresentando falhas principalmente no que diz respeito a representação e aquisição do conhecimento.

Na prática, uma das mais importantes características de um Sistema Especialista é que o mesmo faz uso de uma ampla base de conhecimentos, esta acumulada durante toda a

construção do mesmo. Este conhecimento deve ser explícito, acessível e organizado para que o sistema obtenha sucesso no decorrer do processo de decisão. Schwabe (1987) faz questão de enfatizar que “A acumulação e codificação de conhecimento é um dos mais importantes aspectos de um sistema especialista”.

Um dos fatores mais marcantes que distingue os Sistemas Especialistas dos sistemas tradicionais é a forma como é feita a interação homem - máquina. Esta interação, como descreve Rich (1988), deve ter as seguintes capacitações: explicação de seu raciocínio, aquisição de conhecimento novo e modificação do conhecimento antigo. Da maneira como o perito explicaria qual raciocínio utilizou para chegar à conclusão, o Sistema Especialista pode ser capaz de explicar as inferências que foram feitas e como foi deduzida tal resposta. Este é um fator extremamente importante quando o objetivo do sistema é ensinar pessoas não especialistas no tópico em questão.

Quanto às alterações na base de conhecimentos, aqui cabe ressaltar que, por serem estes sistemas que dependem exclusivamente do conhecimento armazenado, tal base de conhecimentos deve ser o mais completa e precisa possível. Mas em alguns casos não é possível codificar toda a perícia do especialista e transpô-la para a base de conhecimentos, portanto a interação com o especialista passa a ser uma questão fundamental para a garantia de um bom funcionamento do sistema.

Outra característica marcante que diferencia os Sistemas Especialistas é que estes fazem uso de heurísticas, isto é, utilizam experiências anteriores na solução de problemas considerados análogos com intuito de limitar a busca pelas soluções. A utilização destas faz com que a resolução do problema seja encontrada mais eficientemente do que se fossem feitas pesquisas exaustivas por todo o domínio do sistema. A diferença marcante entre a utilização de um algoritmo e a de um método heurístico é que o primeiro destes garante uma solução correta para o problema, mesmo que esta demore horas ou dias; enquanto que um método heurístico produz uma solução aceitável na maioria das vezes.

O processo de construção de um Sistema Especialista é comumente chamado de Engenharia do Conhecimento. Esta caracteriza-se por um contato direto entre o construtor do software, chamado de engenheiro do conhecimento, e um ou mais peritos na área em que tal sistema se desenvolve. O engenheiro do conhecimento tem como tarefa elaborar as perguntas relevantes para a construção de tal sistema, enquanto que o(s) perito(s) as responderá(ão) de acordo com sua experiência e conhecimento adquirido. Cabe ainda ao engenheiro do conhecimento transformar tais informações em regras e fatos que serão inseridos no sistema para posteriormente serem utilizados no processo de inferência.

Atualmente são as mais variadas as áreas de aplicação dos Sistemas Especialistas. Como apresenta Waterman (1986), as categorias mais comuns são:

- diagnósticos;
- interpretações de situações observadas;
- projeto de produtos sob certas condições;

- prescrições médicas;
- controle do comportamento de determinados sistemas;
- predição de conseqüências.

No item seis deste capítulo serão apresentados alguns dos Sistemas Especialistas mais conhecidos que pertencem a diferentes áreas de conhecimento, reforçando assim a afirmação do referido autor.

### 3.5. Arquitetura de um Sistema Especialista

De acordo com Rich (1988), um Sistema Especialista é composto basicamente por três módulos principais, quais sejam, uma Base de Regras, uma Memória de Trabalho e um Motor de Inferência. Adicionalmente destaca-se como outro módulo importante do SE a Interface que este irá apresentar. A Base de Regras juntamente com a Memória de Trabalho formam o que chama-se de Base de Conhecimentos. Pode-se representar tal arquitetura como o esquema que segue:

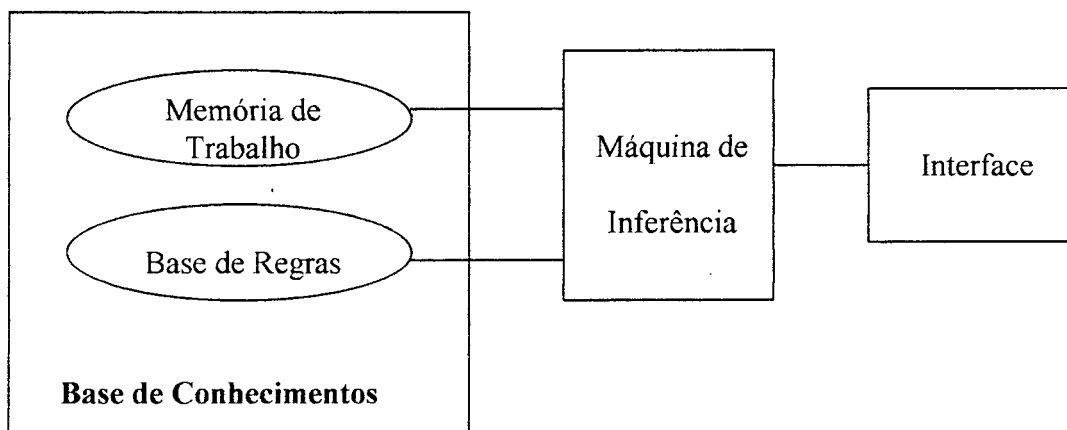


Figura 3.1 - Arquitetura de um Sistema Especialista.

#### 3.5.1. Base de Conhecimentos

A Base de Conhecimentos é, como o próprio nome indica, a parte do sistema que armazena o conhecimento usado na tomada de decisões. Em sistemas baseados em regras de produção - regras da forma Se ... então, este conhecimento é expresso por um conjunto de regras e por um conjunto de fatos, armazenados respectivamente na Base de Regras e na Memória de Trabalho. Fatos são elementos que declaram que algo é verdadeiro sobre um objeto ou uma relação entre objetos. Regras definem relações entre os fatos.



Na Base de Regras estas são armazenadas respeitando a forma das regras de produção, as quais são constituídas por uma premissa e uma conclusão, sendo que a ação especificada na conclusão somente é acionada quando a regra for considerada no decorrer do processo de decisão e a premissa for verdadeira. Tais regras são obtidas a partir de heurísticas aplicadas pelos operadores do sistema em questão.

Na Memória de Trabalho são armazenadas informações temporárias mas relevantes, as quais o sistema grava e apaga de acordo com sua utilização durante o processo de inferência. Quando for necessário o sistema consulta este módulo a procura de fatos importantes para que se chegue a uma solução desejada. No início de tal consulta a Memória de Trabalho está vazia. No decorrer do processo, através de fatos e regras armazenados na base de conhecimentos combinados com as respostas fornecidas pelo usuário, novos fatos são adicionados a esta. Caso estes fatos sejam considerados relevantes na continuação do processo de inferência, eles devem ser salvos e armazenados em uma memória rígida (*winchester* ou mesmo disquetes).

### **3.5.2. Máquina de Inferência**

A Máquina de Inferência, também conhecida como motor ou processador de inferência, é um módulo programado para processar fatos e regras, isto é, é neste módulo que será especificado como tais fatos e regras serão combinados para gerar novos fatos ou para atingir certa conclusão. Tal processamento ocorre em ciclos, sendo que cada ciclo é constituído por três fases, quais sejam:

1. Correspondência de Dados, onde as regras que casam com a atual situação do problema são selecionadas;
2. Resolução de Conflito, onde são escolhidas, dentre aquelas selecionadas na primeira fase, as regras que efetivamente serão executadas;
3. Ação, onde as regras são de fato aplicadas.

Mas a questão mais delicada quanto a Máquina de Inferência é como esta deve ser programada, pois isto depende da natureza do problema e de como o conhecimento é representado e organizado.

Em *shells*, como, por exemplo, a *shell* KAPPA que será aqui utilizada, o Motor de Inferência já está incorporado para interpretar a base de conhecimentos criada pelos construtores do softwares. Já linguagens de programação, como, por exemplo, o LISP, requerem que tal mecanismo seja projetado e implementado pelos engenheiros do conhecimento.

### **3.5.3. A Interface**

A Interface é a maneira como o sistema irá interagir com as pessoas e vice versa. Esta deve ser bastante satisfatória para que não hajam erros de interpretação de ambos os lados, já que a linguagem que ambos utilizam, linguagem de programação e linguagem natural respectivamente, são distintas. Muitas pesquisas vem sendo feitas na área de processamento de linguagem natural para que esta interface ocorra cada vez de modo mais simples e claro. Este módulo do Sistema Especialista pode então ser subdividido em dois, quais sejam, o da interface com o especialista e o da interface com o usuário.

A maneira como o sistema se apresenta para a equipe de desenvolvimento de um Sistema Especialista é de extrema importância para o bom desenvolvimento do sistema em questão. Por esse motivo é que a interface com o(s) especialista(s) deve ser personalizada, ou seja, deve-se customizar o ambiente de forma que o sistema fique o mais idêntico possível àquele ambiente no qual o especialista está acostumado a atuar. Esta interação é feita unilateralmente no sentido especialista - sistema, sendo que o primeiro pode adicionar, excluir ou alterar fatos e regras que estejam armazenados.

A interface com o usuário final do sistema também é fundamental para o sucesso deste. Esta deve ser ergonomicamente bem projetada, visando o bom entendimento por parte do usuário de todas as informações que o sistema lhe oferecer ou pedir, e ainda levando em conta o quanto este usuário está familiarizado com o domínio de trabalho e com os computadores em geral. Para que esta interface seja amigável, pode-se fazer uso de janelas, menus gráficos, animação e outros recursos atraentes. Esta deve ainda fornecer ao usuário a possibilidade de interromper a execução do sistema em qualquer momento e enviar mensagens de erro claras e informativas para que os mesmos possam ser evitados futuramente.

Diferentemente da interface do sistema com o especialista, a interface deste com o usuário é feita bilateralmente, tanto no sentido sistema - usuário como no sentido inverso. No primeiro sentido o sistema apresenta perguntas, tira conclusões e apresenta sugestões ao usuário; já no segundo, o usuário é capaz de recuperar informações contidas na base de conhecimentos.

### **3.6. Alguns Sistemas Especialistas**

Como já foi aqui comentado, as aplicações práticas dos Sistemas Especialistas se dão nas mais diversas áreas de conhecimento e cada vez mais novos sistemas são desenvolvidos para auxiliar as pessoas não especializadas a organizar e aplicar tal conhecimento em situações reais.

Alguns destes Sistemas Especialistas merecem maior consideração por terem marcado a evolução desta área de aplicação da Inteligência Artificial. Estes são o DENDRAL, o MYCIN e o PROSPECTOR.

O DENDRAL, como já foi dito quando da explanação sobre IA, foi considerado como sendo o primeiro dos Sistemas Especialistas. Este foi desenvolvido na Universidade de Stanford no ano de 1964. Seu objetivo era auxiliar na descoberta da estrutura molecular de um composto químico. Mais tarde este sistema foi por muitos contestado, tanto quanto ao que se refere a sua utilidade quanto às suas bases científicas.

O MYCIN também foi desenvolvido em Stanford, mas no ano de 1972. Sua área de aplicação é a química e tem por objetivo principal diagnosticar doenças infecciosas. Testes foram realizados comprovando a eficácia do mesmo comparando suas respostas com as respostas dadas por médicos; seus diagnósticos inclusive foram mais acertados do que aqueles dados por médicos não especialistas da área.

Já o PROSPECTOR foi desenvolvido no ano de 1978 no Stanford Research Institute e tem como área de aplicação a geologia. Este serve de auxiliar nas pesquisas sobre depósitos de certos minérios em particular. Sua viabilidade foi comprovada em 1982 quando o mesmo inferiu que sob uma região do estado de Washington existiam alguns depósitos de molibdênio. Os geologistas a princípio discordaram, mas quando foram ao local fazer as perfurações, constataram que o PROSPECTOR estava correto.

Waterman (1986) apresenta uma listagem com mais de cento e oitenta Sistemas Especialistas existentes até o ano da publicação de seu referido livro. Cada qual é brevemente descrito, abordando qual o tipo de ferramenta utilizada na sua implementação, qual o tipo de raciocínio que este aplica, onde e quando o mesmo foi desenvolvido; além de apresentar por fim uma referência bibliográfica sobre cada um destes. Tais sistemas são agrupados de acordo com o domínio de conhecimento em que os mesmos mais comumente são aplicados. Essas áreas de aplicação vão desde a medicina (uma das que possui o maior número de aplicações), passando pela geologia, pelas ciências exatas, até áreas mais recentes como as de eletrônica e sistemas computacionais.

Alguns destes Sistemas Especialistas, pode-se dizer que os mais conhecidos e citados na literatura, são sucintamente descritos no Quadro 3.2 apresentado a seguir.

<b>Sistema Especialista</b>	<b>Aplicação</b>	<b>Desenvolvimento</b>
ACE	Identifica problemas na rede de telefonia	Bell Laboratories
ANALYST	Dá assistência em diversas situações num campo de batalha	MITRE Corporation
BABY	Auxilia no cuidado de recém-nascidos internados na UTI	University of Illinois
BLUE BOX	Analisa paciente depressivo e recomenda tratamento	Stanford University
DART	Diagnostica falhas em computadores	Stanford University
DELTA	Auxilia na manutenção de locomotivas	General Electric Company
FOLIO	Determina os melhores investimentos de cada clientes	Stanford University
INTERNIST I	Auxilia a diagnosticar o estado geral de um paciente	University of Pittsburg
JUDITH	Auxilia na argumentação sobre casos de lei civil	Heidelberg University e Darmstadt University
MACSYMA	Manipula expressões algébricas, integra e resolve equações	Massachusetts Institute of Technology
MATHLAB 68	Manipula simbolicamente expressões algébricas	Massachusetts Institute of Technology
MES	Auxilia a diagnosticar problemas em aeronaves	Air Force Institute of Technology
MOLGEM	Auxilia em experimentos de clonagem de genes	Stanford University
POMME	Ajuda o agricultor a melhorar o cultivo de maçãs	Virginia Polytechnic Institute
XCON	Configura sistemas computacionais para clientes	Carnegie-Mellon University / DEC

**Quadro 3.2 - Sistemas Especialistas nas diversas áreas de conhecimento.**

### **3.7. Fases de Desenvolvimento de um Sistema Especialista**

A construção de um Sistema Especialista é uma tarefa árdua que consome muito tempo. Diferentemente da construção de sistemas computacionais tradicionais, onde as fases são consideradas distintas e adotadas de forma seqüencial, o desenvolvimento dos Sistemas Especialistas se dá a partir de um protótipo e as etapas não ocorrem linearmente, sendo que os resultados obtidos em uma das fases poderão conduzir a revisão de etapas anteriores. De acordo com Waterman (1986), a seqüência formal das etapas a seguir para a construção de um Sistema Especialista é dada pelas fases de: Identificação, Conceituação, Formalização, Implementação, Teste e Avaliação, e Revisão.

### 3.6.1. Fase de Identificação

Nesta primeira fase deverão ser especificadas todas as informações necessárias para a caracterização do sistema, informando assim sobre os participantes, sobre os recursos envolvidos, sobre as características relevantes do problema e os objetivos a se atingir.

Quanto aos participantes do projeto, duas pessoas deverão ser destacadas: primeiro deve-se definir quem será o “dono” do sistema, ou seja, aquele que tem a última palavra no que diz respeito ao corpo do sistema; posteriormente identifica-se o engenheiro de conhecimento, como definido anteriormente neste texto, aquele que elabora a base de conhecimentos e apresenta o sistema em uma forma passível de implementação. Outra questão crucial quanto aos participantes refere-se a escolha do especialista, que deve ser aquela pessoa com maior conhecimento e experiência no assunto, que seja capaz de assumir tal responsabilidade e que tenha tempo para se dedicar a tal projeto, pois caso sua participação seja insatisfatória, a margem de erros do sistema aumentará consideravelmente. Os outros participantes serão escolhidos de acordo com as características e também do porte do sistema.

Quanto aos recursos envolvidos, devem ficar claros nesta etapa todos os recursos computacionais, tanto *hardwares* quanto *softwares*, que estarão disponíveis aos participantes, bem como os recursos bibliográficos aos quais estes poderão consultar. Também deve-se definir neste momento um cronograma básico a ser seguido, estipulando um limite de tempo no qual o protótipo deve ser finalizado.

Quanto às características do problema e ao objetivo do sistema, algumas perguntas devem ser respondidas para que o engenheiro de conhecimento e o especialista possam efetivamente entender o problema e definir uma metodologia para solucionar o mesmo. Segundo Hayes-Roth, Waterman e Lenat (1983), algumas destas perguntas são as seguintes:

- a) Quais são os dados do problema?
- b) O que é uma solução deste problema?
- c) Que aspectos o perito humano acha importante na solução?
- d) Que situações costumam impedir a solução?

### 3.6.2. Fase de Conceituação

Nesta etapa do desenvolvimento do sistema deverão ser definidos, pelo engenheiro de conhecimento e pelo especialista, os conceitos, as relações e os mecanismos de controle necessários para se produzir uma solução. Estes serão responsáveis ainda pelo estabelecimento do grau de refinamento, ou granularidade segundo Waterman (1986), que determinará a que nível de detalhamento o conhecimento deverá ser representado. Cabe aqui dizer que nesta fase ainda não há necessidade de uma especificação detalhada do problema, sendo que esta fase provavelmente será retomada após a implementação do protótipo para que sejam feitas novas considerações sobre o mesmo.

Um alerta vindo de Rabuske (1995) é que esta fase “não dispensa uma olhada para frente, verificando quais as possíveis representações e ferramentas que poderão servir de base para a implementação”, mas deve-se tomar o cuidado de não viciar as fases seguintes, ou seja, de não impor restrições para que tais ferramentas e representações sejam de fato as adotadas.

Como para a fase de identificação, Hayes-Roth, Waterman e Lenat (1986) formularam algumas perguntas cujas respostas servirão para definir a base conceitual do sistema. Algumas destas são:

- a) Que tipos de dados estão disponíveis?
- b) O que é dado e o que é inferido?
- c) Como se interrelacionam os objetos do domínio?
- d) A solução do problema compreende quais processos?

### **3.6.3. Fase de Formalização**

Tais conceitos definidos na fase anterior deverão ser agora representados de maneira formal utilizando a linguagem do Sistema Especialista (uma linguagem de programação ou aquela utilizada pela *shell* que tiver sido adotada)

Nesta fase de desenvolvimento do sistema, o engenheiro de conhecimentos é considerado de extrema importância, sendo ele o responsável pela escolha das ferramentas mais adequadas para serem aplicadas ao problema em questão. Este deve concentrar sua atenção em três pontos importantes, quais sejam:

- a) o espaço de hipóteses, estabelecendo as características dos conceitos e como estes se interligam;
- b) o modelo subjacente, determinando de que maneira as soluções são geradas;
- c) as características dos dados, definindo aspectos tipo precisão e consistência.

### **3.6.4. Fase de Implementação**

Esta fase de desenvolvimento do Sistema Especialista se inicia com a escolha da linguagem de programação ou a *shell* a ser utilizada. Após uma destas ter sido definida, passa-se então para a implementação propriamente dita, ou seja, todas as informações obtidas nas fases anteriores serão aqui codificadas (de acordo com a linguagem ou *shell* escolhida) para que façam parte da base de conhecimentos do sistema.

Um cuidado adicional deve ser tomado aqui quanto a integração das informações que estão sendo editadas a fim de eliminar desigualdades entre as estruturas de dados, regras ou

estratégias de controle, pois caso exista alguma controvérsia a base de conhecimentos torna-se inconsistente, afetando assim todo o raciocínio do sistema.

Recomenda-se então o desenvolvimento de um protótipo do sistema, um programa simplificado que atenda as necessidades básicas do sistema e conseqüentemente reduza a probabilidade de fracasso do mesmo. Utilizar-se-á este para verificar, através de exemplos mais genéricos, a efetividade das decisões tomadas nas fases iniciais do sistema. Para avaliar seu desempenho e utilidade antes que o mesmo seja mais elaborado e detalhado, serão feitos testes em diversas situações para verificar a coerência e veracidade dos resultados que o mesmo oferece.

### **3.6.5. Fase de Teste e Avaliação**

Esta fase de desenvolvimento do sistema é de suma importância, pois esta análise é feita considerando tanto o corpo do sistema que está sendo trabalhado quanto a razão de ser do mesmo.

O sistema deve ser testado e avaliado frequentemente, ou seja, todas as fases de desenvolvimento do sistema, desde o projeto até sua aplicação, devem ser de alguma forma testadas e avaliadas para que possíveis erros não se acumulem nas fases seguintes e gerem uma falha gigantesca no final do sistema. Deve-se avaliar a correção das respostas, a explicação que será dada ao usuário quando este pedir, a interação homem - máquina, a eficiência, a precisão e a credibilidade do sistema.

A maneira mais simples do sistema ser avaliado é comparar o seu conhecimento armazenado com o do especialista ou comparar a forma com que ambos resolvem o problema, mas estas comparações não fornecem resultados muito confiáveis. Uma forma de avaliação que pode ser útil e que representará como os futuros usuários do sistema reagirão ao utilizar o mesmo, é a utilização do mesmo por pessoas que não participaram do desenvolvimento do sistema e que portanto não conhecem como o mesmo trabalha.

### **3.6.6. Fase de Revisão**

Dependendo do resultado dos testes e da avaliação feita sobre o sistema em questão, algumas modificações deverão ser feitas no corpo do sistema com intuito de melhorar o seu desempenho futuro. Inserções de novas regras ou fatos e pequenas alterações poderão ser feitas sem afetar todo o conhecimento já armazenado pelo sistema, desde que não haja informações controversas. Mas caso tais alterações sejam significativas, como, por exemplo, a mudança do objetivo do sistema ou ainda uma alteração nas características do problema, possivelmente haverá necessidade de se reavaliar todas as fases deste sistema.

Resumidamente, pode-se caracterizar as fases de desenvolvimento de um Sistema Especialistas através da Figura 3.2 apresentada a seguir.

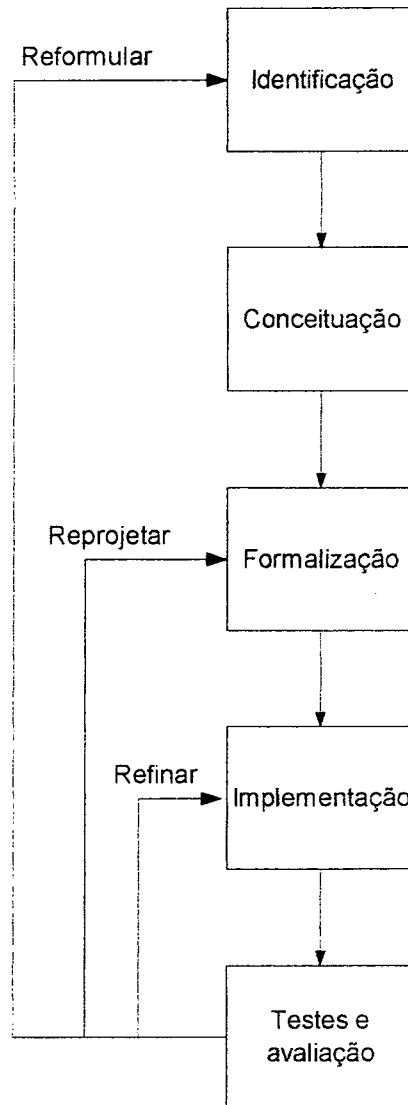


Figura 3.2 - Fases do desenvolvimento de um Sistema Especialista.

Para desenvolver o protótipo de Sistema Especialista proposto no presente trabalho, utilizou-se a *shell* KAPPA, que será apresentada no capítulo seguinte.



## ***A Shell KAPPA***

O presente capítulo descreve as principais características da shell KAPPA e está baseado nos manuais desta shell (ver KAPPA User's Guide (1991) e KAPPA Reference Manual (1991)) e na experiência pessoal anterior de trabalho com a mesma.

### **4.1. Informações Gerais**

A *shell* KAPPA é uma ferramenta a ser utilizada na construção de softwares que simulam sistemas complexos, como aqueles freqüentemente encontrados nas situações reais, proporcionando uma maior facilidade na representação do conhecimento sobre estes sistemas.

Uma definição dada pelos fabricantes da tal *shell*, a *Intelicorp Corporation*, é a seguinte: "O *software* KAPPA™ pretende produzir um computador pessoal instruível oferecendo aos programadores uma ferramenta útil para auxiliar no desenvolvimento de sistemas baseados no conhecimento. Para desenvolver tais sistemas, deve-se fornecer as informações, regras e procedimentos requeridos para criar o sistema resultante. Deve-se esperar que se tenha que testar, dar assistência e manutenção e ainda monitorar o sistema que foi desenvolvido utilizando o *software*."

A *shell* KAPPA é usada para construir aplicações baseadas no conhecimento, isto é, sistemas que armazenam o conhecimento necessário para entender a complexidade de certos campos de ação. Com estes sistemas os computadores são capazes de incorporar o conhecimento de *experts* humanos para assim resolver problemas de grande dificuldade. Além da quantidade significativa de conhecimento que pode ser armazenado diretamente nestes sistemas, a máquina tem habilidade de "raciocinar" além do conhecimento diretamente armazenado. Esta habilidade dos computadores, quando munidos de uma *shell* como esta, é chamada de inferência.

Um bom candidato para ser modelado dentro dos padrões da base de conhecimentos da *shell* KAPPA deve ser um sistema onde são conhecidos claramente os seus componentes, como estes se comportam, se os mesmos interagem e, em caso afirmativo, como interagem. No caso do sistema proposto para ser desenvolvido no presente trabalho, estas características são satisfeitas, tornando então sua construção viável dentro dos moldes da *shell* em questão.

O sistema que será aqui apresentado para exemplificar elementos ou rotinas da referida *shell* será o Poli 1.0 - Um Sistema que Classifica Polígonos, que, como o próprio nome diz, classifica formas poligonais, utilizando para tanto as medidas dos lados e dos ângulos das mesmas. Tal sistema foi desenvolvido no ano de 1995 no GEIAAM - Grupo de Estudos de Inteligência Artificial Aplicada à Matemática - por Cleide Regina Lentz Paladini, Flávia Silveira Correia e Karin Cristina Siqueira (ver, por exemplo, Correia et al.(1995a) e Correia et al. (1995b)).

## 4.2. Diferenças entre KAPPA e Processadores Convencionais

A primeira grande diferença entre a *shell* KAPPA e outras linguagens de programação é aquela que diferencia a Inteligência Artificial da programação convencional, qual seja, a manipulação de símbolos ao invés de números. Além disso, esta *shell* pode armazenar complexas relações entre objetos, conceitos e processos.

Nesta *shell* o armazenamento e o processamento dos dados são traduzidos em modelos e raciocínio. Os modelos são utilizados na armazenagem do conhecimento que foi inserido dentro do sistema, enquanto que o raciocínio se encarrega de processar tais conhecimentos. Em ambos os casos tal representação é qualitativamente diferente dos sistemas convencionais de processamento de dados.

Outra grande qualidade e por conseguinte grande diferença entre a *shell* e os processadores convencionais é quanto à transparência da primeira. Os programas em geral são como “caixas pretas”, onde o usuário não tem condições de acompanhar passo a passo o que foi utilizado no processo até a obtenção da solução final. Em problemas complexos como, por exemplo, a demonstração de teoremas matemáticos, uma visão geral dos passos seguidos pelo programa pode ser de suma importância para o entendimento das respostas dadas ao usuário.

A interface com o usuário, no caso em questão o programador, ocorre através de janelas e ícones, da maneira convencional de todos os programas em ambiente *Windows*. Assim o usuário se sente mais familiarizado com o ambiente e não tem receio de cometer graves erros, já que os comandos principais como Novo, Abrir, Salvar, Copiar e outros são semelhantes aos por ele conhecidos.

Combinando a transparência da linha de raciocínio que o programa segue e a interface amigável que a *shell* oferece, o usuário pode conferir a posição de cada mudança no sistema bem como os passos conseqüentes desta mudança. Para tanto este pode fazer uso de um dos aplicativos do KAPPA, o *Debugging*, que possibilita uma visão do interior do programa em ação. Neste aplicativo uma função é analisada passo a passo e por isso este é um excelente recurso para se detectar erros. Em cada passo então são testadas algumas informações contidas na base de conhecimentos, e quando existem erros de programação ou de lógica, imediatamente aparece uma mensagem de alerta. Tal aplicativo será apresentado posteriormente neste capítulo.

### 4.3. Definição dos Objetos, Classes e Slots

Para que seja possível construir a base de conhecimentos de um certo sistema, são necessárias ferramentas para representar as coisas concretas do mundo, coisas estas que serão as componentes do problema.

Diferentemente de outras ferramentas de sistemas chamados espertos (*Expert Systems Tools*) a *shell* KAPPA permite que se programe mais do que simplesmente regras. Nela os componentes do domínio do sistema são apresentados por estruturas chamadas Objetos. Estes podem representar algo em particular ou podem representar um grupo de elementos com certas características em comum. No sistema especialista Poli 1.0 - Um Sistema que Classifica Polígonos, objetos podem ser os triângulos isósceles ou mais genericamente podem ser os próprios triângulos.

Os objetos podem ser Classes ou Instâncias, que são usadas para representar coisas concretas ou conceitos. Uma classe é um objeto mais geral, enquanto uma instância é um item mais específico. Outra diferença entre ambas é que as classes podem possuir subclasses, enquanto que as instâncias, devido à especificidade, não podem. Do mesmo exemplo anterior, pode-se dizer que Triângulos representa uma classe, enquanto que Triângulo Isósceles representa uma instância (que é também uma subclasse da classe Triângulos).

As relações entre os objetos podem ser representadas encadeando-os como em uma árvore, numa estrutura hierárquica representada na janela *Object Browser*. As classes ou instâncias são organizadas de acordo com o nível hierárquico, respeitando as relações de antecessores e decedentes. A classe imediatamente acima de qualquer subclasse é dita ser a classe Parente desta.

Tal árvore a qual se estava referindo, é construída a partir da classe Raiz (*Root*), sendo esta já pré definida e, portanto, todas as classes e instâncias que forem criadas serão suas decedentes. Existe ainda uma instância pré definida chamada *Global*, onde pode-se definir variáveis de caráter temporário, e uma subclasse chamada *Image* designada para armazenar objetos gráficos que poderão vir a ser criados. Esta estrutura hierárquica será útil quanto a representação de atributos a serem herdados pelos decedentes, mas esta questão será abordada mais especificamente no próximo tópico.

Parte da estrutura hierárquica do sistema Poli 1.0 a qual está-se referindo pode ser vista na Figura 4.1.

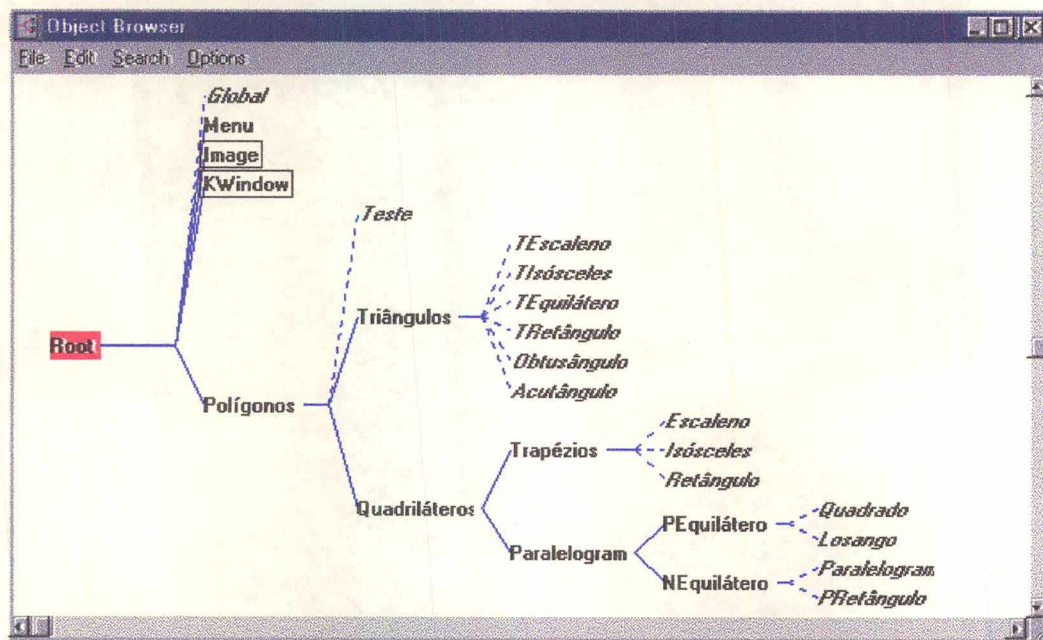


Figura 4.1 - Estrutura hierárquica do sistema Poli 1.0.

#### 4.4. A Caracterização Através de Slots

Cada objeto, classe ou instância, possui o que costuma-se chamar de *Slots*, que são entidades que descrevem as características de tal objeto. Estes adicionam detalhes, atributos e propriedades ao objeto em questão. Cada uma destas descrições é representada por um *slot* e pelo valor deste *slot*, podendo este valor ser um número, uma lista, um texto, um valor booleano (verdadeiro ou falso) e até mesmo um outro objeto.

A partir do objeto no qual um *slot* foi definido, ele, mas nem sempre o seu valor, será herdado por todas as classes decedentes. Quando um *slot* é herdado de uma classe anterior, ele aparece precedido de um asterisco (\*). Isto não impede que sejam criados *slots* locais, nestes casos não precedidos de asterisco.

Pode-se desta forma especificar as propriedades gerais apenas uma vez, escolhendo para isso a classe mais apropriada, para que a partir desta todas subclasses recebam tal *slot*. Isto proporciona muitas vantagens, já que a base de conhecimentos torna-se mais fácil de ser criada e manipulada, além de o conhecimento ser mais facilmente armazenado, pois as propriedades gerais são declaradas em apenas uma classe e aplicadas a todos os seus decedentes.

Usando então a hereditariedade, pode-se assumir os valores dos objetos abaixo na hierarquia, mais rápido e facilmente. Fazer mudanças nestes valores também torna-se muito mais simples; usando a opção de *slot* (*Slot Option*), pode-se fornecer informações mais detalhadas sobre tal *slot* para assim obter uma aplicação mais precisa. Ali se escolhe o tipo de *slot*, os limites dos valores que o usuário poderá entrar e outras especificações para tal entidade.

A figura a seguir apresenta os *slots* locais e os herdados referentes à classe Triângulos do sistema Poli 1.0.

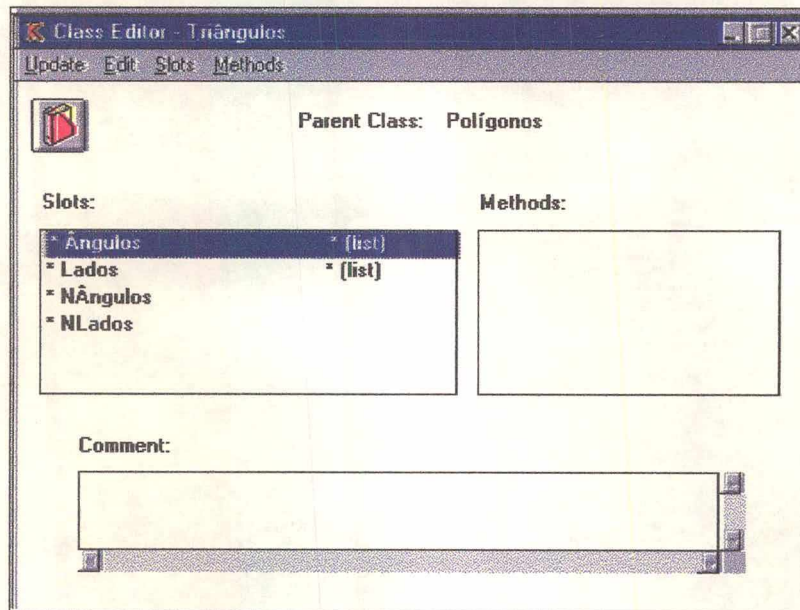


Figura 4.2 - Slots da classe Triângulos.

#### 4.5. Regras, Funções e Métodos

Na *shell* KAPPA, os processos do domínio são representados de três diferentes maneiras: regras, funções e métodos. Todas estas estruturas são escritas em linguagem KAL, a linguagem própria da *shell* KAPPA.

A primeira maneira de representar os processos em KAPPA envolve a utilização de regras para representar as etapas do processo. Uma regra especifica as condições que devem ser satisfeitas para que a inferência seja aplicada. Elas podem especificar interações complexas entre vários dos componentes do sistema.

O raciocínio baseado em regras é usado para resolver muitos problemas que possuem afirmações condicionadas. As regras possuem a forma *if - then* (se - então), onde na parte *if* estão contidas as premissas, representando um teste ou uma condição que deve ser satisfeita; na parte *then* as conclusões, representando a ação que deve ser desencadeada caso o teste tenha sido bem sucedido ou a condição tenha se verificado. Elas são manipuladas por uma máquina de inferência que liga premissas de uma regra com conclusões de outra.

O exemplo de uma regra do sistema ao qual está se referindo, o Poli 1.0, é o seguinte:

SE o polígono tem três lados e a soma das medidas de dois destes lados é menor do que a medida do terceiro lado, ENTÃO este polígono é realmente um triângulo.

A segunda maneira de representação dos processos na *shell* KAPPA envolve funções. A *shell* fornece uma biblioteca com duzentos e quarenta (240) funções pré definidas que ajudam a manipular a base de conhecimentos. Estas funções vão de um simples operador lógico como o operador aditivo “+”, até funções mais complexas como “Enquanto...”. Usando a linguagem KAL ou também a linguagem de programação C (que em muito se assemelham), pode-se construir outras funções de que se pode necessitar. Todas estas funções encadeadas é que formarão o seqüenciamento lógico de ações que conseqüentemente permitirão a obtenção de alguma solução para o problema.

Uma função que foi utilizada no sistema Poli 1.0 foi aquela que descrevia o Teorema de Pitágoras através das funções soma e produto internas à *shell* KAPPA.

A terceira e última maneira envolve a especificação do comportamento dos *slots*, que é feita através da criação dos métodos. Estes são escritos na linguagem de programação KAL e sua função é a de especificar como um objeto pode se comportar, isto é, eles definem como um objeto deve agir, dando a eles habilidade para tal. Cada ação que um objeto pode executar em algum momento é descrita através de um método. Assim, eles são procedimentos que são representados como atributos de um certo objeto.

Quando um objeto recebe uma mensagem correspondente a um de seus métodos, este é ativado, e todos os procedimentos que tal objeto deve seguir serão especificados pelo método e com isso, tais procedimentos serão imediatamente adotados.

Esta técnica de armazenar o comportamento de um objeto é um dos atributos que fazem parte da técnica de programação chamada Programação Orientada ao Objeto. Um programa deste tipo é composto por uma coleção de estruturas de dados que são os objetos. Cada objeto contém dois tipos básicos de informação: a primeira é a informação que descreve o objeto em si e a segunda é a informação que especifica o que tal objeto pode fazer.

Este tipo de Programação Orientada ao Objeto que a *shell* KAPPA faz uso, além de associar o comportamento dos objetos através dos métodos, também prevê uma interface uniforme que será disparada a diferentes objetos com diferentes comportamentos. Este tipo de programação tem se mostrado bem mais eficiente do que a utilização de regras nos casos em que se deseja expressar procedimentos bem definidos.

Quanto aos métodos, cabe ainda lembrar que os mesmos, tal como acontece com os *slots*, podem ser herdados, bem como podem ser adicionados métodos locais em qualquer etapa da programação.

Para finalizar a descrição das três maneiras de representar os processos do domínio, cabe aqui um comentário. Em muitos casos pode ocorrer que mais de uma regra, função ou método seja possível de ser aplicada, isto é, que as condições para que a mesmas sejam desencadeadas dentro do processo tenham sido satisfeitas. Nestes casos podem ser definidas prioridades de escolha, dando valores mais altos àquelas regras, funções ou métodos que sejam mais importantes ou que levem mais facilmente à solução.

## 4.6. Formas de Raciocínio

Existem duas maneiras básicas de se raciocinar: a primeira é quando se tem uma condição inicial e a partir dela são testadas regras até que se chegue a uma condição final, no caso, a solução; já a outra maneira é quando se tem o objetivo final e a partir dele são testadas regras de maneira inversa tal que se encontre qual a condição inicial que deve ser satisfeita para que se possa chegar a tal meta. Estes tipos de raciocínio são chamados respectivamente de Raciocínio para Frente (*Forward Chaining*) e Raciocínio para Trás (*Backward Chaining*).

Como dito, no Raciocínio para Frente, também chamado de Encadeamento Dirigido por Dados, o sistema vai testando as regras selecionadas e conforme estas forem sendo satisfeitas, novas informações vão sendo introduzidas na memória de trabalho e com isso o sistema vai processando os resultados. Neste caso não é necessário o estabelecimento de uma meta (*goal*). Este tipo de raciocínio é usado para determinar as conseqüências de um novo fato, sendo então mais apropriado quando é necessário entrar com novos fatos (dados) e encontrar suas conseqüências.

Para melhor entender esta maneira de raciocinar, supõe-se que sejam conhecidas as seguintes informações e regras:

Informação 1: Existe incêndio.

Informação 2: O material é inflamável.

Informação 3: O fogo não é dominado.

Regra 1: Se existe incêndio e o material é inflamável  
Então tente apagar o fogo.

Regra 2: Se tentar apagar o fogo e o fogo não é dominado  
Então saia o mais rápido possível.

As informações 1 e 2 fazem com que a regra 1 seja acionada, já que as duas condições para tal são satisfeitas. Com isso, a conclusão da regra 1 passa a ser uma informação conhecida, e esta combinada com a informação 3 desencadeia a regra 2, inferindo então que se deve sair o mais rápido possível.

O Raciocínio para Trás, também chamado de Encadeamento Dirigido por Objetivos, por sua vez exige, como já foi dito, o estabelecimento de uma meta, utilizando esta para responder as questões propostas e que serão testadas; ou seja, o seu objetivo é encontrar uma situação que satisfaça sua questão meta. Um objetivo pode ser satisfeito diretamente por um elemento na memória de trabalho, ou podem existir regras que permitam inferir algum dos objetivos correntes. Este tipo de raciocínio é utilizado em momentos em que são necessárias informações específicas e um raciocínio mais direcionado.

Considerando que as mesmas informações do exemplo anterior são conhecidas, reestruturam-se as regras 1 e 2 da seguinte maneira:

Regra 1: Tentar apagar o fogo  
Se existe um incêndio e o material é inflamável.

Regra 2: Sair o mais rápido possível  
Se tentar apagar o fogo e o fogo não é dominado.

O raciocínio para trás pode ser usado neste caso para responder a pergunta: “deve-se sair o mais rápido possível?”. Para sair o mais rápido possível as condições de tentar apagar o fogo e o fogo não é dominado devem ser satisfeitas (de acordo com a regra 2). A informação 3 garante que o fogo não é dominado, enquanto que o fato de se tentar apagar o fogo decorrer da regra 1. As condições para que a regra 1 seja acionada estão na base de conhecimentos (informação 1 e 2), e portanto a mesma pode ser acionada, concluindo assim que se deve sair o mais rápido possível.

O tipo de encadeamento a ser adotado normalmente é definido de acordo com as características do problema a ser resolvido. Problemas de planejamento, classificações e simulações geralmente utilizam o encadeamento para frente, enquanto problemas de diagnóstico, onde existem apenas algumas saídas possíveis mas um grande número de estados iniciais, utilizam o encadeamento para trás.

#### 4.7. A Linguagem KAL

A linguagem de programação KAL, própria da *shell* KAPPA, é o caminho mais rápido para acessar todo potencial da *shell* utilizando a interface entre esta e o programador.

A *shell* KAPPA proporciona um mecanismo para definir expressões similares ao da programação com linguagem clássica. Estas expressões devem seguir os padrões da linguagem KAL. As expressões formuladas nesta linguagem podem ser escritas diretamente dentro de uma janela chamada *KAL Interpreter*, que será descrita posteriormente.

A linguagem KAL permite:

- Adicionar novas informações à base de conhecimentos, como, por exemplo, para criar um novo objeto ou *slot*, para modificar o valor de um certo *slot*, ou criar uma ligação entre dois objetos;
- Remover informações da base de conhecimentos, como, por exemplo, para remover uma ligação entre dois objetos;
- Fazer perguntas sobre as informações contidas na base de conhecimentos, por exemplo, recuperando o valor de um *slot*, encontrando *slots* que têm um certo valor ou recuperar as instâncias de um objeto;



- Programar a interface entre o computador e o usuário, que irá ocorrer através de perguntas que devem ser respondidas pelo usuário e mensagens conclusivas ou de erro e alerta.

Esta linguagem fornece desde operadores simples como os das quatro operações matemáticas básicas (+, -, \* e /), quanto operadores de associação e atribuição(= e +=), operadores lógicos (*And*, *Or* e *Nor*), até expressões especiais como *For*, *While* e *If*.

Mas as ferramentas mais potentes e conseqüentemente mais poderosas da linguagem KAL são as funções. Tudo o que tiver que ser feito em qualquer das janelas, pode ser feito mais rápido e facilmente se estas forem usadas. Como já foi dito neste texto, a *shell* KAPPA proporciona uma biblioteca com duzentos e quarenta funções já prontas, podendo estas serem usadas para definir expressões ou criar outras funções combinando estas funções padrão. Esta biblioteca permite os seguintes tipos de operação:

- Criar, acessar e modificar os elementos da base;
- Avaliar expressões lógicas e matemáticas;
- Manipular listas e arquivos;
- Controlar o processamento do conhecimento e outros.

A regra apresentada como exemplo anteriormente, será escrita da seguinte maneira na linguagem KAL:

#### Linguagem Natural:

Se o polígono tem três lados e a soma das medidas de dois destes lados é menor do que a medida do terceiro lado,

**Então** este polígono é realmente um triângulo.

#### Linguagem KAL:

```

If {
  GetNthElem( Pol:Lados,1) < ( GetNthElem( Pol:Lados,2) + GetNthElem( Pol:Lados,3))
  And GetNthElem(Pol:Lados,2) < (GetNthElem(Pol:Lados,1)+ GetNthElem(Pol:Lados,3))
  And GetNthElem(Pol:Lados,3)< (GetNthElem(Pol:Lados,1)+ GetNthElem(Pol:Lados,2))
};
Then {
  PostMessage( "Este polígono realmente é um triângulo. Você sabe como cheguei a esta conclusão?" );
  PostMessage( "É SIMPLES: Nenhum dos 3 lados é maior, nem igual a soma dos outros dois." );
  Pol:ÉTriângulo = TRUE;
};

```

## 4.8. Interface

Além de todas as ferramentas aqui descritas, a *shell* KAPPA proporciona uma rica e poderosa interface para o desenvolvimento das aplicações, como foi comentado anteriormente. Esta interface consiste em ferramentas para visualização e modificação de vários elementos desta *shell*. O sistema deve fornecer todos os recursos necessários para que o programador (neste caso é deste usuário que estamos falando e não do usuário final do sistema) entre com os dados, formule questões e dê respostas. A interface com o usuário é a parte do sistema que administra esta troca de informações.

Há duas razões diferentes para que se faça uso desta interface. A primeira é quanto às facilidades que são proporcionadas ao programador que constrói a aplicação, já que nesta interface a linguagem KAL constitui um modo conveniente e simples de modificar a base de conhecimentos e fazer perguntas sobre esta. A segunda razão para se fazer uso desta interface é que a mesma possibilita o programador construir a interface que será feita entre o sistema e o usuário final do mesmo, dando a este último fácil acesso a uma aplicação completa e mostrando somente os detalhes que a ele interessam.

Como a *shell* KAPPA trabalha em ambiente *Windows*, as janelas são as componentes básicas da interface. Estas janelas são áreas separadas umas das outras onde aparecem textos, gravuras e listas de opções, que são os típicos menus. A janela principal do KAPPA (*Main Window*) apresenta nove outras janelas, quais sejam:

- *Object Browser*,
- *Session*,
- *Edit Tools*,
- *KAL Interpreter*,
- *KAL View Debugger*,
- *Find Replace*,
- *Rule Relations*,
- *Rule Trace*,
- *Inference Browser*.

As cinco primeiras janelas são as mais utilizadas e portanto dar-se-á enfoque especial sobre as mesmas.

### 4.8.1. *Object Browser*

Nesta janela, a árvore representando a hierarquia entre os objetos é apresentada. Nela podem ser criadas, excluídas e renomeadas classes e instâncias, bem como manipulados *slots* e métodos de cada uma destas. Aparecem ainda, além da instância *Global* e da subclasse *Image*, as subclasses *KWindow* e *Menu*, que referem-se, respectivamente, à ligação entre a shell KAPPA e outros aplicativos do *Windows*, e à lista de opções de escolha.

Na Figura 4.1, como dito, tem-se representada a estrutura hierárquica das classes e instâncias do sistema Poli 1.0. Tal estrutura é criada pelo programador na janela *Object Browser* da maneira como a referida figura apresenta.

### 4.8.2. *Session*

O programador, dentro desta janela, poderá personalizar as telas da interface entre o sistema e o usuário final deste, fazendo uso de gráficos, textos e botões para que esta interface seja a mais ergonômica possível.

Para tanto, faz-se uso principalmente da opção *Layout Mode* (dentro do menu *Options*) para criar a aparência final do sistema. Com esta opção pode-se criar várias telas diferentes que o usuário vai se deparar quando utilizar o sistema.

A entrada de dados no sistema é aqui elaborada, podendo esta ser feita respondendo perguntas, selecionando alternativas apresentadas ou utilizando o *slider* (onde o usuário desliza o cursor até o valor mais próximo da sua resposta).

### 4.8.3. *Edit Tools*

Esta é a janela mais utilizada pelo programador, pois a partir dela são definidos os editores de classes (*Class*), instâncias (*Instance*), funções (*Function*), regras (*Rule*) e metas (*Goal*). Os dois primeiros editores, de classes e instâncias, ainda contém mais três editores internos, o de *slots*, de opções e de métodos. Ao lado de cada editor destes aparece ainda, entre parênteses, o número de itens que cada um possui na aplicação corrente, como pode ser visto na Figura 4.3.

As funções, regras e métodos são então criadas e editadas nesta janela. Cada um destes itens quando criado, deverá ser identificado com um nome que o programador irá escolher, pois quando dentro de um processo de inferência o mesmo tiver que ser acionado, basta indicá-lo por este nome.

Para que o programador tenha acesso às funções pré-definidas pela *shell*, basta, dentro de algum destes itens, clicar com o botão direito do *mouse* e selecionar qual a categoria da função que a ele interessa, por exemplo, se esta é uma função matemática, lógica, outra função já criada pelo programador. Após aparecerão as funções que a shell oferece dentro da referida categoria, bastando o programador selecionar o comando que a ele interessa.



Figura 4.3 – Janela *Edit Tools*.

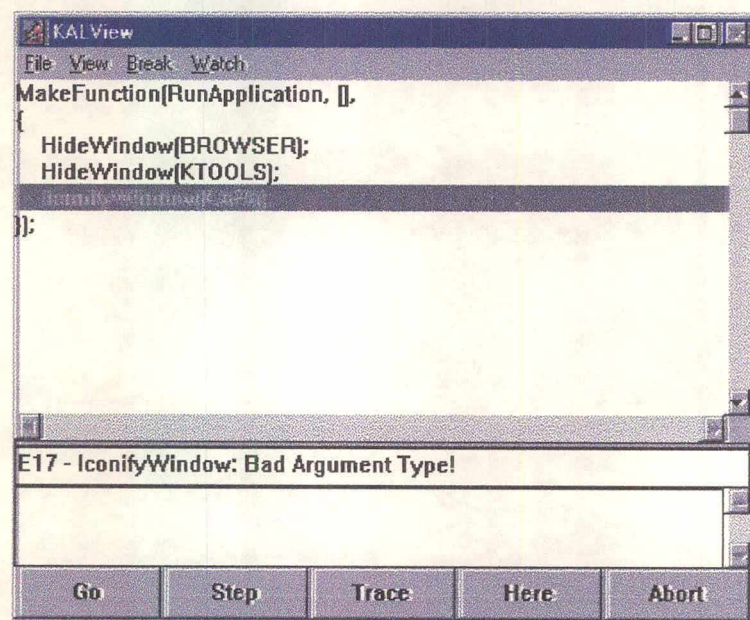
#### 4.8.4. KAL *Interpreter*

Nesta janela e em outras onde se tenha feito uso da linguagem KAL, pode-se utilizar um verificador de sintaxe existente na *shell*, bastando para tanto selecionar *Check Syntax* no menu *Edit*. A resposta será “*Expression OK*” caso não haja nenhum erro de sintaxe, ou então uma mensagem aparecerá especificando qual o tipo de erro cometido e em que linha o mesmo se encontra.

O programador também pode fazer uso da biblioteca de funções da shell KAPPA dentro desta janela, de maneira semelhante a feita na janela *Edit Tools*.

#### 4.8.5. KAL *View Debugger*

A última das cinco janelas consideradas mais importantes, a *KAL View Debugger*, é aquela onde, como já foi dito no início deste texto, o programador pode acompanhar um por um os passos seguidos pelo programa e com isso detectar possíveis erros mais facilmente. Caso exista um erro, de programação ou de lógica, é enviada ao programador a seguinte mensagem de alerta: “*Bad Argument Type*”. Um exemplo desta mensagem de alerta é mostrado na Figura 4.4, como segue:



**Figura 4.4 – Mensagem de alerta do KAL View Debugger.**

Esta *shell*, na versão 2.0, foi escolhida para o presente trabalho pelo fato de a mesma ser desenvolvida em *Windows*, que é um ambiente bastante conhecido e divulgado, tornando assim o sistema mais atraente e conseqüentemente de mais fácil utilização para o futuro usuário. Porém ela apresenta algumas deficiências como, por exemplo, quanto a representação de movimento e quanto a elaboração de funções e métodos com passagem de parâmetros.

Fazendo uso então desta ferramenta computacional, foi desenvolvido um protótipo de Sistema Especialista de apoio ao planejamento de serviços urbanos, o qual será apresentado no capítulo a seguir.

## *O Protótipo Proposto*

Neste capítulo é apresentado o ForPlan, um protótipo de Sistema Especialista de apoio na formação de planejadores de serviços urbanos. A maior parte do conhecimento utilizado na elaboração deste protótipo foi obtida através da literatura, sendo as principais contribuições provenientes de estudos de autores como Larson e Odoni (1981), Eilon *et al* (1971) e Novaes (ver, por exemplo, Novaes (1989) e Novaes (1993)).

### **5.1. Definição do Problema**

Como já foi dito no presente trabalho, no planejamento inicial de serviços urbanos são necessárias medidas de distância para escolher, por exemplo, qual melhor roteiro seguir para ir de um ponto a outro da região. Nesta fase do planejamento não é viável medir distâncias reais e sim utilizar uma estimativa destes valores. O problema para o planejador está então em como obter uma boa estimativa para tal.

Na bibliografia existente, algumas situações específicas são abordadas, mas de maneira pouco didática para planejadores em formação, já que os resultados obtidos por estes autores muitas vezes são apenas indicados, deixando o desenvolvimento do cálculo para encontrar os mesmos como exercício para o leitor. Além do mais, a complexidade dos cálculos utilizados para obtenção destes resultados exige que o planejador tenha conhecimento de cálculo matemático, diferencial e integral, bem como noções de probabilidade.

Devido estas dificuldades encontradas principalmente por planejadores iniciantes, surgiu a proposta da construção de um protótipo de Sistema Especialista que reunisse parte deste conhecimento, sendo este conhecimento repassado do protótipo ao planejador de formas simples e clara.

Dada a grande quantidade de combinações entre a forma da região, a localização da unidade de serviço e o tipo de barreira de viagem existente, algumas restrições foram feitas para facilitar o trabalho do engenheiro de conhecimento. Com isso, o caso onde o espaço é contínuo e a região e a métrica são retangulares apresenta resultados mais especificados do que em outras situações (isto se dá devido à maior facilidade de cálculo nestes casos).

## 5.2. Objetivos do Protótipo

O protótipo de Sistema Especialista ForPlan tem como objetivo principal servir como consultor para planejadores em formação. Através da interação com o ForPlan, o usuário poderá identificar as características relevantes para a resolução do seu problema, já que o mesmo fornece explicações sobre as respostas obtidas e sobre algumas perguntas dirigidas ao mesmo.

O protótipo ainda tem como finalidade indicar, quando possível, valores importantes para o planejamento inicial de serviços urbanos, como uma estimativa da distância esperada de viagem e um coeficiente de correção.

## 5.3. Recursos Necessários

Quanto à equipe de desenvolvimento do protótipo, a autora participou como engenheiro de conhecimento e teve participação também na fase de implementação, sendo nesta última fase auxiliada pelo acadêmico de Ciências da Computação da Universidade Federal de Santa Catarina André Antunes Meurer.

Quanto ao equipamento utilizado para implementação do protótipo, foi utilizado um microcomputador PC rodando Windows 95, munido da *shell* KAPPA 2.0. Para utilização do protótipo pelo usuário, o mesmo deve ter um micro com configuração semelhante (já que o auto executável desta *shell* ainda não está disponível).

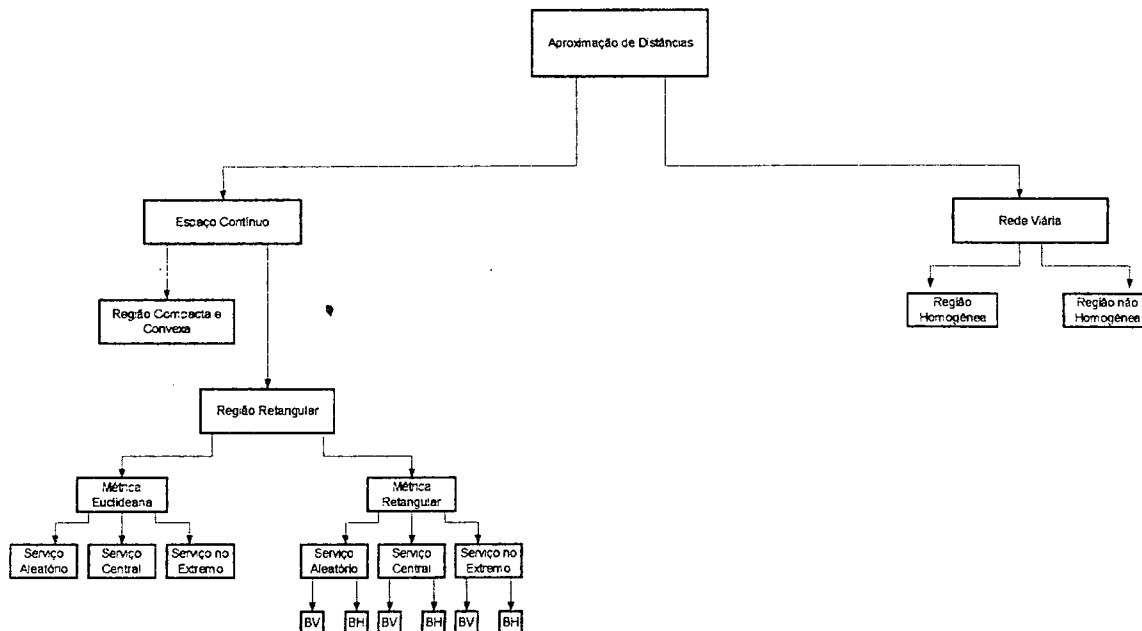
## 5.4. Estrutura do Protótipo

O protótipo ForPlan foi desenvolvido em dois módulos, quais sejam, o módulo da Introdução e o de Aproximação de Distâncias. O usuário deverá logo no início optar por iniciar pela introdução ou se já partirá direto para a estimativa dos valores desejados.

No módulo da Introdução apresenta-se uma revisão quanto às características da região de estudos que serão relevantes no processo de estimativa da distância e do coeficiente de correção. Esta revisão é apresentada através de textos explicativos e figura ilustrando alguns casos que serão abordados no módulo seguinte. Algumas telas da introdução serão mostradas mais adiante neste capítulo.

No módulo de Aproximação de Distâncias são feitas perguntas que devem ser respondidas pelo usuário através de entrada de dados ou seleção de alternativas. O protótipo é então capaz, de acordo com as respostas dadas, de indicar as estimativas desejadas ou de sugerir ao planejador outra maneira de encontrar tais valores.

Através da Figura 5.1 pode-se visualizar como foi estruturado o módulo de aproximação de distâncias.



**Figura 5.1 - Estrutura do módulo Aproximação de Distâncias**

Primeiramente o usuário (leia-se o planejador em formação) deve optar por trabalhar sobre um espaço contínuo, onde qualquer movimento para se deslocar de um ponto a outro é permitido, ou sobre a grade viária, onde os movimentos ficam restritos às vias de tráfego disponíveis.

Se a opção for pela grade viária, ele tem mais uma escolha a fazer, essa quanto ao tipo da região. Se a opção for por uma região homogênea, então o protótipo é capaz de indicar uma estimativa para a distância viajada entre dois pontos da grade. Se a opção for por uma região não homogênea, então o sistema sugere que o usuário colete dados reais e faça um ajuste por regressão linear, tal como em Novaes (1993).

Já se a opção inicial tenha sido o espaço contínuo, o usuário também terá que informar a topologia da região, podendo optar por considerar uma região retangular ou uma região compacta e convexa. Se a escolha for pela segunda alternativa, então o protótipo indica uma estimativa da distância que depende da área da região e um coeficiente de correção, tal como foi apresentado na sessão 2.5 deste trabalho.

Se a opção for pela região retangular, como já foi dito neste texto e pode-se verificar através da Figura 5.1, os resultados podem ser mais precisos, tendo o usuário que fornecer maiores informações quanto ao problema que o mesmo deseja resolver.

Neste caso, a escolha seguinte é quanto a métrica a ser adotada para estimar a distância, podendo esta ser a métrica euclidiana ou a métrica retangular. Caso a primeira tenha sido escolhida, então basta o usuário informar a localização da unidade de serviço, se esta é aleatória, central ou localizada num extremo da região. Estando esta localizada no centro ou num extremo, o sistema fornece uma estimativa da distância esperada de viagem, indicando um coeficiente de correção aproximado de 1,3 (ver experimentos feitos por Novaes



(1989)). Caso o serviço esteja aleatoriamente posicionado na região, o protótipo novamente sugere uma coleta de dados para ajustar uma reta de regressão para então obter um coeficiente de correção adequado, tal como indica Eilon (1971).

Se a métrica escolhida para estimar a distância tenha sido a retangular, o usuário além de informar a localização da unidade de serviço deve informar também se existe ou não barreiras de viagem. Em caso afirmativo o mesmo deve informar ainda se esta é vertical ou horizontal, representadas por BV e BH respectivamente na Figura 5.1, e escolher posteriormente a configuração da mesma (de acordo com as figuras 2.7 e 2.8). Dadas estas informações, o protótipo tem condições de apresentar ao usuário uma estimativa mais precisa da distância esperada da viagem entre o ponto de demanda e a unidade de serviço.

## **5.5. Representação do Conhecimento**

Como dito no início deste capítulo, as regras, funções e métodos que compõem a base de conhecimentos do sistema baseiam-se em resultados obtidos pelos autores referenciados, tal como foi apresentado detalhadamente no segundo capítulo.

Na fase de estruturação do protótipo, o engenheiro do conhecimento tem como principal tarefa identificar e representar o conhecimento de maneira clara para que seja possível implementá-lo do modo mais simples possível.

Como foi apresentado no capítulo 3 através da Figura 3.1, um Sistema Especialista é composto por três módulos, quais sejam, Base de Conhecimentos, Máquina de Inferência e Interface. Aqui serão detalhados o primeiro e terceiro módulo, pois a máquina de inferência já está incorporada na *shell* KAPPA.

### **5.5.1. A Base de Conhecimentos**

A base de conhecimentos de um Sistema Especialista é composta por uma memória de trabalho, onde são armazenados os fatos relevantes, e por uma base de regras, onde são explicitadas as relações entre os fatos.

A memória de trabalho do protótipo ForPlan é composta por vinte e dois (22) *slots* (pertencentes à única classe definida, chamada Problema). Destes *slots*, alguns são entradas de dados (ED) ou seleção de alternativas (SA) feitas pelo usuário durante a utilização do protótipo. Sendo uma entrada de dados, o *slot* pode assumir valores numéricos diversos, enquanto que se for uma seleção de alternativas, este pode assumir apenas valores referentes às alternativas dadas pelo sistema.

Outros *slots* são usados somente para facilitar a estruturação da base de regras. Estes *slots* internos (SI) geralmente assumem os valores verdadeiro (TRUE) e falso (FALSE).

Há ainda aqueles *slots* que, de acordo com as informações que o usuário fornecer ao sistema, serão dados como resposta (RS) para o problema que se está resolvendo com auxílio do protótipo.

No Quadro 5.1 são listados estes *slots*, informando o tipo (de acordo com a classificação apresentada acima) e os possíveis valores que estes podem assumir.

Slot	Tipo	Valores que este pode assumir
A	ED	No. Real (coordenada da barreira)
a	ED	No. Real (tamanho do quarteirão)
$a_0$	ED	No. Real (área da região)
área	SI	TRUE, FALSE
B	ED	No. Real (coordenada da barreira)
barreiras	SI	TRUE, FALSE
E.D.	RS	No. Real (valor esperado da distância)
homogênea	SA	homogênea, não homogênea
m	RS ou ED	No. Inteiro (No. vias horizontais)
métrica	SA	retangular, euclideana
n	RS ou ED	No. Inteiro (No. vias verticais)
redeHomogênea	SI	TRUE, FALSE
temBarreira	SI	TRUE, FALSE
temDimensões	SI	TRUE, FALSE
temLocalBarreira	SI	TRUE, FALSE
tipoDeBarreira	SA	horizontal, vertical
tipoDeEspaço	SA	contínuo, rede viária
tipoDeRegião	SA	retangular, compacta e convexa
unidadeDeServiço	SA	aleatória, central, no extremo
valores	SA	i), ii), iii)
$x_0$	ED	No. Real (dimensão da região)
$y_0$	ED	No. Real (dimensão da região)

**Quadro 5.1 - Slots do ForPlan.**

A base de regras do ForPlan é composta por trinta e três regras, cento e quarenta funções e onze métodos. Para chegar a todos estes, primeiramente estruturou-se toda a base de conhecimentos em regras de produção da forma SE (condições) ENTÃO (conclusões), pois deste modo reproduzem fielmente a maneira que o ser humano raciocina.

Para serem implementadas em linguagem KAL, estas regras iniciais sofreram algumas mudanças e divisões para tornar o trabalho mais fácil para o implementador bem como para tornar o processo de inferência mais simplificado. Com isso, a base de regras do protótipo Forplan ficou sendo composta por trinta e três regras, cento e quarenta funções e onze métodos.

Em nível de ilustração, apresenta-se as regras *ÁREACC1* e *RedeViária3* que compõem a base de conhecimentos do ForPlan, como esta foi escrita em linguagem natural e como está estruturada em linguagem KAL. A primeira destas regras foi mantida, sendo apenas transcrita na linguagem de programação.

### **Regra *ÁREACC1***

#### Linguagem Natural:

Se a métrica é euclideana, a unidade de serviço está posicionada aleatoriamente e o usuário conhece a área da região de estudos,

Então calcular o valor esperado da distância:  $E.D. = 0,52\sqrt{a_0}$ .

#### Linguagem KAL:

```
If {  
( Problema:unidadeDeServico #= Aleatoriamente )  
  And ( Problema:metrica #= Euclídeana );  
};
```

```
Then {  
Problema:E.D. = Sqrt( Problema:a0 )  
  * 0.52;  
};
```

Já a regra *RedeViária3* foi dividida, sendo que o cálculo dos valores de m (número de vias horizontais) e n (número de vias verticais) foi feito separadamente em uma função chamada Aproximação Calcula MN.

### **Regra *RedeViária3***

#### Linguagem Natural

Calcular o número de vias horizontais:  $m = \frac{y_0}{a}$

Calcular o número de vias verticais:  $n = \frac{x_0}{a}$

Se n e m são inteiros

**Então** escreva a mensagem “Com isso você tem (valor de n) vias verticais e (valor de m) vias horizontais”.

#### Linguagem KAL

```
If {  
Integer?( Problema:m ) And Integer?( Problema:n );  
};  
  
Then {  
ClearTranscriptImage( APXtranscript16 );  
DisplayText( APXtranscript16, FormatValue( "Com isso você tem %s vias verticais      e\n",  
Problema:n ) );  
DisplayText( APXtranscript16, FormatValue( "%s vias horizontais.", Problema:m ) );  
SetValue( APXpbContinuar2:Title, Continuar );  
SetValue( APXpbContinuar2:Action, Aproximacao17 );  
};
```

Todas as regras iniciais escritas em linguagem natural do ForPlan estão apresentadas em anexo.

Cabe aqui ressaltar que, havendo alguma alteração ou acréscimo a ser feito na base de conhecimentos, em praticamente nada vão alterar as regras já existentes. Com isso, quando o protótipo for continuado, todas estas regras serão utilizadas da maneira como foram estruturadas.

#### **5.5.2. A Interface com o Usuário**

A interface do protótipo ForPlan com o usuário foi elaborada visando obter um sistema bastante ergonômico, para que o usuário consiga entender claramente as informações que o sistema venha a pedir ou fornecer.

Esta interface é composta por textos explicativos, figuras que representam certas situações e botões. A interação homem - máquina ocorre de maneira bastante amigável. O usuário fornece as informações necessárias através de entrada de dados, como quando o sistema pede as dimensões da região de estudos, ou através da seleção de alternativas, como quando o usuário deve escolher qual a métrica adotar para estimar a distância. Já o sistema responde fornecendo estimativas e as fórmulas de onde estas foram derivadas, ou ainda, sugerindo outros caminhos a serem seguidos quando se torna inviável calcular tais valores.

O protótipo permite, através de um botão de Voltar que o usuário retorne às telas anteriores e modifique valores assumidos ou opções antes escolhidas sem prejudicar em maneira nenhuma a continuidade do sistema. O ForPlan fornece ainda ao usuário a possibilidade de interrupção do sistema em qualquer momento.

De acordo com uma das principais características de um Sistema Especialista, a capacidade de explanação, o ForPlan é capaz de enviar mensagens de erro claras e

informativas para que os mesmos erros não tornem a se repetir. Quando, por exemplo, o usuário entra com o número de vias horizontais ou verticais de uma rede viária ou mesmo quando o sistema calcula tais valores e estes não são números inteiros, o protótipo envia uma mensagem de alerta para o usuário informando que estes devem ser números inteiros.

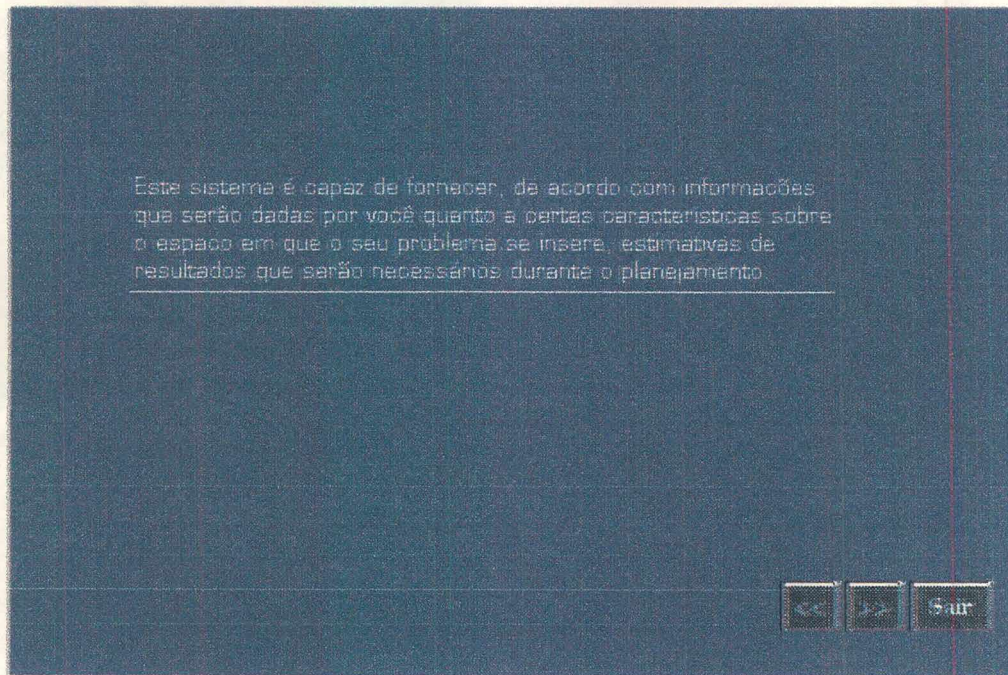
A seguir serão mostradas algumas telas do protótipo ForPlan com intuito de ilustrar como este será apresentado, mesmo que preliminarmente, ao planejador em formação.

A tela inicial do ForPlan apenas apresenta o sistema e seu autor, como mostra a Figura 5.2 abaixo.



**Figura 5.2 - Tela inicial do ForPlan.**

A partir daí algumas informações sobre o sistema são esclarecidas, como os seus objetivos e os dados necessários para a utilização do mesmo. A Figura 5.3. mostra parte desta explicação:

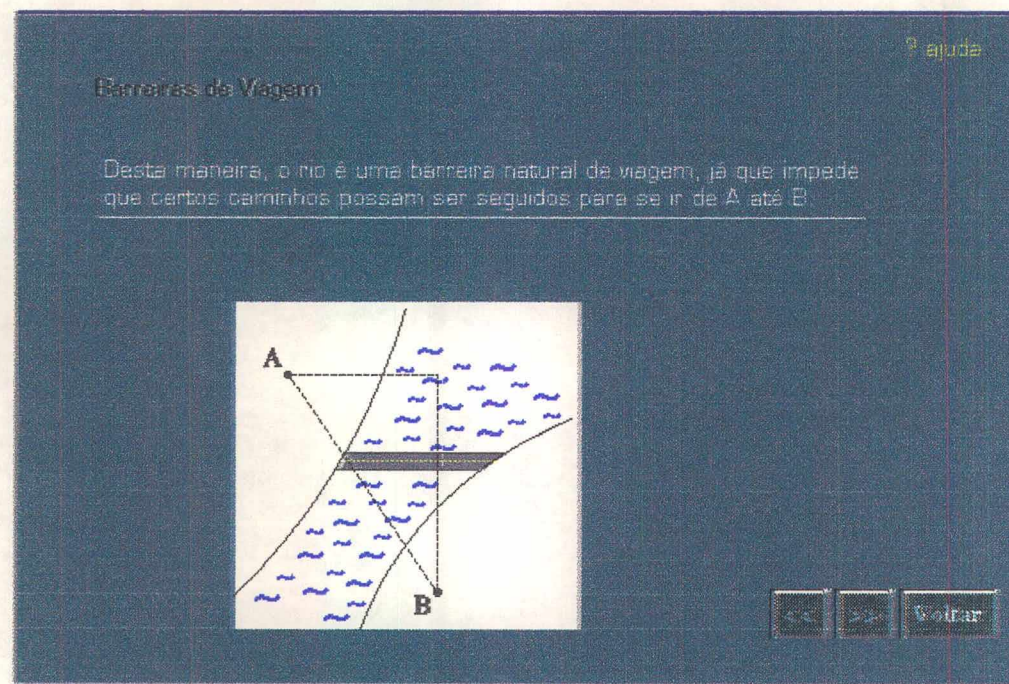


**Figura 5.3 - Apresentação do ForPlan.**

Depois de ser informado sobre as capacidades do protótipo, o usuário deve escolher, de acordo com seu grau de conhecimento sobre o assunto que está se estudando, qual o módulo irá seguir, se o módulo Introdução ou o módulo Aproximação de Distâncias. Ambos os módulos foram detalhados no início deste capítulo e portanto são considerados conhecidos do leitor.

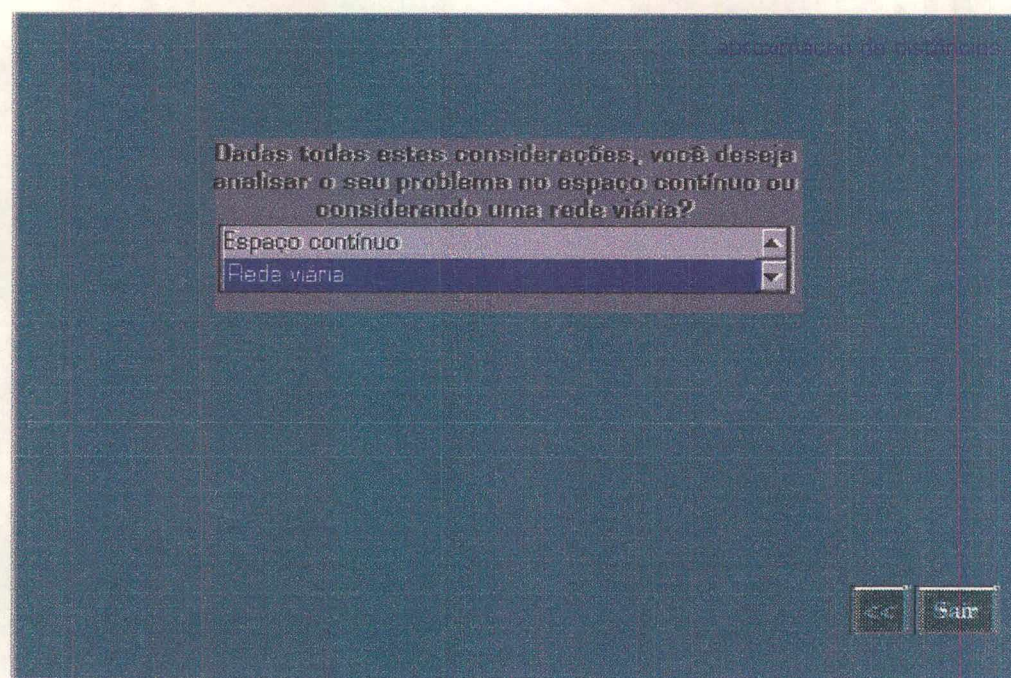
Caso a escolha tenha sido iniciar pelo módulo da introdução, novos textos explicativos são apresentados, bem como teclas de ajuda que explicam mais detalhadamente as escolhas que o usuário deve fazer. São mostradas também figuras que ilustram situações específicas.

Através da Figura 5.4 pode-se ver uma destas telas de ajuda, esta referente a uma parte da explicação do que vem a ser uma barreira de viagem e como esta afeta a estimativa da distância.



**Figura 5.4 - Ajuda sobre barreiras de viagem.**

Já no módulo de aproximação de distâncias, uma primeira escolha que o usuário deve fazer se dá quanto ao espaço em que o problema se encaixa. Esta escolha se dá através da seleção de alternativas, como segue:



**Figura 5.5 - Escolha sobre o espaço.**

Escolhendo o espaço como sendo a rede viária, uma figura bastante esclarecedora que aparece para o usuário é a que representa os valores que este precisa conhecer, como segue:

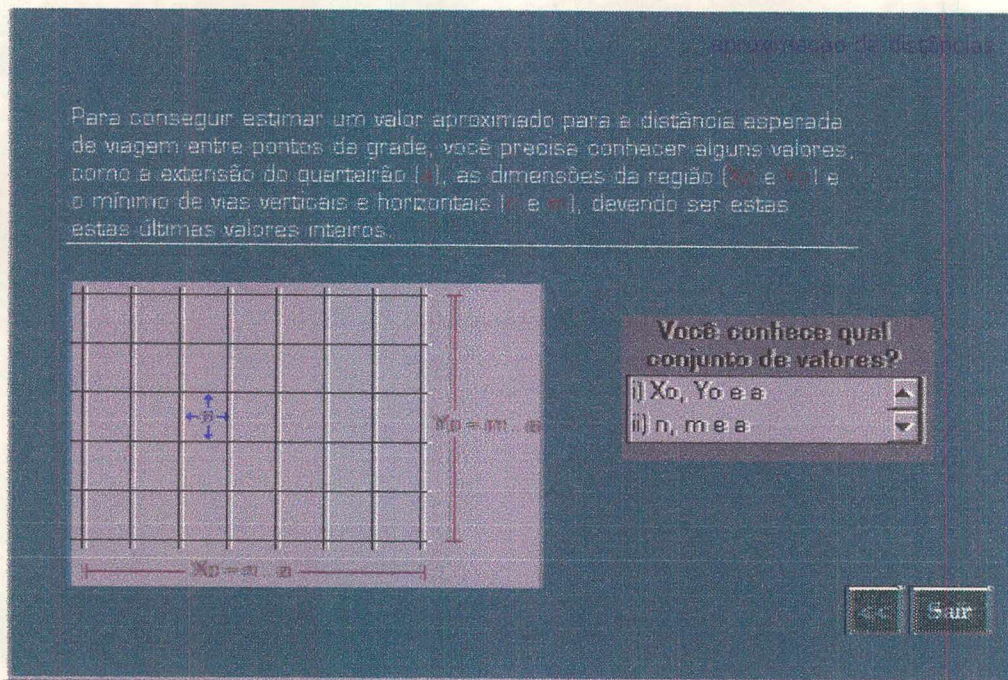


Figura 5.6 - Ilustração da rede viária.

Em seguida, uma entrada de dados que aparece ao usuário é a seguinte:

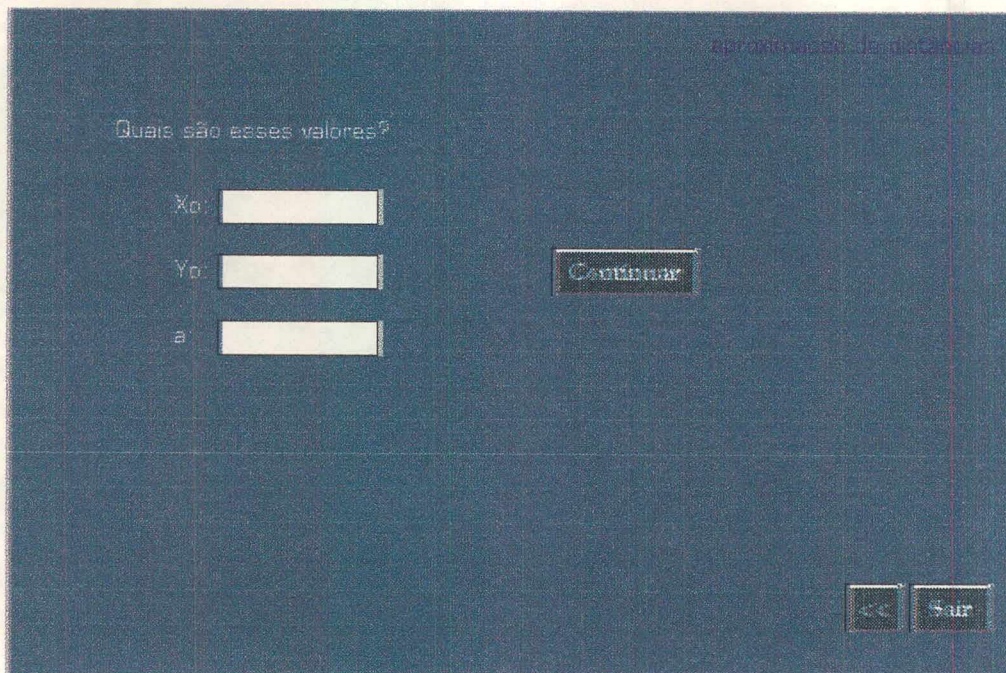
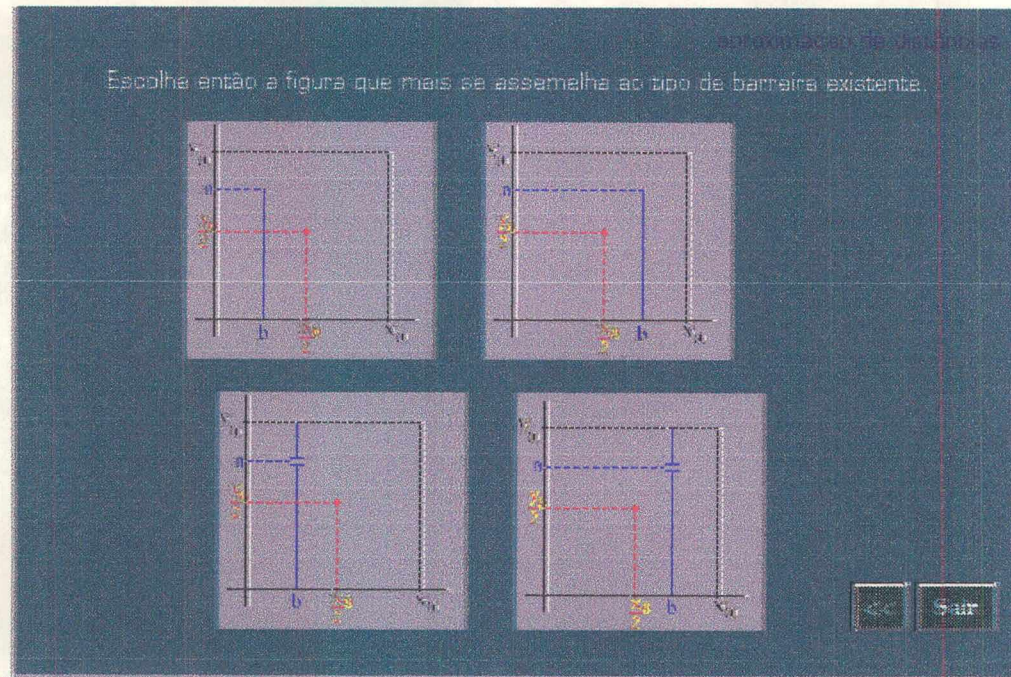


Figura 5.7 - Entrada de dados referente à rede viária.



Quando o espaço é contínuo e a forma da região e a métrica a ser adotada para estimar distâncias são retangulares, têm-se resultados mais precisos para diversas situações. Por exemplo, quando da existência de uma barreira de viagem vertical e a unidade prestadora de serviço está posicionada no centro da região de estudos, aparecem as seguintes figuras para o usuário escolher qual a configuração que mais se assemelha com seu problema real. A Figura 5.8 mostra esta interface:

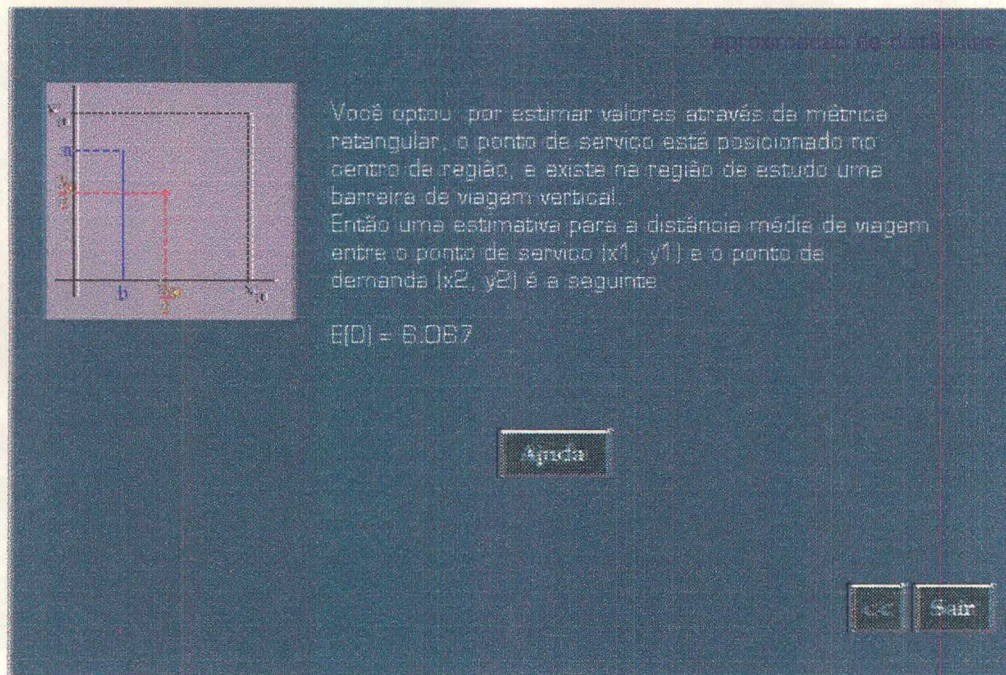


**Figura 5.8 - Escolha quanto à configuração da barreira de viagem**

Se o usuário opta, por exemplo, por considerar a primeira configuração para a barreira, ele deve clicar o mouse sobre a figura e esta fica a mostra até o processo de estimativa ser finalizado.

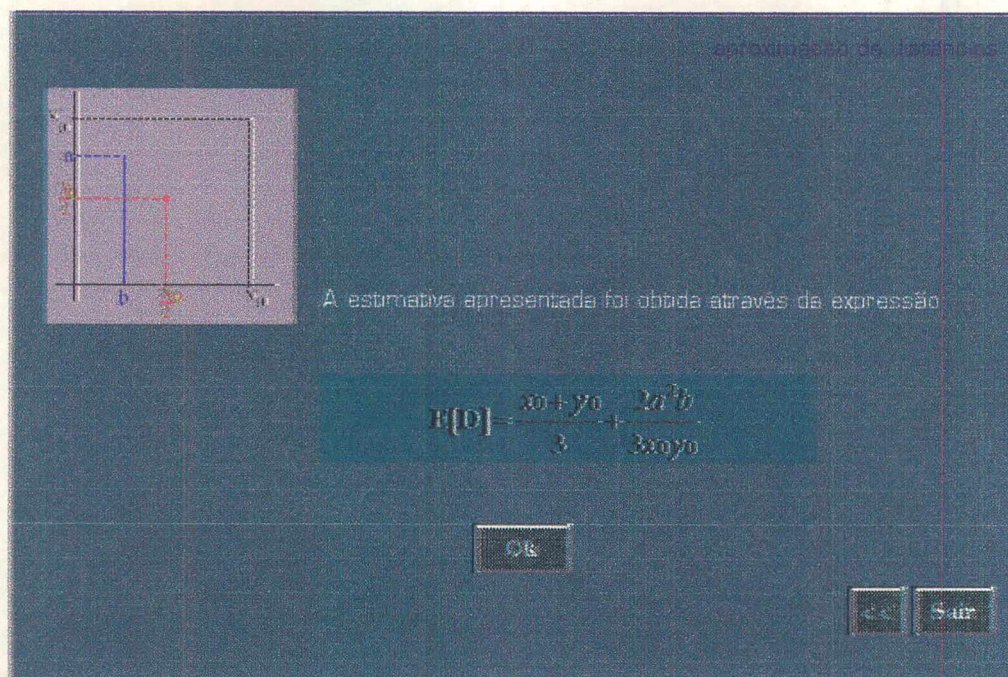
Em seguida o sistema pede informações quanto às coordenadas da barreira. Caso o usuário as conheça, estas devem ser informadas com uma nova entrada de dados.

A resposta final fornecendo uma estimativa para a distância esperada de viagem entre os pontos numa certa região de estudos, é apresentada ao usuário como mostra a Figura 5.9 a seguir:



**Figura 5.9 - Resposta dada pelo sistema.**

Caso o usuário deseje saber de onde foi derivado o valor dado como resposta, o mesmo pode acessar a tecla de ajuda que o sistema imediatamente apresenta a fórmula relativa a esta situação. Esta fórmula, como mostra a Figura 5.10, também é apresentada ao usuário quando este diz desconhecer os dados necessários para estimar o valor desejado.



**Figura 5.10 - Fórmula de onde foi derivado o resultado.**

As demais situações que o sistema contempla são apresentadas de forma análogo às que foram aqui exibidas, sempre fazendo uso de textos explicativos e figuras ilustrativas.

Para finalizar esta explanação sobre o protótipo de Sistema Especialista ForPlan proposto nesta dissertação, cabe aqui um comentário. Da mesma maneira que para as regras da base de conhecimentos, o *layout* das telas de interação com tamanho ou tipo da fonte, cores de fundo e posicionamento de botões e textos, podem facilmente ser modificados. Logo, para a elaboração final do Sistema Especialista, se algumas mudanças tiverem que ser feitas para melhorar ou enriquecer a interface e tornar a relação homem - máquina o mais natural possível, estas podem ser realizadas de maneira bastante simples e rápida.

Quanto à continuidade do protótipo, sugestões para melhorias e possíveis trabalhos futuros serão apresentados no capítulo a seguir.

## *Conclusões e Perspectivas*

Na seqüência serão analisadas as vantagens trazidas ao planejador em formação com a utilização do protótipo ForPlan desenvolvido no presente trabalho. Inicialmente será apresentado de forma sucinta um resumo do trabalho desenvolvido nesta dissertação. Posteriormente serão levantadas algumas propostas de continuidade e aperfeiçoamento deste trabalho.

### **6.1. Resumo e Contribuições do Trabalho**

Com esta dissertação propunha-se a construção de um protótipo de sistema Especialista de apoio na formação de serviços urbanos. Como principal objetivo desejava-se fornecer ao planejador iniciante uma ferramenta de apoio que auxiliasse de maneira simples e clara a obter estimativas de valores essenciais para o planejamento inicial de certo serviço.

Para alcançar este objetivo foi necessário inicialmente uma pesquisa bibliográfica quanto aos conceitos de medidas geométricas de distância como sobre os métodos existentes para obtenção de estimativas da distância média de viagem entre pontos da região de estudos e do coeficiente de correção nas mais diversas situações.

Com isso, uma primeira e importante contribuição deste trabalho foi quanto a reunião de resultados obtidos por autores diversos para estimar tais valores. Tais resultados foram apresentados no capítulo 2 deste trabalho.

Considerando a situação específica onde a região é retangular, a métrica adotada para medir distâncias é retangular e a unidade de serviço está localizada num extremo da região, foi desenvolvido um estudo do efeito de uma barreira de viagem inclinada sobre a distância esperada de viagem entre a unidade de serviço e o ponto de demanda. Utilizando probabilidade geométrica e os métodos adotados pelos autores pesquisados, conseguiu-se um resultado bastante satisfatório e generalizado, já que barreiras de viagem verticais e barreiras de viagem horizontais, cujos efeitos sobre a distância foram apresentados pelos referenciados autores, são na verdade casos especiais de barreiras de viagem inclinadas.

Antes de passar à fase de implementação do protótipo proposto inicialmente, foi necessário introduzir os conceitos básicos de Inteligência Artificial e Sistemas Especialistas, como estes são desenvolvidos, suas diferenças para com a programação convencional e as

principais áreas de aplicação destes sistemas. Foi necessário também fazer uma detalhada exposição da *shell* KAPPA, a ferramenta utilizada nesta fase de desenvolvimento do Sistema Especialista. Nesta exposição foi apresentado como representar o conhecimento através de regras, funções e métodos, como caracterizar os objetos através dos *slots* e, principalmente, como a *shell* se apresenta para o programador, mostrando quais as ferramentas ela proporciona para o mesmo desenvolver seu trabalho mais facilmente. Estes assuntos quanto a Inteligência Artificial e Sistemas Especialistas e a *shell* KAPPA foram apresentados respectivamente nos capítulos 3 e 4 deste trabalho.

A estruturação e construção do protótipo proposto, a maneira como foram elaboradas as regras e funções e como o conhecimento armazenado ou inferido foi passado ao usuário (neste caso o planejador) foram apresentados com maiores detalhes no quinto capítulo. Uma árvore representando um dos módulos do protótipo serviu para visualizar como o mesmo foi estruturado e quais as situações específicas o sistema tem condições de abordar, já que algumas restrições foram feitas devido à complexidade dos cálculos na obtenção das estimativas.

Exemplos de regras e funções também foram apresentados para mostrar como foi representado o conhecimento e como este deve ser passado ao usuário. Telas do protótipo foram acrescentadas ao texto para mostrar como ocorre a interação entre o sistema e o planejador, que esta é amigável e se dá através de linguagem natural e figuras, facilitando assim o entendimento do problema a ser resolvido por um planejador muitas vezes inexperiente.

Resumidamente então, podem ser destacadas as seguintes contribuições do presente trabalho:

- levantamento das características relevantes para a estimativa da distância esperada de viagem e do coeficiente de correção;
- levantamento dos resultados obtidos pelos autores referenciados no texto para estas estimativas em configurações específicas da rede de transporte;
- desenvolvimento de um estudo referente ao efeito de barreiras de viagem inclinadas sobre a distância esperada de viagem em uma configuração específica da rede;
- entendimento e aplicação de importantes conceitos de Inteligência Artificial e de Sistemas Especialistas;
- apresentação da *shell* KAPPA, uma ferramenta própria para a construção de Sistemas Especialistas, sua estrutura e vantagens para com outras ferramentas e
- desenvolvimento de um instrumento de apoio para o planejador em formação que indique valores importantes para o planejamento inicial de serviços urbanos, visando o aprendizado do mesmo.

## **6.2. Propostas de Continuidade de Trabalho e Perspectivas Futuras**

Uma primeira possibilidade de continuação deste trabalho se refere ao estudo mais aprofundado dos efeitos das barreiras de viagem inclinadas sobre a distância esperada de viagem. Como feito neste trabalho para uma configuração específica, propõe-se fazer o mesmo estudo para diversas situações, como quando a unidade de serviço está localizada no centro ou está aleatoriamente posicionada em uma região retangular.

Uma outra possibilidade de continuidade do trabalho apresentado nesta dissertação consiste no aperfeiçoamento do protótipo ForPlan, acrescentando ao mesmo a possibilidade de o usuário fazer ajustes através de regressões, como o próprio protótipo sugere em situações especiais onde não é viável desenvolver cálculos bastante extensos para estimar valores. Ainda, como aprimoramento do protótipo, deseja-se acrescentar ao mesmo um exemplo resolvido com dados reais que sirva de referência para planejadores iniciantes entenderem como o sistema funciona.

Finalmente, como perspectiva de um trabalho futuro, propõe-se a validação do protótipo ForPlan, levando-o a ser utilizado por vários planejadores em formação como, por exemplo, alunos de pós-graduação do PPGEPS na área de Transporte e Logística. A partir dos resultados obtidos por estes e das críticas e sugestões que forem apresentadas, poderão ser feitas pequenas alterações no protótipo para que o mesmo satisfaça as necessidades dos usuários.

## ***Referências Bibliográficas***

- (01) BARR, A.; FEIGENBAUM, E. A. **The Handbook of Artificial Intelligence**. USA: Addison - Wesley Publishing Company, 1986.
- (02) CORREIA, F.; SIQUEIRA, K.; PALADINI, C. Aplicações de POLI 1.0 - Um Sistema que Classifica Polígonos. Anais do V Seminário Catarinense de Iniciação Científica, p 414, Florianópolis: 1995a.
- (03) CORREIA, F.; FLEMMING, D.; GONÇALVES, M.; PEREIRA, M.; SIQUEIRA, K.; PALADINI, C. Sistemas Especialistas - Um recurso didático para o ensino de matemática. Anais do VI Simpósio Brasileiro de Informática na Educação, pp 351-362, Florianópolis: 1995b.
- (04) EILON, S.; WATSON-GANDY, C. D. T.; CHRISTOFIDES, N. **Distribution Managements Mathematical Modelling and Practical Analysis**. London: Griffin, 1971.
- (05) HAYES-ROTH, F.; WATERMAN, D.; LENAT, D. B. **Building Expert Systems**. Vol.1. USA: Addison - Wesley Publishing Company, 1983.
- (06) KAPPA **User's Guide**. New York: IntelliCorp Inc., 1991.
- (07) KAPPA **Reference Manual**. New York: IntelliCorp Inc., 1991.
- (08) LARSON, R.; ODONI, A. R. **Urban Operations Research**. London: Prentice-Hall, 1981.
- (09) NOVAES, A. G. **Relações Métricas e Aproximações em Redes Logísticas**. VII ANPET, Congresso de Pesquisa e Ensino em Transportes, vol. 2, pp 579-591, São Paulo: 1993.
- (10) \_\_\_\_\_. **Sistemas Logísticos: Transporte, Armazenagem e Distribuição Física de Produtos**. São Paulo: Editora Edgar Blucher Ltda., 1989.
- (11) \_\_\_\_\_. **Using Geographical Information Systems in Conjunction with Vehicle Routing Problems**. IFORS 96, The International Federation of Operational Research Societies. Canadá: 1996.
- (12) PEDERSEN, K. **Expert Systems Programming - Pratical Techniques for Rule - Based Systems**. John Wiley & Sons Inc. New York: 1989.
- (13) RABUSKE, R. A. **Inteligência Artificial**. Florianópolis: Editora da UFSC, 1995.

- (14) RIBEIRO, H. C. S. **Introdução aos Sistemas Especialistas**. Rio de Janeiro: Livros Técnicos e Científicos Editora AS, 1987.
- (15) RICH, E. **Inteligência Artificial**. São Paulo: McGraw Hill, 1988.
- (16) SCHWABE, D.; DE CARVALHO, R. L. **Engenharia de Conhecimento e Sistemas Especialistas**. Argentina: Editorial KAPELUSZ S. A., 1987.
- (17) SIMON, H. A. **As Ciências do Artificial**. Coleção STVDIVM, 2ª. Edição. Coimbra: Armínio Amado Editor Sucessor, 1981.
- (18) SIMONS, G. L. **Introdução à Inteligência Artificial**. Clássica Editora, 1984.
- (19) SIQUEIRA, K. C.; GONÇALVES, M. B. **Distância em Redes de Transporte: Um Enfoque Usando Sistemas Especialistas**. Anais do XI ANPET, Congresso de Pesquisa e Ensino em Transportes, Dissertações em Andamento, pp 53-57, Rio de Janeiro: 1997a.
- (20) SIQUEIRA, K. C.; GONÇALVES, M. B. **Aproximação de Distâncias - Um Enfoque Matemático**. Anais do CNMAC, Congresso Nacional de Matemática Aplicada e Computacional, Gramado, 1997b.
- (21) WATERMAN, D. A. **A Guide to Expert Systems**. USA: Addison - Wesley Publishing Company Inc., 1986.



## ***Bibliografia Complementar***

- (01) BELTRAMI, E. J. **Models for Public Systems Analysis**. London: Academic Press, 1977.
- (02) BRASILEIRO, A. ; FERREIRA, S. **REENGE & Ensino de Transportes: O Caso da UFPE**. Revista Transportes, Vol. 4, No. 1 e No. 2, pp. 107-113, Novembro, 1996.
- (03) BRIMBERG, J. ; LOVE, R. F. **A New Distance Function for Modeling Travel Distances in a Transportation Network**. Transportation Science, Operations Research Society of America, Vol. 26, No. 2, Maio, 1992.
- (04) CALFAT, R. A. **Um Sistema Especialista de Apoio na Seleção de Técnicas de Sequenciação**. Dissertação de Mestrado. São Paulo: Escola Politécnica da USP, 1990.
- (05) CHORAFAS, D. N. **Sistemas Especialistas**. São Paulo: McGraw Hill, 1988.
- (06) GENARO, S. **Sistemas Especialistas - O Conhecimento Artificial**. Rio de Janeiro: Livros Técnicos e Científicos S.A., 1987.
- (07) JAMES, M. **Inteligência Artificial em Basic**. 2ª Edição. Rio de Janeiro: Editora Campus, 1986.
- (08) KASTNER, J. K.; HONG, S. J. **A Review of Expert Systems**. European Journal of Operational Research, No. 18, pp 285-292, 1984.
- (09) LARSON, R. **Urban Police Patrol Analysis**. Massachusetts: Kingsport Press, 1972.
- (10) LEAL, J. E.; MARTINS, J. A. **Questões para a definição de uma política de pesquisa em transportes no Brasil**. Revista Transportes, Vol. 5, No. 1, pp. 31-54, Maio, 1997.
- (11) LEVINE, R.; DRANG, D.; EDELSON, B. **Inteligência Artificial e Sistemas Especialistas - Aplicações e Exemplos Práticos**. São Paulo: McGraw - Hill, 1988.
- (12) LOVE, R.; MORRIS, J.; WESOLOWSKYY, G. **Facilities Location**. New York: Elsevier Science Publishing Co. Inc., 1988.
- (13) NAYLOR, C. **Building Your Own Expert System**. New York: Halsted Presss, 1985.

- (14) NOVAES, A. G.; ALVARENGA A. C. **Logística Aplicada - Suprimento e Distribuição Física**. 2<sup>a</sup>. Edição. São Paulo: 1994.
- (15) NOVAES, A. G. **Using Geographical Information Systems in Conjunction with Vehicle Routing Problems**. IFORS 96, The International Federation of Operational Research Societies. Canadá: 1996.
- (16) PASSOS, E. L. **Inteligência Artificial e Sistemas Especialistas**. São Paulo: LTC Editora, 1989.
- (17) PEETERS, D.; RUGHIGIRA, J. D. **Using The Transportation Problem To Measure The Proximity Between Two Sets Of Weighted Points In A Geographical Space**. The Journal of the RSAI, Regional Science Association International, Vol. 72, Número 3, 1993.
- (18) SCHANK, R. C. **Where'S the AI?** The AI Magazine, Vol. 12, 1991.
- (19) SIMON, H. A.; NEWELL, A. **Human Problem Solving**. New Jersey: Prentice-Hall, 1972.
- (20) WEISS, S. M.; KULIKOWSKI, C. A. **A Practical Guide to Designing Expert Systems**. USA: Rowman & Allanheld, 1984.

## ***Anexo***

## 1. Regras para uma região compacta e convexa

### Área CC0

Se o usuário desconhece a área da região de estudos,

**Então** escreva a mensagem:

“Uma estimativa para a distância esperada entre pontos da região é:  $E.D. = k\sqrt{a_0}$ , onde  $k$  é um coeficiente de correção que depende da localização da unidade de serviço e da métrica adotada, como mostra o quadro abaixo:

		Regiões Compactas e Convexas
Unidade de Serviço Aleatória	Métrica euclideana	0.52
	Métrica retangular	0.67
Unidade de Serviço Central	Métrica euclideana	0.38
	Métrica retangular	0.50

### Área CC1

Se a métrica é euclideana, a unidade de serviço está posicionada aleatoriamente e o usuário conhece a área da região de estudos,

**Então** calcular o valor esperado da distância:

$$E.D. = 0,52\sqrt{a_0} .$$

### Área CC2

Se a métrica é retangular, a unidade de serviço está posicionada aleatoriamente e o usuário conhece a área da região de estudos,

**Então** calcular o valor esperado da distância:

$$E.D. = 0,67\sqrt{a_0} .$$

### Área CC3

Se a métrica é euclideana, a unidade de serviço está posicionada no centro e o usuário conhece a área da região de estudos,

**Então** calcular o valor esperado da distância:

$$E.D. = 0,38\sqrt{a_0} .$$

## Área CC4

Se a métrica é retangular, a unidade de serviço está posicionada no centro e o usuário conhece a área da região de estudos,

**Então** calcular o valor esperado da distância:

$$E.D. = 0,50\sqrt{a_0}.$$

## **2. Regras para uma rede viária**

### Rede Viária 0

Se o usuário desconhece os conjuntos de valores  $X_0$ ,  $Y_0$  e  $a$  ou  $n$ ,  $m$  e  $a$ ,

**Então** escreva a mensagem:

“Então você deve considerá-los genericamente, e com isso, a expressão para calcular uma estimativa da distância esperada entre pontos é:

$$E[d_G] = \frac{a(n+m)}{3} + \frac{4anm(n+1)(m+1)}{3[(n+1)m + (m+1)n]^2}$$

### Rede Viária 1

Calcular o número de vias horizontais:  $m = \frac{y_0}{a}$

Calcular o número de vias verticais:  $n = \frac{x_0}{a}$

Se  $n$  e  $m$  são nulos,

**Então** escreva a mensagem:

“Lembre-se que  $m$  e  $n$  devem ser inteiros”

Voltar para a tela de entrada de dados.

### Rede Viária 2

Calcular o número de vias horizontais:  $m = \frac{y_0}{a}$

Calcular o número de vias verticais:  $n = \frac{x_0}{a}$

Se  $n$  e  $m$  não são números inteiros,

**Então** escreva a mensagem:

“Lembre-se que  $m$  e  $n$  devem ser inteiros”

Voltar para a tela de entrada de dados.

### Rede Viária 3

Calcular o número de vias horizontais:  $m = \frac{y_0}{a}$

Calcular o número de vias verticais:  $n = \frac{x_0}{a}$

Se n e m são inteiros,

**Então** escreva a mensagem:

“Com isso você tem (valor de n) vias verticais e (valor de m) vias horizontais”.

## **3. Regras para um espaço contínuo**

### **3.1. Quando não existem barreiras de viagem**

#### Serviço Aleatório 1

Se a métrica é euclidiana e a unidade de serviço está posicionada aleatoriamente,

**Então** escreva a mensagem:

“Infelizmente o cálculo da distância esperada de viagem entre dois pontos aleatoriamente distribuídos na região é bastante complexo. Para conseguir neste caso uma estimativa para este valor, deve-se fazer uso de simulações”

#### Serviço Central 1

Se a métrica é euclidiana, a unidade de serviço está posicionada no centro e o usuário conhece as dimensões da região,

**Então** escreva a mensagem:

“Então, como você optou por estimar valores através da métrica euclidiana e a unidade de serviço está posicionada no centro da região, uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_1, y_1)$  e o ponto de demanda é a seguinte:

$$E.D. = \frac{r_1}{3} + \frac{X_0}{24} \left( \frac{X_0}{Y_0} \ln H_1 + \frac{Y_0^2}{X_0^2} \ln H_2 \right)$$

onde

$$H_1 = \frac{B + r_1}{r_2 - y_1}, H_2 = \frac{A + r_1}{r_3 - x_1}, r_1 = \sqrt{A^2 + B^2}, r_2 = \sqrt{A^2 + y_1^2}, r_3 = \sqrt{x_1^2 + B^2} \quad A = X_0 - x_1, \\ B = Y_0 - y_1 \text{ e } X_0 = (\text{valor de } x_0) \text{ e } Y_0 = (\text{valor de } y_0).”$$

#### Serviço Central 1'

Se a métrica é euclidiana, a unidade de serviço está posicionada no centro e o usuário não conhece as dimensões da região,

**Então** escreva a mensagem:

“Então, como você optou por estimar valores através da métrica euclidiana e a unidade de serviço está posicionada no centro da região, uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_1, y_1)$  e o ponto de demanda é a seguinte:

$$\text{E.D.} = \frac{r_1}{3} + \frac{X_0}{24} \left( \frac{X_0}{Y_0} \ln H_1 + \frac{Y_0^2}{X_0^2} \ln H_2 \right)$$

onde  $H_1 = \frac{B+r_1}{r_2-y_1}$ ,  $H_2 = \frac{A+r_1}{r_3-x_1}$ ,  $r_1 = \sqrt{A^2+B^2}$ ,  $r_2 = \sqrt{A^2+y_1^2}$ ,  $r_3 = \sqrt{x_1^2+B^2}$

$A = X_0 - x_1$ ,  $B = Y_0 - y_1$  e  $X_0$  e  $Y_0$  são as dimensões da região de estudos.”

### Serviço Extremo 1

Se a métrica é euclidiana, a unidade de serviço está posicionada no extremo e o usuário conhece as dimensões da região,

**Então** escreva a mensagem:

“Então, como você optou por estimar valores através da métrica euclidiana e a unidade de serviço está posicionada no extremo da região, uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_1, y_1)$  e o ponto de demanda é a seguinte:

$$\text{E.D.} = \frac{1}{6X_0Y_0} \left( 2X_0Y_0r_1 + X_0^3 \ln \frac{Y_0+r_1}{X_0} + Y_0^3 \ln \frac{X_0+r_1}{Y_0} \right)$$

onde  $r_1 = \sqrt{(X_0 - x_1)^2 + (Y_0 - y_1)^2}$  e  $X_0 = (\text{valor de } x_0)$  e  $Y_0 = (\text{valor de } y_0)$ .”

### Serviço Extremo 1'

Se a métrica é euclidiana, a unidade de serviço está posicionada no extremo e o usuário não conhece as dimensões da região,

**Então** escreva a mensagem:

“Então, como você optou por estimar valores através da métrica euclidiana e a unidade de serviço está posicionada no extremo da região, uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_1, y_1)$  e o ponto de demanda é a seguinte:

$$\text{E.D.} = \frac{1}{6X_0Y_0} \left( 2X_0Y_0r_1 + X_0^3 \ln \frac{Y_0+r_1}{X_0} + Y_0^3 \ln \frac{X_0+r_1}{Y_0} \right)$$

onde  $r_1 = \sqrt{(X_0 - x_1)^2 + (Y_0 - y_1)^2}$  e  $X_0$  e  $Y_0$  são as dimensões da região de estudos.”

### Serviço Aleatório 2

Se a métrica é retangular, a unidade de serviço está posicionada aleatoriamente e o usuário conhece as dimensões da região,

Então calcular o valor esperado da distância:

$$E.D. = \frac{x_0 + y_0}{4}$$

### Serviço Aleatório 2'

Se a métrica é retangular, a unidade de serviço está posicionada aleatoriamente e o usuário não conhece as dimensões da região,

Então escreva a mensagem:

“Você optou por estimar valores através da métrica retangular, a unidade de serviço está posicionada aleatoriamente e não existe na região de estudo nenhuma barreira de viagem; então uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_l, y_l)$  e o ponto de demanda é a seguinte:

$$E.D. = \frac{X_0 + Y_0}{4}$$

onde  $X_0$  e  $Y_0$  são as dimensões da região de estudos.”

### Serviço Central 2

Se a métrica é retangular, a unidade de serviço está posicionada no centro e o usuário conhece as dimensões da região,

Então calcular o valor esperado da distância:

$$E.D. = \frac{x_0 + y_0}{3}$$

### Serviço Central 2'

Se a métrica é retangular, a unidade de serviço está posicionada no centro e o usuário não conhece as dimensões da região,

Então escreva a mensagem:

“Você optou por estimar valores através da métrica retangular, a unidade de serviço está posicionada no centro a região e não existe na região de estudo nenhuma barreira de viagem; então uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_l, y_l)$  e o ponto de demanda é a seguinte:

$$E.D. = \frac{X_0 + Y_0}{3}$$



onde  $X_0$  e  $Y_0$  são as dimensões da região de estudos.”

### Serviço Extremo 2

Se a métrica é retangular, a unidade de serviço está posicionada no extremo e o usuário conhece as dimensões da região,

**Então** calcular o valor esperado da distância:

$$E.D. = \frac{x_0 + y_0}{2}$$

### Serviço Extremo 2'

Se a métrica é retangular, a unidade de serviço está posicionada no extremo e o usuário não conhece as dimensões da região,

**Então** escreva a mensagem:

“Você optou por estimar valores através da métrica retangular, a unidade de serviço está posicionada no extremo a região e não existe na região de estudo nenhuma barreira de viagem; então uma estimativa para a distância média de viagem entre a unidade de serviço ( $x_1, y_1$ ) e o ponto de demanda é a seguinte:

$$E.D. = \frac{X_0 + Y_0}{2}$$

onde  $X_0$  e  $Y_0$  são as dimensões da região de estudos.”

## **3.2. Quando existem barreiras de viagem**

### BVA1

Se a métrica é retangular, a unidade de serviço está posicionada aleatoriamente, existe uma barreira de viagem vertical do tipo 1 e o usuário conhece as dimensões da região,

Então calcular o valor esperado da distância:

$$E.D. = \frac{x_0 + y_0}{4} + \frac{4A^3B(x_0 - B)}{3x_0^2y_0^2}$$

### BVA1'

Se a métrica é retangular, a unidade de serviço está posicionada aleatoriamente, existe uma barreira de viagem vertical do tipo 1 e o usuário não conhece as dimensões da região,

Então escreva a mensagem:

“Você optou por estimar valores através da métrica retangular, a unidade de serviço está posicionada aleatoriamente e existe na região de estudo uma barreira de viagem vertical; então uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_1, y_1)$  e o ponto de demanda é a seguinte:

$$E.D. = \frac{X_0 + Y_0}{4} + \frac{4a^3 b(X_0 - b)}{3X_0^2 Y_0^2} .”$$

### BVA2

Se a métrica é retangular, a unidade de serviço está posicionada aleatoriamente, existe uma barreira de viagem vertical do tipo 2 e o usuário conhece as dimensões da região, Então calcular o valor esperado da distância:

$$E.D. = \frac{x_0 + y_0}{4} + \frac{4B(x_0 - B)}{3x_0^2 y_0^2} [A^3 + (y_0 - A)^3].$$

### BVA2'

Se a métrica é retangular, a unidade de serviço está posicionada aleatoriamente, existe uma barreira de viagem vertical do tipo 2 e o usuário não conhece as dimensões da região, Então escreva a mensagem:

“Você optou por estimar valores através da métrica retangular, a unidade de serviço está posicionada aleatoriamente e existe na região de estudo uma barreira de viagem vertical; então uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_1, y_1)$  e o ponto de demanda é a seguinte:

$$E.D. = \frac{X_0 + Y_0}{4} + \frac{4b(X_0 - b)}{3X_0 Y_0^2} [a^3 + (Y_0 - a)^3].”$$

### BVE1

Se a métrica é retangular, a unidade de serviço está posicionada no extremo, existe uma barreira de viagem vertical e o usuário conhece as dimensões da região, Então calcular o valor esperado da distância:

$$E.D. = \frac{x_0 + y_0}{2} + \frac{A^2(x_0 - B)}{x_0 y_0} .$$

### BVE1'

Se a métrica é retangular, a unidade de serviço está posicionada no extremo, existe uma barreira de viagem vertical e o usuário não conhece as dimensões da região, Então escreva a mensagem:

“Você optou por estimar valores através da métrica retangular, a unidade de serviço está posicionada no extremo e existe na região de estudo uma barreira de viagem vertical; então

uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_l, y_l)$  e o ponto de demanda é a seguinte:

$$E.D. = \frac{X_0 + Y_0}{2} + \frac{a^2(X_0 - b)}{X_0 Y_0} .”$$

### BVC1

Se a métrica é retangular, a unidade de serviço está posicionada no centro, existe uma barreira de viagem vertical do tipo 1 e o usuário conhece as dimensões da região,

Então calcular o valor esperado da distância:

$$E.D. = \frac{x_0 + y_0}{3} + \frac{2A^2B}{3x_0y_0} .$$

### BVC1'

Se a métrica é retangular, a unidade de serviço está posicionada no centro, existe uma barreira de viagem vertical do tipo 1 e o usuário não conhece as dimensões da região,

Então escreva a mensagem:

“Você optou por estimar valores através da métrica retangular, a unidade de serviço está posicionada no centro e existe na região de estudo uma barreira de viagem vertical; então uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_l, y_l)$  e o ponto de demanda é a seguinte:

$$E.D. = \frac{X_0 + Y_0}{3} + \frac{2a^2b}{3X_0Y_0} .$$

### BVC2

Se a métrica é retangular, a unidade de serviço está posicionada no centro, existe uma barreira de viagem vertical do tipo 2 e o usuário conhece as dimensões da região,

Então calcular o valor esperado da distância:

$$E.D. = \frac{x_0 + y_0}{3} + \frac{2A^2(x_0 - B)}{3x_0y_0} .$$

### BVC2'

Se a métrica é retangular, a unidade de serviço está posicionada no centro, existe uma barreira de viagem vertical do tipo 2 e o usuário não conhece as dimensões da região,

Então escreva a mensagem:

“Você optou por estimar valores através da métrica retangular, a unidade de serviço está posicionada no centro e existe na região de estudo uma barreira de viagem vertical; então uma

estimativa para a distância média de viagem entre a unidade de serviço  $(x_i, y_i)$  e o ponto de demanda é a seguinte:

$$E. D. = \frac{X_0 + Y_0}{3} + \frac{2a^2(X_0 - b)}{3X_0Y_0}$$

### BVC3

Se a métrica é retangular, a unidade de serviço está posicionada no centro, existe uma barreira de viagem vertical do tipo 3 e o usuário conhece as dimensões da região,

Então calcular o valor esperado da distância:

$$E. D. = \frac{x_0 + y_0}{3} + \frac{2B}{3x_0y_0} [A^2 + (y_0 - A)^2]$$

### BVC3'

Se a métrica é retangular, a unidade de serviço está posicionada no centro, existe uma barreira de viagem vertical do tipo 3 e o usuário não conhece as dimensões da região,

Então escreva a mensagem:

“Você optou por estimar valores através da métrica retangular, a unidade de serviço está posicionada no centro e existe na região de estudo uma barreira de viagem vertical; então uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_i, y_i)$  e o ponto de demanda é a seguinte:

$$E. D. = \frac{X_0 + Y_0}{3} + \frac{2b}{3X_0Y_0} [a^2 + (Y_0 - a)^2]$$

### BVC4

Se a métrica é retangular, a unidade de serviço está posicionada no centro, existe uma barreira de viagem vertical do tipo 2 e o usuário conhece as dimensões da região,

Então calcular o valor esperado da distância:

$$E. D. = \frac{x_0 + y_0}{3} + \frac{2(x_0 - B)}{3x_0y_0} [A^2 + (y_0 - A)^2]$$

### BVC4'

Se a métrica é retangular, a unidade de serviço está posicionada no centro, existe uma barreira de viagem vertical do tipo 4 e o usuário não conhece as dimensões da região,

Então escreva a mensagem:

“Você optou por estimar valores através da métrica retangular, a unidade de serviço está posicionada no centro e existe na região de estudo uma barreira de viagem vertical; então uma

estimativa para a distância média de viagem entre a unidade de serviço  $(x_i, y_i)$  e o ponto de demanda é a seguinte:

$$E.D. = \frac{X_0 + Y_0}{3} + \frac{2(X_0 - b)}{3X_0Y_0} [a^2 + (Y_0 - a)^2].$$

### BHA1

Se a métrica é retangular, a unidade de serviço está posicionada aleatoriamente, existe uma barreira de viagem horizontal do tipo 1 e o usuário conhece as dimensões da região,

**Então** calcular o valor esperado da distância:

$$E.D. = \frac{x_0 + y_0}{4} + \frac{4B^3 A(y_0 - A)}{3x_0^2 y_0^2}.$$

### BHA1'

Se a métrica é retangular, a unidade de serviço está posicionada aleatoriamente, existe uma barreira de viagem horizontal do tipo 1 e o usuário não conhece as dimensões da região,

**Então** escreva a mensagem:

“Você optou por estimar valores através da métrica retangular, a unidade de serviço está posicionada aleatoriamente e existe na região de estudo uma barreira de viagem horizontal; então uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_i, y_i)$  e o ponto de demanda é a seguinte:

$$E.D. = \frac{X_0 + Y_0}{4} + \frac{4b^3 a(Y_0 - a)}{3X_0^2 Y_0^2}.”$$

### BHA2

Se a métrica é retangular, a unidade de serviço está posicionada aleatoriamente, existe uma barreira de viagem horizontal do tipo 2 e o usuário conhece as dimensões da região,

**Então** calcular o valor esperado da distância:

$$E.D. = \frac{x_0 + y_0}{4} + \frac{4A(y_0 - A)}{3x_0^2 y_0^2} [B^3 + (x_0 - B)^3].$$

### BHA2'

Se a métrica é retangular, a unidade de serviço está posicionada aleatoriamente, existe uma barreira de viagem horizontal do tipo 2 e o usuário não conhece as dimensões da região,

**Então** escreva a mensagem:

“Você optou por estimar valores através da métrica retangular, a unidade de serviço está posicionada aleatoriamente e existe na região de estudo uma barreira de viagem horizontal;

então uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_l, y_l)$  e o ponto de demanda é a seguinte:

$$E.D. = \frac{X_0 + Y_0}{4} + \frac{4a(Y_0 - a)}{3X_0^2 Y_0^2} [b^3 + (X_0 - b)^3].$$

#### BHE1

Se a métrica é retangular, a unidade de serviço está posicionada no extremo, existe uma barreira de viagem horizontal e o usuário conhece as dimensões da região,

**Então** calcular o valor esperado da distância:

$$E.D. = \frac{x_0 + y_0}{2} + \frac{B^2(y_0 - A)}{x_0 y_0}.$$

#### BHE1'

Se a métrica é retangular, a unidade de serviço está posicionada no extremo, existe uma barreira de viagem horizontal e o usuário não conhece as dimensões da região,

**Então** escreva a mensagem:

“Você optou por estimar valores através da métrica retangular, a unidade de serviço está posicionada no extremo e existe na região de estudo uma barreira de viagem horizontal; então uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_l, y_l)$  e o ponto de demanda é a seguinte:

$$E.D. = \frac{X_0 + Y_0}{2} + \frac{b^2(Y_0 - a)}{X_0 Y_0}.$$

#### BHC1

Se a métrica é retangular, a unidade de serviço está posicionada no centro, existe uma barreira de viagem horizontal do tipo 1 e o usuário conhece as dimensões da região,

**Então** calcular o valor esperado da distância:

$$E.D. = \frac{x_0 + y_0}{3} + \frac{2B^2 A}{3x_0 y_0}.$$

#### BHC1'

Se a métrica é retangular, a unidade de serviço está posicionada no centro, existe uma barreira de viagem horizontal do tipo 1 e o usuário não conhece as dimensões da região,

**Então** escreva a mensagem:

“Você optou por estimar valores através da métrica retangular, a unidade de serviço está posicionada no centro e existe na região de estudo uma barreira de viagem horizontal; então uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_l, y_l)$  e o ponto de demanda é a seguinte:

$$E. D. = \frac{X_0 + Y_0}{3} + \frac{2b^2 a}{3X_0 Y_0}.$$

### BHC2

Se a métrica é retangular, a unidade de serviço está posicionada no centro, existe uma barreira de viagem horizontal do tipo 2 e o usuário conhece as dimensões da região,

**Então** calcular o valor esperado da distância:

$$E. D. = \frac{x_0 + y_0}{3} + \frac{2B^2(y_0 - A)}{3x_0 y_0}.$$

### BHC2'

Se a métrica é retangular, a unidade de serviço está posicionada no centro, existe uma barreira de viagem horizontal do tipo 2 e o usuário não conhece as dimensões da região,

**Então** escreva a mensagem:

“Você optou por estimar valores através da métrica retangular, a unidade de serviço está posicionada no centro e existe na região de estudo uma barreira de viagem horizontal; então uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_l, y_l)$  e o ponto de demanda é a seguinte:

$$E. D. = \frac{X_0 + Y_0}{3} + \frac{2b^2(Y_0 - a)}{3X_0 Y_0}.$$

### BHC3

Se a métrica é retangular, a unidade de serviço está posicionada no centro, existe uma barreira de viagem horizontal do tipo 3 e o usuário conhece as dimensões da região,

**Então** calcular o valor esperado da distância:

$$E. D. = \frac{x_0 + y_0}{3} + \frac{2A}{3x_0 y_0} [B^2 + (x_0 - B)^2].$$

### BHC3'

Se a métrica é retangular, a unidade de serviço está posicionada no centro, existe uma barreira de viagem horizontal do tipo 3 e o usuário não conhece as dimensões da região,

**Então** escreva a mensagem:

“Você optou por estimar valores através da métrica retangular, a unidade de serviço está posicionada no centro e existe na região de estudo uma barreira de viagem horizontal; então uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_l, y_l)$  e o ponto de demanda é a seguinte:

$$E. D. = \frac{X_0 + Y_0}{3} + \frac{2a}{3X_0Y_0} [b^2 + (X_0 - b)^2].$$

#### BHC4

Se a métrica é retangular, a unidade de serviço está posicionada no centro, existe uma barreira de viagem horizontal do tipo 2 e o usuário conhece as dimensões da região,

**Então** calcular o valor esperado da distância:

$$E. D. = \frac{x_0 + y_0}{3} + \frac{2(y_0 - A)}{3x_0y_0} [B^2 + (x_0 - B)^2].$$

#### BHC4'

Se a métrica é retangular, a unidade de serviço está posicionada no centro, existe uma barreira de viagem horizontal do tipo 4 e o usuário não conhece as dimensões da região,

**Então** escreva a mensagem:

“Você optou por estimar valores através da métrica retangular, a unidade de serviço está posicionada no centro e existe na região de estudo uma barreira de viagem horizontal; então uma estimativa para a distância média de viagem entre a unidade de serviço  $(x_l, y_l)$  e o ponto de demanda é a seguinte:

$$E. D. = \frac{X_0 + Y_0}{3} + \frac{2(Y_0 - a)}{3X_0Y_0} [b^2 + (X_0 - b)^2].$$