

Sistema Inteligente de Locomoção de Viaturas para Atendimento na Rede de Distribuição

G.Lambert Torres (UNIFEI), L.E. Borges da Silva (UNIFEI), A.R. Aoki (UNIFEI), C.H.V. Moraes (UNIFEI), B.R. Costa (CEB) e J.A. Barbosa (CEB)

RESUMO

Este artigo apresenta os desenvolvimentos realizados no Programa de Pesquisa e Desenvolvimento Tecnológico da Companhia Energética de Brasília (CEB) sob o mesmo título deste artigo. Este projeto desenvolveu uma metodologia de análise automática e um pacote computacional com a finalidade de estabelecer um conjunto de rotinas inteligentes de busca para determinar a melhor viatura e o melhor caminho que esta deve efetuar para chegar a um ponto de defeito na rede de distribuição de energia elétrica. O melhor caminho entre a localização da viatura e a região de defeito deve ser encontrado levando-se em consideração aspectos tais como: vias disponíveis na cidade, tráfego, horários de pico e engarrafamentos, entre outros.

PALAVRAS-CHAVE

Operação, Locomoção de Viaturas, Suporte à Decisão, Técnicas de Busca Heurísticas, Sistemas Inteligentes.

I. INTRODUÇÃO

A CEB possui um sistema contínuo de localização de suas viaturas. O projeto descrito neste artigo tem a função de otimizar os deslocamentos dessas viaturas, no sentido de minimizar o tempo de restauração do sistema de distribuição. Um dos problemas das empresas de cumprir, ou mesmo melhorarem, seus índices de DEC é o tempo de deslocamento até a área atingida por um defeito. A idéia deste projeto é estabelecer uma série de rotinas inteligentes para determinar qual a viatura que deve atender o defeito. Para isto, diversos fatores devem ser levados em consideração, por exemplo:

- (a) nem sempre a viatura que está mais próxima possui todos os equipamentos necessários para a correção do defeito. Porém será que ela não pode começar a tomar algumas medidas corretivas, antes da próxima viatura chegar para complementar o serviço?
- (b) nem sempre a viatura que está mais próxima, mesmo possuindo todos os equipamentos necessários para a correção do defeito, é a melhor, pois problemas como tempo já trabalhado pela equipe no dia (gerando horas-extras ou problemas no reparo por estafa da equipe) ou outra emergência que deva ser atendida pode mudar esta escolha.

Diversos outros exemplos poderiam ser dados, que mostrariam que esta escolha não é óbvia. Acoplado a isto deve-se também verificar o horário que o defeito ocorreu, pois por vezes não é a viatura mais próxima, linearmente falando, que chegará na área de defeito mais rapidamente.

Problemas de engarrafamento ou simples aglomeração de horários de pico podem também alterar a solução da melhor viatura.

A idéia central deste projeto é minimizar o tempo de atendimento a um defeito na rede elétrica, que é composto pelo tempo de deslocamento mais o tempo de reparo.

O pacote computacional para a locomoção de viaturas para atendimento a emergências e chaveamentos na rede de distribuição da CEB foi desenvolvido com o objetivo de auxiliar os operadores durante a condução das modificações no sistema. Este pacote computacional foi baseado em processos de busca heurística e outras técnicas de inteligência artificial.

Inicialmente, este artigo apresenta uma visão geral dos programas computacionais desenvolvidos. Em seguida, são apresentadas as ferramentas computacionais utilizadas, para então descrever sucintamente a técnica de busca escolhida, com as bases de dados utilizadas. Daí, apresentam-se os dois pacotes computacionais frutos deste projeto, denominados: programa de desenvolvimento e programa operacional. O primeiro foi utilizado conforme especifica seu nome e, posteriormente, definido para ser um pacote de treinamento. O segundo, o programa operacional, está em operação no Centro de Controle da CEB. Exemplos práticos são mostrados para ilustrar o artigo.

II. VISÃO GERAL DO PROJETO

Para que os objetivos do projeto pudessem ser cumpridos e as técnicas inteligentes híbridas fossem testadas, foram desenvolvidos dois pacotes computacionais, denominados: programa de desenvolvimento e programa operacional.

O primeiro, como o próprio nome sugere (programa de desenvolvimento), foi formulado para que as várias características e objetivos pudessem ser testados e que os resultados ficassem mais facilmente disponíveis para análise. O segundo, o programa operacional, que está disponível dentro do centro de controle e despacho da CEB, apresenta ao operador somente as respostas solicitadas sem análises ou respostas intermediárias.

G. Lambert Torres, L.E. Borges da Silva, A.R. Aoki e C.H.V. Moraes são afiliados à Universidade Federal de Itajubá (e-mail: {germano, leborges, aoki, valerio}@iee.efei.br).

B.R. Costa e J.A. Barbosa trabalham na Companhia Energética de Brasília (e-mail: Belmiro, jair@ceb.com.br).

Esta divisão em dois programas acompanha estudos e levantamentos realizados pelo Grupo 39 da CIGRÉ [1], que mostraram a necessidade de só se apresentar ao operador/despachante a solução do problema, já devidamente ordenada em ordem de prioridade, sem que critérios possam ser modificados ou avaliados no momento. Isto decorre do fato de que, normalmente, o programa será executado durante uma interrupção de energia em ramos do sistema de distribuição e toda a atenção do operador deve estar na solução do problema e não em critérios internos de um programa de auxílio. Por este motivo, telas com perguntas ou encaminhamentos alternativos, que constam do programa de desenvolvimento, foram suprimidas nesta versão e substituídas por conhecimentos adquiridos em testes realizados durante o desenvolvimento do programa. A versão final do pacote de desenvolvimento será também útil para o processo de transferência tecnológica.

Com este encaminhamento foi possível criar um pacote computacional com uma análise automática de deslocamento de viaturas que atingisse os objetivos do projeto, sendo ao mesmo tempo flexível e operacional. Assim, ele possibilita verificar as melhores viaturas que podem se deslocar até o ponto com problema, ordenando-as e calculando um índice (nota) de desempenho, de 0 a 10.

Vale a pena aqui relatar o problema encontrado para a operacionalização do programa que irá ser executado no centro de controle. Uma das bases de dados que o programa utiliza para fornecer a resposta ao usuário é a de cadastro de todos os trechos das ruas da cidade de Brasília e das 19 regiões administrativas (definidas conforme a Secretaria do Estado de Desenvolvimento Urbano e Habitação do Distrito Federal) servidas com energia pela CEB. Este cadastro foi feito, porém sua atualização não é “on-line” e depende de uma intervenção específica para ocorrer. Isto faz com que problemas momentâneos com o tráfego devam ser inseridos individualmente nesta base de dados. Isto ocorre pois ainda não existe a possibilidade de se ter este tipo de dados disponíveis em Brasília e, por conseqüência, na CEB. Este problema foi contornado com a inclusão de “regiões com problemas típicos” ou “horários típicos de problemas de tráfego”, e que podem a qualquer momento ser alterados pelos operadores do sistema. Não obstante este fato, o programa foi construído de maneira a que quando houver este processo de aquisição de dados “on-line”, ele não necessite ser alterado.

III. FERRAMENTAS COMPUTACIONAIS UTILIZADAS

O programa de desenvolvimento foi implementado em Visual-Basic e Turbo-Prolog. A primeira linguagem, Visual-Basic, serviu para a modelagem das bases de dados, para o desenvolvimento de rotinas numéricas e para a confecção da interface com o usuário. A segunda ferramenta, Turbo-Prolog, serviu para o desenvolvimento dos sistemas espe-

cialistas para o auxílio no processo de tomada de decisão.

O programa operacional foi desenvolvido somente em Visual-Basic. As rotinas em Turbo-Prolog foram convertidas e otimizadas para o problema em questão. Isto foi feito para se ter uma resposta mais rápida na solução do problema, característica fundamental para um programa a ser utilizado durante uma pane.

IV. SELEÇÃO E ESCOLHA DA TÉCNICA DE BUSCA

Este projeto teve como objetivo desenvolver algoritmos de busca que possam integrar diversos tipos de técnicas, criando sistemas híbridos. Para tal, foram confeccionados três programas que pudessem concorrer entre si, são eles:

(a) Processo de busca baseado em heurísticas

Heurísticas são critérios, métodos ou princípios utilizados para se decidir qual entre diversos caminhos alternativos de solução promete ser o mais eficaz a fim conseguir algum objetivo. Elas representam um compromisso entre duas exigências: a necessidade estabelecer critérios simples e confiáveis e, ao mesmo tempo, o desejo vê-los discriminar corretamente entre escolhas boas e más [2, 3]. No caso deste projeto, as heurísticas foram regras práticas de locomoção utilizadas no trânsito.

Outras regras utilizadas no programa estabeleciam estratégias para se escolher um caminho entre diversos possíveis. O programa deve decidir por um dado caminho que pareça mais eficaz que outros por resultar em uma posição mais próxima de seu objetivo final. Entretanto, nem sempre este fato garante a seleção do movimento mais eficaz e alguns caminhos devem ser cortados [4]. Assim, verificou-se que a utilização de uma heurística mesma correta, não leva com certeza a solução de um problema, pois outros fatores externos não levados em consideração podem agir e não possibilitar sua solução, como, por exemplo, uma rua sem saída ou um lago.

Outro elemento a considerar é que a busca de caminhos por busca em árvores é um problema do tipo NP-complexo [5], requerendo assim uma avaliação de um número imenso de possibilidades para determinar uma solução ótima. O tempo requerido, para se encontrar e determinar que uma solução é ótima, é freqüentemente muito grande. Nestes casos, as heurísticas desempenham um papel eficaz em tais problemas indicando uma maneira reduzir o número das avaliações e obter soluções dentro dos condicionantes de tempo requerido.

(b) Processo de busca via sistemas especialistas difusos

O sistema especialista desenvolvido neste encaminhamento foi escrito em Prolog, que é uma linguagem declarativa a qual utiliza o raciocínio dedutivo para solucionar o problema apresentado. Isto contrasta com as lingua-

gens computacionais tradicionais, tais como Fortran e Basic, que são linguagens de procedimento, nas quais o programador deve dizer ao computador exatamente como fazer para obter a solução de um dado problema, fornecendo, passo-a-passo, as instruções necessárias. Em outras palavras, o programador deve saber solucionar o problema antes que o computador possa fazê-lo. Já o programador de Prolog precisa apenas de fornecer uma descrição do problema, usando um número de fatos e das regras básicas para solucioná-lo. No caso deste projeto, as bases de dados das ruas/quadras e regras de tráfego.

A partir daí, o compilador Prolog é deixado livre para determinar como agir de forma a encontrar a solução do problema descrito. Isto significa que, uma vez que o programador tenha descrito o que deve ser computado, ou seja, a posição de origem e a posição final, o próprio sistema organiza como a forma da computação, a qual é conduzida para se obter a solução do problema descrito.

Neste projeto utilizou-se o compilador Turbo-Prolog, que tem a característica de que quando executado tenta achar todos os conjuntos de valores que satisfaça à meta dada, ou seja, todos os caminhos possíveis, dentro da faixa desejada.

O Turbo-Prolog utiliza o mecanismo de procura “Encadeamento para Trás” (backtracking), uma vez que uma solução tenha sido encontrada, impele o Prolog a reavaliar quaisquer hipóteses ou suposições feitas de forma a verificar se algum valor novo pode fornecer novas soluções.

Incluiu-se também deste procedimento uma avaliação realizada através da teoria dos conjuntos difusos. As variáveis de entrada em um sistema difuso são em geral mapeadas dentro de conjuntos de funções consecutivas - o processo de conversão de um valor de entrada intermediário em um valor difuso é chamado de “fuzzificação” [6]. Em seguida, estas variáveis difusas são passadas por um sistema de regras, no caso deste projeto de avaliação. Assim, a decisão é baseada neste conjunto de regras. Vale lembrar que todas as regras existentes são invocadas, usando as funções consecutivas e valores verdadeiros obtidos das entradas, para determinar o resultado da regra. Daí, será mapeada dentro da função consecutiva o valor difuso da variável saída. Estes resultados são combinados para gerar uma resposta específica, a qual será a resposta difusa do sistema. Neste ponto é aplicado um procedimento conhecido como “defuzzificação”, gerando a resposta final. A combinação de operações difusas e do sistema de regras compõem o sistema especialista difuso.

(c) Processo de busca integrando técnicas de algoritmos genéticos

Existem diversas possibilidades de integrar os algoritmos genéticos aos sistemas especialistas difusos. Um dos possíveis é um sistema híbrido que envolva módulos separados como parte de um sistema global. Os

módulos baseados em algoritmos genéticos e na lógica difusa podem ser agrupados isoladamente ou com outros subsistemas de programas computacionais inteligentes ou convencionais que formam o programa.

Neste projeto, utilizou-se uma metodologia de refino do processo através do uso de cromossomos que incluem informações sobre as condições e ações correspondentes às regras difusas. Incluí-las no tratamento genético permite ao sistema aprender ou refinar as regras difusas.

O procedimento geral deste processo é mostrado na Figura 1. Os cromossomos são definidos como sendo uma concatenação dos valores de todas as funções de pertinência. Quando as funções triangulares são usadas para representar as funções de pertinência de avaliação (descritas acima), os parâmetros são os centros e as larguras para cada conjunto difuso. De uma gama inicial de valores de parâmetros possíveis, o sistema difuso é rodado para determinar o quanto ele funciona bem. Essas informações são usadas para determinar o ajuste de cada cromossomo e para estabelecer a nova população. O ciclo é repetido até que seja encontrado o melhor conjunto de valores para os parâmetros das funções de pertinência.

Em seguida são apresentados alguns elementos utilizados neste procedimento:

- 1) O cromossomo foi definido como sendo a concatenação dos valores de ajuste das funções de pertinência.
- 2) Os parâmetros são os centros e as larguras de cada conjunto difuso. Estes parâmetros compõem os genes do cromossomo.
- 3) De uma gama inicial de valores de parâmetros possíveis, o sistema difuso é rodado para determinar o quanto ele funciona bem.
- 4) Essas informações são usadas para determinar o ajuste de cada cromossomo (adaptabilidade) e estabelecer assim uma nova população.
- 5) O ciclo é repetido até que se complete o número de gerações definidas pelo usuário. A cada geração é encontrado o melhor conjunto de valores para os parâmetros das funções de pertinência.



FIGURA 1 – Processo global para uso de um algoritmo genético a fim de melhorar o desempenho de um sistema difuso

Como dito anteriormente, colocou-se os três processos de busca para concorrerem. Entretanto, de uma forma geral, uma regra prática não pode garantir escolher a me-

lhor alternativa, mas cria possibilidade de se ter uma solução “boa o bastante” para o tempo solicitado de resposta. É importante compreender que existe uma estreita relação entre a solução requerida e o tempo que se tem para obtê-la. Senão vejamos. Se um operador de um sistema elétrico tiver um tempo infinito para solucionar um problema de sobretensão no sistema, ele poderia estabelecer diversos estudos, consultar especialistas, ter reuniões técnicas, formar grupos de estudo, entre outros visando encontrar a melhor solução para o problema. Entretanto, na prática ele possui apenas alguns momentos para tomar a decisão. Isto, entretanto, não quer dizer que a decisão não seja apropriada, ou mesmo que não seja a obtida depois de vários estudos.

Aqui também ocorre este fenômeno e as avaliações feitas nos algoritmos levaram isto em consideração. Todas estas implementações foram feitas nas versões de teste do programa de desenvolvimento. Mais algumas características foram agregadas a este processo com a finalidade de aumentar sua eficiência. Uma delas que vale a pena relatar foi aquela referente a otimização do processo devido ao tamanho das bases de dados envolvidas.

V. BASES DE DADOS UTILIZADAS

V.1 Base de Dados das Chaves do Sistema da CEB

O banco de dados do sistema de distribuição da CEB foi capturado diretamente do sistema existente no centro de controle da empresa. Foi estabelecido um procedimento para que se pudesse obter estes dados e converter para o programa operacional todas as vezes que se fizerem necessárias. Assim, foi disponibilizado em um arquivo texto com diversos dados tais como: identificação da chave, tipo (NA de recurso, NA de socorro externo, NA de socorro interno, entre outras), número do circuito, localização, proprietário, coordenadas X e Y. Em seguindo esta metodologia, foi desenvolvido um programa computacional conversor deste tipo de arquivo para o formato utilizado na base de dados do programa operacional. Desta forma estabeleceu-se, o procedimento padrão que será utilizado pelos profissionais da CEB para a utilização regular o pacote computacional.

V.2 Base de Dados do Sistema Viário de Brasília

O banco de dados do sistema viário de Brasília e arredores foi feito determinando-se quadra-a-quadra existentes. Primeiramente, buscou-se organizar o mapa de Brasília segundo o padrão adotado pela Secretaria do Estado de Desenvolvimento Urbano e Habitação (SEDUH) a qual divide o Distrito federal em dezenove regiões administrativas: I – Brasília; II – Gama; III – Taguatinga; IV – Brazlândia; V – Sobradinho; VI – Planaltina; VII – Paranoá; VIII – Núcleo Bandeirante; IX – Ceilândia; X – Guará; XI – Cruzeiro; XII – Samambaia; XIII – Santa Maria; XIV – São Sebastião; XV – Recanto

das Emas; XVI – Lago Sul; XVII – Riacho Fundo; XVIII – Lago Norte; XIX – Candangolândia.

Em seguida, para cada região o mapa foi digitalizado a partir de um software específico, e ainda foi desenvolvido um interpretador dos dados digitalizados para um banco de dados do tipo Access que contém as seguintes tabelas:

- i) Bairros – nessa tabela está contida o identificador do bairro e o nome do bairro.
- ii) Pontos – nessa tabela está contida todas as esquinas encontradas no mapa, armazenadas através de um identificador do ponto, meridiano de longitude WGr, paralelo de latitude sul, ambos em graus, e um identificador se esse ponto é de conexão entre bairros.
- iii) Ruas – nessa tabela está contida todas as ruas encontradas no mapa, armazenadas através de um identificador da rua, o nome da rua e o código de endereçamento postal.
- iv) Quadras – nessa tabela está contida todas as quadras encontradas no mapa, armazenadas através de um identificador da quadra, um identificador do bairro, um identificador da rua, um identificador da esquina de origem, um identificador da esquina de destino, o tamanho da quadra em metros, e por fim o sentido de trânsito.

A partir desse banco de dados foi desenvolvido um tradutor que transforma essa informação em uma base de dados empregada pelo sistema inteligente. A Figura 2 apresenta uma pequena parte da base de dados. Essa base de dados contém as seguintes informações para cadastrar uma quadra, Figura 3:

Bairro – Nome do bairro da cidade onde a quadra da rua se encontra;

Rua – Nome da rua da quadra que está sendo adicionada;

Sentido – Sentido de trânsito na quadra (1 – mão única e 2 – mão dupla);

Número da Esquina De – Número da esquina de origem da quadra;

Número da Esquina Para – Número da esquina de destino da quadra;

Peso – Peso atribuído para quadras de acordo com os fatores:

Semáforos – Quantidade (de 0 a 5);

Escolas – Quantidade (de 0 a 5);

Redutores de velocidade – Quantidade (de 0 a 5);

Faixa de pedestres – Quantidade (de 0 a 5);

Congestionamento no sentido “De” para “Para” – indica a existência de congestionamento de acordo com o período do dia: 0 – nenhum; 1 – matutino; 2 – vespertino; 3 – noturno; 4 – horários de pico e 5 – dia todo;

Congestionamento no sentido “Para” para “De” – indica a existência de congestionamento de acordo com o período do dia: 0 – nenhum; 1 – matutino; 2 – vespertino; 3 – noturno; 4 – horários de pico e 5 – dia todo;

[BAIRRO]	[RUA]	Am	Noite		
2	280	281	0;0;1;1;0;0;4	55,0	
2	319	320	1;0;0;0;0;0;0	42,4	
2	231	287	0;1;0;0;0;0;0	51,7	
2	285	286	0;0;0;0;0;0;0	149,3	
2	233	285	0;0;0;1;0;1	253,7	
2	234	284	2;0;0;0;0;0;0	63,6	
2	284	235	0;1;0;1;0;0;0	51,3	
2	288	284	0;0;0;1;0;0;0	217,0	3
2	283	280	0;0;0;0;1;0;0	192,1	
2	283	278	0;0;0;0;1;0;0	43,4	
2	282	283	0;0;0;1;0;0;3	53,8	
2	282	156	0;0;0;0;0;0;0	53,4	
2	229	288	0;0;1;1;0;0;0	97,2	
2	281	171	0;0;0;1;0;0;0	70,0	
2	229	289	2;0;0;0;1;0;0	84,5	1
2	279	280	0;0;1;1;0;0;0	54,8	
2	279	173	0;0;1;0;0;0;0	72,5	
2	278	279	0;0;0;0;1;0;0	196,6	
2	155	278	1;0;1;1;0;0;0	52,3	
2	277	154	0;0;0;0;0;0;0	199,9	
2	153	276	0;0;0;0;1;0;0	195,8	
2	275	152	0;0;0;0;1;0;0	208,6	2
2	151	274	0;0;0;0;0;0;0	189,1	

FIGURA 2 – Base de dados do sistema inteligente

Distância [m] – Distância da quadra em metros;

Quadra Preferencial – Indica se há na quadra uma esquina que é um caminho preferencial para cruzar de um bairro para outro de acordo com um indicador: 1 – Esquina “De”; 2 – Esquina “Para” e 3 – Ambas as esquinas. Em caso de nenhuma ser não precisa aparecer na base de dados.

FIGURA 3 – Janela de cadastramento de quadras

VI. PROGRAMA DE DESENVOLVIMENTO

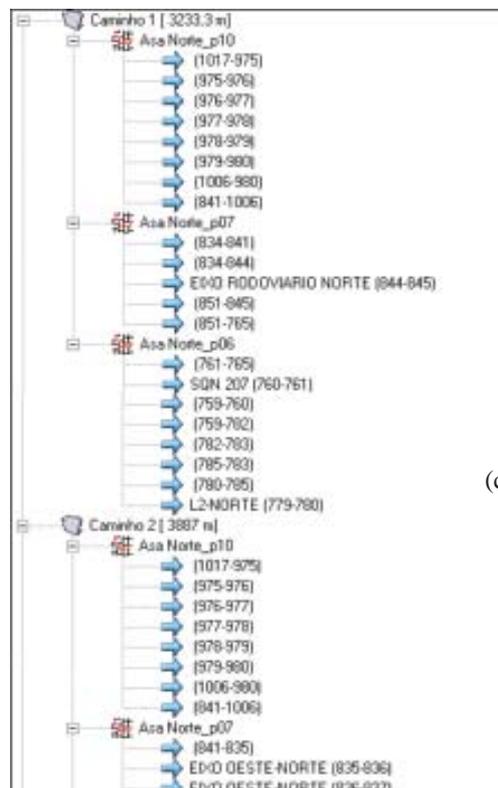
Algumas telas do programa de desenvolvimento, onde foram testadas as metodologias, são apresentadas a seguir. A Figura 4(a) mostra a tela de cadastro das quadras. Na Figura 4(b), é apresentada uma tela com a quadra de origem (onde está situada a viatura) e a chave de destino (local onde a viatura quer chegar). O elemento 1.1 (em português 1,1) é um fator de expansão de soluções alternativas. Feito isso, o usuário deverá então acionar o programa, clicando no botão “Percurso”, para

buscar os melhores percursos para locomover uma viatura da quadra de origem até a chave. Essa busca é feita primeiramente por bairros, determinando por quais bairros deve-se passar para atingir o objetivo. Em seguida, busca-se de bairro em bairro os melhores caminhos para percorrê-los. A resposta apresentada pelo programa está mostrada na Figura 4(c).

Bairro	Rua	Esquina De	Esquina Para	Componentes	Ligações
Asa Norte	FISCOMA-EPNA	12	13	49,2	
Asa Norte	FISCOMA-EPNA	1	2	146,3	1
Asa Norte	FISCOMA-EPNA	13	14	125,5	2
Asa Norte	FISCOMA-EPNA	11	12	288,8	1
Asa Norte	FISCOMA-EPNA	30	11	188,8	3
Asa Norte	FISCOMA-EPNA	9	18	195,5	2
Asa Norte	FISCOMA-EPNA	6	7	78,7	2
Asa Norte	FISCOMA-EPNA	8	9	76,9	2
Asa Norte	FISCOMA-EPNA	4	5	174,3	2
Asa Norte	FISCOMA-EPNA	2	3	49,8	1
Asa Norte	FISCOMA-EPNA	3	4	249,1	1
Asa Norte	FISCOMA-EPNA	8	9	81,6	3
Asa Norte	FISCOMA-EPNA	14	15	71,5	2
Asa Norte	FISCOMA-EPNA	1	16	128,8	1
Asa Norte	FISCOMA-EPNA	20	21	128,8	2
Asa Norte	FISCOMA-EPNA	21	22	88,8	2
Asa Norte	FISCOMA-EPNA	22	23	100,8	2

(a)

(b)



(c)

FIGURA 4 – Exemplo de um processo de busca (a) Dados de busca, (b) Dados das Quadras e (c) Resposta da busca de percurso

VII. PROGRAMA OPERACIONAL

VII.1 Integração o Programa com o SAO

A integração do SAO, sistema de apoio à operação que é utilizado pelos operadores e despachantes da CEB, com o programa operacional de locomoção de viaturas é feita através de troca de arquivos via NFS, sendo que o SAO atua como servidor NFS e o programa como cliente NFS.

O SAO sendo o servidor NFS disponibiliza e exporta um arquivo do sistema (file system) nas máquinas servidoras da base de dados. Como o sistema lê e escreve no mesmo diretório, não é permitida a realização de operações simultâneas de leitura/escrita pelo sistema.

Para o controle da concorrência, visando evitar que o SAO e o programa operacional tentem acessar os arquivos de integração simultaneamente, deve ser implementado um método de exclusão mútua baseado na existência ou não de arquivos auxiliares: EAL.LOC (External Application Link) e VIATURA.LOC. A Figura 5 são apresentados os fluxogramas para controle da concorrência.

Formato do arquivo SAIDA.TXT:

[Identificador do solicitante]

Listagem das viaturas ordenadas do mais apto para o menos apto ao atendimento da solicitação:

[Identificador da viatura] [Classificador]

O classificador é um índice que dá suporte à tomada de decisão.

Formato do arquivo DADOS.TXT:

[Identificador do solicitante]

[Identificador da chave a ser operada]

Listagem das viaturas:

[Identificador da viatura] [X] [Y]

VII.2 Procedimento para Importação dos Dados de Chaves

Para que se possa atualizar os dados de chaves no aplicativo de Locomoção de Viaturas tem sido desenvolvido um procedimento para que o usuário possa quando julgar necessário efetuar tal operação. Esse procedimento é feito de modo manual por usuário devidamente treinado, e está disponível na sua interface para Windows somente.

VII.3 Interfaces para o Windows

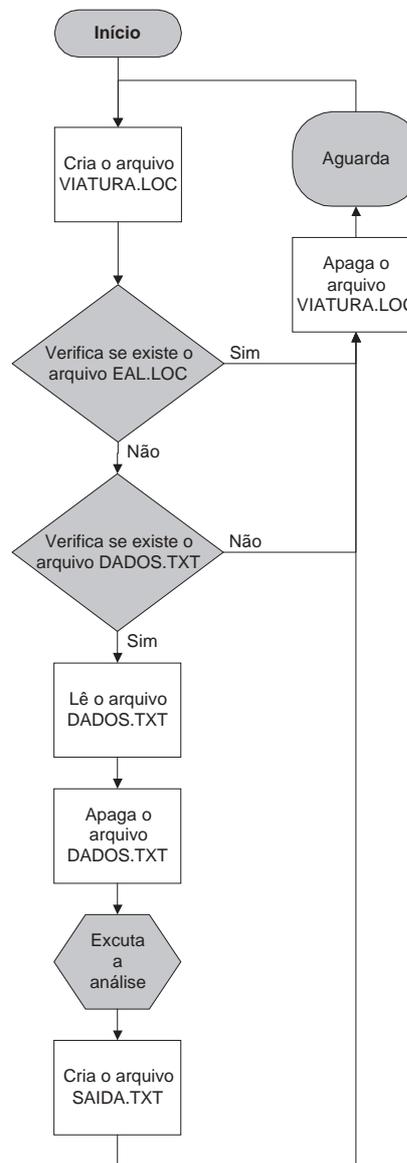
O programa operacional funciona em modo minimizado, apresentando-se apenas como um ícone na Barra de Tarefas, porém quando solicitada uma análise o mesmo apresenta sinais de processamento através da mudança do ícone do mouse para ponteiro com ampulheta ao lado.

Ao terminar a análise o mesmo apresenta a solução em forma de janela utilizando apresentação já empregada nas versões anteriores e os menus apropriados para a operação do programa. Ao terminar de visualizar o resultado da análise o usuário deve clicar em um botão "OK" para que o aplicativo retorne ao seu estado minimizado.

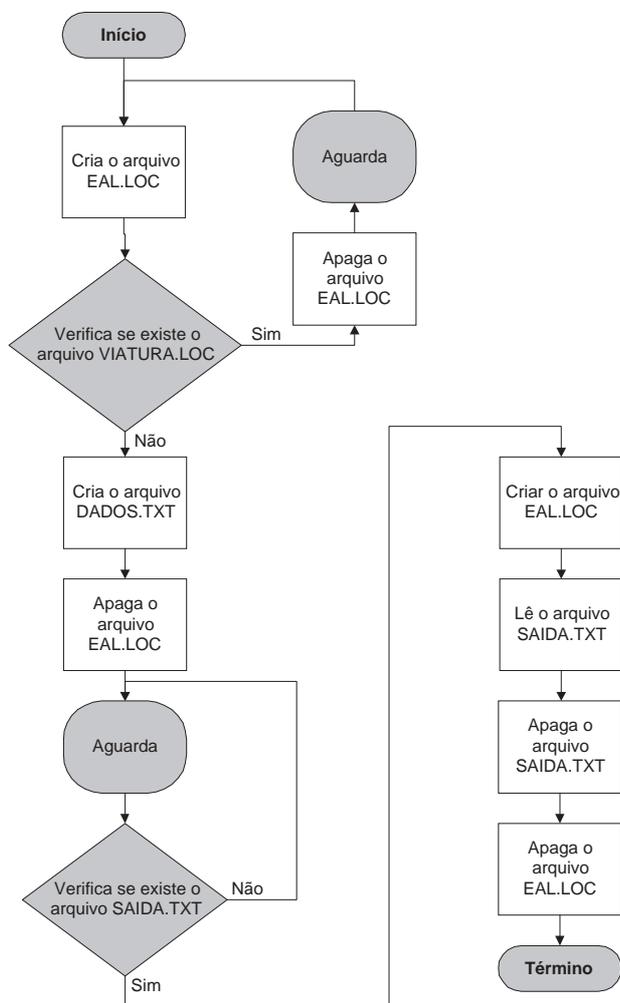
VII.4 Procedimentos Operacionais para o Aplicativo de Locomoção de Viaturas

Adotou-se a seguinte metodologia de operação:

- O SAO fornece a chave que se deseja operar e a posição de todas as viaturas da CEB;
- O programa operacional de locomoção de viaturas analisa as rotas para o atendimento da solicitação a partir de cada uma das viaturas;
- É fornecido como resultado uma listagem ordenada das melhores viaturas para o atendimento da solicitação, levando em conta a distância das mesmas até a chave;
- É atribuído a cada viatura um classificador para que o usuário possa mensurar sua viabilidade.



(a)



(b)

FIGURA 5 – Fluxogramas das Informações: (a) Sistema de Locomoção de Viaturas → SAO, (b) SAO → Sistema de Locomoção de Viaturas

VIII. CONCLUSÕES

Este trabalho representa mais um passo no desenvolvimento de ferramentas que utilizam técnicas de Inteligência Artificial para dar apoio às atividades na operação de Sistemas Elétricos de Potência, cujo objetivo final é a busca constante da otimização das ações e procedimentos de supervisão e controle do sistema.

Este projeto foi dividido em duas partes: um Programa de Desenvolvimento e o Programa Operacional. O programa de desenvolvimento para locomoção de viaturas, é uma versão “off-line”, e teve como finalidade testar a abordagem proposta, porém ainda pode ser aplicado para o treinamento de operadores dos centros de operação.

O programa operacional é a versão para o centro de operação e é uma ferramenta que deve ser utilizada conjuntamente com um sistema automatizado de supervisão e controle da CEB.

IX. REFERÊNCIAS BIBLIOGRÁFICAS

- [1] G. Lambert Torres, J. Haddad, C.I.A. Costa, V.H. Quintana e G.M. Januzzi - “A Fuzzy-Logic Tool to Support Medium and Long-Term Decision-Making Planning”, *International Journal of Power & Energy Systems*, Vol. 18, No.3, pp. 167-175, 1998
- [2] C.C. Liu and CIGRÉ TF38-06-03 - “Practical Use of Expert System in Planning and Operation of Power System”, *Electra*, 1993.
- [3] B. Valiquette, G. Lambert-Torres e D. Mukhedkar - “An Tool for Teaching Power System Operation Emergency Control Strategies”, *IEEE Trans. on Power Systems*, Vol.6, No.3, pp. 1315-1322, August 1991.
- [4] K. Parsaye e M. Chignelli - “Expert System for Experts”, John Wiley & Sons, Inc., QA76E95P27, p.117-126, 1988.
- [5] G. Lambert Torres, G.M. Ribeiro, C.I.A. Costa, A.P. Alves da Silva e V.H. Quintana - “Knowledge Engineering Tool for Training Power-Substation Operators”, *IEEE Transactions on Power Systems*, Vol. 12, No. 2, pp. 694-699, April 1997.
- [6] S. Russel e P. Norvig, *Artificial Intelligence – A Modern Approach*, Prentice Hall, 1995.