

FACULDADE DE TECNOLOGIA E CIÊNCIAS - FTC

MARCELO CAMERA OLIVEIRA

**PROJETO DE VIABILIZAÇÃO DE CANAL DE TV COM
AUXÍLIO DO MATLAB**

Salvador - Bahia

Fevereiro - 2006

MARCELO CAMERA OLIVEIRA

**PROJETO DE VIABILIZAÇÃO DE CANAL DE TV COM
AUXÍLIO DO MATLAB**

Monografia apresentada ao Curso de Engenharia Elétrica com Ênfase em Telecomunicações e Computação da Faculdade de Tecnologia e Ciências - FTC, para obtenção do título de Bacharel em Engenharia Elétrica com habilitação em Telecomunicações e Computação.

Salvador - Bahia

Fevereiro – 2006

MARCELO CAMERA OLIVEIRA

PROJETO DE VIABILIZAÇÃO DE CANAL DE TV COM AUXÍLIO DO MATLAB

Monografia apresentada ao Curso de Engenharia Elétrica com Ênfase em Telecomunicações e Computação da Faculdade de Tecnologia e Ciências - FTC, para obtenção do título de Bacharel em Engenharia Elétrica com habilitação em Telecomunicações e Computação.

Aprovada em ____/____/____

BANCA EXAMINADORA

Prof. Roberto da Costa e Silva
(Orientador)

AGRADECIMENTOS

A meus pais, que acreditaram em mim durante todos estes anos.

A minha namorada, por ter compreendido a minha ausência durante a elaboração deste trabalho e ajuda quando mais precisei.

Aos amigos, professores e funcionários da FTC, que proporcionaram momentos enriquecedores durante a graduação.

A meu amigo Otávio, pela ajuda mútua no decorrer dos nossos trabalhos.

Ao prof. Roberto da Costa e Silva, pela orientação desse trabalho.

RESUMO

Trata-se da criação de um programa para execução de projetos de viabilização de canais de TV. O programa foi feito baseado na Regulamentação nº 284 de 7 de dezembro de 2001, que tem como anexo o Regulamento Técnico para a Prestação do Serviço de Radiodifusão de Sons e Imagens e do Serviço de Retransmissão de Televisão. O programa é denominado de TVMAT e foi construído através do Matlab, uma ferramenta computacional que foi de suma importância para a elaboração do mesmo. O enfoque do trabalho foi usar o método da interpolação para estimar os valores das curvas de intensidade de campo utilizadas nos cálculos de viabilização. O TVMAT será útil para engenheiros e propõe-se a executar um projeto deste tipo em um tempo hábil e preciso.

Palavras-Chave: Matlab, Interpolação, TVMAT, TV.

ABSTRACT

One is about the creation of a program for execution of projects of TV channels. The program was made based in the Regulation nº 284 of 7 of December of 2001, that it has as attached the Regulation Technician for the Installment of the Service of Broadcasting of Sounds and Images and of the Service of Retransmission of Television. The program is called of TVMAT and was constructed through the Matlab, a computational tool that was of utmost importance for the elaboration of exactly. The approach of the work was to use the method of the interpolation esteem the values of the curves of used intensity of field in the calculations of project. The TVMAT will be useful for engineers and considers to execute it a project of this type in a skillful and necessary time.

Keywords: Matlab, Interpolation, TVMAT, TV.

LISTA DE FIGURAS

Figura 1 – Faixa de frequência TV.....	12
Figura 2 – Determinação da rugosidade de um perfil.	14
Figura 3 – Janela inicial do programa.	21
Figura 4 – Janela principal do programa.	23
Figura 5 – Janela das estações.	24
Figura 6 – Janela das radiais.....	26
Figura 8 – Exemplo ilustrado no TVMAT.	35

LISTA DE TABELAS

Tabela 1 – Classificação das estações.	12
Tabela 2 – Valores de intensidade de campo E (50,50) para os contornos de serviço.....	15
Tabela 3 – Valores de intensidade de campo para contornos protegidos.....	15
Tabela 4 – Relações de proteção para VHF e UHF.....	16
Tabela 5 – Relações de proteção para UHF.	16
Tabela 6 – Variáveis do sistema.....	22
Tabela 7 – Vetor intensidade de campo x altura da antena x distância.....	30
Tabela 8 – Relação dos modelos de interpolação.....	30
Tabela 9 – Dados da Estação Proponente.....	32
Tabela 10 – Cotas das Radiais, suas NMRs e HNMRs.....	33
Tabela 11 – Resultado da Rugosidade em metros Obtida pra cada Radial.....	35
Tabela 12 – Comparação entre os contornos (em Km) calculados e obtidos no TVMAT.....	36
Tabela 13 – Lista das possíveis estações interferentes.....	36

LISTA DE ABREVIATURAS E SIGLAS

ΔF – Fator de correção da rugosidade do terreno

ΔH – Valor de rugosidade média do terreno

A – Atenuação em dB

dB – Relação logarítmica entre duas potências ou intensidades

dB μ – Relação logarítmica entre duas potências ou intensidades referidas a 1 μ V/m

dBd – Relação logarítmica entre duas potências ou intensidades referidas ao ganho de uma antena em relação a um dipolo de meia onda

F – Fator de correção da intensidade de campo

GT – Ganho da antena em dBd

GUI – Interface gráfica com o usuário (Graphical User Interface)

HBT – Cota da base da torre

HNMT – Altura do nível médio do terreno

HT – Altura da antena

NMR – Nível médio da radial

NMT – Nível médio do terreno

UHF – Ultra frequência (Ultra High Frequency)

VHF – Alta frequência (Very High Frequency)

SUMÁRIO

1 OBJETIVO	10
2 TEORIA DE VIABILIZAÇÃO DE CANAL DE TV	11
2.1 Regulamento Técnico	11
2.2 Aspectos Técnicos	11
2.3 Radiais	13
2.4 Determinação dos contornos	14
2.4.1 Contornos de Serviço	15
2.4.2 Contorno Protegido	15
2.4.3 Contorno Interferente	16
2.5 Estudo de Viabilidade.....	16
2.6 Fator de correção	17
3 O MATLAB	19
4 METODOLOGIA UTILIZADA	20
4.1 O TVMAT	20
4.1.1 Inicializando o TVMAT	21
4.1.2 Utilizando o TVMAT	23
4.2 Interpolação	28
4.3 Tratamento de erros	31
5 EXEMPLO PRÁTICO	32
6 CONCLUSÃO.....	48
REFERÊNCIAS	50
ANEXOS.....	51

1 OBJETIVO

Este trabalho apresentará um programa de computador denominado TVMAT, desenvolvido através de uma ferramenta matemática computacional muito conhecida atualmente: o Matlab.

O TVMAT proporcionará ao usuário uma maneira mais fácil e prática de se fazer um Projeto de Viabilização de Canal de TV. Este programa foi realizado baseado na Resolução nº 284 de 7 de dezembro de 2001 (Anexo A) que tem como anexo o Regulamento Técnico para a Prestação do Serviço de Radiodifusão de Sons e Imagens e do Serviço de Retransmissão de Televisão (Anexo B).

O programa foi desenvolvido com duas finalidades. A primeira visou o aprendizado e capacitação do autor nesta importante ferramenta, ampliando assim sua experiência e agregando um instrumento de grande valia na sua vida profissional e acadêmica. E a segunda foi pensando em facilitar a vida do engenheiro de telecomunicações, visto que um Projeto de Viabilização de Canal de TV é muito repetitivo e envolve muitos cálculos.

Serão encontrados nos capítulos posteriores um embasamento teórico sobre o regulamento em questão, uma explanação sobre o programa Matlab, uma abordagem da aplicação do TVMAT, além de um capítulo destinado exclusivamente a um exemplo prático da utilização do programa e sua comparação com um projeto feito sem o uso do mesmo.

2 TEORIA DE VIABILIZAÇÃO DE CANAL DE TV

2.1 Regulamento Técnico

O Projeto de Viabilização de Canal de TV, é um estudo técnico que tem por objetivo definir ou alterar as características técnicas dos serviços de retransmissão de televisão ou inclusão de canal de televisão. Este estudo envolve a verificação da proteção da estação proponente e das estações existentes envolvidas no projeto.

Para um projeto ser aprovado, ele deverá atender a todos os aspectos técnicos da regulamentação em questão, principalmente no que diz respeito a interferências entre estações.

Este capítulo irá mostrar o embasamento teórico presente na regulamentação que foi usada como parâmetro para fazer o programa. Os quesitos referentes às interferências em relação a FM não serão abordados. O programa irá tratar apenas as relações de interferências entre estações de TV.

2.2 Aspectos Técnicos

As siglas VHF e UHF vêm do inglês, *Very High Frequency* e *Ultra High Frequency*, respectivamente, pois a faixa espectral utilizada pela primeira é considerada como sendo de alta frequência, no caso de UHF, ultra.

A faixa espectral reservada para os serviços de Radiodifusão de Sons e Imagens e de Retransmissão de TV em VHF e UHF está compreendida entre 54 e 746 MHz, dividida em 58 canais de 6 MHz de largura de banda (Anexo B, páginas 11, 13 e 14). A faixa de frequência de 608 a 614 MHz, correspondente ao canal 37, está reservada ao Serviço de Radioastronomia internacional, portanto não se deve usá-la.

Na banda de um canal de TV estão presentes três tipos de portadoras: de vídeo, de áudio e de cor. A portadora de vídeo está modulada em AM Vestigial e a de áudio em FM.

Abaixo, a Figura 1 ilustra a faixa de frequência para um canal de TV contendo as suas portadoras com os devidos espaçamentos entre elas.

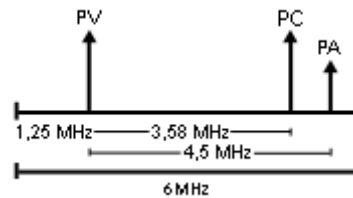


Figura 1 – Faixa de frequência TV.

Existem três tipos de estações de tv:

1. Geradora: precisa ter no mínimo de duas a quatro horas de programação por semana gerada pela própria filial;
2. Retransmissora: recebe o sinal e faz uma limpeza de sincronismo;
3. Repetidora: recebe o sinal, decodifica e remodula, mas não interfere no sincronismo do mesmo.

A estação deverá ser identificada de acordo com o seu propósito. Caso seja uma estação geradora de televisão, ela poderá ser classificada em Especial, Classe A, Classe B ou Classe C. Se for apenas uma estação retransmissora de televisão, ela poderá ser classificada em Classe A, Classe B ou Classe C. Para ambos os propósitos, as estações deverão ser classificadas de acordo com os valores máximos da Potência Efetiva Irradiada (ERP), como mostra a Tabela 1.

Tabela 1 – Classificação das estações.

Classe		Canal	Máxima Potência ERP
TV	RTV		
ESPECIAL	-----	2 - 6	100 kW (20 dBk)
		7 - 13	316 kW (25 dBk)
		UHF	1600 kW(32 dBk)
A	A	2 - 6	10 kW (10 dBk)
		7 - 13	31,6 kW (15 dBk)
		UHF	160 kW (22 dBk)
B	B	2 - 6	1 kW (0 dBk)
		7 - 13	3,16 kW (5 dBk)
		UHF	16 kW (12 dBk)
C	C	2 - 6	0,1 kW (-10 dBk)
		7 - 13	0,316kW (-5 dBk)
		UHF	1,6 kW (2,04 dBk)

A classe da estação é identificada através da radial de maior ERP referida a uma altura do centro de irradiação de 150 metros sobre o nível médio da radial.

2.3 Radiais

A norma técnica exige que sejam traçadas pelo menos 12 radiais com espaçamento angular de 30° e com pelo menos 50 cotas, igualmente espaçadas. As radiais deverão ser traçadas a partir do local da antena e deverão incluir a direção do Norte Verdadeiro. Para o caso do diagrama de irradiação horizontal da estação ser diretivo, as radiais deverão situar-se dentro do(s) setor(res) de irradiação com espaçamento angular de até 15° entre si, nas direções de irradiação, a partir da direção do ganho máximo.

De posse dos perfis traçados, calcula-se o nível médio da radial (NMR) para cada uma delas. O nível médio da radial é a média aritmética de todas as cotas da radial, compreendidas no trecho entre 3 e 15 quilômetros. De posse dos resultados dos NMRs, faz-se necessário calcular o nível médio do terreno (NMT), que é a média aritmética das NMRs, e faz com que o terreno seja um plano de altura conhecida.

Para a utilização das curvas de intensidade de campo, será necessário obter a altura do nível médio do terreno (HNMT), que é definido através da Expressão 1.

$$HNMT = HBT + HT - NMT \quad (1)$$

Onde:

HBT é a altura da base da torre em relação ao nível do mar; e

HT é a altura da antena.

A obtenção da rugosidade do terreno será necessária para o caso do projeto ser inviável. Para efeito de cálculos, a rugosidade de um perfil é dada como um plano de altura conhecida.

“A rugosidade de um perfil é dada pela distância existente em metros entre duas retas. Estas duas retas têm distâncias idênticas a do perfil a ser estudado sendo paralelas entre si e a distância retilínea do perfil. A reta R1 é traçada de tal modo que apenas 10% (10 por cento) da distância total do perfil tenham alturas superiores à mesma. A reta R2 é traçada de tal forma que apenas 10% (10 por cento) da distância total do perfil tenham alturas inferiores à mesma”. (COSTA E SILVA, 1998, p. 49).

A rugosidade é obtida entre o trecho de 10 até 50 quilômetros. A Figura 2, ilustra esta definição.

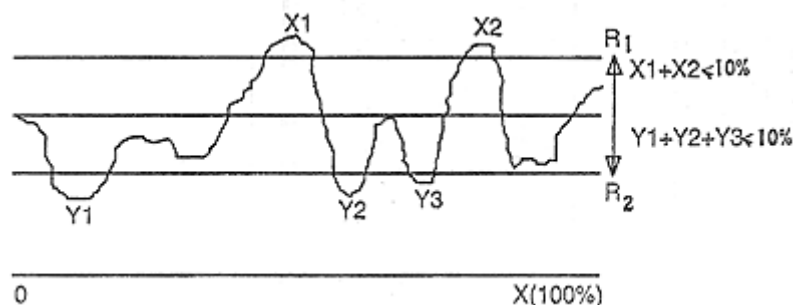


Figura 2 – Determinação da rugosidade de um perfil.

2.4 Determinação dos contornos

Em anexo à regulamentação, existem dois tipos de curvas de intensidade de campo: a curva E (50,50) e a E (50,10).

As curvas E (50,50) serão usadas para calcular as distâncias aos contornos de serviço e contorno protegido, além de servir para corrigir valores de ERP para a altura do nível médio do terreno (HNMT) de 150m, e calcular o valor da intensidade de campo do sinal desejado. Estas curvas fornecem os valores de intensidade de campo excedidos em 50% dos locais, e durante 50% do tempo.

As curvas E (50,10) são utilizadas para determinar as distâncias aos contornos interferentes e calcular a intensidade de campo do sinal interferente. Elas fornecem os valores de intensidade de campo excedidos em 50% dos locais, e durante 10% do tempo.

“Estas curvas indicam os valores de intensidade de campo em dB acima de $1\mu\text{V/m}$ ($\text{dB}\mu$), para uma ERP de 1kW, irradiada de um dipolo de meia onda no espaço livre, que produz uma intensidade de campo não atenuada de 221,4 m V/m (aproximadamente 107 dB μ) a 1 km.” (ANATEL, 2001, p.21).

2.4.1 Contornos de Serviço

A cobertura da estação deverá estar aliada aos seus contornos de serviço. Existem três tipos de contornos: 1, 2 e 3. O contorno 1, ou primário, deverá abranger a maior parte da zona central da localidade; o contorno 2, ou secundário ou urbano, deverá compreender a maior parte possível da sua zona urbana; e o contorno 3, ou rural, está vinculado às estações instaladas em áreas rurais. Todos estes três tipos estão limitados pelos contornos de intensidade de campo E (50,50), em dBμ, conforme a Tabela 2, variando de acordo com a faixa de canais. As distâncias aos contornos de serviço são calculadas para cada radial existente, formando assim um diagrama não isotrópico dos mesmos.

Tabela 2 – Valores de intensidade de campo E (50,50) para os contornos de serviço.

CANAIS	CONTORNO 1 (dBμ)	CONTORNO 2 (dBμ)	CONTORNO 3 (dBμ)
VHF - 2 a 6	74	68	54
VHF - 7 a 13	77	71	60
UHF - 14 a 83	80	74	70

2.4.2 Contorno Protegido

O contorno protegido da estação deverá ser calculado através da curva de intensidade de campo E (50,50) para a HNMT da estação proponente. Ele delimita a sua área de proteção onde deverá estar livre de interferências e corresponde ao lugar geométrico onde o seu sinal tem valores definidos para cada faixa de canal, como mostra a Tabela 3.

Tabela 3 – Valores de intensidade de campo para contornos protegidos.

CANAL	2 a 6	7 a 13	14 a 59
CONTORNO PROTEGIDO E (50,50) em dBμ	58	64	70

2.4.3 Contorno Interferente

O contorno interferente de uma estação é obtido através das curvas de intensidade E (50,10) e pode ser entendido como um lugar geométrico onde os seus sinais têm valores definidos para cada faixa de canal e relações de proteção. Ele é determinado pela aplicação da relação de proteção correspondente a cada situação de interferência sobre o valor de intensidade de campo a proteger.

“A proteção das estações será considerada como assegurada para um serviço livre de interferências quando, no seu contorno protegido, a relação entre o sinal desejado e cada um dos sinais interferentes tiver, no mínimo, o valor indicado na Tabela 4 para VHF e UHF e na Tabela 5 somente para UHF, em função do canal do sinal interferente.” (ANATEL, 2001, p.23).

Tabela 4 – Relações de proteção para VHF e UHF.

INTERFERÊNCIA	CANAL INTERFERENTE	RELAÇÃO DE PROTEÇÃO (dB)
Co-canal	n	+ 45
Co-canal com decalagem	n + n –	+ 28
Canais adjacentes	n + 1	–12
	n – 1	– 6

Tabela 5 – Relações de proteção para UHF.

INTERFERÊNCIA	CANAL INTERFERENTE	RELAÇÃO DE PROTEÇÃO (dB)
Frequência imagem de vídeo	n + 15	+ 3
Frequência imagem de áudio	n + 14	– 6
Oscilador Local	n + 7 e n – 7	– 6
Batimento de FI	n + 8 e n – 8	– 12

2.5 Estudo de Viabilidade

O contorno interferente de uma estação nunca deverá ultrapassar o contorno protegido de outra, e vice-versa. O resultado disso seria um projeto inviável, visto que haveria interferência entre as estações.

Quando os valores obtidos pelas curvas E (50,50) e E (50,10) determinarem a inviabilidade de uma situação proposta em projeto técnico, não significa que a inclusão daquele canal é inviável. As curvas de intensidade de campo admitem uma rugosidade do terreno intrínseca de 50 metros, portanto poderá ser aplicada uma correção neste valor para o caso dele ser maior que 50 metros.

Esta correção (ΔF), relativa à rugosidade do terreno, é determinada pela Expressão 2:

$$\Delta F = C - 0,03\Delta H \left(1 + \frac{f}{300} \right) \quad (2)$$

Onde :

f é a frequência da portadora de vídeo em MHz;

ΔH é a rugosidade média do terreno; e,

C é a constante cujo valor é específico para cada uma das faixas:

1,9 para canais de TV de 2 a 6

2,5 para canais de TV de 7 a 13

4,8 para canais de TV de 14 a 59

Se aplicada a correção, e o projeto ainda apresentar inviabilidade, pode-se ainda tentar provar a sua viabilidade através de um cálculo mais realista, baseado no perfil traçado a partir da estação interferente até a emissora. Tenta-se provar com isso que a atenuação sofrida pelo sinal devido ao relevo é superior ao encontrado com a utilização das curvas.

2.6 Fator de correção

É subtraído dos contornos um determinado fator de correção, responsável pela adequação na utilização das curvas de intensidade E (50,50) e E (50,10), visto que, elas foram construídas com base em valores pré-definidos já discutidos no Item 2.4. Este fator é definido pela Expressão 3.

$$F = 10 \cdot \log\left(\frac{P}{1000}\right) + Gt - (A_{\text{extras}}) \quad (3)$$

Onde:

P é a potência do transmissor em Watts;

Gt é o ganho da antena transmissora em dBd; e

Aextras são as atenuações da estação em questão. Ex.: cabo, conectores, etc.

3 O MATLAB

O Matlab é uma ferramenta computacional poderosa e muito completa. Ele não foi desenvolvido somente para resolução de cálculos matemáticos, como muitos acreditam. Hoje ele dispõe de uma quantidade enorme de recursos adicionais que englobam desde a área de telecomunicações, automação até a área financeira.

O programa supracitado trata todos os dados como sendo vetores, isto faz com que a manipulação dos mesmos seja rápida e prática. A simplicidade e facilidade nos comandos são bem próximos ao qual escrevemos matematicamente, e fazem do Matlab uma ferramenta indispensável e acessível ao leigo.

Ele possui um depurador que executa todas as funções, expressões e rotinas contidas em um arquivo que possui extensão *.m*, e por isso é chamado de Arquivo M. A programação que poderá estar contida nestes arquivos é bem próxima da linguagem C, Basic, ou similares, isto quer dizer que o usuário não terá que aprender uma nova linguagem para poder programar no Matlab.

A programação também poderá ser feita com a ajuda de uma interface gráfica com o usuário (GUI), que é uma ferramenta muito útil para o desenvolvimento de aplicativos. Esta GUI lembra bem as linguagens Visual Basic e Delphi, tornando assim a programação mais fácil. Foi com base nesta GUI que o programa TVMAT foi desenvolvido.

Uma grande diferença da programação no Matlab com outras linguagens é que as variáveis utilizadas nele não precisam ser declaradas ou até mesmo dimensionadas. Isto é, elas poderão ser criadas ou dimensionadas a qualquer momento do programa, sem que isto interfira no seu código, diferentemente de como acontece com outras linguagens em que o programador deverá dar uma atenção especial ao tratamento de dados. Com todas essas facilidades, o programador só precisará dar ênfase ao seu projeto e não à manipulação de dados.

Existem inúmeros programas matemáticos, tais como, o *Mathematica*, *Maple V*, *MathCAD*, dentre outros. Foi escolhido o Matlab para o desenvolvimento do TVMAT por causa de todas as facilidades citadas acima, e além de ser o programa mais conhecido comercialmente.

4 METODOLOGIA UTILIZADA

4.1 O TVMAT

O programa TVMAT, consta basicamente de oito arquivos onde se encontram as rotinas a serem executadas. São quatro arquivos com extensão *.m* e quatro arquivos com extensão *.fig*. São eles: *tv.m*, *tv.fig*, *radiais.m*, *radiais.fig*, *estacoes_interferentes.m*, *estacoes_interferentes.fig*, *sobre.m* e *sobre.fig*. Há ainda arquivos com extensão *.mat*, em que o usuário poderá utilizar quantos forem necessários e que serão criados de acordo com o funcionamento do programa.

Nos arquivos de terminologia *.m* estão as rotinas que compõem todo o funcionamento do TVMAT (Anexo C). Nos arquivos *.mat* estão localizadas as variáveis do programa. Nos arquivos terminados em *.fig*, estão organizados os objetos gráficos que compõem a interface gráfica com o usuário.

Todas as rotinas relacionadas à primeira janela de interface do programa estão localizadas no primeiro arquivo citado. O tratamento dos dados principais da estação proponente está localizado nesta janela, bem como todas as suas sub-rotinas referentes à mesma. O arquivo *radiais.m* inclui as rotinas de edição e informações sobre o relevo do terreno. A rotina de cálculo dos níveis médios das radiais está localizada também neste arquivo. No arquivo *estações_interferentes.m* encontram-se as rotinas de tratamento de dados referente às estações existentes que são consideradas como possíveis interferências à estação proponente e vice-versa. E por fim, o arquivo *sobre.m* constam apenas informações resumidas sobre o programa e o autor.

Todos estes quatro arquivos citados acima se comunicam através de um simples arquivo com terminologia *.mat* que pode ser de qualquer nome, ex.: *projeto1.mat*. Nele estão localizadas todas as variáveis que o programa necessita para o seu funcionamento. Tudo que o usuário entra como dado, é inserido neste arquivo, bem como as variáveis que são criadas de acordo com algumas rotinas. Quando o usuário inicia o programa, ele deverá entrar com um nome para este arquivo, relacionado ao projeto em questão.

Os arquivos com extensão *.fig* não serão citados, pois os mesmos são de código fechado do Matlab e só podem ser abertos e acessados através do mesmo. Como dito anteriormente, consta apenas a interface gráfica do programa.

4.1.1 Inicializando o TVMAT

Todos os arquivos que compõem o programa deverão ser copiados para a pasta de trabalho do Matlab, que por padrão é a *C:\Matlab\work*. O programa deverá ser inicializado a partir da linha de comando do Matlab digitando-se: *tv*. Uma interface gráfica irá se abrir conforme ilustrado na Figura 3. No menu *Arquivo*, o usuário poderá optar por iniciar um novo projeto ou abrir um já arquivado para uma possível alteração nos dados. O menu *Editar* permanece inicialmente indisponível até que o usuário inicie ou carregue um projeto.



Figura 3 – Janela inicial do programa.

Ao iniciar um novo projeto, o programa irá criar um arquivo *.mat* com todas as variáveis que o programa irá necessitar com o nome que o usuário definiu. Estas variáveis serão criadas com seus valores padrões, normalmente zero, e serão atualizadas de acordo com o funcionamento do programa. Segue abaixo uma tabela descritiva (Tabela 6) sobre as variáveis utilizadas.

Tabela 6 – Variáveis do sistema.

temp_file	Nome do arquivo <i>.mat</i>
estacoes	Estrutura de dados referente às estações existentes interferentes
radial	Estrutura de dados referente às radiais
entidade	Nome da entidade requisitante
descricao	Uma breve descrição sobre o projeto
cidade	Cidade da estação proponente
estado	Estado da estação proponente
latitude	Latitude da estação proponente
longitude	Longitude da estação proponente
canal	Canal da estação proponente
decalagem	Decalagem do canal (+ ou -), se tiver
canal_tipo	VHF ou UHF
classe	Especial, A, B ou C
frequencia_video	Frequência da portadora de vídeo
frequencia_som	Frequência da portadora de áudio
potencia	Potência da estação em Watts
ganho_antena	Ganho da antena em dBd
altura_antena	Altura da antena em metros
hbt	Altura da base da torre em metros
atenuacao_cabo	Atenuação do cabo para dB/100m
comprimento_cabo	Comprimento do cabo em m
atenuacao_extras	Atenuações extras em dB
fator_correcao	Fator de correção em dBu
hnmnt	Altura do nível médio do terreno
cpc	Contorno protegido corrigido em dBu
contorno_primario	Contorno primário em dBu
contorno_secundario	Contorno secundário em dBu
contorno_rural	Contorno rural em dBu
contorno_protegido	Contorno protegido em dBu
CPP	Distância do contorno protegido em km
rugosidade_terreno	Rugosidade média do terreno

As variáveis *estacoes* e *radial* são do tipo estrutura. Nelas constam campos relativos a determinados dados referentes a várias estações ou radiais, tais como: *descricao*, *latitude*, *longitude*, *canal*, *decalagem*, *tipo_canal*, *potencia*, *azimute*, *distancia*, *CPE*, *ci*, *CIE*, *CIP*, *cpec*, *ci ec*, *cipc*, *CPE_CIP*, *CPP_CIE*, *interferente*, *fator_correcao*, *tipo_interferencia*, *frequencia_video*, *frequencia_audio*, para a variável de estrutura *estacoes*; e *descricao*, *cota*,

azimute, nmr, hnmr, dist_primario, dist_secundario, dist_rural, radial_dist, rug, para a variável de estrutura *radial*.

4.1.2 Utilizando o TVMAT

A janela principal do programa (Figura 4) será mostrada logo que o usuário determinar um nome para o projeto em questão. Nela constam os campos de dados a serem definidos referentes à estação proponente. Estes campos poderão ser salvos no arquivo de variável do projeto clicando sobre o menu *Arquivo* e logo após em *Salvar Projeto*. O arquivo de variáveis também será atualizado ao abrir a janela de estações ou de radiais, que são acessadas através do menu *Editar -> Estações* e *Editar -> Radiais*, respectivamente.

Figura 4 – Janela principal do programa.

Ao digitar o número do canal, o programa irá definir o seu contorno protegido, primário, secundário e rural, referente ao canal escolhido de acordo com as Tabelas 2 e 3, e irá mostrá-los nos campos específicos ao lado. O campo com as frequências de vídeo e de áudio também serão atualizados de acordo com o canal escolhido, conforme tabela em anexo (Anexo B, páginas 11, 13 e 14).

O fator de correção será atualizado e mostrado no campo específico ao lado após qualquer alteração nos campos de potência, ganho da antena, comprimento e atenuação do

cabo e atenuações extras, desde que os mesmos não estejam vazios. É importante lembrar que no Matlab, por ser um programa de procedência americana, a utilização de vírgulas para as casas decimais se dá através da utilização do ponto.

Estes campos citados e os demais, serão utilizados durante a execução do programa. Depois de preenchidos todos os campos desta janela, será necessário editar as possíveis estações interferentes existentes e as radiais da estação proponente. Ao abrir a janela das estações, pode-se notar que os campos estão bloqueados (Figura 5).

Figura 5 – Janela das estações.

Ao clicar no botão *Adicionar*, uma primeira estação será selecionada e os demais campos serão desbloqueados. Poderão ser adicionadas quantas estações forem necessárias ao projeto, sendo que, quanto mais estações, mais o programa demorará a processar todos os cálculos necessários à verificação das viabilizações. Esta janela contém todas as informações necessárias para que o usuário possa comparar os resultados obtidos pelo programa, com as curvas de intensidade de campo (Anexo B, página 89), tornando assim o programa mais didático. Tal como na janela anterior, ao digitar o número do canal, o programa irá definir o seu contorno protegido; e ao digitar a potência em Watts em seu devido campo, o programa calculará o seu fator de correção baseado na Expressão 4.

$$F = 10 \cdot \log\left(\frac{P}{1000}\right) \quad (4)$$

Sua distância será calculada a partir dos campos de latitude e longitude. Ao digitar os valores referentes a estes campos, o programa irá verificar se os mesmos não se encontram

vazios e também se existem as variáveis de latitude e longitude referentes à estação proponente. Feito isso, o programa irá processar o seu cálculo utilizando como base a Expressão 5, denominada de distância ortodômica.

$$D = 111,1775 \cdot \arccos\{\sin(\text{lat1}) \cdot \sin(\text{lat2}) + [\cos(\text{lat1}) \cdot \cos(\text{lat2}) \cdot \cos(\text{long1} - \text{long2})]\} \quad (5)$$

Onde:

Lat 1 – Latitude da cidade da estação proponente;

Lat 2 – Latitude da cidade da estação existente;

Long 1 – Longitude da cidade da estação proponente; e

Long 2 – Longitude da cidade da estação existente.

Caso as coordenadas da estação proponente não tiverem sido definidas, o programa não calculará a distância. Será necessário retornar à janela anterior e definir estes campos. Após isso será necessário entrar com as coordenadas de todas as estações novamente para que o seu valor de distância seja atualizado.

Algumas informações desta janela por enquanto permanecem sem valores. Os mesmos serão atualizados somente após os cálculos do projeto serem iniciados (último passo do programa).

O usuário poderá inclusive, se desejar, importar as estações de um determinado projeto feito através deste mesmo programa. Será útil quando o usuário for adicionar mais de uma estação proponente na mesma área geográfica. Não será necessário digitar os campos de todas as estações novamente, basta clicar no botão *Importar* e selecionar o arquivo que contém as estações desejadas. Caso queira remover ou adicionar alguma estação, basta clicar nos seus respectivos botões. Ao importar as estações, é possível que alguns campos tais como distância, tipo de interferência, interfere, CIP, CIE, CPE+CIP e CPP+CIE, estejam desatualizados. Isto acontece porque a importação se baseia nos dados da estação do projeto a ser importado, os quais geralmente não são iguais aos dados deste projeto em questão. Para atualizar o dado da distância, basta editar qualquer campo de coordenadas com o mesmo número, assim o valor será atualizado. Para os demais valores será necessário executar o projeto no menu *Arquivo -> Iniciar Projeto*, mas isto só depois das radiais terem sido editadas, caso contrário, o programa irá mostrar uma mensagem de erro.

Ao clicar em *Aplicar*, o programa irá salvar todos estes dados em variáveis no mesmo arquivo *.mat* definido. Ao clicar em *Cancelar*, todas as modificações feitas serão

descartadas. Para ambas as opções, a janela de estações irá se fechar e voltará para a janela principal.

O próximo passo será abrir a janela de radiais para editá-las (Figura 6). Os campos de edição estarão inicialmente bloqueados, iguais ao da janela de estações, bastando apenas clicar no botão *Adicionar* para os mesmos serem desbloqueados.

Figura 6 – Janela das radiais.

Nesta janela constam informações sobre os contornos e níveis médios do terreno para cada radial. Estes contornos são atualizados de acordo com a execução do projeto na janela principal (*Arquivo -> Iniciar Projeto*), portanto, quando as cotas são adicionadas eles não serão atualizados.

Como na janela de estações, aqui o usuário também poderá importar as cotas de um outro projeto já feito através deste programa. Isto poderá ser útil no caso de um estudo de viabilidade para diferentes canais de TV. Contudo, ao importar as radiais de um outro projeto, os dados de contorno primário, secundário, rural e a HNMT estarão desatualizados. Estes dados dependem do número do canal para definição dos contornos e, da cota da torre e altura da antena para atualizar a HNMT da estação proponente. Estes campos foram definidos anteriormente na janela principal e geralmente são diferentes dos campos do projeto proposto.

Ao inserir uma cota, o programa irá verificar se a mesma está definida para uma distância acima de três quilômetros. Isto porque para o cálculo do nível médio do terreno, a distância deverá estar compreendida em um limite de três a quinze quilômetros. Depois de verificar sua consistência, o programa irá calcular o nível médio do terreno e a sua rugosidade e mostrará em seus devidos campos.

Ao clicar em *Aplicar*, o programa irá salvar todos estes dados em variáveis no mesmo arquivo *.mat* definido anteriormente. Ao clicar em *Cancelar*, todas as modificações feitas serão descartadas. Para ambas, a janela de radiais irá se fechar e voltará à janela principal.

O último passo para a conclusão do projeto é proceder com a execução da viabilização do canal escolhido. Como dito anteriormente, basta clicar sobre o menu *Arquivo* e logo depois em *Iniciar Projeto* para que o programa possa dar início a todos os cálculos e processamentos necessários.

Primeiramente, ele irá salvar todos os dados da janela principal no arquivo de variáveis já definido, e irá fazer uma checagem delas à medida em que há necessidade de usá-las nos cálculos seguintes. As cotas serão checadas uma a uma, e caso alguma delas não tenha sido definida, o programa irá dar uma mensagem de erro e interromperá o processo dos demais cálculos. O usuário terá que corrigir o erro editando alguma variável que não tenha sido definida antes, para logo após executar os cálculos do projeto novamente.

É neste momento que ele irá definir a HNMR, mas para isso, primeiramente, irá verificar se existem as variáveis da HBT e da HT, pois são os campos necessários para o cálculo da HNMR (Expressão 6). Para esta equação será necessário também estar presente o nível médio de cada radial (NMR), mas isto já foi checado na passagem anterior.

$$HNMR = HBT + HT - NMR \quad (6)$$

Tendo como base o NMR, o programa calcula o nível médio do terreno como sendo a média aritmética da NMR de cada radial. Utiliza-se o resultado deste cálculo para se obter a altura do nível médio do terreno, definida pela Expressão 1.

Esta expressão será válida para uma HNMT de 30 a 1000 metros, devido à limitação das tabelas de intensidade de campo definidas pela norma (Anexo B, página 89). Se o valor da HNMT for menor do que o seu limite inferior, ele será definido como sendo o próprio limite inferior, que no caso será de 30 metros. Caso o valor da HNMT calculado ultrapasse os 1000 metros, este será substituído pelo seu limite superior de 1000 metros.

Depois de obtida a HNMT, o próximo passo do programa é calcular os contornos primários, secundários e rurais para cada radial a partir da HNMT em questão, e também o contorno protegido da estação proponente. Estes contornos são obtidos a partir das curvas de intensidade de campo definidas pela norma, que se diferem de acordo com o canal definido. O programa utiliza uma matriz de dados com alguns valores da curva em questão e os interpola para a obtenção dos valores adequados. Este processo de interpolação será explicado adiante com mais detalhes.

Em seguida o programa irá comparar os canais das estações existentes com o canal da estação proponente para verificar qual o tipo de interferência que poderá ocorrer entre as estações e colocará a sua referência na variável *estacoes* no campo *tipo_interferencia*. De posse dessa referência, o programa irá calcular o contorno interferente corrigido de cada estação combinado com o fator de correção da estação existente e a relação de proteção de acordo com a estação proponente. É importante ressaltar que este valor de campo será limitado conforme a matriz de dados da curva de intensidade inerente ao programa. Seu limite inferior é de -15 dBμ e valor máximo de 80 dBμ. Caso estes valores se excedam, toma-se como base os valores limites para cada caso. Definido o contorno interferente, sua distância se dará através das curvas de intensidade E (50,10) através da interpolação de dados.

O último passo do programa será determinar a viabilidade de um canal. De acordo com os resultados dos contornos protegidos e interferentes das estações, obtidos através das rotinas acima descritas, o programa dará o seu parecer conclusivo comparando as suas distâncias. Caso haja inviabilidade, ele aplicará uma devida correção na rugosidade do terreno e fará novamente o processo de obtenção das distâncias dos contornos protegidos e interferentes das estações, conforme explicado no Item 2.5. Será mostrado em seguida a janela de estações para verificação da viabilidade de cada estação.

Todos esses resultados são guardados através de variáveis no arquivo de destino, e mostrados em suas respectivas janelas no programa.

4.2 Interpolação

As curvas de intensidade de campo foram construídas a partir de um modelo experimental e portanto, não seguem uma função específica. Com elas podemos obter tanto a distância para um nível de intensidade de campo de um determinado contorno, quanto o próprio nível, através de uma distância conhecida.

Faz-se necessário o uso dos dados obtidos por meio da interpolação dos mesmos, já que as curvas não obedecem a uma função específica. Método este, presente no Matlab e que será de grande valia para a execução do programa em questão.

Interpolação é um método para estimar valores a partir de uma matriz de dados pré-estabelecida. Com o Matlab é possível obter interpolações em qualquer número de dimensões, usando estruturas multidimensionais. Existem várias curvas de intensidade de campo em um mesmo gráfico, o que faz com que a utilização da interpolação unidimensional seja descartada. Uma interpolação bidimensional será utilizada a partir dos eixos das ordenadas e abscissas conhecidos.

No eixo das abscissas, constam os valores da altura da antena transmissora em metros. No eixo das ordenadas encontram-se os valores de intensidade de campo em dB. No eixo paralelo ao das ordenadas estão localizados os valores referentes à distância em quilômetros. Esta distância que se deseja obter irá variar de acordo com a altura da antena e a intensidade de campo.

Será demonstrado em seguida como o Matlab irá fazer a interpolação dos dados. Como exemplo tem-se os seguintes vetores referentes à altura da antena transmissora em relação a HNMT (Expressão 7) e intensidade de campo (Expressão 8):

$$H=[30 \ 50 \ 100 \ 150 \ 300 \ 500 \ 1000] \quad (7)$$

$$E=[-15 \ -10 \ 0 \ 10 \ 20 \ 40 \ 60 \ 80 \ 100] \quad (8)$$

Para cada combinação dos vetores acima há uma distância resultante do gráfico em questão. Veja que o vetor H só aceitará valores a partir de 30 metros, referente à altura mínima limitada pelo gráfico. Apesar do gráfico aceitar uma altura máxima de 1600 metros, a interpolação só irá ocorrer para valores H até 1000 metros. Valor este, suficiente para a execução do projeto, afinal, uma altura superior a este valor é sinal de que alguma coisa está errada.

De acordo com a Expressão 8, o vetor de intensidade de campo deverá estar compreendido entre -15 e 100 dB para contorno protegido, e -15 e 80 dB para contorno interferente, devido à limitação das curvas de intensidade de campo. Para efeito de demonstração toma-se com exemplo a matriz de distância (Expressão 9) para o contorno protegido E (50,50) e um intervalo de canais de 2 a 6 VHF.

$$\begin{aligned}
 \text{CP} = & \begin{bmatrix} 290 & 250 & 185 & 130 & 75 & 29 & 10 & 3.5 & 0.5 \\
 & 293 & 257 & 190 & 135 & 83 & 38 & 12 & 4.5 & 0.6 \\
 & & 305 & 265 & 205 & 144 & 97 & 51 & 15 & 6 & 1.1 \\
 & & & 310 & 275 & 210 & 150 & 105 & 59 & 20 & 7.5 & 1.5 \\
 & & & & 35 & 290 & 225 & 170 & 125 & 70 & 29 & 10 & 1.8 \\
 & & & & & 360 & 310 & 245 & 185 & 85 & 37.5 & 12 & 12 & 2 \\
 & & & & & & 400 & 370 & 273 & 215 & 165 & 105 & 57 & 14 & 2 \end{bmatrix}
 \end{aligned}
 \tag{9}$$

Esta matriz foi definida através da obtenção de valores nas curvas de intensidade de campo para os vetores H e E. Para valores não definidos nesta matriz, e dentro dos limites pré-estabelecidos acima, o Matlab irá fazer a interpolação adequada e mostrar o resultado. A Tabela 7 mostra como o Matlab irá trabalhar com a matriz em questão.

Tabela 7 – Vetor intensidade de campo x altura da antena x distância.

E/H	-15	-10	0	10	20	40	60	80	100
30	290	250	185	130	75	29	10	3.5	0.5
50	293	257	190	135	83	38	12	4.5	0.6
100	305	265	205	144	97	51	15	6	1.1
150	310	275	210	150	105	59	20	7.5	1.5
300	35	290	225	170	125	70	29	10	1.8
500	360	310	245	185	85	37.5	12	12	2
1000	400	370	273	215	165	105	57	14	2

Há quatro modelos de interpolações possíveis no Matlab: linear, cúbico, splines cúbicos e vizinho mais próximo. A Tabela 8 mostra os resultados de uma comparação dos modelos de interpolação citados para E=5dB e H=150m.

Tabela 8 – Relação dos modelos de interpolação.

Linear	CP = 180 dBu
Cúbico	CP = 178.8586 dBu
Splines cúbicos	CP = 178.8586 dBu
Vizinho mais próximo	CP = 150 dBu

O modelo linear é usado como padrão no Matlab. Ele interpola linearmente os pontos dados na matriz. Os modelos cúbicos e splines cúbicos são muito próximos entre si, pois calculam através de polinômios de grau três, ainda que de modos diferentes. No exemplo apresentado ocorreu resultarem em um mesmo valor, mas isso foi devido ao fato da matriz possuir poucos dados. Para o caso de muitos dados, estes valores terão resultados diferentes, mas ainda assim bem parecidos. O modelo vizinho mais próximo, simplesmente pega o valor

mais próximo da matriz aos valores desejados. Para o problema em questão, este modelo não é viável, pois o seu resultado foi muito abaixo do esperado, ocasionando um erro relevante. Apesar de não servir para este problema, ele ainda é muito usado para casos onde existem muitos dados a serem processados em pouco tempo disponível, isto é, onde necessitam de uma maior velocidade de processamento.

Verifica-se que o resultado do modelo linear, comparado com os modelos cúbico e splines cúbicos são bem próximos, chegando a ter uma diferença desprezível. Considerando o fato de que estes dois últimos modelos citados necessitam de um maior tempo de processamento, escolheu-se então o modelo de interpolação linear como o mais adequado ao programa em questão.

4.3 Tratamento de erros

Sem dúvida alguma, a parte mais difícil e demorada de se fazer um programa é o tratamento de erros. Nem sempre é possível identificar todos os erros que possam ocorrer durante o funcionamento de um programa, mas deve-se ao máximo prevê-los e tentar evitá-los. Eles são tratados através de rotinas que verificam a validade de um determinado campo, limitam uma certa variável de acordo com valores pré-estabelecidos, ou simplesmente verificam a existência de alguma variável.

Não há programa sem erros. Pode-se constatar isso a todo o momento, como por exemplo no sistema operacional Windows da Microsoft® e as suas famosas telas azuis que aparecem quando o sistema trava. Travamento este ocorrido geralmente devido a um ou mais erros que não foram analisados adequadamente.

Busca-se ao máximo o tratamento adequado destes erros justamente para fazer um programa, mais prático, ágil e funcional aos seus usuários.

5 EXEMPLO PRÁTICO

Neste capítulo será apresentado um exemplo prático fictício para melhor compreensão da forma de como o TVMAT funciona. Será tomado como exemplo uma emissora a ser instalada no canal 14 UHF na cidade de Itapitanga – BA. Seguem abaixo os dados da estação proponente:

Tabela 9 – Dados da Estação Proponente

Entidade Requerente:	Televisão ABC S/A
Descrição:	Viabilização do canal 14 em Itapitanga
Cidade:	Itapitanga
Estado:	BA
Latitude:	14°25'20" S
Longitude:	39°33'53" O
Canal:	14 UHF
Potência do Transmissor:	45W
Classe:	C
Frequência de Vídeo:	471,25 MHz
Frequência de Áudio:	475,75 MHz
Altura da Base da Torre:	630 metros
Altura da Antena:	25 metros
Ganho da Antena:	3 dBd
Comprimento do Cabo:	30 metros
Atenuação do Cabo:	5 dB/100m
Atenuações Extras:	1 dB

Como primeiro passo, deverá ser calculado o Fator de Correção através da Expressão 3, necessário à utilização nas curvas de intensidade de campo. Este valor foi o mesmo encontrado pelo TVMAT e armazenado na variável *fator_correcao* (Expressão 10). A Figura 7, ilustra o exemplo.

$$F = 10 \cdot \log\left(\frac{45}{1000}\right) + 3 - (5 \cdot 30/100 + 1) = -12,9679 \text{ dB} \quad (10)$$

TVMAT - Projeto de Viabilização de Canal de TV

Arquivo Editar Ajuda

Dados:

Entidade Requerente:

Descrição:

Cidade: Estado:

Latitude: g m s

Longitude: g m s

Canal: Decalagem: ☒ Nao ☐ + ☐ -

Potencia Transmissor: W Classe:

Frequencia: Video: MHz Som: MHz

Altura Base Torre: m Altura Antena: m

Ganho Antena: dBd Comp. Cabo: m

Atenuação Cabo: dB / 100m Extras: dB

Contornos:

Fator de Correção:	-12.96	dBu
Contorno Protegido:	70	dBu
Contorno Primario:	80	dBu
Contorno Secundario:	74	dBu
Contorno Rural:	70	dBu
CPC:	0	dBu
CPP:	0	Km
HNMT:	0	m
Rugosidade Media:	0	m

Figura 7 – Tela principal do TVMAT mostrando exemplo.

As cotas das radiais são exemplificadas na Tabela 10. Após isso, serão calculadas as NMRs através da Expressão 11, as HNMRs através da Expressão 12, e as rugosidades das radiais através da sua definição no Item 2.3.

Os resultados obtidos referentes as NMRs e HNMRs foram iguais para ambos e também iguais ao calculado pelo TVMAT, visto que o cálculo das NMRs envolve uma expressão de média aritmética simples e são ilustrados na Tabela 10. As HNMRs também são obtidas através de uma expressão simples. As rugosidades obtidas pelo TVMAT (Tabela 11) não foram muito precisas devido ao espaçamento das cotas. Elas serão mais precisas quanto mais as distâncias entre as cotas tenderem a zero. A Figura 8 ilustra o exemplo no TVMAT.

$$NMR = \frac{\sum \text{cotas}}{n^{\circ} \text{ de cotas}} (3 \text{ a } 15 \text{ km}) \quad (11)$$

$$HNMR = HBT + HT - NMR \quad (12)$$

Tabela 10 – Cotas das Radiais, suas NMRs e HNMRs.

DIST.	0°	30°	60°	90°	120°	150°	180°	210°	240°	270°	300°	330°
3.0	640	560	680	640	640	520	520	560	600	720	520	560
3.2	640	560	680	640	600	480	560	560	600	680	520	560
3.4	640	600	640	640	560	480	600	560	600	680	520	520
3.6	600	640	640	640	560	480	640	560	600	680	520	520
3.8	600	640	640	640	560	480	680	560	600	720	520	520

4.0	600	640	640	640	560	480	680	560	600	720	560	520
4.2	600	680	600	640	520	480	520	560	600	680	600	520
4.4	560	640	600	640	520	520	560	520	600	680	560	520
4.6	560	640	600	640	520	560	560	560	560	680	520	520
4.8	480	600	600	640	520	520	560	560	560	640	520	520
5.0	480	600	600	640	520	560	560	560	560	640	520	520
5.2	480	600	600	640	480	560	600	560	560	640	520	520
5.4	520	560	640	600	480	560	600	560	560	680	560	480
5.6	520	560	640	600	480	520	600	560	560	680	560	520
5.8	520	560	680	560	480	520	560	560	560	640	560	520
6.0	480	560	680	520	480	520	560	600	560	600	560	520
6.2	480	560	680	520	480	560	560	600	560	600	560	520
6.4	480	520	640	520	440	560	560	600	560	600	560	520
6.6	480	520	640	480	470	560	560	600	560	560	560	520
6.8	480	520	600	480	470	560	560	600	560	520	560	520
7.0	480	520	560	520	470	560	560	600	560	520	520	520
7.2	480	520	520	480	470	560	560	600	560	520	560	520
7.4	480	520	520	480	470	600	560	640	560	480	560	520
7.6	480	520	520	480	470	600	560	680	560	480	560	520
7.8	480	520	480	480	470	600	520	680	560	480	520	480
8.0	480	560	480	480	520	600	520	680	560	480	520	480
8.2	480	560	480	480	520	600	520	680	560	480	520	480
8.4	480	560	440	480	520	600	520	680	560	480	520	480
8.6	480	560	440	440	520	600	520	680	560	480	520	480
8.8	480	520	440	440	520	600	560	680	560	480	520	480
9.0	440	520	440	440	520	600	560	680	560	480	520	480
9.2	440	520	440	520	520	600	520	640	560	480	520	480
9.4	440	520	440	440	520	640	520	680	560	440	520	480
9.6	440	520	440	440	520	640	520	680	560	480	520	480
9.8	440	520	440	440	520	600	520	640	560	480	520	440
10.0	440	480	440	440	480	600	520	640	520	480	480	440
10.2	440	480	440	440	480	600	520	600	560	480	480	440
10.4	440	480	440	440	480	600	520	600	560	480	480	440
10.6	440	480	440	440	440	600	560	600	560	440	480	440
10.8	440	480	440	440	440	600	600	600	560	440	480	440
11.0	440	480	440	440	440	600	600	600	560	440	440	440
11.2	440	480	440	440	440	600	600	600	560	480	440	440
11.4	440	520	440	480	440	640	600	600	560	480	440	440
11.6	440	520	440	480	440	640	600	600	560	480	440	440
11.8	440	520	480	480	480	640	560	600	560	480	440	440
12.0	440	520	480	480	480	640	560	640	520	480	480	440
12.2	440	520	480	480	480	640	560	640	560	480	480	440
12.4	440	520	520	480	480	600	600	640	560	480	480	440
12.6	440	520	520	480	480	600	600	680	560	480	480	440
12.8	440	520	520	440	480	600	600	640	560	480	480	440
13.0	440	520	560	480	520	600	600	640	520	480	480	480
13.2	480	520	560	480	520	600	640	640	520	520	480	480
13.4	480	520	560	480	520	600	640	640	480	520	480	480
13.6	520	520	560	480	520	600	600	640	480	520	480	480
13.8	520	520	520	480	520	600	600	640	480	520	480	480
14.0	520	480	520	480	520	600	600	640	480	520	480	480
14.2	520	440	520	480	520	600	600	640	480	520	480	480
14.4	520	440	520	480	520	600	600	600	480	520	480	500
14.6	520	440	520	480	560	600	600	600	480	520	480	500

14.8	560	440	520	480	560	600	600	600	480	520	480	500
15.0	560	440	520	480	560	600	600	600	480	520	480	500
NMR	493,11	533,11	535,08	510,81	503,11	578,36	572,45	613,77	550,81	541,63	509,50	487,21
HNMR	161,89	121,89	119,92	144,18	151,89	76,64	82,54	41,23	104,18	113,36	145,49	167,79

Tabela 11 – Resultado da Rugosidade em metros Obtida pra cada Radial.

	0°	30°	60°	90°	120°	150°	180°	210°	240°	270°	300°	330°
RUG:	80	80	120	40	120	40	80	40	80	80	40	60

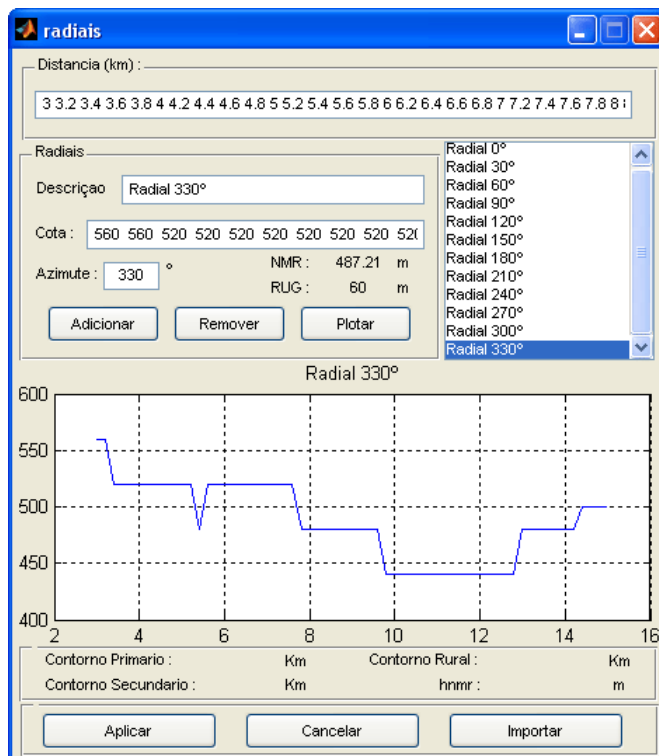


Figura 8 – Exemplo ilustrado no TVMAT.

O HNMT será calculado a partir da Expressão 1, onde temos uma NMT de 535,75 metros conforme a Expressão 12.

$$HNMT = 630 + 25 - 535,75 = 119,25 \quad (12)$$

Os contornos primários, secundários e rurais das radiais, deverão ser obtidos através dos valores referentes na Tabela 2 e o contorno protegido através da Tabela 3. Como o canal exemplificado é o 14 UHF, então os seus valores são: 80, 74, 70 e 70 dBμ, respectivamente. Aplica-se a estes valores o fator de correção calculado anteriormente e, através da Expressão 13, obtêm-se os novos valores dos contornos corrigidos: 92,97; 86,97; 82,97; e 82,97 dBμ.

$$C_{\text{corrigido}} = C_{\text{nominal}} - F \quad (13)$$

Estes contornos corrigidos são utilizados junto com as HNMRs de cada radial para obter os valores das distâncias em quilômetros aos seus referidos contornos nas curvas de intensidade de campo E (50,50). A Tabela 12 ilustra uma comparação entre os valores do exemplo calculado e os obtidos pelo TVMAT.

Tabela 12 – Comparação entre os contornos (em Km) calculados e obtidos no TVMAT.

	0°	30°	60°	90°	120°	150°	180°	210°	240°	270°	300°	330°
Contorno Primário Calculado	3,5	3,2	3,2	3,5	3,5	2,5	2,6	1,5	2,8	3,0	3,2	3,4
Contorno Primário TVMAT	3,57	3,17	3,17	3,43	3,51	2,49	2,68	1,80	2,99	3,11	3,43	3,60
Contorno Secundário Calculado	5,2	4,5	4,5	4,9	5,0	2,8	3,9	2,8	4,2	4,3	4,9	5,2
Contorno Secundário TVMAT	5,31	4,71	4,70	5,11	5,23	3,76	4,01	2,83	4,43	4,60	5,10	5,36
Contorno Rural Calculado	6,5	5,8	5,8	6,2	6,3	4,8	4,9	3,7	5,5	5,8	6,2	6,4
Contorno Rural TVMAT	6,48	5,73	5,72	6,23	6,37	4,61	4,90	3,52	5,39	5,60	6,22	6,53

Pode-se perceber que as distâncias aos contornos obtidas não são totalmente iguais, mas são bem próximas umas das outras. Esta diferença pode ser explicada devido à precisão utilizada nas curvas de intensidade de campo do exemplo calculado ser menor do que a utilizada pelo TVMAT. A precisão do programa é muito superior devido à utilização da interpolação como método para calcular as distâncias.

Os cálculos das interferências são feitos através de uma lista das possíveis estações existentes. Esta lista está ilustrada na Tabela 13.

Tabela 13 – Lista das possíveis estações interferentes.

Localidade	Coordenadas Geográficas	Canal	Potência (W)	Dist.(Km)
Almadina – BA	14°42'18" S 39°38'13" O	13 VHF	100	32,38
Buerarema – BA	14°57'33" S 39°17'58" O	14 UHF	20	66,16

Ipiaú – BA	14°07'35" S 39°45'10" O	14+ UHF	54	38,63
Aurelino Leal – BA	14°18'59" S 39°19'38" O	21 UHF	20	28,15
Inema – BA	14°30'16" S 39°28'25" O	29 UHF	200	13,40

Para cada estação são feitos estudos para determinar sua viabilidade. No exemplo atual, serão feitos 6 estudos de viabilidade que serão mostrados em seguida.

1. Estudo: Itapitanga e Almadina.

- Verificação de interferência de Almadina em Itapitanga:

O contorno protegido corrigido da estação proponente já foi calculado e o seu valor é de 82,97 dBμ. Com base neste valor e no HNMT calculado temos a distância ao seu contorno protegido a partir das curvas de intensidade de campo E (50,50). Estas distâncias estão definidas na Expressão 14, referente ao exemplo calculado, e na Expressão 15, referente ao obtido pelo TVMAT.

$$\left. \begin{array}{l} 82,97dB\mu \\ 119,25m \\ E(50,50) \end{array} \right\} CPP = 5,7Km \quad (14)$$

$$\left. \begin{array}{l} 82,97dB\mu \\ 119,25m \\ E(50,50) \end{array} \right\} CPP = 5,71Km \quad (15)$$

O canal da estação existente em Almadina é o 13 VHF, portanto poderá ocasionar em uma interferência de canal adjacente inferior. Está prevista pela norma uma relação de proteção de -6 dB conforme Tabela 4. A distância ao contorno interferente da estação existente será calculada com base no sinal útil menos a sua relação de proteção em referência à estação proponente e menos o seu fator de correção (Expressão 16), demonstrado na Expressão 17.

$$FCE = 10 \cdot \log\left(\frac{P}{1000}\right) = 10 \cdot \log\left(\frac{100}{1000}\right) = -10dB \quad (16)$$

$$SI = SU - RP - FCE = 70 - (-6) - (-10) = 86dB\mu \quad (17)$$

O TVMAT limitou o sinal interferente encontrado para 80 dB, isto ocorreu por causa do limite existente nas curvas de intensidade de campo. Ele limita a distância mínima a 4,5 Km para a HNMT em questão, no caso 150 metros.

Com estas distâncias pode-se verificar a sua viabilidade no estudo em questão através da Expressão 18. O seu resultado é demonstrado na Expressão 19 para o exemplo calculado e na Expressão 20 obtida pelo TVMAT.

$$CPP + CIE < \text{Distância entre as estações} \quad (18)$$

$$5,7 + 4,5 = 10,2 < 32,38 \text{ Km} \quad (19) \quad 5,71 + 4,5 = 10,21 < 32,38 \text{ Km} \quad (20)$$

Observa-se que a estação existente não interfere na estação proponente em ambos os cálculos. Pode-se verificar também, que o resultado obtido pelo TVMAT foi bem próximo ao do exemplo calculado.

- Verificação de interferência de Itapitanga em Almadina:

O contorno protegido da estação existente é definido através da Tabela 3, e no caso, seu valor é de 64 dBμ, pois se trata do canal 13 VHF. Aplicado o seu fator de correção (FCE) temos um contorno protegido corrigido no valor de 74 dBμ. Como não temos a HNMT da referida estação usaremos o valor de 150 metros. Os resultados obtidos são mostrados nas Expressões 21 e 22, para exemplo calculado e obtido através do TVMAT, respectivamente.

$$\left. \begin{array}{l} 74dB\mu \\ 150m \\ E(50,50) \end{array} \right\} CPE = 11Km \quad (21)$$

$$\left. \begin{array}{l} 74dB\mu \\ 150m \\ E(50,50) \end{array} \right\} CPE = 13,45Km \quad (22)$$

Calcula-se agora o contorno interferente da estação proponente com base na Expressão do sinal interferente (23).

$$SI = SU - RP - FC = 70 - (-6) - (-12,97) = 88,97dB\mu \quad (23)$$

Conforme explicado anteriormente, todos os valores de campos interferentes obtidos acima de 80 são limitados a este valor. Então, o contorno interferente da estação proponente (CIP) será de 4,192 Km para a HNMT de 119,25 metros. O cálculo da viabilidade

deverá obedecer à Expressão 24 e tem como resultado as Expressões 25 e 26, para exemplo calculado e obtido pelo TVMAT, respectivamente.

$$CPE + CIP < \text{Distância entre as estações} \quad (24)$$

$$11 + 4,192 = 15,19 < 32,38 \text{ Km} \quad (25) \quad 13,45 + 4,192 = 17,64 < 32,38 \text{ Km} \quad (26)$$

Observa-se que em nenhum caso a distância limite foi atingida, portanto a estação em Itapitanga não interfere em Almadina e vice-versa. Em consequência disso o projeto é viável.

2. Estudo: Itapitanga e Buerarema.

- Verificação de interferência de Buerarema em Itapitanga:

Não será necessário calcular novamente o contorno protegido da estação proponente, pois isto já foi feito. Será calculado apenas o contorno interferente da estação existente. A estação citada possui o canal 14 UHF, portanto o mesmo canal da estação proponente, o que poderá ocasionar uma interferência do tipo co-canal. A norma admite uma relação de proteção no valor de 45 dB conforme a Tabela 4. Com base na Expressão 16, calcula-se o FCE da estação proponente que tem uma potência de 20 W, o que resultará em uma FCE de -16,99 dB.

De acordo com o sinal útil, com a relação de proteção e com o fator de correção da estação, tem-se o contorno interferente da estação existente, demonstrado na Expressão 27.

$$SI = SU - RP - FCE = 70 - (45) - (-16,99) = 41,99 \text{ dB}\mu \quad (27)$$

De posse do contorno interferente iremos calcular a sua distância utilizando as curvas de intensidade de campo E (50,10) e uma HNMT de 150 metros (Expressões 28 e 29).

$$\left. \begin{array}{l} 41,99 \text{ dB}\mu \\ 150 \text{ m} \\ E(50,10) \end{array} \right\} CIE = 53 \text{ Km} \quad (28)$$

$$\left. \begin{array}{l} 41,99 \text{ dB}\mu \\ 150 \text{ m} \\ E(50,10) \end{array} \right\} CIE = 54,61 \text{ Km} \quad (29)$$

Com todos os dados em mãos, pode-se calcular a sua viabilidade através da Expressão 18 e como resultado as Expressões 30 e 31.

$$5,7 + 53 = 58,7 < 66,13 \text{ Km} \quad (30) \quad 5,71 + 54,61 = 60,32 < 66,13 \text{ Km} \quad (31)$$

De acordo com os dados apresentados e a partir da demonstração feita, o projeto resulta em sua viabilidade.

- Verificação de interferência de Itapitanga em Buerarema:

O contorno protegido corrigido da estação existente, foi calculado como a soma de seu valor resultante da Tabela 3 (70 dBμ) com o seu fator de correção (-16,99 dB), o que nos resulta em 86,99 dBμ. Este valor será usado com a HNMT de 150 metros para se obter a sua distância (Expressões 32 e 33).

$$\left. \begin{array}{l} 86,99dB\mu \\ 150m \\ E(50,50) \end{array} \right\} CPE = 4,9Km \quad (32)$$

$$\left. \begin{array}{l} 86,99dB\mu \\ 150m \\ E(50,50) \end{array} \right\} CPE = 5,2Km \quad (33)$$

Calcula-se o contorno interferente da estação proponente com base no sinal interferente (Expressão 23). Aplicando-se o fator de correção da estação proponente (-12,97dB), a relação de proteção em questão (45 dB) e o sinal útil, será obtido na Expressão 34 o seu valor de contorno.

$$SI = 70 - (45) - (-12,97) = 37,97dB\mu \quad (34)$$

Calcula-se então a distância ao seu contorno (Expressões 35 e 36). É calculado em seguida, nas Expressões 37 e 38, os cálculos de viabilidade baseados na Expressão 24.

$$\left. \begin{array}{l} 37,97dB\mu \\ 119,25m \\ E(50,10) \end{array} \right\} CIP = 58Km \quad (35)$$

$$\left. \begin{array}{l} 37,97dB\mu \\ 119,25m \\ E(50,10) \end{array} \right\} CIP = 62,39Km \quad (36)$$

$$4,9 + 58 = 62,9 < 66,13 \text{ Km} \quad (37) \quad 5,2 + 62,39 = 67,59 > 66,13 \text{ Km} \quad (38)$$

Apesar de ser pequena a diferença entre o somatório das distâncias (4,39 Km) ela causou uma certa dúvida na viabilidade do projeto. A precisão neste caso deverá ser um fator importante a ser levado em consideração, visto que, a precisão a “olho nú” nas curvas de

intensidade, foi responsável pela viabilidade do projeto, e a precisão através da interpolação foi responsável pela inviabilidade do mesmo. Deve-se levar em conta que a precisão pelo método da interpolação é bem mais precisa que o outro caso e portanto o projeto deverá ser considerado inviável.

Como explicado no Item 2.5, pode-se corrigir a rugosidade do terreno para depois verificar novamente a sua viabilidade. Essa correção poderá ser feita através da Expressão 2 para as seguintes constantes:

$$C = 4,8; \Delta H = 71,67 \text{ m} ; e f = 471,25 \text{ MHz} \quad (39)$$

$$\Delta F = 4,8 - 0,03 \cdot 71,67 \cdot \left(1 + \frac{471,25}{300}\right) = -0,73 \text{ dB} \quad (40)$$

Como somente há interferência da estação proponente na existente, o valor do campo do contorno interferente da estação proponente deverá ser atualizado (Expressão 41). Como não se temo valor da rugosidade média do terreno da estação existente, não pode-se atualizar o valor do campo protegido desta estação, como é o caso deste exemplo prático.

$$CIP = CIP - \Delta F = 37,97 - (-0,73) = 38,69 \text{ dB}\mu \quad (41)$$

Abaixo, através das Expressões 42 e 43, é demonstrado o cálculo da distância ao contorno interferente da estação proponente com a sua devida correção relativa à rugosidade. A partir destes dados obtidos será calculada a viabilidade do projeto na Expressão 24, resultando nas Expressões 44 e 45. Sendo assim, considerando os cálculos, o projeto foi considerado viável.

$$\left. \begin{array}{l} 38,69 \text{ dB}\mu \\ 119,25 \text{ m} \\ E(50,10) \end{array} \right\} CIP = 55 \text{ Km} \quad (42)$$

$$\left. \begin{array}{l} 38,69 \text{ dB}\mu \\ 119,25 \text{ m} \\ E(50,10) \end{array} \right\} CIP = 59,49 \text{ Km} \quad (43)$$

$$4,9 + 55 = 59,9 < 66,13 \text{ Km} \quad (44)$$

$$5,2 + 59,49 = 64,69 < 66,13 \text{ Km} \quad (45)$$

3. Estudo: Itapitanga e Ipiaú.

- Verificação de interferência de Ipiaú em Itapitanga:

A estação interferente localizada em Ipiaú utiliza o canal 14 decalado para mais, portanto, tem-se uma possível interferência de co-canal com decalagem, o que define uma relação de proteção no valor de 28 dB. Seu FCE foi calculado para a potência de 54 W e resultado em -12,68 dB. Seu sinal interferente será calculado na Expressão 46.

$$SI = SU - RP - FCE = 70 - (28) - (-12,68) = 54,68dB\mu \quad (46)$$

As distâncias aos contornos interferentes são calculadas nas Expressões 47 e 48. De acordo com a Expressão 18, observa-se o cálculo de viabilidade, com o resultado sendo demonstrado nas Expressões 49 e 50.

$$\left. \begin{array}{l} 54,68dB\mu \\ 150m \\ E(50,10) \end{array} \right\} CIE = 30Km \quad (47)$$

$$\left. \begin{array}{l} 54,68dB\mu \\ 150m \\ E(50,10) \end{array} \right\} CIE = 33,05Km \quad (48)$$

$$5,7 + 30 = 35,7 < 38,63 \text{ Km} \quad (49) \quad 5,71 + 33,05 = 38,76 > 38,63 \text{ Km} \quad (50)$$

Outra divergência na viabilidade do projeto para o exemplo calculado e o obtido pelo TVMAT foi encontrada. Como explicado anteriormente, deve-se aceitar o resultado obtido pelo TVMAT, isto é, declarar o projeto inviável.

Tendo a inviabilidade do projeto como resultado, o fator de correção da rugosidade do terreno no contorno protegido da estação proponente deverá então, ser aplicado. O resultado disto será o mesmo obtido pela Expressão 40, visto que as constantes são as mesmas.

$$CPP = CPP - \Delta F = 82,97 - (-0,73) = 83,7dB\mu \quad (51)$$

Está demonstrada nas Expressões 52 e 53 a atualização da distância aos contornos em questão. Foi calculada novamente a viabilidade com o novo valor de campo protegido nas Expressões 54 e 55.

$$\left. \begin{array}{l} 83,7dB\mu \\ 119,25m \\ E(50,50) \end{array} \right\} CPP = 5,5Km \quad (52)$$

$$\left. \begin{array}{l} 83,7dB\mu \\ 119,25m \\ E(50,50) \end{array} \right\} CPP = 5,52Km \quad (53)$$

$$5,5 + 30 = 35,5 < 38,63 \text{ Km} \quad (54) \quad 5,52 + 33,05 = 38,57 < 38,63 \text{ Km} \quad (55)$$

Com a correção do campo nota-se que o projeto é viável para ambos os cálculos.

- Verificação de interferência de Itapitanga em Ipiáu:

Com base nos valores de contorno protegido da estação existente igual a 70 dBμ, e FCE igual a -12,68 dB, foi obtido o contorno protegido corrigido no valor de 82,68 dB. Este valor será usado com o HNMT de 150 metros para resultar na distância ao contorno.

$$\left. \begin{array}{l} 82,68dB\mu \\ 150m \\ E(50,50) \end{array} \right\} CPE = 6,5Km \quad (56)$$

$$\left. \begin{array}{l} 82,68dB\mu \\ 150m \\ E(50,50) \end{array} \right\} CPE = 6,44Km \quad (57)$$

Calcula-se o contorno interferente da estação proponente com base na Expressão do sinal interferente (23). Aplicando-se o fator de correção da estação proponente (-12,97dB), a relação de proteção em questão (28 dB) e o sinal útil, pode-se encontrar na Expressão 34 o valor do contorno.

$$SI = 70 - (28) - (-12,97) = 54,97dB\mu \quad (59)$$

Foi calculada então a distância ao seu contorno (Expressões 60 e 61). Será obtido em seguida, nas Expressões 62 e 63, os cálculos de viabilidade baseados na Expressão 24.

$$\left. \begin{array}{l} 54,97dB\mu \\ 119,25m \\ E(50,10) \end{array} \right\} CIP = 28Km \quad (60)$$

$$\left. \begin{array}{l} 54,97dB\mu \\ 119,25m \\ E(50,10) \end{array} \right\} CIP = 29,78Km \quad (61)$$

$$6,5 + 28 = 34,5 < 38,63 \text{ Km} \quad (62) \quad 6,44 + 29,78 = 36,22 < 38,63 \text{ Km} \quad (63)$$

Portanto, a estação proponente não interfere na existente e o projeto é viável.

4. Estudo: Itapitanga e Aurelino Leal.

- Verificação de interferência de Aurelino Leal em Itapitanga:

O canal da estação existente em Aurelino Leal é o 21 UHF, portanto poderá ocasionar uma interferência do tipo oscilador local. A norma define a sua relação de proteção no valor de -6 dB conforme Tabela 5. A distância ao CIE será calculada com base no sinal interferente definido pela Expressão 17. A sua potência é de 20 W, o que resulta em um FCE no valor de -16,99 dB. Com base nesses dados tem-se o valor do sinal interferente da estação existente demonstrado na Expressão 64.

$$SI = SU - RP - FCE = 70 - (-6) - (-16,99) = 92,99dB\mu \quad (64)$$

Como explicado anteriormente este valor será limitado para 80 dBμ. O que resulta em uma distância ao seu contorno de 4,5 Km para uma HNMT de 150 metros.

Com estas distâncias será possível verificar a viabilidade do estudo em questão através da Expressão 18. O seu resultado é demonstrado na Expressão 65 para o exemplo calculado e na Expressão 66 obtida pelo TVMAT.

$$5,7 + 4,5 = 10,2 < 28,15 \text{ Km} \quad (65) \quad 5,71 + 4,5 = 10,21 < 28,15 \text{ Km} \quad (66)$$

Pode-se observar que a estação existente não interfere na estação proponente em ambos os cálculos.

- Verificação de interferência de Itapitanga em Aurelino Leal:

O contorno protegido da estação existente é definido através da Tabela 3, e no caso seu valor é 70 dBμ, pois se trata do canal 21 UHF. Aplicado o seu fator de correção (FCE) obtem-se um contorno protegido corrigido no valor de 86,99 dBμ. Como não há a HNMT da referida estação será usado o valor de 150 metros. Os resultados obtidos são mostrados nas Expressões 67 e 68, para exemplo calculado e obtido através do TVMAT, respectivamente.

$$\left. \begin{array}{l} 86,99dB\mu \\ 150m \\ E(50,50) \end{array} \right\} CPE = 4,9Km \quad (67)$$

$$\left. \begin{array}{l} 86,99dB\mu \\ 150m \\ E(50,50) \end{array} \right\} CPE = 5,21Km \quad (68)$$

Calcula-se agora o CIP com base na Expressão do sinal interferente (23).

$$SI = SU - RP - FC = 70 - (-6) - (-12,97) = 88,97dB\mu \quad (69)$$

Este valor será limitado a 80 dB μ , e para uma HNMT de 119,25 metros, resultará em uma distância ao seu contorno interferente de 4,192 Km. O cálculo da viabilidade deverá obedecer a Expressão 24 e tem como resultado nas Expressões 70 e 71, para exemplo calculado e obtido pelo TVMAT, respectivamente.

$$4,9 + 4,192 = 9,1 < 28,15 \text{ Km} \quad (70) \quad 5,21 + 4,192 = 9,4 < 28,15 \text{ Km} \quad (71)$$

Observa-se que em nenhum caso a distância limite foi atingida, portanto a estação em Itapitanga não interfere em Aurelino Leal e vice-versa. Em consequência disso o projeto é viável.

5. Estudo: Itapitanga e Inema.

- Verificação de interferência de Inema em Itapitanga:

A estação interferente localizada em Inema utiliza o canal 29 UHF, portanto há uma possível interferência de frequência de imagem de vídeo, o que define uma relação de proteção no valor de 3 dB. Seu FCE foi calculado para a potência de 200 W e resultado em - 6,99 dB. Seu sinal interferente será calculado na Expressão 72.

$$SI = SU - RP - FCE = 70 - (3) - (-6,99) = 73,99dB\mu \quad (72)$$

Suas distâncias aos contornos interferentes são calculadas nas Expressões 73 e 74. De acordo com a Expressão 18 observa-se seu cálculo de viabilidade.

$$\left. \begin{array}{l} 73,99dB\mu \\ 150m \\ E(50,10) \end{array} \right\} CIE = 10Km \quad (73)$$

$$\left. \begin{array}{l} 73,99dB\mu \\ 150m \\ E(50,10) \end{array} \right\} CIE = 10,36Km \quad (74)$$

$$5,7 + 10 = 15,7 > 13,40 \text{ Km} \quad (75) \quad 5,71 + 10,36 = 16,07 > 13,40 \text{ Km} \quad (76)$$

Para ambos os cálculos o projeto foi considerado inviável. A correção do contorno relativo à rugosidade deve ser feita. O resultado do fator de correção é o mesmo obtido pela

Expressão 40, visto que as constantes são as mesmas. O CPP já foi calculado na Expressão 51 e o seu valor é de 83,7 dBμ. Os resultados obtidos pelo exemplo calculado e o TVMAT foram demonstrados nas Expressões 52 e 53, e os seus valores são 5,5 e 5,52 Km, respectivamente.

Com o contorno calculado será feito novamente o cálculo de viabilidade. As Expressões 77 e 78 mostram os resultados

$$5,5 + 10 = 15,5 > 13,40 \text{ Km} \quad (77) \quad 5,52 + 10,36 = 15,88 > 13,40 \text{ Km} \quad (78)$$

Mesmo com a correção do campo nota-se que o projeto continua inviável para ambos os meios. Seria aconselhável partir para o próximo método através dos perfis, mencionado no Item 2.5, mas que não é abrangido pelo TVMAT.

- Verificação de interferência de Itapitanga em Inema:

Com base nos valores de contorno protegido da estação existente igual a 70 dBμ, e FCE igual a -6,99 dB, o contorno protegido corrigido é de 76,99 dB. Este valor será usado com o HNMT de 150 metros para se obter a sua distância ao contorno.

$$\left. \begin{array}{l} 76,99dB\mu \\ 150m \\ E(50,50) \end{array} \right\} CPE = 8,7Km \quad (79)$$

$$\left. \begin{array}{l} 76,99dB\mu \\ 150m \\ E(50,50) \end{array} \right\} CPE = 9,2Km \quad (80)$$

Calcula-se o contorno interferente da estação proponente com base na Expressão do sinal interferente (23). Aplicando-se o fator de correção da estação proponente (-12,97dB), a relação de proteção em questão (3 dB) e o sinal útil temos na Expressão 81 o seu valor de contorno.

$$SI = 70 - (3) - (-12,97) = 79,97dB\mu \quad (81)$$

Calcula-se então a distância ao seu contorno nas Expressões 82 e 83, e em seguida, nas Expressões 84 e 85, os cálculos de viabilidade baseados na Expressão 24.

$$\left. \begin{array}{l} 79,97dB\mu \\ 119,25m \\ E(50,10) \end{array} \right\} CIP = 4Km \quad (82)$$

$$\left. \begin{array}{l} 79,97dB\mu \\ 119,25m \\ E(50,10) \end{array} \right\} CIP = 4,22Km \quad (83)$$

$$4 + 8,7 = 12,7 < 13,40 \text{ Km} \quad (84) \quad 4,22 + 9,2 = 13,42 > 13,40 \text{ Km} \quad (85)$$

Os valores de distância aos contornos obtidos divergiram quanto ao método aplicado. Será considerada a inviabilidade do projeto conforme explicação anterior.

O resultado do fator de correção é o mesmo obtido pela Expressão 40, visto que as constantes são as mesmas. Com base neste fator será atualizado o contorno interferente da estação proponente, pois somente ela interfere na existente, como demonstrado na Expressão 86.

$$CIP = CIP - \Delta F = 79,97 - (-0,73) = 80,7 \text{ dB}\mu \quad (86)$$

O valor obtido foi bem próximo do seu limite de 80 dB μ . Para um contorno de 80dB μ a uma HNMT de 119,25 metros, tem-se uma distância ao contorno interferente de 4,192 Km.

$$4,192 + 8,7 = 12,89 < 13,40 \text{ Km} \quad (87) \quad 4,192 + 9,2 = 13,392 < 13,40 \text{ Km} \quad (88)$$

Contudo, com a devida correção na rugosidade do terreno, o projeto passou a ser viável. Porém a estação existente localizada em Inema continua interferindo na emissora proponente localizada em Itapitanga.

6 CONCLUSÃO

O presente trabalho, através de tudo que foi apresentado, conseguiu atender o seu objetivo. Com o uso de gráficos, tabelas, expressões e ilustrações, foi explicado o programa TVMAT. Pôde-se constatar como seria mais prático e fácil a realização de um projeto de viabilização de canal de TV usando este programa. Isto seria facilmente notado devido à precisão dos cálculos e à proximidade dos resultados do TVMAT quando comparados ao que foi simplesmente feito no exemplo calculado.

A tarefa de fazer todo o projeto manualmente foi diminuída pela inclusão das curvas de intensidade de campo, que reduz o esforço do usuário, podendo este, economizar tempo, além de possuir valores mais fiéis e próximos da realidade.

Ferramentas do Matlab, como a interpolação, foram incluídas no TVMAT e fazem com que este projeto se torne diferente e dinâmico. A interpolação, podendo estimar os valores das curvas de intensidade de campo se torna o ponto primordial do programa.

Todos os aspectos abordados no decorrer do TVMAT realçam sua importância e validade, e seus resultados satisfatórios permitem levantar propostas de continuidade do mesmo. A fim de tornar este projeto mais completo, propõe-se um novo trabalho, que será descrito em seguida.

Poderá ser feito um relatório impresso com dados do projeto. Este relatório seria construído com o Matlab Report Generator, uma funcionalidade do Matlab que incorpora a criação de relatórios, inclusive para publicação na Internet. Nele, poderá inclusive, constar também os perfis das radiais do projeto.

Este trabalho foi feito concomitantemente a outro que abrange a viabilização de canais de FM. Sugere-se fazer um programa mais completo, abrangendo os dois conteúdos, incluindo as interferências entre TV e FM, e vice-versa.

Os contornos primário, secundários e rurais para cada radial foram feitos. Com base neles, poderá ser traçado o diagrama dos contornos da estação através da interpolação dos mesmos.

A importação das radiais que só funcionam para projetos feitos no TVMAT, poderá ser feita através do método de tabulação, permitindo assim a importação através de qualquer arquivo texto. Este método consiste em localizar as tabulações existentes entre as cotas e transformá-las em colunas, ou seja, definir cada cota à sua devida distância.

Para a realização deste novo programa, será necessário o aperfeiçoamento do usuário no Matlab e uma nova busca de fontes bibliográficas, que o elucidará, enriquecendo e tornando o trabalho viável.

REFERÊNCIAS

_____. **NBR 14724**: informação e documentação: trabalhos acadêmicos: apresentação. Rio de Janeiro, 2002.

BRASIL. ANATEL. **Resolução n. 284, de 7 de dezembro de 2001**. Regulamento Técnico para a Prestação do Serviço de Radiodifusão de Sons e Imagens e do Serviço de Retransmissão de Televisão.

COSTA E SILVA, Roberto da. **Eletromagnetismo Aplicado**. 1. ed. Salvador: Universidade Federal da Bahia, 1998.

COSTA E SILVA, Roberto da. **Disciplina: Sistemas de Comunicação III**. Faculdade de Tecnologia e Ciências – FTC. Salvador, 2005.1.

HANSELMANN, Duane; LITTLEFIELD, Bruce. **MatLab 6 Curso Completo**. São Paulo: Prentice Hall, 2003.

ANEXOS

ANEXO A

AGÊNCIA NACIONAL DE TELECOMUNICAÇÕES

RESOLUÇÃO Nº 284, DE 7 DE DEZEMBRO DE 2001

Aprova o Regulamento Técnico para a Prestação do Serviço de Radiodifusão de Sons e Imagens e do Serviço de Retransmissão de Televisão.

O CONSELHO DIRETOR DA AGÊNCIA NACIONAL DE TELECOMUNICAÇÕES - ANATEL, no uso das atribuições que lhe foram conferidas pelo art. 22, da Lei nº 9.472, de 16 de julho de 1997, e art.35 do Regulamento da Agência Nacional de Telecomunicações, aprovado pelo Decreto nº 2.338, de 7 de outubro de 1997,

CONSIDERANDO os comentários recebidos em decorrência da Consulta Pública nº 179, de 27 de setembro de 1999, publicada no Diário Oficial da União de 28 de setembro de 1999;

CONSIDERANDO o disposto nos arts 19, 157, 159,160,211 e 214 da Lei n.º 9.472/97;

CONSIDERANDO deliberação tomada por meio do Circuito Deliberativo nº 244, de 7 de dezembro de 2001, resolve:

Art. 1º Aprovar o Regulamento Técnico para a Prestação do Serviço de Radiodifusão de Sons e Imagens e do Serviço de Retransmissão de Televisão, na forma do Anexo a esta Resolução.

Art. 2º Esta Resolução entra em vigor na data de sua publicação.

RENATO NAVARRO GUERREIRO

Presidente do Conselho

ANEXO B

ANEXO C


```

function varargout = tv(varargin)tv.m
%PROJETO DE VIABILIZACAO DE CANAL DE TV COM AUXILIO DO MATLAB
%
%FACULDADE DE TECNOLOGIA E CIENCIAS - FTC
%TRABALHO DE CONCLUSAO DO CURSO DE ENGENHARIA ELETRICA COM ENFASE
%EM TELECOMUNICACOES E COMPUTACAO
%ORIENTADOR: PROF. ROBERTO DA COSTA E SILVA
%AUTOR: MARCELO CAMERA OLIVEIRA
%EMAIL: mcelooliveira@gmail.com
%SALVADOR - BAHIA - JANEIRO - 2006

% Begin initialization code
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn',   @tv_OpeningFcn, ...
                  'gui_OutputFcn',    @tv_OutputFcn, ...
                  'gui_LayoutFcn',    [], ...
                  'gui_Callback',     []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
if nargin
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code

function tv_OpeningFcn(hObject, eventdata, handles, varargin)

handles.output = hObject;
guidata(hObject, handles);

set(handles.uipanel_dados,'Visible','off');
set(handles.uipanel_contornos,'Visible','off');
set(handles.mnu_save,'Enable','off');
set(handles.mnu_edit,'Enable','off');
set(handles.mnu_iniciar,'Enable','off');

global CI02_50_10 CI07_50_10 CI14_50_10 CP02_50_50 CP07_50_50 CP14_50_50
%Carregar as tabelas:
%CURVA D CANAL 2 A 6 (50,10)
CI02_50_10=[355   315   245   185   130   36    7    2
            365   325   250   190   135   47   12    3
            372   332   265   205   146   60   18    4
            380   340   272   210   153   67   24   4.5
            395   358   287   227   172   85   33    7
            420   378   305   245   188   105  43   11
            450   415   345   282   220   129  62   17 ];
%CURVA D CANAL 7 A 13(50,10)
CI07_50_10=[355   315   245   185   130   44   10    2
            360   320   250   190   135   52   15    3
            370   332   260   200   145   65   24    4
            380   340   270   210   150   72   30   4.5
            400   355   285   225   170   90   40    8
            420   378   305   245   190   108  53   14
            450   415   340   280   218   132  73   20 ];
%CURVA D CANAL 14 A 59(50,10)
CI14_50_10=[345   305   230   170   118   36   7.5    2
            345   305   238   180   120   41   13    3

```

```

        360    320    250    190    130    52    20    4
        365    325    257    197    140    58    24    4.5
        385    345    277    215    157.5    70    33    7.5
        405    365    295    230    175    88    40    12
        440    400    330    265    205    112.5    52    16 ];
%CURVA D CANAL 2 A 6 (50,50)
CP02_50_50=[290    250    185    130    75    29    10    3.5    0.5
            293    257    190    135    83    38    12    4.5    0.6
            305    265    205    144    97    51    15    6    1.1
            310    275    210    150    105    59    20    7.5    1.5
            35    290    225    170    125    70    29    10    1.8
            360    310    245    185    85    37.5    12    12    2
            400    370    273    215    165    105    57    14    2 ];
%CURVA D CANAL 7 A 13 (50,50)
CP07_50_50=[295    255    185    132    75    35    10    4.5    0.5
            297    260    190    135    85    47    13    5.4    1
            305    270    205    145    102    58    18    7    1.5
            310    280    215    150    109    64    25    8.5    1.6
            335    303    230    170    130    72    35    10    1.7
            345    310    250    187    147.5    91    50    13    1.8
            385    350    295    223    170    111    65    15    1.9 ];
%CURVA D CANAL 14 A 59 (50,50)
CP14_50_50=[265    228    155    90    62    28    10    3.5    0.5
            270    233    160    97    67    32    12    4.5    0.7
            282    245    167    115    72    47    15.5    6    1.3
            290    255    175    124    80    51    20.5    7.2    1.5
            305    270    195    140    90    60    28    9    1.9
            335    285    212    160    120    69    33    11.5    2
            360    330    245    185    148    85    47    14    2 ];

function varargout = tv_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;

function edit_atcabo_Callback(hObject, eventdata, handles)
atualiza_contorno(hObject, eventdata, handles);

function edit_pot_Callback(hObject, eventdata, handles)
atualiza_contorno(hObject, eventdata, handles);

function edit_pot_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

function popup_canal_Callback(hObject, eventdata, handles)
if str2num(get(handles.edit_canal,'String'))>1 &&
str2num(get(handles.edit_canal,'String'))<14
    set(handles.popup_canal,'Value',1);
else
    set(handles.popup_canal,'Value',2);
end
atualiza_contorno(hObject, eventdata, handles);

function popup_canal_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

```

```

function edit_atextras_Callback(hObject, eventdata, handles)
atualiza_contorno(hObject, eventdata, handles);

function edit_atextras_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

% -----
function mnu_new_Callback(hObject, eventdata, handles)
% Inicializacao de um novo projeto.
% Criando arquivo e definindo suas variáveis.
global tv_file
tv_file = uiputfile('*.mat','Digite o nome do arquivo:');

if ~isequal(tv_file,0)

estacoes=struct('descricao',{},'latitude',{},'longitude',{},'canal',{},'decalagem',{},'tipo_canal',{},'potencia',{},'azimute',{},'distancia',{},'CPP',{},'CPE',{},'ci',{},'CIE',{},'CIP',{},'cpec',{},'ciiec',{},'cipc',{},'cppc',{},'CPE_CIP',{},'CPP_CIE',{},'interferente',{},'fator_correcao',{},'tipo_interferencia',{},'frequencia_video',{},'frequencia_audio',{});

radial=struct('descricao',{},'cota',{},'azimute',{},'nmr',{},'hnmr',{},'dist_primario',{},'dist_secundario',{},'dist_rural',{},'rug',{},'radial_dist={};
    entidade=[];descricao=[];cidade=[];estado=[];latitude=[];longitude=[];
    canal=[];decalagem='Nao';canal_tipo=[1];classe=[1];
    frequencia_video=[];frequencia_som=[];c=[];
    potencia=[];ganho_antena=[];altura_antena=[];hbt=[];
    atenuacao_cabo=[];comprimento_cabo=[];atenuacao_extras=[];
    fator_correcao=0;hnmt=0;cpc=0;rugosidade_terreno=0;
    contorno_primario=0;contorno_secundario=0;contorno_rural=0;
    contorno_protegido=0;CPP=0;
    set(handles.edit_entidade,'String',entidade);
    set(handles.edit_desc,'String',descricao);
    set(handles.edit_cidade,'String',cidade);
    set(handles.edit_estado,'String',estado);
    set(handles.edit_lat_graus,'String',latitude);
    set(handles.edit_lat_min,'String',latitude);
    set(handles.edit_lat_seg,'String',latitude);
    set(handles.edit_long_graus,'String',longitude);
    set(handles.edit_long_min,'String',longitude);
    set(handles.edit_long_seg,'String',longitude);
    set(handles.edit_canal,'String',canal);

set(handles.uipanel_decalagem,'SelectedObject',handles.radiobutton_dec_nao);
set(handles.popup_canal,'Value',canal_tipo);
set(handles.popup_classe,'Value',classe);
set(handles.edit_frequencia_video,'String',frequencia_video);
set(handles.edit_frequencia_som,'String',frequencia_som);
set(handles.edit_pot,'String',num2str(potencia));
set(handles.edit_ganho_antena,'String',num2str(ganho_antena));
set(handles.edit_altura_antena,'String',num2str(altura_antena));
set(handles.edit_hbt,'String',num2str(hbt));
set(handles.edit_atcabo,'String',num2str(atenuacao_cabo));
set(handles.edit_compcabo,'String',num2str(comprimento_cabo));
set(handles.edit_atextras,'String',num2str(atenuacao_extras));
set(handles.text_fator_correcao,'String',num2str(fator_correcao));

```

```

set(handles.text_protegido,'String',num2str(contorno_protegido));
set(handles.text_primario,'String',num2str(contorno_primario));
set(handles.text_secundario,'String',num2str(contorno_secundario));

set(handles.text_rural,'String',num2str(contorno_rural));
set(handles.text_CPP,'String',num2str(CPP));
set(handles.text_cpc,'String',num2str(cpc));
set(handles.text_hnmt,'String',num2str(hnmt));
set(handles.text_rug_media,'String',num2str(rugosidade_terreno));

set(handles.mnu_iniciar,'Enable','on');
set(handles.uipanel_dados,'Visible','on');
set(handles.uipanel_contornos,'Visible','on');
set(handles.mnu_save,'Enable','on');
set(handles.mnu_edit,'Enable','on');

save(tv_file,'estacoes','radial','radial_dist','CPP','cpc','hnmt','entidade',
'hbt','descricao','cidade','estado','latitude','longitude','canal','decalagem',
','canal_tipo','classe','frequencia_video','frequencia_som','potencia','ganho',
_antena','altura_antena','atenuacao_cabo','comprimento_cabo','atenuacao_extra',
s','fator_correcao','contorno_protegido','contorno_primario','contorno_secund',
ario','contorno_rural','rugosidade_terreno','c');
end

% -----
function mnu_open_Callback(hObject, eventdata, handles)
% Inicializacao de um projeto existente.
% Atualizando variáveis.
global tv_file
tv_file = uigetfile('*.mat','Abrir Projeto');

if ~isequal(tv_file,0)
    load(tv_file);

    set(handles.uipanel_dados,'Visible','on');
    set(handles.uipanel_contornos,'Visible','on');
    set(handles.mnu_save,'Enable','on');
    set(handles.mnu_edit,'Enable','on');
    set(handles.mnu_iniciar,'Enable','on');

    set(handles.edit_entidade,'String',entidade);
    set(handles.edit_desc,'String',descricao);
    set(handles.edit_cidade,'String',cidade);
    set(handles.edit_estado,'String',estado);
    set(handles.edit_lat_graus,'String',num2str(fix(latitude)));
    set(handles.edit_lat_min,'String',num2str(fix((latitude-
fix(latitude))*60)));
    set(handles.edit_lat_seg,'String',num2str(fix(((latitude-
fix(latitude))*60)-fix((latitude-fix(latitude))*60))*60)));
    set(handles.edit_long_graus,'String',num2str(fix(longitude)));

    set(handles.edit_long_min,'String',num2str(fix((longitude-
fix(longitude))*60)));
    set(handles.edit_long_seg,'String',num2str(fix(((longitude-
fix(longitude))*60)-fix((longitude-fix(longitude))*60))*60)));
    set(handles.edit_canal,'String',num2str(canal));
    if decalagem=='Nao'

set(handles.uipanel_decalagem,'SelectedObject',handles.radiobutton_dec_nao);
elseif decalagem=='+'

```

```

set(handles.uipanel_decalagem,'SelectedObject',handles.radiobutton_dec_mais);
    elseif decalagem=='-'

set(handles.uipanel_decalagem,'SelectedObject',handles.radiobutton_dec_menos)
;
    end
    set(handles.popup_canal,'Value',canal_tipo);
    set(handles.popup_classe,'Value',classe);
    set(handles.edit_frequencia_video,'String',frequencia_video);
    set(handles.edit_frequencia_som,'String',frequencia_som);
    set(handles.edit_pot,'String',potencia);
    set(handles.edit_ganho_antena,'String',num2str(ganho_antena));
    set(handles.edit_altura_antena,'String',num2str(altura_antena));
    set(handles.edit_hbt,'String',num2str(hbt));
    set(handles.edit_atcabo,'String',num2str(atenuacao_cabo));
    set(handles.edit_compcabo,'String',num2str(comprimento_cabo));
    set(handles.edit_atextras,'String',num2str(atenuacao_extras));
    set(handles.text_fator_correcao,'String',num2str(fator_correcao));
    set(handles.text_protegido,'String',num2str(contorno_protegido));
    set(handles.text_primario,'String',num2str(contorno_primario));
    set(handles.text_secundario,'String',num2str(contorno_secundario));
    set(handles.text_rural,'String',num2str(contorno_rural));
    set(handles.text_cpc,'String',num2str(cpc));
    set(handles.text_CPP,'String',num2str(CPP));
    set(handles.text_hnmt,'String',num2str(hnmt));
    set(handles.text_rug_media,'String',num2str(rugosidade_terreno));
end

% -----
function mnu_save_Callback(hObject, eventdata, handles)

global tv_file

entidade=get(handles.edit_entidade,'String');
descricao=get(handles.edit_desc,'String');
cidade=get(handles.edit_cidade,'String');
estado=get(handles.edit_estado,'String');
latitude=str2num(get(handles.edit_lat_graus,'String'))+str2num(get(handles.ed
it_lat_min,'String'))/60+str2num(get(handles.edit_lat_seg,'String'))/3600;
longitude=str2num(get(handles.edit_long_graus,'String'))+str2num(get(handles.
edit_long_min,'String'))/60+str2num(get(handles.edit_long_seg,'String'))/3600
;
canal=str2num(get(handles.edit_canal,'String'));
decalagem=get(get(handles.uipanel_decalagem,'SelectedObject'),'string');
canal_tipo=get(handles.popup_canal,'Value');
classe=get(handles.popup_classe,'Value');
frequencia_video=get(handles.edit_frequencia_video,'String');
frequencia_som=get(handles.edit_frequencia_som,'String');
potencia=str2num(get(handles.edit_pot,'String'));
ganho_antena=str2num(get(handles.edit_ganho_antena,'String'));
altura_antena=str2num(get(handles.edit_altura_antena,'String'));
hbt=str2num(get(handles.edit_hbt,'String'));
atenuacao_cabo=str2num(get(handles.edit_atcabo,'String'));
comprimento_cabo=str2num(get(handles.edit_compcabo,'String'));
atenuacao_extras=str2num(get(handles.edit_atextras,'String'));
fator_correcao=str2num(get(handles.text_fator_correcao,'String'));
contorno_protegido=str2num(get(handles.text_protegido,'String'));
contorno_primario=str2num(get(handles.text_primario,'String'));
contorno_secundario=str2num(get(handles.text_secundario,'String'));
contorno_rural=str2num(get(handles.text_rural,'String'));

```

```

cpc=str2num(get(handles.text_cpc,'String'));
CPP=str2num(get(handles.text_CPP,'String'));
hnmt=str2num(get(handles.text_hnmt,'String'));
rugosidade_terreno=str2num(get(handles.text_rug_media,'String'));
save(tv_file,'entidade','descricao','cidade','estado','latitude','longitude',
'canal','decalagem','canal_tipo','classe','frequencia_video','frequencia_som',
'potencia','ganho_antena','altura_antena','hbt','atenuacao_cabo','compriment
o_cabo','atenuacao_extras','fator_correcao','contorno_primario','contorno_sec
undario','contorno_protegido','contorno_rural','CPP','cpc','hnmt','rugosidade
_terreno','-append');

% -----

function edit_compcabo_Callback(hObject, eventdata, handles)
atualiza_contorno(hObject, eventdata, handles);

function edit_compcabo_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

function edit_ganho_antena_Callback(hObject, eventdata, handles)
atualiza_contorno(hObject, eventdata, handles);

function atualiza_contorno(hObject, eventdata, handles)
if ~isempty(get(handles.edit_pot,'String')) |
~isempty(get(handles.edit_ganho_antena,'String')) |
~isempty(get(handles.edit_atcabo,'String')) |
~isempty(get(handles.edit_atextras,'String')) |
~isempty(get(handles.edit_compcabo,'String'))
    %fc=10*log(P/1kW) + Gant - comp.*(atenuacao_cabo/100) - atenuacoes_extras

fator_correcao=10*log10(str2num(get(handles.edit_pot,'String'))/1000)+str2num
(get(handles.edit_ganho_antena,'String'))-
str2num(get(handles.edit_compcabo,'String'))*str2num(get(handles.edit_atcabo,
'String'))/100-str2num(get(handles.edit_atextras,'String'));
else
    fator_correcao=0;
end
if str2num(get(handles.edit_canal,'String')) >= 2 &
str2num(get(handles.edit_canal,'String')) <= 6 &
get(handles.popup_canal,'Value') == 1
    contorno_protegido=58;
    contorno_primario=74;
    contorno_secundario=68;
    contorno_rural=54;
elseif str2num(get(handles.edit_canal,'String')) >= 7 &
str2num(get(handles.edit_canal,'String')) <= 13 &
get(handles.popup_canal,'Value') == 1
    contorno_protegido=64;
    contorno_primario=77;
    contorno_secundario=71;
    contorno_rural=60;
else
    contorno_protegido=70;
    contorno_primario=80;
    contorno_secundario=74;
    contorno_rural=70;
    set(handles.popup_canal,'Value',2);

```

```
end
set(handles.text_fator_correcao,'String',fator_correcao);
set(handles.text_protegido,'String',contorno_protegido);
set(handles.text_primario,'String',contorno_primario);
set(handles.text_secundario,'String',contorno_secundario);
set(handles.text_rural,'String',contorno_rural);

function edit_canal_Callback(hObject, eventdata, handles)

if str2num(get(handles.edit_canal,'String'))>1 &&
str2num(get(handles.edit_canal,'String'))<14
    set(handles.popup_canal,'Value',1);
elseif str2num(get(handles.edit_canal,'String'))==37
    errordlg('Canal reservado ao Serviço de Radioastronomia. Erro!');
    set(handles.edit_canal,'String','');
    set(handles.edit_frequencia_video,'String','');
    set(handles.edit_frequencia_som,'String','');
    return
elseif str2num(get(handles.edit_canal,'String'))>13 &&
str2num(get(handles.edit_canal,'String'))<=59 &&
str2num(get(handles.edit_canal,'String'))~=37
    set(handles.popup_canal,'Value',2);
else
    uiwait(errordlg('Canal Inválido. (2 a 59)','Erro!'));
    set(handles.edit_canal,'String','');
end
atualiza_contorno(hObject, eventdata, handles);

switch get(handles.edit_canal,'String')
    case ' ', frequencia_video=0;frequencia_som=0;
    case '2', frequencia_video=55.25;frequencia_som=59.75;
    case '3', frequencia_video=61.25;frequencia_som=65.75;
    case '4', frequencia_video=67.25;frequencia_som=71.75;
    case '5', frequencia_video=77.25;frequencia_som=81.75;
    case '6', frequencia_video=83.25;frequencia_som=87.75;
    case '7', frequencia_video=175.25;frequencia_som=179.75;
    case '8', frequencia_video=181.25;frequencia_som=185.75;
    case '9', frequencia_video=187.25;frequencia_som=191.75;
    case '10', frequencia_video=193.25;frequencia_som=197.75;
    case '11', frequencia_video=199.25;frequencia_som=203.75;
    case '12', frequencia_video=205.25;frequencia_som=209.75;

    case '13', frequencia_video=211.25;frequencia_som=215.75;
    case '14', frequencia_video=471.25;frequencia_som=475.75;
    case '15', frequencia_video=477.25;frequencia_som=481.75;
    case '16', frequencia_video=483.25;frequencia_som=487.75;
    case '17', frequencia_video=489.25;frequencia_som=493.75;
    case '18', frequencia_video=495.25;frequencia_som=499.75;
    case '19', frequencia_video=501.25;frequencia_som=505.75;
    case '20', frequencia_video=507.25;frequencia_som=511.75;
    case '21', frequencia_video=513.25;frequencia_som=517.75;
    case '22', frequencia_video=519.25;frequencia_som=523.75;
    case '23', frequencia_video=525.25;frequencia_som=529.75;
    case '24', frequencia_video=531.25;frequencia_som=535.75;
    case '25', frequencia_video=537.25;frequencia_som=541.75;
    case '26', frequencia_video=543.25;frequencia_som=547.75;
    case '27', frequencia_video=549.25;frequencia_som=553.75;
    case '28', frequencia_video=555.25;frequencia_som=559.75;
    case '29', frequencia_video=561.25;frequencia_som=565.75;
    case '30', frequencia_video=567.25;frequencia_som=571.75;
    case '31', frequencia_video=573.25;frequencia_som=577.75;
```

```

case '32', frequencia_video=579.25;frequencia_som=583.75;
case '33', frequencia_video=585.25;frequencia_som=589.75;
case '34', frequencia_video=591.25;frequencia_som=595.75;
case '35', frequencia_video=597.25;frequencia_som=601.75;
case '36', frequencia_video=603.25;frequencia_som=607.75;
case '38', frequencia_video=615.25;frequencia_som=619.75;
case '39', frequencia_video=621.25;frequencia_som=625.75;
case '40', frequencia_video=627.25;frequencia_som=631.75;
case '41', frequencia_video=633.25;frequencia_som=637.75;
case '42', frequencia_video=639.25;frequencia_som=643.75;
case '43', frequencia_video=645.25;frequencia_som=649.75;
case '44', frequencia_video=651.25;frequencia_som=655.75;
case '45', frequencia_video=657.25;frequencia_som=661.75;
case '46', frequencia_video=663.25;frequencia_som=667.75;
case '47', frequencia_video=669.25;frequencia_som=673.75;
case '48', frequencia_video=675.25;frequencia_som=679.75;
case '49', frequencia_video=681.25;frequencia_som=685.75;
case '50', frequencia_video=687.25;frequencia_som=691.75;
case '51', frequencia_video=693.25;frequencia_som=697.75;
case '52', frequencia_video=699.25;frequencia_som=703.75;
case '53', frequencia_video=705.25;frequencia_som=709.75;
case '54', frequencia_video=711.25;frequencia_som=715.75;
case '55', frequencia_video=717.25;frequencia_som=721.75;
case '56', frequencia_video=723.25;frequencia_som=727.75;
case '57', frequencia_video=729.25;frequencia_som=733.75;
case '58', frequencia_video=735.25;frequencia_som=739.75;
case '59', frequencia_video=741.25;frequencia_som=745.75;
end
set(handles.edit_frequencia_video,'String',frequencia_video);
set(handles.edit_frequencia_som,'String',frequencia_som);

function edit_desc_Callback(hObject, eventdata, handles)

% -----
function mnu_iniciar_Callback(hObject, eventdata, handles)
global tv_file y CI02_50_10 CI07_50_10 CI14_50_10 CP02_50_50 CP07_50_50
CP14_50_50 rc_rug estacoes hnmt CP CI contorno_protegido fator_correcao

%Salvando projeto antes de executar:
mnu_save_Callback(hObject, eventdata, handles);
%Verificação das variáveis. - RADIAIS:
load(tv_file);
if isempty(radial) | isempty(radial_dist)
    uiwait(errordlg('As radiais não foram editadas. Por favor entre com as
    radiais e tente novamente.','Erro!'));
    return
else
    numradiais=length(radial);
    %Verifica se existem as cotas e nmrs:
    for i=1:numradiais
        if isempty(radial(numradiais).cota)
            uiwait(errordlg('Algumas cotas não foram definidas!','Erro!'));
            return
        end
        if isempty(radial(numradiais).nmr)
            uiwait(errordlg('Algumas NMRs não foram definidas! Favor editar
as cotas novamente','Erro!'));
            return
        end
        if isempty(radial(numradiais).rug)

```



```

        uiwait(errordlg('Algumas rugosidades não foram definidas! Favor
editar as cotas novamente','Erro!'));
        return
    end
end
%Calcula as HNMRS e Rugosidade média
nmt=0;rugosidade_terreno=0;
for i=1:numradiais
    %HNMR=HBT+HT-NMR
    if ~isempty(altura_antena)
        if ~isempty(hbt)
            radial(i).hnmr=hbt+altura_antena-radial(i).nmr;
        else
            errordlg('A altura da base da torre não foi
definida!','Erro!');
            return
        end
    else
        errordlg('A altura da antena não foi definida!','Erro!');
        return
    end
    nmt=nmt+radial(i).nmr;
    rugosidade_terreno=rugosidade_terreno+radial(i).rug;
end
end
nmt=nmt/numradiais;rugosidade_terreno=rugosidade_terreno/numradiais;
set(handles.text_rug_media,'String',num2str(rugosidade_terreno));
hnmt=hbt+altura_antena-nmt;
%limita hnmt para interpolação:
if hnmt<30
    hnmt=30;
elseif hnmt>1000
    hnmt=1000;
end
set(handles.text_hnmt,'String',num2str(hnmt));

%Calcula os contornos interferentes possíveis da proponente:
H=[ 30 50 100 150 300 500 1000 ];E=[ -15 -10 0 10 20 40 60 80 ];
if isempty(canal)
    errordlg('O canal não foi definido!','Erro!');
    return
end
if isempty(fator_correcao)
    errordlg('O fator de correção não foi calculado!','Erro!');
    return
end
if (canal>=2 & canal<=6)
    CP=CP02_50_50;CI=CI02_50_10;c=1.9;
    CIP_cocanal=13-fator_correcao;%Co-canal sem decalagem
    CIP_cocanal_dec=30-fator_correcao;%Co-canal com decalagem
    CIP_adj_sup=70-fator_correcao;%Canal adjacente superior
    CIP_adj_inf=64-fator_correcao;%Canal adjacente inferior
    CIP_osc=0;CIP_freq_aud=0;CIP_freq_vid=0;CIP_fi=0;
elseif (canal>=7 & canal<=13)
    CP=CP07_50_50;CI=CI07_50_10;c=2.5;
    CIP_cocanal=19-fator_correcao;%Co-canal sem decalagem
    CIP_cocanal_dec=36-fator_correcao;%Co-canal com decalagem
    CIP_adj_sup=76-fator_correcao;%Canal adjacente superior
    CIP_adj_inf=70-fator_correcao;%Canal adjacente inferior
    CIP_osc=0;CIP_freq_aud=0;CIP_freq_vid=0;CIP_fi=0;
elseif (canal>=14 & canal<=59)

```

```

    CP=CP14_50_50;CI=CI14_50_10;c=4.8;
    CIP_cocanal=25-fator_correcao;%Co-canal sem decalagem
    CIP_cocanal_dec=42-fator_correcao;%Co-canal com decalagem
    CIP_adj_sup=82-fator_correcao;%Canal adjacente superior
    CIP_adj_inf=76-fator_correcao;%Canal adjacente inferior
    CIP_osc=76-fator_correcao;%Oscilador Local
    CIP_freq_aud=76-fator_correcao;%Frequencia Imagem de Áudio
    CIP_freq_vid=67-fator_correcao;%Frequencia Imagem de Vídeo
    CIP_fi=82-fator_correcao;%Batimento de FI
else
    errordlg('Número de canal Inválido','Erro!');
    return
end

%Cacula os contornos primários, secundários e rurais de cada radial:
H=[ 30 50 100 150 300 500 1000 ];E=[ -15 -10 0 10 20 40 60 80 100 ];

for i=1:numradiais
    radial(i).dist_primario=interp2(E,H,CP,(contorno_primario-
fator_correcao),radial(i).hnmr);
    radial(i).dist_secundario=interp2(E,H,CP,(contorno_secundario-
fator_correcao),radial(i).hnmr);
    radial(i).dist_rural=interp2(E,H,CP,(contorno_rural-
fator_correcao),radial(i).hnmr);
end
cpc=contorno_protegido-fator_correcao;
%limita cpc
if cpc>100
    cpc=100;
elseif cpc<-15
    cpc=-15;
end
CPP=interp2(E,H,CP,cpc,hnmt);
set(handles.text_cpc,'String',num2str(cpc));
set(handles.text_CPP,'String',num2str(CPP));
rc_rug=0;
%Estações:
if isempty(estacoes)
    errordlg('As estações ainda não foram definidas.','Erro!');
    return
end
numestacoes=length(estacoes);
for i=1:numestacoes
    %Calcula a distancia das estacoes:
    if ~isempty(estacoes(numestacoes).latitude) &&
~isempty(estacoes(numestacoes).longitude) && ~isempty(latitude) &&
~isempty(longitude)

estacoes(numestacoes).distancia=111.1775*(acos(sin(latitude*pi/180)*sin(estac
oes(numestacoes).latitude*pi/180)+cos(latitude*pi/180)*cos(estacoes(numestaco
es).latitude*pi/180)*cos((longitude-
estacoes(numestacoes).longitude)*pi/180))/pi*180);
    else
        errordlg('Algumas coordenadas não foram definidas!','Erro!');
        return
    end
    estacoes(i).CPP=CPP;
    estacoes(i).cpc=cpc;
end

for y=1:numestacoes

```

```

%Verifica o tipo de possíveis interferencias:
if ~isempty(estacoes(y).canal)
    %VHF e UHF
    if canal==estacoes(y).canal
        if estacoes(y).decalagem=='Nao'
            estacoes(y).tipo_interferencia='Co-canal';
            estacoes(y).cipc=CIP_cocanal;
            estacoes(y).rp=-45;
            func_cip(hObject, eventdata, handles);
            func_cie(hObject, eventdata, handles);
        else
            estacoes(y).tipo_interferencia='Co-canal com decalagem';
            estacoes(y).cipc=CIP_cocanal_dec;
            estacoes(y).rp=-28;
            func_cip(hObject, eventdata, handles);
            func_cie(hObject, eventdata, handles);
        end
    elseif canal==(estacoes(y).canal-1)
        estacoes(y).tipo_interferencia='Canal adjacente +1';
        estacoes(y).cipc=CIP_adj_sup;
        estacoes(y).rp=12;
        func_cip(hObject, eventdata, handles);
        func_cie(hObject, eventdata, handles);
    elseif canal==(estacoes(y).canal+1)
        estacoes(y).tipo_interferencia='Canal adjacente -1';
        estacoes(y).cipc=CIP_adj_inf;
        estacoes(y).rp=6;
        func_cip(hObject, eventdata, handles);
        func_cie(hObject, eventdata, handles);

    %UHF
    elseif estacoes(y).tipo_canal==2 && canal==(estacoes(y).canal-15)
        estacoes(y).tipo_interferencia='Frequencia imagem de vídeo';
        estacoes(y).cipc=CIP_freq_vid;
        estacoes(y).rp=-3;
        func_cip(hObject, eventdata, handles);
        func_cie(hObject, eventdata, handles);
    elseif estacoes(y).tipo_canal==2 && canal==(estacoes(y).canal-14)
        estacoes(y).tipo_interferencia='Frequencia imagem de audio';
        estacoes(y).cipc=CIP_freq_aud;
        estacoes(y).rp=6;
        func_cip(hObject, eventdata, handles);
        func_cie(hObject, eventdata, handles);
    elseif estacoes(y).tipo_canal==2 && canal==(estacoes(y).canal-7) ||
canal==(estacoes(y).canal+7)
        estacoes(y).tipo_interferencia='Oscilador Local';
        estacoes(y).cipc=CIP_osc;
        estacoes(y).rp=6;
        func_cip(hObject, eventdata, handles);
        func_cie(hObject, eventdata, handles);
    elseif estacoes(y).tipo_canal==2 && canal==(estacoes(y).canal-8) ||
canal==(estacoes(y).canal+8)
        estacoes(y).tipo_interferencia='Batimento de FI.';
        estacoes(y).cipc=CIP_fi;
        estacoes(y).rp=12;
        func_cip(hObject, eventdata, handles);
        func_cie(hObject, eventdata, handles);
    else
        estacoes(y).tipo_interferencia='Não interfere.';
    end
end
else

```

```

        errordlg('Algum canal das estações não foram definidos.','Erro!');
        return
    end
end

%verifica se há interferencias:
for y=1:numestacoes
    estacoes(y).CPP=CPP;estacoes(y).cppc=cpc;
    estacoes(y).CPE_CIP=estacoes(y).CPE+estacoes(y).CIP;
    estacoes(y).CPP_CIE=estacoes(y).CPP+estacoes(y).CIE;
    %PROPONENTE -> EXISTENTE
    if estacoes(y).CPE_CIP>estacoes(y).distancia
        rc_rug=0;t=1;
        if rugosidade_terreno>50
            rc_rug=c-
0.03*rugosidade_terreno*(1+str2num(frequencia_video)/300);
            func_cip(hObject, eventdata, handles);
            estacoes(y).CPE_CIP=estacoes(y).CPE+estacoes(y).CIP;
        end
        if estacoes(y).CPE_CIP>estacoes(y).distancia
            t=1;
        else
            estacoes(y).interferente='VIÁVEL (CORRIGIDO)';t=2;
        end
        if estacoes(y).CPP_CIE>estacoes(y).distancia
            rc_rug=0;
            if rugosidade_terreno>50
                rc_rug=c-
0.03*rugosidade_terreno*(1+str2num(frequencia_video)/300);
                func_cpp(hObject, eventdata, handles);
                estacoes(y).CPP_CIE=estacoes(y).CPP+estacoes(y).CIE;
            end
        end
        if estacoes(y).CPP_CIE>estacoes(y).distancia
            if t==1
                estacoes(y).interferente='INVIÁVEL: P <=> E';
            else
                estacoes(y).interferente='INVIÁVEL: P <- E';
            end
        else
            if t==1
                estacoes(y).interferente='INVIÁVEL: P -> E';
            else
                estacoes(y).interferente='VIÁVEL (CORRIGIDO)';
            end
        end
        %EXISTENTE -> PROPONENTE
        elseif estacoes(y).CPP_CIE>estacoes(y).distancia
            rc_rug=0;
            if rugosidade_terreno>50
                rc_rug=c-
0.03*rugosidade_terreno*(1+str2num(frequencia_video)/300);
                estacoes(y).CPP=CPP;estacoes(y).cppc=cpc;
                func_cpp(hObject, eventdata, handles);
                estacoes(y).CPP_CIE=estacoes(y).CPP+estacoes(y).CIE;
            end
            if estacoes(y).CPP_CIE>estacoes(y).distancia
                estacoes(y).interferente='INVIÁVEL: P <- E';
            else
                estacoes(y).interferente='VIÁVEL (CORRIGIDO)';
            end
        end
    end
end

```

```

        end
    else
        estacoes(y).interferente='VIÁVEL';
    end
end

save(tv_file,'c','radial','estacoes','cpc','CPP','CIP_cocanal','CIP_cocanal_d
ec','CIP_adj_sup','CIP_adj_inf','CIP_osc','CIP_freq_aud','CIP_freq_vid','CIP_
fi','rugosidade_terreno','-append');
clear estacoes radial;
estacoes_interferentes;

function func_cpp(hObject, eventdata, handles)
global y rc_rug hnmt CP estacoes
H=[ 30 50 100 150 300 500 1000 ];E=[ -15 -10 0 10 20 40 60 80 100 ];
estacoes(y).cppc=estacoes(y).cppc-rc_rug;
%limita cpc
if estacoes(y).cppc>100
    estacoes(y).cppc=100;
elseif estacoes(y).cppc<-15
    estacoes(y).cppc=-15;
end
estacoes(y).CPP=interp2(E,H,CP,estacoes(y).cppc,hnmt);

function func_cip(hObject, eventdata, handles)
global y estacoes hnmt rc_rug CI
E=[ -15 -10 0 10 20 40 60 80 ];H=[ 30 50 100 150 300 500 1000 ];
estacoes(y).cipc=estacoes(y).cipc-rc_rug;
if estacoes(y).cipc>80
    estacoes(y).cipc=80;
elseif estacoes(y).cipc<-15
    estacoes(y).cipc=-15;
end
estacoes(y).CIP=interp2(E,H,CI,estacoes(y).cipc,hnmt);

function func_cie(hObject, eventdata, handles)
global y estacoes CI02_50_10 CI07_50_10 CI14_50_10
E=[ -15 -10 0 10 20 40 60 80 ];H=[ 30 50 100 150 300 500 1000 ];
estacoes(y).ciiec=estacoes(y).cpec+estacoes(y).rp;
if estacoes(y).ciiec>80
    estacoes(y).ciiec=80;
elseif estacoes(y).ciiec<-15
    estacoes(y).ciiec=-15;
end
estacoes(y).CIE=interp2(E,H,eval(estacoes(y).ci),estacoes(y).ciiec,150);

function edit_entidade_Callback(hObject, eventdata, handles)
function edit_entidade_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit_cidade_Callback(hObject, eventdata, handles)
function edit_cidade_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
end

```

```
function edit_estado_Callback(hObject, eventdata, handles)
function edit_estado_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit_frequencia_video_Callback(hObject, eventdata, handles)
function edit_frequencia_video_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit_frequencia_som_Callback(hObject, eventdata, handles)
function edit_frequencia_som_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function popup_classe_Callback(hObject, eventdata, handles)
function popup_classe_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit_lat_graus_Callback(hObject, eventdata, handles)
function edit_lat_graus_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit_lat_min_Callback(hObject, eventdata, handles)
function edit_lat_min_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit_lat_seg_Callback(hObject, eventdata, handles)
function edit_lat_seg_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit_altura_antena_Callback(hObject, eventdata, handles)
function edit_altura_antena_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit_hbt_Callback(hObject, eventdata, handles)
function edit_hbt_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
```

```
        set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
    end
    function edit_long_graus_Callback(hObject, eventdata, handles)
    function edit_long_graus_CreateFcn(hObject, eventdata, handles)
    if ispc
        set(hObject,'BackgroundColor','white');
    else
        set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
    end
    function edit_long_min_Callback(hObject, eventdata, handles)
    function edit_long_min_CreateFcn(hObject, eventdata, handles)
    if ispc
        set(hObject,'BackgroundColor','white');
    else
        set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
    end
    function edit_long_seg_Callback(hObject, eventdata, handles)
    function edit_long_seg_CreateFcn(hObject, eventdata, handles)
    if ispc
        set(hObject,'BackgroundColor','white');
    else
        set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
    end
    function uipanel_decalagem_SelectionChangeFcn(hObject, eventdata, handles)
    function mnu_about_Callback(hObject, eventdata, handles)
    sobre;
    function mnu_estacoes_Callback(hObject, eventdata, handles)
    mnu_save_Callback(hObject, eventdata, handles);
    estacoes_interferentes;
    function mnu_radiais_Callback(hObject, eventdata, handles)
    mnu_save_Callback(hObject, eventdata, handles);
    radiais;
```

```

function varargout = radiais(varargin)
% Begin initialization code
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn',   @radiais_OpeningFcn, ...
                  'gui_OutputFcn',    @radiais_OutputFcn, ...
                  'gui_LayoutFcn',    [] , ...
                  'gui_Callback',     []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code

function radiais_OpeningFcn(hObject, eventdata, handles, varargin)
global tv_file

handles.output = hObject;
guidata(hObject, handles);

load(tv_file, '-regexp', 'radial');
if ~isempty(radial_dist)
    set(handles.edit_radial_dist, 'String', num2str(radial_dist));
end
if ~isempty(radial)
    set(handles.listbox_radial, 'String', strvcat(radial.descricao));
    set(handles.edit_descricao, 'String', radial(1).descricao);
    set(handles.edit_cota, 'String', num2str(radial(1).cota));
    set(handles.edit_azimute, 'String', num2str(radial(1).azimute));
    set(handles.text_nmr, 'String', num2str(radial(1).nmr));
    set(handles.text_hnmr, 'String', num2str(radial(1).hnmr));
    set(handles.text_rug, 'String', num2str(radial(1).rug));

    set(handles.text_contorno_primario, 'String', num2str(radial(1).dist_primario))
    ;

    set(handles.text_contorno_secundario, 'String', num2str(radial(1).dist_secundario));
    set(handles.text_contorno_rural, 'String', num2str(radial(1).dist_rural));
else
    set(handles.edit_descricao, 'Enable', 'off');
    set(handles.edit_cota, 'Enable', 'off');
    set(handles.edit_azimute, 'Enable', 'off');
    set(handles.listbox_radial, 'Enable', 'off');
    set(handles.pushbutton_remove, 'Enable', 'off');
    set(handles.pushbutton_plotar, 'Enable', 'off');
end

function varargout = radiais_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;

function pushbutton_aplicar_Callback(hObject, eventdata, handles)
global tv_file radial radial_dist
save(tv_file, '-regexp', 'radial', '-append');
clear radial radial_dist;
close;

```



```

function pushbutton_cancelar_Callback(hObject, eventdata, handles)
clear radial radial_dist;
close;

function pushbutton_adicionar_Callback(hObject, eventdata, handles)
global radial
if isempty(get(handles.listbox_radial,'String'))
    radial(1).descricao='Radial1';
    radial(1).cota='';

    radial(1).azimute='';
    radial(1).nmr='';
    radial(1).rug='';
    radial(1).hnmr='';
    radial(1).dist_primario='';
    radial(1).dist_secundario='';
    radial(1).dist_rural='';
    set(handles.edit_descricao,'String','Radial1','Enable','on');
    set(handles.listbox_radial,'String','Radial1','Value',1,'Enable','on');
    set(handles.edit_cota,'Enable','on');
    set(handles.edit_azimute,'Enable','on');
    set(handles.pushbutton_aplicar,'Enable','on');
    set(handles.pushbutton_remove,'Enable','on');
    set(handles.pushbutton_plotar,'Enable','on');
    numradiais=1;
else
    [numradiais,temp]=size(get(handles.listbox_radial,'String'));
    numradiais=numradiais+1;
    set(handles.listbox_radial,'String',
strvcat(char(get(handles.listbox_radial,'String')),strcat('Radial',num2str(numradiais))), 'Value', numradiais);
    radial(numradiais).descricao=strcat('Radial',num2str(numradiais));
    set(handles.edit_descricao,'String',radial(numradiais).descricao);
    set(handles.edit_cota,'String','');
    set(handles.edit_azimute,'String','');
    set(handles.text_nmr,'String','');
    set(handles.text_hnmr,'String','');
    set(handles.text_rug,'String','');
    set(handles.text_contorno_primario,'String','');
    set(handles.text_contorno_secundario,'String','');
    set(handles.text_contorno_rural,'String','');
end

function pushbutton_remove_Callback(hObject, eventdata, handles)
global radial
entries = cellstr(get(handles.listbox_radial,'String'));
value = uint8(get(handles.listbox_radial,'Value'));

entries(value) = [];radial(value)= [];
if isempty(entries)
    nentries=0;
else
    nentries = size(entries,1);
end
if value > nentries
    value = value-1;
    set(handles.listbox_radial,'Value',value);
end
set(handles.listbox_radial,'String',entries);
if nentries==0

```

```

    set(handles.listbox_radial,'Value',0,'Enable','off');
    set(handles.pushbutton_remove,'Enable','off');
    set(handles.pushbutton_plotar,'Enable','off');
    set(handles.edit_descricao,'String','', 'Enable','off');
    set(handles.edit_cota,'String','', 'Enable','off');
    set(handles.edit_azimute,'String','', 'Enable','off');
    set(handles.text_nmr,'String','');
    set(handles.text_hnmr,'String','');
    set(handles.text_rug,'String','');
    set(handles.text_contorno_primario,'String','');
    set(handles.text_contorno_secundario,'String','');
    set(handles.text_contorno_rural,'String','');
end
listbox_radial_Callback(hObject, eventdata, handles);

function listbox_radial_Callback(hObject, eventdata, handles)
global radial
if ~isempty(get(handles.listbox_radial,'String'))

set(handles.edit_descricao,'String',radial(get(handles.listbox_radial,'Value')
)).descricao);

set(handles.edit_cota,'String',num2str(radial(get(handles.listbox_radial,'Val
ue')).cota));

set(handles.edit_azimute,'String',num2str(radial(get(handles.listbox_radial,'
Value')).azimute));

set(handles.text_nmr,'String',num2str(radial(get(handles.listbox_radial,'Valu
e')).nmr));

set(handles.text_hnmr,'String',num2str(radial(get(handles.listbox_radial,'Val
ue')).hnmr));

set(handles.text_rug,'String',num2str(radial(get(handles.listbox_radial,'Valu
e')).rug));

set(handles.text_contorno_primario,'String',num2str(radial(get(handles.listbo
x_radial,'Value')).dist_primario));

set(handles.text_contorno_secundario,'String',num2str(radial(get(handles.list
box_radial,'Value')).dist_secundario));

set(handles.text_contorno_rural,'String',num2str(radial(get(handles.listbox_r
adial,'Value')).dist_rural));
end

function listbox_radial_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

function edit_descricao_Callback(hObject, eventdata, handles)
global radial
radial(get(handles.listbox_radial,'value')).descricao=get(hObject,'String');
set(handles.listbox_radial,'String',strvcat(radial.descricao));

function edit_descricao_CreateFcn(hObject, eventdata, handles)
if ispc

```

```

        set(hObject,'BackgroundColor','white');
    else
        set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
    end

    function edit_cota_Callback(hObject, eventdata, handles)
    global radial radial_dist
    radial(get(handles.listbox_radial,'value')).cota=str2num(get(hObject,'String'
    ));
    %Calcula a NMR:
    if isempty(radial_dist)
        return
    end
    %Limita o vetor distancia entre 3 e 15km:
    n=length(radial_dist);
    for i=1:n
        if radial_dist(i)<=3
            w=i;
        end
        if i>1
            if radial_dist(i-1)<15
                k=i;
            end
        end
    end
    if ~exist('w') || radial_dist(w)<3
        uiwait(errordlg('Impossível calcular a NMR. O vetor distancia é inferior
a 3km.','Erro!'));
        return
    end
    if ~isempty(w) && exist('w')
        y=k;
        %Limita a radial para no caso do comprimento dela ser inferior a K
        if y>length(radial(get(handles.listbox_radial,'value')).cota)
            y=length(radial(get(handles.listbox_radial,'value')).cota);
        end
        if w>length(radial(get(handles.listbox_radial,'value')).cota)
            errordlg('Esta cota está fora dos parametros!','Erro!');
            return
        end
        radlim=radial(get(handles.listbox_radial,'value')).cota(w:y);%vetor
        radial limitado entre 3 e 15km

        radial(get(handles.listbox_radial,'value')).nmr=sum(radlim)/length(radlim);%s
        omatório das radiais / total de radiais

        set(handles.text_nmr,'String',radial(get(handles.listbox_radial,'value')).nmr
        );
    else
        errordlg('Esta cota está fora do padrão!','Erro!');
    end

    %Calcula a rugosidade:
    inter=radial_dist(2)-radial_dist(1);
    for i=1:n
        if radial_dist(i)<=10
            u=i;
        end
        if i>1
            if radial_dist(i-1)<50
                v=i;
            end
        end
    end

```

```

        end
    end
end
if radial_dist(u)<10
    errordlg('Impossível calcular a rugosidade. O vetor distancia é inferior
a 10km.','Erro!');
    return
end
if v>length(str2num(get(handles.edit_cota,'String')))
    v=length(str2num(get(handles.edit_cota,'String')));
end

yl=radial(get(handles.listbox_radial,'value')).cota(u:v);
xlc=radial_dist(u:v);
ylc=sort(yl);
comp=inter*length(yl)-inter;
dl0=0.1*comp;

for i=1:length(yl)-1
    if xlc(i+1)-xlc(1)<=dl0
        hl0=ylc(i+1);
    end
    if xlc(length(yl))-xlc(length(yl)-i)<=dl0
        h90=ylc(length(yl)-i);
    end
end
radial(get(handles.listbox_radial,'value')).rug=h90-hl0;
set(handles.text_rug,'String',radial(get(handles.listbox_radial,'value')).rug
);

function edit_cota_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

function edit_azimute_Callback(hObject, eventdata, handles)
global radial
radial(get(handles.listbox_radial,'value')).azimute=str2num(get(hObject,'Stri
ng'));

function edit_azimute_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

function pushbutton_plotar_Callback(hObject, eventdata, handles)
dist=str2num(get(handles.edit_radial_dist,'String'));
cota=str2num(get(handles.edit_cota,'String'));
ncota=length(cota);
ndist=length(dist);
if ncota>ndist
    errordlg('A quantidade de cotas é superior a quantidade de valores da
distancia!','Erro!');
    set(handles.edit_cota,'String','');
else
    axes(handles.grafico);
    plot(dist(1:ncota),cota);

```

```
        grid on;
        title(get(handles.edit_descricao,'String'));
    end

function pushbutton_importar_Callback(hObject, eventdata, handles)
tv_file_temp = uigetfile('*.mat','Importar Radiais');

if ~isequal(tv_file_temp,0)
    load(tv_file_temp,'-regexp','radial');
else
    return
end
if ~isempty(radial_dist)
    set(handles.edit_radial_dist,'String',num2str(radial_dist));
end
if ~isempty(radial)
    set(handles.listbox_radial,'String',strvcat(radial.descricao));
    set(handles.edit_descricao,'Enable','on');
    set(handles.edit_cota,'Enable','on');
    set(handles.edit_azimute,'Enable','on');
    set(handles.listbox_radial,'Enable','on');
    set(handles.pushbutton_remove,'Enable','on');
    set(handles.pushbutton_plotar,'Enable','on');
    set(handles.edit_descricao,'String',radial(1).descricao);
    set(handles.edit_cota,'String',num2str(radial(1).cota));
    set(handles.edit_azimute,'String',num2str(radial(1).azimute));
    set(handles.text_nmr,'String',num2str(radial(1).nmr));
    set(handles.text_rug,'String',num2str(radial(1).rug));
    set(handles.text_hnmr,'String',num2str(radial(1).hnmr));

    set(handles.text_contorno_primario,'String',num2str(radial(1).dist_primario))
    ;

    set(handles.text_contorno_secundario,'String',num2str(radial(1).dist_secundario));
    set(handles.text_contorno_rural,'String',num2str(radial(1).dist_rural));

end

function edit_radial_dist_Callback(hObject, eventdata, handles)
global radial_dist
radial_dist=str2num(get(hObject,'String'));
```

```

function varargout = estacoes_interferentes(varargin)
% Begin initialization code
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn',   @estacoes_interferentes_OpeningFcn, ...
                  'gui_OutputFcn',    @estacoes_interferentes_OutputFcn, ...
                  'gui_LayoutFcn',    [], ...
                  'gui_Callback',     []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code

% --- Executes just before estacoes_interferentes is made visible.
function estacoes_interferentes_OpeningFcn(hObject, eventdata, handles,
varargin)
global tv_file estacoes

handles.output = hObject;
guidata(hObject, handles);

load(tv_file,'estacoes');
if ~isempty(estacoes)
    set(handles.listbox_estacoes,'String',strvcat(estacoes.descricao));
    set(handles.edit_desc,'String',estacoes(1).descricao);
    set(handles.edit_canal,'String',estacoes(1).canal);
    if estacoes(1).decalagem=='Nao'

set(handles.uipanel_decalagem,'SelectedObject',handles.radiobutton_dec_nao);
    elseif estacoes(1).decalagem=='+'

set(handles.uipanel_decalagem,'SelectedObject',handles.radiobutton_dec_mais);
    elseif estacoes(1).decalagem=='-'

set(handles.uipanel_decalagem,'SelectedObject',handles.radiobutton_dec_menos)
;
    end
    set(handles.popupmenu_tipo_canal,'Value',estacoes(1).tipo_canal);
    set(handles.edit_potencia,'String',estacoes(1).potencia);
    set(handles.edit_azimute,'String',estacoes(1).azimute);
    set(handles.text_distancia,'String',estacoes(1).distancia);
    set(handles.text_CPE,'String',estacoes(1).CPE);
    set(handles.text_CIE,'String',estacoes(1).CIE);
    set(handles.text_CIP,'String',estacoes(1).CIP);
    set(handles.text_CPP,'String',estacoes(1).CPP);
    set(handles.text_cpec,'String',estacoes(1).cpec);
    set(handles.text_ciec,'String',estacoes(1).ciec);
    set(handles.text_cipc,'String',estacoes(1).cipc);
    set(handles.text_cppc,'String',estacoes(1).cppc);
    set(handles.text_CPE_CIP,'String',estacoes(1).CPE_CIP);
    set(handles.text_CPP_CIE,'String',estacoes(1).CPP_CIE);
    set(handles.text_fc_estacoes,'String',estacoes(1).fator_correcao);

set(handles.text_tipo_interferencia,'String',estacoes(1).tipo_interferencia);
    set(handles.text_interfere,'String',estacoes(1).interferente);

```

```

        set(handles.edit_lat_graus,'String',num2str(fix(estacoes(1).latitude)));
        set(handles.edit_lat_min,'String',num2str(fix((estacoes(1).latitude-
fix(estacoes(1).latitude))*60)));
        set(handles.edit_lat_seg,'String',num2str(fix((((estacoes(1).latitude-
fix(estacoes(1).latitude))*60)-fix((estacoes(1).latitude-
fix(estacoes(1).latitude))*60))*60)));

set(handles.edit_long_graus,'String',num2str(fix(estacoes(1).longitude)));
        set(handles.edit_long_min,'String',num2str(fix((estacoes(1).longitude-
fix(estacoes(1).longitude))*60)));
        set(handles.edit_long_seg,'String',num2str(fix((((estacoes(1).longitude-
fix(estacoes(1).longitude))*60)-fix((estacoes(1).longitude-
fix(estacoes(1).longitude))*60))*60)));
else
    set(handles.edit_desc,'Enable','off');
    set(handles.edit_canal,'Enable','off');
    set(handles radiobutton_dec_nao,'Enable','off');
    set(handles radiobutton_dec_mais,'Enable','off');
    set(handles radiobutton_dec_menos,'Enable','off');
    set(handles.popupmenu_tipo_canal,'Enable','off');
    set(handles.edit_lat_graus,'Enable','off');
    set(handles.edit_lat_min,'Enable','off');
    set(handles.edit_lat_seg,'Enable','off');
    set(handles.edit_long_graus,'Enable','off');
    set(handles.edit_long_min,'Enable','off');
    set(handles.edit_long_seg,'Enable','off');
    set(handles.edit_potencia,'Enable','off');
    set(handles.edit_azimute,'Enable','off');
    set(handles.listbox_estacoes,'Enable','off');
    set(handles.pushbutton_aplicar,'Enable','off');
    set(handles.pushbutton_remove,'Enable','off');
end

function varargout = estacoes_interferentes_OutputFcn(hObject, eventdata,
handles)
varargout{1} = handles.output;

function edit_desc_Callback(hObject, eventdata, handles)
global estacoes
estacoes(get(handles.listbox_estacoes,'value')).descricao=get(hObject,'String
');
set(handles.listbox_estacoes,'String',strvcat(estacoes.descricao));

function edit_desc_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

function edit_canal_Callback(hObject, eventdata, handles)
global estacoes
x=get(handles.listbox_estacoes,'Value');
if ~isempty(get(handles.edit_canal,'String'))
    if str2num(get(handles.edit_canal,'String'))>=2 &&
str2num(get(handles.edit_canal,'String'))<=6
        set(handles.popupmenu_tipo_canal,'Value',1);
        estacoes(get(handles.listbox_estacoes,'value')).cp=58;

estacoes(get(handles.listbox_estacoes,'value')).canal=str2num(get(hObject,'St
ring'));

```

```
elseif str2num(get(handles.edit_canal,'String'))>=7 &&
str2num(get(handles.edit_canal,'String'))<=13
    set(handles.popupmenu_tipo_canal,'Value',1);
    estacoes(get(handles.listbox_estacoes,'value')).cp=64;

estacoes(get(handles.listbox_estacoes,'value')).canal=str2num(get(hObject,'String'));
elseif str2num(get(handles.edit_canal,'String'))>=14 &&
str2num(get(handles.edit_canal,'String'))<=59
    set(handles.popupmenu_tipo_canal,'Value',2);
    estacoes(get(handles.listbox_estacoes,'value')).cp=70;

estacoes(get(handles.listbox_estacoes,'value')).canal=str2num(get(hObject,'String'));
else
    uiwait(errordlg('Canal Inválido. (2 a 59)','Erro!'));
    set(handles.edit_canal,'String','');
end
estacoes(x).tipo_canal=get(handles.popupmenu_tipo_canal,'Value');
atualiza_contorno_estacoes(hObject, eventdata, handles);
else
    estacoes(x).canal=[];
end

switch get(handles.edit_canal,'String')
    case '' ,estacoes(x).frequencia_video=0;estacoes(x).frequencia_audio=0;
    case
'2',estacoes(x).frequencia_video=55.25;estacoes(x).frequencia_audio=59.75;
    case
'3',estacoes(x).frequencia_video=61.25;estacoes(x).frequencia_audio=65.75;
    case
'4',estacoes(x).frequencia_video=67.25;estacoes(x).frequencia_audio=71.75;
    case
'5',estacoes(x).frequencia_video=77.25;estacoes(x).frequencia_audio=81.75;
    case
'6',estacoes(x).frequencia_video=83.25;estacoes(x).frequencia_audio=87.75;
    case
'7',estacoes(x).frequencia_video=175.25;estacoes(x).frequencia_audio=179.75;
    case
'8',estacoes(x).frequencia_video=181.25;estacoes(x).frequencia_audio=185.75;
    case
'9',estacoes(x).frequencia_video=187.25;estacoes(x).frequencia_audio=191.75;
    case
'10',estacoes(x).frequencia_video=193.25;estacoes(x).frequencia_audio=197.75;
    case
'11',estacoes(x).frequencia_video=199.25;estacoes(x).frequencia_audio=203.75;
    case
'12',estacoes(x).frequencia_video=205.25;estacoes(x).frequencia_audio=209.75;
    case
'13',estacoes(x).frequencia_video=211.25;estacoes(x).frequencia_audio=215.75;
    case
'14',estacoes(x).frequencia_video=471.25;estacoes(x).frequencia_audio=475.75;
    case
'15',estacoes(x).frequencia_video=477.25;estacoes(x).frequencia_audio=481.75;
    case
'16',estacoes(x).frequencia_video=483.25;estacoes(x).frequencia_audio=487.75;
    case
'17',estacoes(x).frequencia_video=489.25;estacoes(x).frequencia_audio=493.75;
    case
'18',estacoes(x).frequencia_video=495.25;estacoes(x).frequencia_audio=499.75;
```



```
case
'19',estacoes(x).frequencia_video=501.25;estacoes(x).frequencia_audio=505.75;
case
'20',estacoes(x).frequencia_video=507.25;estacoes(x).frequencia_audio=511.75;
case
'21',estacoes(x).frequencia_video=513.25;estacoes(x).frequencia_audio=517.75;
case
'22',estacoes(x).frequencia_video=519.25;estacoes(x).frequencia_audio=523.75;
case
'23',estacoes(x).frequencia_video=525.25;estacoes(x).frequencia_audio=529.75;
case
'24',estacoes(x).frequencia_video=531.25;estacoes(x).frequencia_audio=535.75;
case
'25',estacoes(x).frequencia_video=537.25;estacoes(x).frequencia_audio=541.75;
case
'26',estacoes(x).frequencia_video=543.25;estacoes(x).frequencia_audio=547.75;
case
'27',estacoes(x).frequencia_video=549.25;estacoes(x).frequencia_audio=553.75;
case
'28',estacoes(x).frequencia_video=555.25;estacoes(x).frequencia_audio=559.75;
case
'29',estacoes(x).frequencia_video=561.25;estacoes(x).frequencia_audio=565.75;
case
'30',estacoes(x).frequencia_video=567.25;estacoes(x).frequencia_audio=571.75;
case
'31',estacoes(x).frequencia_video=573.25;estacoes(x).frequencia_audio=577.75;
case
'32',estacoes(x).frequencia_video=579.25;estacoes(x).frequencia_audio=583.75;
case
'33',estacoes(x).frequencia_video=585.25;estacoes(x).frequencia_audio=589.75;
case
'34',estacoes(x).frequencia_video=591.25;estacoes(x).frequencia_audio=595.75;
case
'35',estacoes(x).frequencia_video=597.25;estacoes(x).frequencia_audio=601.75;
case
'36',estacoes(x).frequencia_video=603.25;estacoes(x).frequencia_audio=607.75;
case
'37',estacoes(x).frequencia_video=609.25;estacoes(x).frequencia_audio=613.75;
case
'38',estacoes(x).frequencia_video=615.25;estacoes(x).frequencia_audio=619.75;
case
'39',estacoes(x).frequencia_video=621.25;estacoes(x).frequencia_audio=625.75;
case
'40',estacoes(x).frequencia_video=627.25;estacoes(x).frequencia_audio=631.75;
case
'41',estacoes(x).frequencia_video=633.25;estacoes(x).frequencia_audio=637.75;
case
'42',estacoes(x).frequencia_video=639.25;estacoes(x).frequencia_audio=643.75;
case
'43',estacoes(x).frequencia_video=645.25;estacoes(x).frequencia_audio=649.75;
case
'44',estacoes(x).frequencia_video=651.25;estacoes(x).frequencia_audio=655.75;
case
'45',estacoes(x).frequencia_video=657.25;estacoes(x).frequencia_audio=661.75;
case
'46',estacoes(x).frequencia_video=663.25;estacoes(x).frequencia_audio=667.75;
case
'47',estacoes(x).frequencia_video=669.25;estacoes(x).frequencia_audio=673.75;
case
'48',estacoes(x).frequencia_video=675.25;estacoes(x).frequencia_audio=679.75;
```

```
case
'49',estacoes(x).frequencia_video=681.25;estacoes(x).frequencia_audio=685.75;
case
'50',estacoes(x).frequencia_video=687.25;estacoes(x).frequencia_audio=691.75;
case
'51',estacoes(x).frequencia_video=693.25;estacoes(x).frequencia_audio=697.75;
case
'52',estacoes(x).frequencia_video=699.25;estacoes(x).frequencia_audio=703.75;
case
'53',estacoes(x).frequencia_video=705.25;estacoes(x).frequencia_audio=709.75;
case
'54',estacoes(x).frequencia_video=711.25;estacoes(x).frequencia_audio=715.75;
case
'55',estacoes(x).frequencia_video=717.25;estacoes(x).frequencia_audio=721.75;
case
'56',estacoes(x).frequencia_video=723.25;estacoes(x).frequencia_audio=727.75;
case
'57',estacoes(x).frequencia_video=729.25;estacoes(x).frequencia_audio=733.75;
case
'58',estacoes(x).frequencia_video=735.25;estacoes(x).frequencia_audio=739.75;
case
'59',estacoes(x).frequencia_video=741.25;estacoes(x).frequencia_audio=745.75;
end

function edit_canal_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

function edit_potencia_Callback(hObject, eventdata, handles)
global estacoes
estacoes(get(handles.listbox_estacoes,'value')).potencia=str2num(get(hObject,
'String'));
atualiza_contorno_estacoes(hObject, eventdata, handles);

function edit_potencia_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

function edit_azimute_Callback(hObject, eventdata, handles)
global estacoes
estacoes(get(handles.listbox_estacoes,'value')).azimute=str2num(get(hObject,
'String'));

function edit_azimute_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

function popupmenu_tipo_canal_Callback(hObject, eventdata, handles)
global estacoes
if str2num(get(handles.edit_canal,'String'))>1 &&
str2num(get(handles.edit_canal,'String'))<14
    set(handles.popupmenu_tipo_canal,'Value',1);
```

```

else
    set(handles.popupmenu_tipo_canal,'Value',2);
end
estacoes(get(handles.listbox_estacoes,'value')).tipo_canal=get(hObject,'value');
atualiza_contorno_estacoes(hObject, eventdata, handles);

function popupmenu_tipo_canal_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

function pushbutton_aplicar_Callback(hObject, eventdata, handles)
global tv_file estacoes
save(tv_file,'estacoes','-append');
clear estacoes;
close;

function pushbutton_remover_Callback(hObject, eventdata, handles)
global estacoes
entries = cellstr(get(handles.listbox_estacoes,'String'));
value = uint8(get(handles.listbox_estacoes,'Value'));
entries(value) = [];estacoes(value)= [];
if isempty(entries)
    nentries=0;
else
    nentries = size(entries,1);
end
if value > nentries
    value = value-1;
    set(handles.listbox_estacoes,'Value',value)
end
set(handles.listbox_estacoes,'String',entries)
if nentries==0
    set(handles.listbox_estacoes,'Value',0)
    set(handles.pushbutton_remover,'Enable','off')
    set(handles.edit_desc,'String','');
    set(handles.edit_canal,'String','');
    set(handles.popupmenu_tipo_canal,'Value',1);
    set(handles.edit_potencia,'String','');
    set(handles.edit_azimute,'String','');
    set(handles.text_distancia,'String','');
    set(handles.text_CPE,'String','');
    set(handles.text_CIE,'String','');
    set(handles.text_CIP,'String','');
    set(handles.text_CPP,'String','');
    set(handles.text_cpec,'String','');
    set(handles.text_ciec,'String','');
    set(handles.text_cipc,'String','');
    set(handles.text_cppc,'String','');
    set(handles.text_CPE_CIP,'String','');
    set(handles.text_CPP_CIE,'String','');
    set(handles.text_fc_estacoes,'String','');
    set(handles.text_tipo_interferencia,'String','');
    set(handles.text_interfere,'String','');
    set(handles.text_tipo_interferencia,'String','');
    set(handles.edit_lat_graus,'String','');
    set(handles.edit_lat_min,'String','');
    set(handles.edit_lat_seg,'String','');

```

```

set(handles.edit_long_graus,'String','');
set(handles.edit_long_min,'String','');
set(handles.edit_long_seg,'String','');
set(handles.edit_desc,'Enable','off');
set(handles.edit_canal,'Enable','off');
set(handles.radiobutton_dec_nao,'Enable','off');
set(handles.radiobutton_dec_mais,'Enable','off');
set(handles.radiobutton_dec_menos,'Enable','off');
set(handles.popupmenu_tipo_canal,'Enable','off');
set(handles.edit_lat_graus,'Enable','off');
set(handles.edit_lat_min,'Enable','off');
set(handles.edit_lat_seg,'Enable','off');
set(handles.edit_long_graus,'Enable','off');
set(handles.edit_long_min,'Enable','off');
set(handles.edit_long_seg,'Enable','off');
set(handles.popupmenu_tipo_canal,'Enable','off');
set(handles.edit_potencia,'Enable','off');
set(handles.edit_azimute,'Enable','off');
set(handles.listbox_estacoes,'Enable','off');
end
listbox_estacoes_Callback(hObject, eventdata, handles)

function pushbutton_importar_Callback(hObject, eventdata, handles)
global estacoes
tv_file_temp = uigetfile('*.mat','Importar Estações');

if ~isequal(tv_file_temp,0)
    load(tv_file_temp,'estacoes');
else
    return
end
if ~isempty(estacoes)
    set(handles.edit_desc,'Enable','on');
    set(handles.edit_canal,'Enable','on');
    set(handles.radiobutton_dec_nao,'Enable','on');
    set(handles.radiobutton_dec_mais,'Enable','on');
    set(handles.radiobutton_dec_menos,'Enable','on');
    set(handles.popupmenu_tipo_canal,'Enable','on');
    set(handles.edit_lat_graus,'Enable','on');
    set(handles.edit_lat_min,'Enable','on');
    set(handles.edit_lat_seg,'Enable','on');
    set(handles.edit_long_graus,'Enable','on');
    set(handles.edit_long_min,'Enable','on');
    set(handles.edit_long_seg,'Enable','on');
    set(handles.edit_potencia,'Enable','on');
    set(handles.edit_azimute,'Enable','on');
    set(handles.listbox_estacoes,'Enable','on');
    set(handles.pushbutton_aplicar,'Enable','on');
    set(handles.pushbutton_remove,'Enable','on');
    set(handles.listbox_estacoes,'String',strvcat(estacoes.descricao));
    set(handles.edit_desc,'String',estacoes(1).descricao);
    set(handles.edit_canal,'String',estacoes(1).canal);
    set(handles.popupmenu_tipo_canal,'Value',estacoes(1).tipo_canal);
    set(handles.edit_potencia,'String',estacoes(1).potencia);
    set(handles.edit_azimute,'String',estacoes(1).azimute);
    set(handles.text_distancia,'String',estacoes(1).distancia);
    set(handles.text_CPE,'String',estacoes(1).CPE);
    set(handles.text_CIE,'String',estacoes(1).CIE);
    set(handles.text_CIP,'String',estacoes(1).CIP);
    set(handles.text_CPP,'String',estacoes(1).CPP);
    set(handles.text_cpec,'String',estacoes(1).cpec);

```

```

        set(handles.text_ciec, 'String', estacoes(1).ciec);
        set(handles.text_cipc, 'String', estacoes(1).cipc);
        set(handles.text_cppc, 'String', estacoes(1).cppc);
        set(handles.text_CPE_CIP, 'String', estacoes(1).CPE_CIP);
        set(handles.text_CPP_CIE, 'String', estacoes(1).CPP_CIE);
        set(handles.text_fc_estacoes, 'String', estacoes(1).fator_correcao);

set(handles.text_tipo_interferencia, 'String', estacoes(1).tipo_interferencia);
set(handles.text_interfere, 'String', estacoes(1).interferente);
set(handles.edit_lat_graus, 'String', num2str(fix(estacoes(1).latitude)));
set(handles.edit_lat_min, 'String', num2str(fix((estacoes(1).latitude-
fix(estacoes(1).latitude))*60)));
set(handles.edit_lat_seg, 'String', num2str(fix((((estacoes(1).latitude-
fix(estacoes(1).latitude))*60)-fix((estacoes(1).latitude-
fix(estacoes(1).latitude))*60))*60)));

set(handles.edit_long_graus, 'String', num2str(fix(estacoes(1).longitude)));
set(handles.edit_long_min, 'String', num2str(fix((estacoes(1).longitude-
fix(estacoes(1).longitude))*60)));
set(handles.edit_long_seg, 'String', num2str(fix((((estacoes(1).longitude-
fix(estacoes(1).longitude))*60)-fix((estacoes(1).longitude-
fix(estacoes(1).longitude))*60))*60)));
end

function listbox_estacoes_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject, 'BackgroundColor', 'white');
else
    set(hObject, 'BackgroundColor', get(0, 'defaultUicontrolBackgroundColor'));
end

function listbox_estacoes_Callback(hObject, eventdata, handles)
global estacoes

if ~isempty(get(handles.listbox_estacoes, 'String'))
    x=get(handles.listbox_estacoes, 'Value');
    set(handles.edit_desc, 'String', estacoes(x).descricao);
    set(handles.edit_canal, 'String', estacoes(x).canal);
    if estacoes(x).decalagem=='Nao'

set(handles.uipanel_decalagem, 'SelectedObject', handles.radiobutton_dec_nao);
    elseif estacoes(x).decalagem=='+'

set(handles.uipanel_decalagem, 'SelectedObject', handles.radiobutton_dec_mais);
    elseif estacoes(x).decalagem=='-'

set(handles.uipanel_decalagem, 'SelectedObject', handles.radiobutton_dec_menos)
;
    end
    set(handles.popupmenu_tipo_canal, 'Value', estacoes(x).tipo_canal);
    set(handles.edit_potencia, 'String', estacoes(x).potencia);
    set(handles.edit_azimute, 'String', estacoes(x).azimute);
    set(handles.text_distancia, 'String', estacoes(x).distancia);
    set(handles.text_fc_estacoes, 'String', estacoes(x).fator_correcao);
    set(handles.text_CPE, 'String', estacoes(x).CPE);
    set(handles.text_CIE, 'String', estacoes(x).CIE);
    set(handles.text_CIP, 'String', estacoes(x).CIP);
    set(handles.text_CPP, 'String', estacoes(x).CPP);
    set(handles.text_cpec, 'String', estacoes(x).cpec);
    set(handles.text_ciec, 'String', estacoes(x).ciec);
    set(handles.text_cipc, 'String', estacoes(x).cipc);

```

```

set(handles.text_cppc, 'String', estacoes(x).cppc);
set(handles.text_CPE_CIP, 'String', estacoes(x).CPE_CIP);
set(handles.text_CPP_CIE, 'String', estacoes(x).CPP_CIE);

set(handles.text_tipo_interferencia, 'String', estacoes(x).tipo_interferencia);
set(handles.text_interfere, 'String', estacoes(x).interferente);
set(handles.edit_lat_graus, 'String', num2str(fix(estacoes(x).latitude)));
set(handles.edit_lat_min, 'String', num2str(fix((estacoes(x).latitude-
fix(estacoes(x).latitude))*60)));
set(handles.edit_lat_seg, 'String', num2str(fix((((estacoes(x).latitude-
fix(estacoes(x).latitude))*60)-fix((estacoes(x).latitude-
fix(estacoes(x).latitude))*60))*60)));

set(handles.edit_long_graus, 'String', num2str(fix(estacoes(x).longitude)));
set(handles.edit_long_min, 'String', num2str(fix((estacoes(x).longitude-
fix(estacoes(x).longitude))*60)));
set(handles.edit_long_seg, 'String', num2str(fix((((estacoes(x).longitude-
fix(estacoes(x).longitude))*60)-fix((estacoes(x).longitude-
fix(estacoes(x).longitude))*60))*60)));
end

function pushbutton_adicionar_Callback(hObject, eventdata, handles)
global estacoes

if isempty(get(handles.listbox_estacoes, 'String'))
    estacoes(1).descricao='Estação1'; estacoes(1).canal='';

set(handles.uipanel_decalagem, 'SelectedObject', handles.radiobutton_dec_nao);
estacoes(1).decalagem='Nao'; estacoes(1).tipo_canal=1;
estacoes(1).potencia=''; estacoes(1).azimute='';
estacoes(1).distancia=''; estacoes(1).ci=''; estacoes(1).cppc='';
estacoes(1).CIP=''; estacoes(1).cpec=''; estacoes(1).ciiec='';
estacoes(1).cipc=''; estacoes(1).interferente=''; estacoes(1).CPE_CIP='';
estacoes(1).CPP_CIE=''; estacoes(1).fator_correcao='';
estacoes(1).tipo_interferencia=''; estacoes(1).interferente='';
set(handles.edit_desc, 'String', 'Estação1', 'Enable', 'on');
set(handles.text_tipo_interferencia, 'String', '');

set(handles.listbox_estacoes, 'String', 'Estação1', 'Value', 1, 'Enable', 'on');
set(handles.text_fc_estacoes, 'String', estacoes(1).fator_correcao);
set(handles.text_CIP, 'String', estacoes(1).CIP);
set(handles.text_CPP, 'String', estacoes(1).CPP);
set(handles.text_cpec, 'String', estacoes(1).cpec);
set(handles.text_ciiec, 'String', estacoes(1).ciiec);
set(handles.text_cipc, 'String', estacoes(1).cipc);
set(handles.text_cppc, 'String', estacoes(1).cppc);
set(handles.text_CPE_CIP, 'String', estacoes(1).CPE_CIP);
set(handles.text_CPP_CIE, 'String', estacoes(1).CPP_CIE);

set(handles.text_tipo_interferencia, 'String', estacoes(1).tipo_interferencia);
set(handles.text_interfere, 'String', estacoes(1).interferente);
set(handles.edit_canal, 'Enable', 'on');
set(handles.radiobutton_dec_nao, 'Enable', 'on');
set(handles.radiobutton_dec_mais, 'Enable', 'on');
set(handles.radiobutton_dec_menos, 'Enable', 'on');
set(handles.popupmenu_tipo_canal, 'Enable', 'on');
set(handles.edit_lat_graus, 'Enable', 'on', 'String', '');
set(handles.edit_lat_min, 'Enable', 'on', 'String', '');
set(handles.edit_lat_seg, 'Enable', 'on', 'String', '');
set(handles.edit_long_graus, 'Enable', 'on', 'String', '');
set(handles.edit_long_min, 'Enable', 'on', 'String', '');

```

```

        set(handles.edit_long_seg,'Enable','on','String','');
        set(handles.edit_potencia,'Enable','on');
        set(handles.edit_azimute,'Enable','on');
        set(handles.pushbutton_aplicar,'Enable','on');
        set(handles.pushbutton_remove,'Enable','on');
        numestacoes=1;
    else
        [numestacoes,temp]=size(get(handles.listbox_estacoes,'String'));
        numestacoes=numestacoes+1;
        set(handles.listbox_estacoes,'String',
strvcat(char(get(handles.listbox_estacoes,'String')),strcat('Estação',num2str
(numestacoes))), 'Value',numestacoes);
        estacoes(numestacoes).descricao=strcat('Estação',num2str(numestacoes));
        estacoes(numestacoes).tipo_canal=1;

set(handles.uipanel_decalagem,'SelectedObject',handles.radiobutton_dec_nao);
estacoes(numestacoes).decalagem='Nao';
set(handles.edit_desc,'String',estacoes(numestacoes).descricao);
set(handles.edit_canal,'String','');
set(handles.popupmenu_tipo_canal,'Value',1);
set(handles.edit_lat_graus,'String','');
set(handles.edit_lat_min,'String','');
set(handles.edit_lat_seg,'String','');
set(handles.edit_long_graus,'String','');
set(handles.edit_long_min,'String','');
set(handles.edit_long_seg,'String','');
set(handles.edit_potencia,'String','');
set(handles.edit_azimute,'String','');
set(handles.text_distancia,'String','');
set(handles.text_CPE,'String','');
set(handles.text_CIE,'String','');
set(handles.text_CIP,'String','');
set(handles.text_CPP,'String','');
set(handles.text_cpec,'String','');
set(handles.text_ciec,'String','');
set(handles.text_cipc,'String','');
set(handles.text_cppc,'String','');
set(handles.text_fc_estacoes,'String','');
set(handles.text_CPE_CIP,'String','');
set(handles.text_CPP_CIE,'String','');
set(handles.text_tipo_interferencia,'String','');
set(handles.text_interfere,'String','');
end

function atualiza_contorno_estacoes(hObject, eventdata, handles)
global estacoes CP02_50_50 CP07_50_50 CP14_50_50

if ~isempty(get(handles.edit_potencia,'String'))
    %fc=10*log(P/1kW)

estacoes(get(handles.listbox_estacoes,'Value')).fator_correcao=10*log10(str2n
um(get(handles.edit_potencia,'String'))/1000);
else
    estacoes(get(handles.listbox_estacoes,'Value')).fator_correcao=0;
end
H=[ 30 50 100 150 300 500 1000 ];E=[ -15 -10 0 10 20 40 60 80 100];

if str2num(get(handles.edit_canal,'String')) >= 2 &
str2num(get(handles.edit_canal,'String')) <= 6 &
get(handles.popupmenu_tipo_canal,'Value') == 1

```

```

    estacoes(get(handles.listbox_estacoes,'Value')).cpec=58-
estacoes(get(handles.listbox_estacoes,'Value')).fator_correcao;

estacoes(get(handles.listbox_estacoes,'Value')).CPE=interp2(E,H,CP02_50_50,(e
stacoes(get(handles.listbox_estacoes,'Value')).cpec),150);
    estacoes(get(handles.listbox_estacoes,'Value')).ci='CI02_50_10';
elseif str2num(get(handles.edit_canal,'String')) >= 7 &
str2num(get(handles.edit_canal,'String')) <= 13 &
get(handles.popupmenu_tipo_canal,'Value') == 1
    estacoes(get(handles.listbox_estacoes,'Value')).cpec=64-
estacoes(get(handles.listbox_estacoes,'Value')).fator_correcao;

estacoes(get(handles.listbox_estacoes,'Value')).CPE=interp2(E,H,CP07_50_50,(e
stacoes(get(handles.listbox_estacoes,'Value')).cpec),150);
    estacoes(get(handles.listbox_estacoes,'Value')).ci='CI07_50_10';
else
    estacoes(get(handles.listbox_estacoes,'Value')).cpec=70-
estacoes(get(handles.listbox_estacoes,'Value')).fator_correcao;

estacoes(get(handles.listbox_estacoes,'Value')).CPE=interp2(E,H,CP14_50_50,(e
stacoes(get(handles.listbox_estacoes,'Value')).cpec),150);
    estacoes(get(handles.listbox_estacoes,'Value')).ci='CI14_50_10';
end

if estacoes(get(handles.listbox_estacoes,'Value')).cpec>100
    estacoes(get(handles.listbox_estacoes,'Value')).cpec=100;
elseif estacoes(get(handles.listbox_estacoes,'Value')).cpec<-15
    estacoes(get(handles.listbox_estacoes,'Value')).cpec=-15;
end

set(handles.text_fc_estacoes,'String',estacoes(get(handles.listbox_estacoes,'
Value')).fator_correcao);
set(handles.text_CPE,'String',estacoes(get(handles.listbox_estacoes,'Value'))
.CPE);
set(handles.text_cpec,'String',estacoes(get(handles.listbox_estacoes,'Value')
).cpec);

function salva_coordenadas(hObject, eventdata, handles)
global estacoes
if ~isempty(get(handles.edit_lat_graus,'String')) &&
~isempty(get(handles.edit_lat_min,'String')) &&
~isempty(get(handles.edit_lat_seg,'String'))

estacoes(get(handles.listbox_estacoes,'Value')).latitude=str2num(get(handles.
edit_lat_graus,'String'))+str2num(get(handles.edit_lat_min,'String'))/60+str2
num(get(handles.edit_lat_seg,'String'))/3600;
end
if ~isempty(get(handles.edit_long_graus,'String')) &&
~isempty(get(handles.edit_long_min,'String')) &&
~isempty(get(handles.edit_long_seg,'String'))

estacoes(get(handles.listbox_estacoes,'Value')).longitude=str2num(get(handles
.edit_long_graus,'String'))+str2num(get(handles.edit_long_min,'String'))/60+s
tr2num(get(handles.edit_long_seg,'String'))/3600;
end

function calcula_distancia(hObject, eventdata, handles)
global tv_file estacoes
load(tv_file,'latitude','longitude');
x=get(handles.listbox_estacoes,'Value');

```



```
if ~isempty(estacoes(x).latitude) && ~isempty(estacoes(x).longitude) &&  
~isempty(latitude) && ~isempty(longitude)  
estacoes(x).distancia=111.1775*(acos(sin(latitude*pi/180)*sin(estacoes(x).lat  
itude*pi/180)+cos(latitude*pi/180)*cos(estacoes(x).latitude*pi/180)*cos((long  
itude-estacoes(x).longitude)*pi/180))/pi*180);  
set(handles.text_distancia,'String',estacoes(x).distancia);  
end
```

```
function edit_lat_graus_Callback(hObject, eventdata, handles)  
salva_coordenadas(hObject, eventdata, handles);  
calcula_distancia(hObject, eventdata, handles);
```

```
function edit_lat_graus_CreateFcn(hObject, eventdata, handles)  
if ispc  
    set(hObject,'BackgroundColor','white');  
else  
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));  
end
```

```
function edit_lat_min_Callback(hObject, eventdata, handles)  
salva_coordenadas(hObject, eventdata, handles);  
calcula_distancia(hObject, eventdata, handles);
```

```
function edit_lat_min_CreateFcn(hObject, eventdata, handles)  
if ispc  
    set(hObject,'BackgroundColor','white');  
else  
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));  
end
```

```
function edit_lat_seg_Callback(hObject, eventdata, handles)  
salva_coordenadas(hObject, eventdata, handles);  
calcula_distancia(hObject, eventdata, handles);
```

```
function edit_lat_seg_CreateFcn(hObject, eventdata, handles)  
if ispc  
    set(hObject,'BackgroundColor','white');  
else  
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));  
end
```

```
function edit_long_graus_Callback(hObject, eventdata, handles)  
salva_coordenadas(handles, eventdata, handles);  
calcula_distancia(hObject, eventdata, handles);
```

```
function edit_long_graus_CreateFcn(hObject, eventdata, handles)  
if ispc  
    set(hObject,'BackgroundColor','white');  
else  
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));  
end
```

```
function edit_long_min_Callback(hObject, eventdata, handles)  
salva_coordenadas(handles, eventdata, handles);  
calcula_distancia(hObject, eventdata, handles);
```

```
function edit_long_min_CreateFcn(hObject, eventdata, handles)  
if ispc  
    set(hObject,'BackgroundColor','white');  
else  
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));  
end
```

end

```
function edit_long_seg_Callback(hObject, eventdata, handles)
salva_coordenadas(handles, eventdata, handles);
calcula_distancia(hObject, eventdata, handles);
```

```
function edit_long_seg_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
```

```
function pushbutton_cancelar_Callback(hObject, eventdata, handles)
clear estacoes;
close;
```

```
function uipanel_decalagem_SelectionChangeFcn(hObject, eventdata, handles)
global estacoes
estacoes(get(handles.listbox_estacoes,'Value')).decalagem=get(get(handles.uipanel_decalagem,'SelectedObject'),'string');
```

```
function varargout = sobre(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn',   @sobre_OpeningFcn, ...
                  'gui_OutputFcn',    @sobre_OutputFcn, ...
                  'gui_LayoutFcn',    [] , ...
                  'gui_Callback',     []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function sobre_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;
guidata(hObject, handles);
set(hObject, 'Color', 'white');

function varargout = sobre_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;
```