



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

SIMULAÇÃO NUMÉRICA DE RELÂMPAGOS: IMPLEMENTAÇÕES

**RELATÓRIO FINAL DE PROJETO DE INICIAÇÃO CIENTÍFICA
(PIBIC/CNPq/INPE)**

Denise Fernandes da Silva (UNITAU, Bolsista PIBIC/CNPq)
E-mail: defersi@hotmail.com

Dr. Odim Mendes Junior (CEA/DGE/INPE, Orientador)
E-mail: odim@dge.inpe.br

Dra. Margarete Oliveira Domingues (CPTEC/ INPE)
E-mail: margaret@cptec.inpe.br

Junho de 2003

Resumo

O relâmpago, embora um fenômeno muito antigo no conhecimento da humanidade, apresenta inúmeros aspectos básicos a serem melhor conhecidos, como o instante exato da sua ocorrência e o caminho que percorrerá. Este trabalho objetiva contribuir para os estudos da eletrodinâmica das descargas elétricas atmosféricas, por meio da simulação numérica do caminho do líder escalonado, que é a etapa precursora na geração desse caminho. Esse caminho fica visualmente bem identificado quando, por exemplo, em uma descarga descendente que atinge o solo, a descarga de retorno, que vem depois da conclusão do caminho, ocorre produzindo luminosidade intensa ao longo da trajetória seguida. No intuito de aprimorar o código de simulação do caminho do relâmpago, no primeiro ano de trabalho, houve o embasamento na física do relâmpago; o manuseio do código de computador já existente (desenvolvido pelo orientador); a pesquisa de uma ferramenta de programação e visualização integradas (o programa Scilab) ; e o ensaio de resultados envolvendo o entendimento de efeitos no campo elétrico atmosférico, por meio do controle de alguns parâmetros atmosféricos (como configuração de cargas, condutividade elétrica atmosférica e condições de fronteira). A metodologia de trabalho consistiu da programação neste ambiente integrado, que caracteriza-se por ser uma ferramenta robusta e de livre utilização. Como resultados, foram gerados no Scilab alguns gráficos para a visualização do ambiente elétrico onde ocorre a formação de relâmpagos. Da análise, concluiu-se que a configuração do campo elétrico na troposfera/estratosfera mostra-se afetada pelos pressupostos físicos do modelo adotado. No segundo ano, a intenção é continuar o aprimoramento do código de simulação e fazer o desenvolvimento em linguagem C++ sob o paradigma POO. Devido à característica de bifurcação do relâmpago, pretende-se introduzir o conceito de árvore binária. A implementação está no estágio de análise e proposição.

Sumário

1	INTRODUÇÃO	2
2	A FÍSICA PARA A SIMULAÇÃO	4
2.1	A Física da Descarga Elétrica	4
2.1.1	Cargas Elétricas	4
2.1.2	Campo Elétrico	5
2.1.3	Potencial Elétrico	5
2.1.4	Atmosfera	6
2.1.5	Condutividade Atmosférica	7
2.1.6	Circuito Elétrico Atmosférico Global (CEAG)	8
2.1.7	Campo Elétrico de uma Nuvem de Tempestade	8
2.1.8	As nuvens cumulonimbus	11
2.1.9	Os Relâmpagos	12
2.2	Relâmpagos Negativos	12
2.2.1	Relâmpagos Nuvem-Solo	12
2.2.2	As etapas do Líder Escalonado	14
3	AS SIMULAÇÕES DA TRAJETÓRIA	16
3.1	O Modelo Físico	16

3.1.1	Exemplo de simulação:	18
3.1.2	Fase Atual do Projeto	19
3.2	Scilab	20
3.2.1	Programação no Scilab	20
3.3	Programação Orientada a Objeto	21
4	Resultados de Simulação	23
4.1	Criação das Funções	23
4.2	Conteúdo dos Arquivos Utilizados para Gerar os Gráficos	26
4.3	Gráficos Gerados	32
5	MODELO 3D	37
5.1	Análise do Problema Físico para Obtenção de um Algoritmo	37
5.2	Idéias para o Desenvolvimento da Programação Orientada a Objeto	39
5.3	Diagrama de Casos de Uso e Diagrama de Classes	40
5.4	Código para Implementação de Árvores Binárias	41
6	Conclusão	49
	Referências Bibliográficas	50
A	INTRODUÇÃO AO SCILAB	52
A.1	Interface	52
A.2	Características	53
A.3	Operações básicas	53
A.3.1	Atribuição de valores	53
A.3.2	Definindo polinômios	54
A.4	Vetores	54

A.4.1	Declaração de vetores	54
A.4.2	Operações com vetores	55
A.5	Operadores:	56
A.6	Matrizes	57
A.7	Gráficos	58
A.7.1	Plotando gráficos	58
A.8	Recursos de Programação	59
A.8.1	Criando funções:	59
A.8.2	Comandos para Iterações:	60
A.8.3	Comandos Condicionais:	61
A.8.4	Ligação do Scilab com Programas em C:	61

Lista de Figuras

2.1	Camadas da atmosfera	7
2.2	Circuito Elétrico Global	9
2.3	Estrutura elétrica de uma cumulonimbus	10
2.4	Estágios de uma nuvem cumulonimbus	12
2.5	Etapas de um relâmpago	15
3.1	Modelo para a simulação	18
3.2	Resultado de uma simulação	19
4.1	Visualização do cenário elétrico na atmosfera em xy utilizando o plot3d	33
4.2	Visualização do cenário elétrico na atmosfera em xy utilizando o fcontour . . .	34
4.3	Visualização do cenário elétrico na atmosfera em xz utilizando o plot3d	34
4.4	Visualização do cenário elétrico na atmosfera em xz utilizando o fcontour . . .	35
4.5	Visualização do cenário elétrico na atmosfera em yz utilizando o plot3d	35
4.6	Visualização do cenário elétrico na atmosfera em yz utilizando o fcontour . . .	36
5.1	Modelo 3D	39
5.2	Diagrama de Casos de Uso	40
5.3	Diagrama de Classes	41
A.1	Ambiente do Scilab	52

A.2 Exemplo de gráfico plotado	58
A.3 Curva plotada no Scilab	59
A.4 Múltiplas curvas plotadas ao mesmo tempo	60

Lista de Tabelas

A.1 Operadores relacionais	56
A.2 Operadores lógicos	57

Capítulo 1

INTRODUÇÃO

Nas mitologias gregas e indo-européia, o relâmpago era um atributo divino, com o qual os deuses poderosos manifestavam sua ira e fulminavam heróis e humanos que se opunham aos seus desígnios (Bordinhão, 2002). Atualmente, sabe-se que os relâmpagos são fenômenos elétricos produzidos por diferenças de potencial elétrico na atmosfera, com energia suficiente para superar a resistência do ar. Na superfície da Terra e de alguns outros planetas, como Júpiter, os relâmpagos restabelecem o equilíbrio elétrico entre as nuvens e o solo, transmitindo abruptamente as cargas elétricas acumuladas em um determinado ponto.

O movimento do ar nas nuvens de tempestades, pelo efeito da gravidade e convecção, faz com que as cargas negativas se acumulem em sua base e as cargas positivas em seu topo. Quando a diferença de potencial elétrico entre as cargas positivas e negativas se torna excessiva, superando a rigidez elétrica do ar, ocorre uma descarga que pode progredir, por exemplo, da nuvem para o solo. Isso gera fenômenos luminosos, em alguns casos marcando claramente caminhos por onde passaram as descargas elétricas, e trovões, que são os estrondos que se ouvem alguns segundos após os relâmpagos, produzidos pela expansão e contração súbita do ar atravessado pela descarga elétrica. Ao processo de descarga como um todo dá-se o nome de relâmpago (Uman, 1987).

Desde séculos passados até os tempos atuais, são feitos estudos e experimentos sobre o fenômeno relâmpago. Atualmente, conta-se com vários recursos tecnológicos, técnicas e metodologias de observação e medição dos relâmpagos (Bent e Lyons, 1984; Krehbiel et al.,

2000), a que se integram outras de simulação numérica das descargas elétricas (Mendes et al., 1997), propiciando abordagens que permitem investigar os parâmetros físicos associados a esse fenômeno. Essas abordagens permitem coletar informações experimentais e desenvolver cálculos, possibilitando, desta forma, estudar o comportamento e a influência de parâmetros físicos associados às descargas elétricas atmosféricas.

Este projeto de iniciação científica tem como objetivo dar prosseguimento à implementação da ferramenta computacional para visualização de dados de relâmpagos e simulação numérica do caminho da descarga elétrica.

Com esse trabalho, está sendo feita a ampliação do modelo físico que rege o fenômeno de descarga elétrica atmosférica, por meio do embasamento do candidato no formalismo eletrodinâmico e em métodos matemáticos; a implementação do algoritmo computacional para a simulação numérica da descarga elétrica, visando uma futura integração no *ThoR*, - *SLA* (Mendes e Domingues, 2000), por meio do embasamento do candidato nos paradigmas computacionais e da utilização inteligente de ferramentas computacionais atuais; e desenvolvimento de uma análise preliminar de resultados para verificação da programação, desenvolvendo com isso o senso crítico científico e refinando as atitudes profissionais do candidato.

No primeiro ano do projeto, foi feita uma revisão bibliográfica envolvendo embasamentos em matemática, física e informática, para trabalhar com a questão do caminho do relâmpago na atmosfera. Também fez-se um estudo mais aprofundado sobre o *SCILAB*, um programa de manipulação de matrizes, funções matemáticas e programação, para verificar como as cargas interagem com o ambiente onde ocorrem os relâmpagos.

No capítulo 2, apresentam-se as linhas gerais da Física para Simulação, envolvendo a física da descarga elétrica atmosférica, o modelo da descarga negativa descendente e o modelo de simulação.

No capítulo 3, apresentam-se as ferramentas de simulações das trajetórias dos relâmpagos.

No capítulo 4, apresentam-se os resultados obtidos na primeira etapa do projeto.

No capítulo 5, apresentam-se os encaminhamentos obtidos na segunda etapa do projeto.

No capítulo 6, apresentam-se as conclusões deste trabalho.

Capítulo 2

A FÍSICA PARA A SIMULAÇÃO

Neste capítulo introduzem-se os elementos básicos associados à Física da descarga elétrica atmosférica e às etapas da própria descarga, de forma a embasar o desenvolvimento do código computacional de simulação numérica a ser realizado.

2.1 A Física da Descarga Elétrica

2.1.1 Cargas Elétricas

A força eletromagnética entre partículas carregadas é uma das forças fundamentais da natureza (Serway, 1996). Na natureza, tudo é formado por cargas elétricas (Festa, 2001). Estas se encontram em dois estados primordiais: positivo (prótons) e negativo (elétrons). Quando duas partículas estacionárias estão eletricamente carregadas, há uma força elétrica que age sobre elas. A força é atrativa se as cargas possuírem tipos de eletrização diferentes e é repulsiva se as cargas possuírem a mesma espécie de eletrização. Essa força é dada pela lei de Coulomb, que foi estabelecida em 1785 por Charles A. de Coulomb (1736-1806). A lei é a seguinte: *A interação eletrostática entre duas partículas carregadas é proporcional às suas cargas e ao inverso do quadrado da distância entre elas e tem a direção da reta que une as duas cargas* (Alonso e Edward, 1967).

$$\vec{F}_{12} = \frac{1}{4\pi\epsilon_0} \frac{q_1 q_2}{r^3} \vec{r}_{12}, \quad (2.1)$$

em que \vec{r} é a distância entre as duas cargas q_1 e q_2 , \vec{F} é a força que atua sobre qualquer das cargas, e ϵ_0 é a constante da permissividade elétrica do vácuo.

2.1.2 Campo Elétrico

Se se considerar uma pequena carga de prova, por exemplo q_2 , pode-se fazer a abstração de um campo físico, denominado campo elétrico, dado por \vec{E} , em qualquer ponto do espaço. Esse campo elétrico produz sobre as cargas elétricas uma força, capaz de atrair ou repelir outras cargas elétricas à distância.

$$\vec{E} = \frac{1}{4\pi\epsilon_0} \frac{q_1}{r^3} \vec{r} \quad (2.2)$$

Pode-se notar que a intensidade em um ponto é igual à força elétrica por unidade de carga colocada nesse ponto.

$$\vec{E} = \frac{\vec{F}_{12}}{q_2} \quad (2.3)$$

Em uma região onde existem partículas ou íons positivos e negativos, o campo elétrico irá mover os corpos carregados positivamente e negativamente em sentidos opostos, resultando em uma separação de cargas, geralmente chamada de polarização. Portanto, se há uma carga positiva, a força sobre a carga tem o mesmo sentido do campo elétrico, mas se a carga é negativa, a força tem o sentido oposto do campo.

2.1.3 Potencial Elétrico

Uma partícula carregada colocada em um campo elétrico, tem energia potencial devido à sua integração com o campo. O potencial elétrico em um ponto é definido como a energia potencial por unidade de carga colocada no ponto.

$$V = \frac{E}{q} \quad (2.4)$$

O potencial elétrico é medido em joules/coulomb, uma unidade chamada volt (V).

2.1.4 Atmosfera

A atmosfera pode ser caracterizada por suas propriedades. Em função da temperatura, a atmosfera é dividida em quatro camadas (Goody e Walker, 1975), que são:

Troposfera - é a camada inferior da atmosfera, onde a temperatura diminui com a altura. Ela se estende em média 12 km de altura. Este valor é de 9 km nos pólos, e de 16 km no Equador. É na troposfera onde quase todas as nuvens e, certamente, toda a precipitação ocorre.

Estratosfera - é a região acima da troposfera, onde a temperatura se mantém constante por aproximadamente 20 km, antes de aumentar rapidamente até a altura de 50 km. Esse aquecimento, é produzido pela absorção dos raios ultravioleta pelo ozônio.

Mesosfera - é a região onde a temperatura volta a diminuir até a mesopausa, região a aproximadamente 80 km acima da superfície.

Termosfera - é a região acima da mesopausa e sem limite superior definido. Nesta camada, extremamente rarefeita, a temperatura aumenta novamente como resultado da absorção da radiação solar por átomos de oxigênio e nitrogênio.

Segundo as características elétricas, divide-se a atmosfera em atmosfera quase neutra (fracamente ionizada), e Ionosfera, região com alta densidade eletrônica, acima de 70 km de altura.

Na Ionosfera, os átomos e moléculas de oxigênio e moléculas de nitrogênio, são ionizados pela absorção da radiação solar de alta energia. Neste processo, cada molécula ou átomo perde um ou mais elétrons e se torna um íon positivo, e os elétrons ficam livres, constituindo correntes elétricas na região (Lutgens e Tarbuck, 1982). Esta região define uma das fronteiras elétricas para os modelos de simulação. A outra fronteira considerada é o solo, pela sua condutividade elétrica. Sendo que o ambiente eletrificado é a atmosfera contida entre essas fronteiras. É na troposfera que desenvolvem-se as tempestades elétricas, pelo desenvolvimento das nuvens Cumulonimbus. O meio em que os relâmpagos ocorrem são a troposfera e a estratosfera, embora os relâmpagos produzam efeitos em outras regiões atmosféricas. A Ionosfera é uma região importante na eletrodinâmica planetária

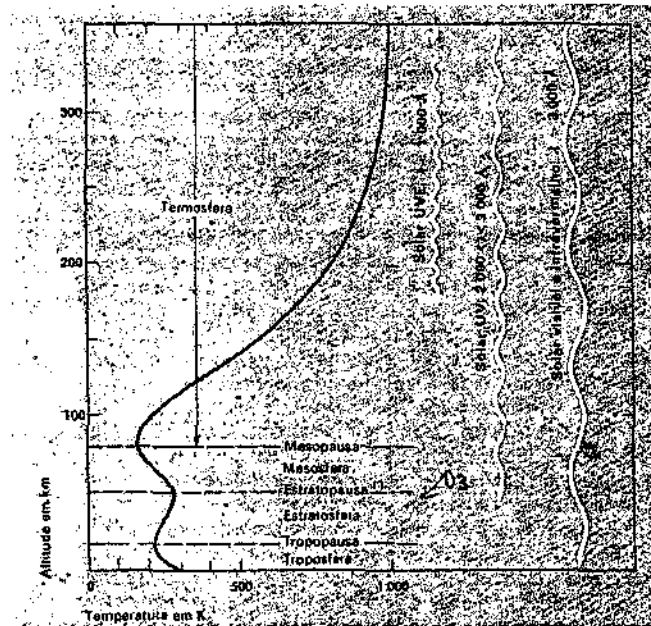


Figura 2.1: Camadas da atmosfera

FONTE: Goody e Walker (1975)

2.1.5 Condutividade Atmosférica

A atmosfera é ionizada de várias formas:

- Próximo ao solo: materiais radioativos emitem radiação beta e gama ionizando as moléculas do ar nos primeiros metros acima da superfície. O gás radônio emite partículas alfa, ionizando o ar.
- Raios cósmicos galácticos: a ionização máxima ocorre em torno de 100 km de altura.

Existem várias fontes de ionização, como os raios-x solares e o ultravioleta extremo, elétrons e prótons de alta energia de origem solar.

O parâmetro condutividade afeta a eletrodinâmica das descargas.

2.1.6 Circuito Elétrico Atmosférico Global (CEAG)

É necessário distinguir duas situações atmosféricas: a de 'tempo bom' e a de 'tempo perturbado'. Na situação de tempo bom, pode haver a presença de nuvens, desde que não haja evidência de perturbações no campo elétrico. Já na situação de tempo perturbado, há a presença de nuvens com processos de separação de cargas e perturbações no campo elétrico. O CEAG consiste de duas superfícies equipotenciais, a superfície da Terra e a Ionosfera, separadas pelo ar atmosférico, que é um meio levemente condutor, onde a condutividade aumenta exponencialmente com a altura. Existe uma diferença de potencial (ddp) entre essas duas superfícies, que é mantida pela atividade de tempestades elétricas. Devido aos geradores de cargas elétricas (as nuvens de tempestade, de acordo com o CEAG), não ocorre o desaparecimento da ddp entre a superfície da Terra e a Ionosfera, pois eles compensam a corrente vertical de condução nas regiões de tempo bom.

2.1.7 Campo Elétrico de uma Nuvem de Tempestade

Na atmosfera existe água na forma de vapor misturado ao ar e é a condensação deste vapor, em torno de partículas suspensas no ar, que origina as nuvens (Gin, 1997). A partir da colisão entre diferentes tipos de partículas, como cristais de gelo e granizo, dentro das nuvens, desenvolvem-se cargas intensas. Na estrutura elétrica das nuvens pode haver camadas de blindagem nos contornos da nuvem: negativa no topo, e positiva na base. Esta blindagem forma-se pela captura por gotículas ou cristais de gelo, de íons atmosféricos dirigindo-se para os centros de carga (Iribarne e Cho, 1980; Gin, 1997). Existem duas formas de eletrização das nuvens: o processo indutivo e o processo termoelétrico. No processo indutivo ocorre a polarização de partículas grandes, como o granizo, devido a presença de um campo elétrico externo. Sendo este campo descendente e havendo colisão na parte inferior do granizo com partículas menores, como o cristal de gelo, o granizo transfere para o cristal cargas positivas em excesso, ficando negativamente carregado. Já no processo termoelétrico, ocorre a transferência de carga dependendo da temperatura de onde ocorre a colisão, denominada de temperatura de inversão de carga (TR). Ocorre entre o granizo e o cristal de gelo. Se a colisão ocorrer a uma temperatura inferior a TR, o granizo transfere carga positiva e torna-se negativamente

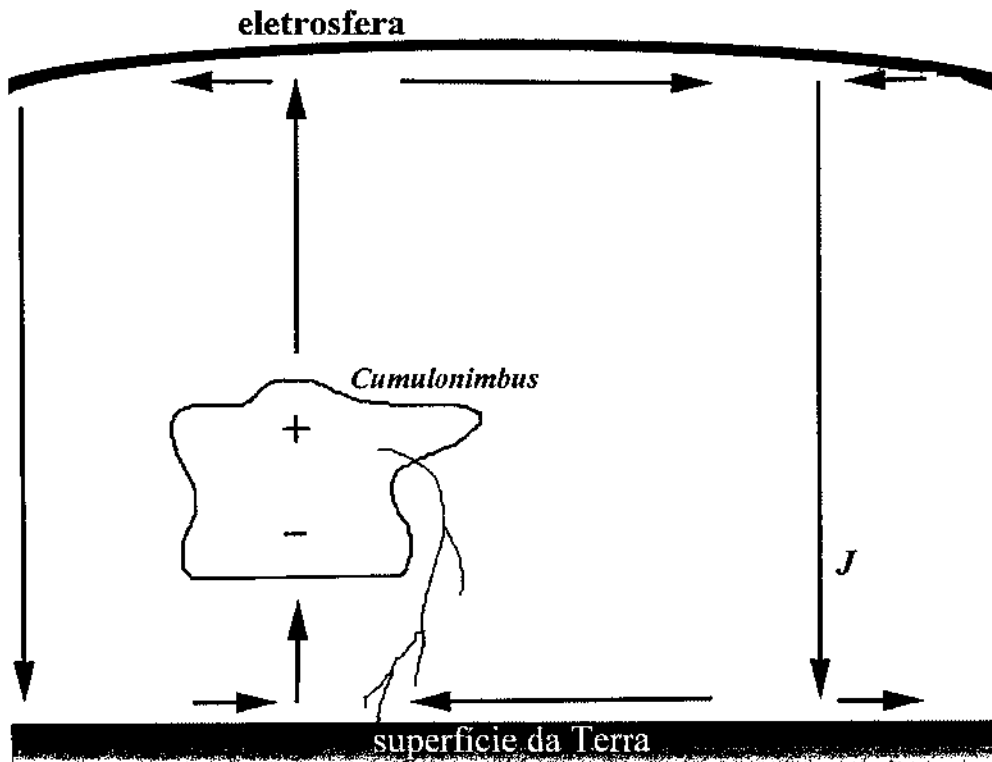


Figura 2.2: Esquema do Circuito Elétrico Global; em que J é a corrente que circula no campo elétrico.

FONTE: Mendes e Domingues (2000)

carregado; se ocorrer a uma temperatura superior a T_R , o granizo transfere carga negativa, ficando positivamente carregado. As cargas, depois de geradas, sofrem ação de processos de transporte dentro da nuvem. Dois processos são conhecidos: o gravitacional (ou de precipitação) e o convectivo. No processo gravitacional, as partículas maiores e mais pesadas associam-se às cargas negativas e/ou positivas, dependendo da temperatura, distribuindo-se na base da nuvem. No processo convectivo, segundo Williams (1988), as cargas positivas são liberadas da superfície da Terra por correntes de ar aquecido e as cargas negativas são produzidas pela interação dos raios cósmicos com a atmosfera associados às partículas já existentes na nuvem. Assim, as cargas positivas são levadas, através de movimentos ascendentes, para o topo da nuvem, e as cargas negativas são levadas, através de movimentos descendentes, para

a base; formando a camada de blindagem da nuvem.

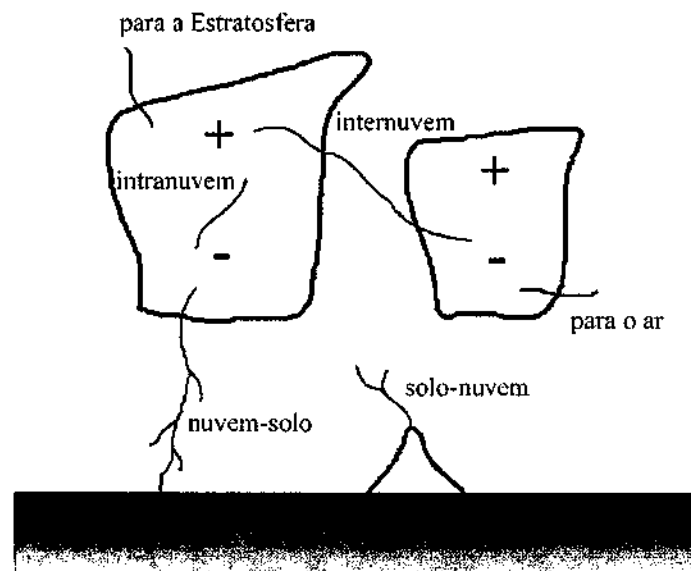


Figura 2.3: Estrutura elétrica de uma cumulonimbus

FONTE: Mendes e Domingues (2000)

Uma nuvem de tempestade elétrica possui uma estrutura interna e apresenta processos de carregamento elétrico.

Esse tipo de nuvem possui ainda, três estágios:

- Inicial(cumulus);
- Maduro;
- Dissipativo.

No estágio *inicial*, a nuvem cessa seu crescimento, a menos que haja um contínuo suprimento de ar úmido. Se isto ocorrer, ocorrerá um rápido desenvolvimento vertical, cujo *updraft*

(corrente ascendente) pode registrar valores substancialmente altos (Saba, 1997). Normalmente, após meia hora, o tamanho médio das gotas de água formadas é muito grande para que o *updraft* as sustente. Assim, a precipitação das mesmas causa um arrasto sobre o ar e inicia o *downdraft* (vento para baixo), que é favorecido pelo influxo do ar seco e frio circunvizinho à nuvem, um processo chamado de entranhamento. Esse processo intensifica o *downdraft* porque o ar adicionado durante o entranhamento é mais frio e mais seco, o que faz parte da precipitação evaporar, esfriando ainda mais o ar dentro do *downdraft*. Desta forma o *downdraft* surge na base da nuvem, a precipitação aparece, caracterizando o início do estágio maduro. No solo o *downdraft* frio se espalha horizontalmente e pode ser percebido antes que a precipitação chegue à superfície. Rajadas de vento estão associadas a este *downdraft* da nuvem. Durante o estágio maduro *updrafts* coexistem com *downdraft*, possibilitando a continuidade do crescimento da nuvem. Quando o crescimento atinge a tropopausa, o *updraft* espalha-se lateralmente produzindo uma bigorna no topo. Uma vez que um *downdraft* se inicia, o processo de entranhamento vai se intensificando até dominar toda a nuvem, iniciando o estágio dissipativo. Se em um suprimento de umidade, a nuvem logo se evapora. Essas nuvens caracterizam-se em função do sentido do movimento vertical predominante das correntes de ar em seu interior. Seu ciclo de vida é de uma a duas horas.

2.1.8 As nuvens cumulonimbus

As nuvens cumulonimbus são denominadas nuvens de tempestades e, como tal, possui estágios, uma estrutura interna e processos de carga, descritos anteriormente. Atuam como um gerador elétrico, recompondo as cargas na atmosfera. Elas são caracterizadas pelo forte movimento convectivo e grande extensão vertical. O processo de formação destas nuvens depende da umidade do ar e da instabilidade atmosférica. Sua estrutura elétrica pode ser caracterizada como um dipolo elétrico vertical com o centro de carga positiva principal na parte superior.

Na cumulonimbus geralmente ocorrem os relâmpagos. Seu tempo de vida é de aproximadamente uma hora, mas ao mover-se, novos suprimentos de ar quente e úmido podem gerar novas células que substituirão aquelas em dissipação (Lutgens e Tarbuck, 1982).

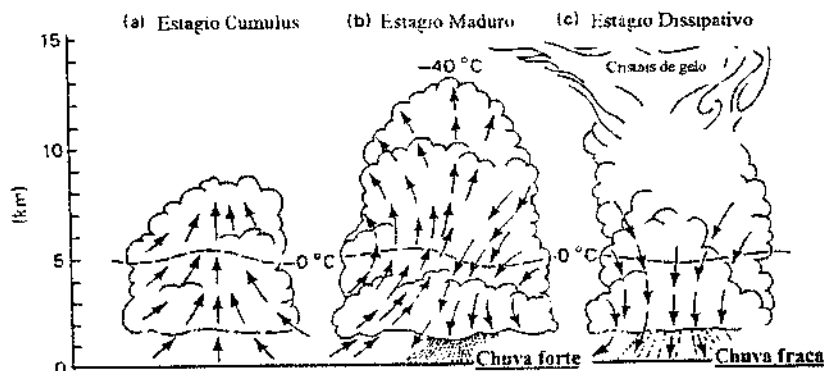


Figura 2.4: Estágios de uma nuvem cumulonimbus

FONTE: Lutgens e Tarbuck (1982)

2.1.9 Os Relâmpagos

Os relâmpagos são descargas elétricas atmosféricas, com milhares de ampères. A duração do relâmpago, em geral, não passa de alguns décimos de segundo. Eles são conseqüências do acúmulo de cargas elétricas, geralmente em nuvens cumulonimbus, e ocorrem quando o campo elétrico excede a capacidade de isolamento do ar. Se os relâmpagos neutralizam as cargas negativas no centro de onde se originam, são denominados relâmpagos negativos; se neutralizam cargas positivas, são relâmpagos positivos. Existem também os relâmpagos bipolares, relâmpagos de múltiplas descargas e que apresentam ambas polaridades (MacGornan e Rust, 1998; Uman, 1987; Volland, 1984).

Os relâmpagos são classificados em: Nuvem-Solo (NS), Solo-Nuvem (SN), Intranuvem (IN), Inter-Nuvem, Horizontais e Para Ionosfera; ou mais basicamente como relâmpagos nuvem-solo e nas nuvens. Os mais freqüentes são o Nuvem-Solo e o Intranuvem. E os relâmpagos de grande interesse e de pesquisa mais fácil são os Nuvem-Solo.

2.2 Relâmpagos Negativos

2.2.1 Relâmpagos Nuvem-Solo

Um típico relâmpago NS pode ser dividido em 6 etapas:

Descarga p-N

Os relâmpagos negativos para o solo começam com uma descarga entre o centro positivo menor na base da nuvem (p) e o centro negativo (N) para a liberação dos elétrons ligados a partículas de água ou gelo (Uman, 1987).

Líder Escalonado (Stepped Leader)

Uma fraca descarga luminosa se propaga em direção ao solo, criando um canal ionizado. Essa descarga, propaga-se para baixo, até aproximadamente 5 a 100 m do solo. O líder escalonado geralmente ramifica-se, mas seus ramos não estendem-se até o solo. A duração dessa etapa é de aproximadamente 20 milissegundos. Essa descarga segue um movimento tortuoso, em busca dos caminhos mais favoráveis à formação do canal. O líder escalonado carrega uma corrente da ordem de 100 ampères, em um canal com diâmetro estimado em 1 cm ao longo do qual uma carga negativa de 5 C é distribuída (Volland, 1984; Mendes et al., 1997).

Descarga Conectante

Uma descarga surge com a intensificação do campo elétrico entre a parte frontal do líder escalonado e o solo. Tal descarga geralmente parte de objetos pontiagudos existentes na superfície, como árvores e edifícios, e desloca-se em direção à parte frontal do líder escalonado.

Descarga de Retorno (Stroke)

Quando o encontro ocorre, a descarga conectante transforma-se em uma descarga forte e de intensa luminosidade, denominada descarga de retorno. Ela é vista como um clarão. Na descarga de retorno, as cargas negativas (elétrons) presentes no canal já ionizado pelo líder

escalonado são aceleradas, criando uma corrente ascendente com picos de corrente de 40.000 ampères. A duração de tal descarga é de aproximadamente 70 microssegundos; e ocorrem, em média, 3 descargas por relâmpago.

Líder Contínuo

Após a primeira descarga de retorno, em um intervalo de 30 a 500 milissegundos, pode ocorrer uma nova descarga, denominada líder contínuo. Ela inicia-se na mesma região onde começou o líder escalonado e, geralmente, propaga-se no canal já ionizado por este. A intensidade do líder contínuo é, normalmente, menor que a do líder escalonado. O processo pode se repetir até o decaimento do canal ionizado. Observa-se que podem existir, com as descargas de retorno subseqüentes uma corrente contínua de aproximadamente 100 ampères e uma duração em torno de 100 milissegundos.

2.2.2 As etapas do Líder Escalonado

Sobre a influência de um campo elétrico estabelecido entre a nuvem e o solo, as cargas negativas movem-se em etapas de dezenas de metros de comprimento, denominadas etapas do líder (Magno, 2002). Cada etapa tem uma duração normal de 1 microssegundo, com uma pausa entre elas de 50 microssegundos. Após alguns milissegundos, o líder escalonado surge na base da nuvem, se movimentando em direção ao solo. Ao longo desse movimento, devido a influência de cargas na atmosfera ao redor do canal, algumas cargas seguem novos caminhos e formam ramificações. As cargas do canal movem-se rumo ao solo em etapas com uma velocidade média de aproximadamente 100 km/s, produzindo uma fraca luminosidade em uma região com um diâmetro de 1 a 10 m ao longo da qual a carga é depositada. A maior parte dessa luminosidade é produzida durante as etapas de 1 microssegundo, não havendo praticamente luminosidade entre as pausas. À medida que as cargas do líder propagam-se ao longo do canal em direção ao solo, são produzidas variações do campo elétrico e magnético. Ao todo, um líder escalonado transporta 10 ou mais coulombs de carga e alcança um ponto perto do solo em dezenas de milissegundos, dependendo da tortuosidade de seu caminho. A corrente média do líder escalonado é de cerca de 1 kA e é transportada em um núcleo central do canal com alguns centímetros de diâmetro.

Quando o canal do líder escalonado aproxima-se do solo, a carga elétrica contida no canal produz um campo elétrico intenso entre a extremidade do líder e o solo, que corresponde a um potencial elétrico de cerca de 100 milhões de volts. Assim que o líder escalonado se aproxima de um objeto condutor, surge uma descarga conectante; que gera uma descarga de retorno, neutralizando a carga elétrica associada ao canal.

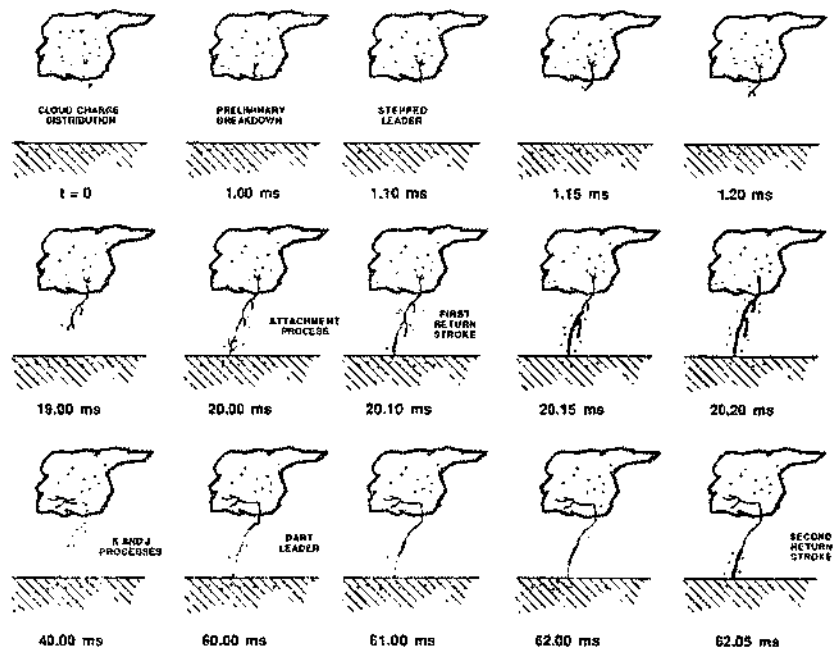


Figura 2.5: Etapas de um relâmpago

FONTE: Uman (1987)

Capítulo 3

AS SIMULAÇÕES DA TRAJETÓRIA

Neste capítulo, apresenta-se o modelo físico que regerá a simulação do canal ionizado do relâmpago e também as ferramentas básicas escolhidas para a implementação do código numérico de simulação da trajetória do líder escalonado.

3.1 O Modelo Físico

Experimentalmente, por meio de uma das técnicas de detecção, os dados dos relâmpagos são identificados através das descargas de retorno. Quando ocorre uma descarga de retorno, estações registram o momento exato da detecção, a localização (longitude e latitude), a polaridade e a intensidade de corrente das descargas. Com isso monta-se uma base de dados das ocorrências de relâmpagos (Mendes et al., 1998), porém isso não permite a visão real do fenômeno. No entanto, existem atualmente alguns modelos para o entendimento do comportamento eletrodinâmico atmosférico (Mendes et al., 1997). Alguns desses modelos lidam com situações específicas e buscam analisar a distribuição de cargas, os campos elétricos e os relâmpagos.

Neste trabalho, para a obtenção de um modelo para a trajetória e comportamento do relâmpago, assume-se que o líder escalonado segue a direção do gradiente do potencial elétrico

existente na sua frente. Assumindo que os campos magnéticos podem ser desconsiderados, que a condutividade elétrica da atmosfera é uma grandeza escalar e cresce exponencialmente com a altura, que a corrente elétrica obedece a lei de Ohm simplificada, e que a Ionosfera e o solo são fronteiras físicas, tem-se a seguinte equação diferencial para o potencial escalar (Mendes e Domingues, 2000).

$$\frac{1}{\tau_c} \nabla^2 \phi - \frac{\sigma}{\epsilon_0} (\nabla^2 \phi + 2\kappa \frac{\partial \phi}{\partial z}) = \frac{1}{\epsilon_0} \nabla \vec{J}_s, \quad (3.1)$$

em que τ_c é o tempo característico do processo; ϵ_0 é a permissividade elétrica do vácuo; $\frac{\sigma}{\epsilon_0}$ é a relaxação elétrica do ar; κ é a constante que representa escala de altura da condutividade elétrica; \vec{J}_s é a densidade superficial de corrente das cargas fontes. Escolheu-se como fronteiras físicas a ionosfera e o solo pelo seguinte motivo: A ionosfera constitui-se de uma região densamente ionizada e portanto eletricamente muito condutora. Por outro lado, o solo embora não seja condutor perfeito, ainda assim é bom condutor. Isso permite uma interação elétrica e a manutenção de um potencial elétrico que é sustentado, em escala planetária, pelas nuvens Cumulonimbus. O potencial calculado na equação 3.1 permitirá determinar o campo elétrico.

Os relâmpagos acontecem em um tempo característico T_c muito pequeno ($<1s$), assim só o primeiro termo do lado esquerdo da equação anterior precisa ser considerado. Como uma condição inicial, que corresponde ao processo de separação de cargas da nuvem, adiciona-se à solução da equação citada acima a solução da equação antes da ocorrência da descarga elétrica. Esta condição inicial é obtida considerando-se apenas o segundo termo do lado esquerdo da equação. Esta solução combinada (Mendes e Domingues, 2000) permite estabelecer uma análise eletrodinâmica atmosférica e, em particular, simular a trajetória do relâmpago.

$$\phi(x, y, z) = V_0 \left[\frac{1 - \exp(-2\kappa z)}{1 - \exp(-2\kappa H)} \right] + q_0 \frac{\exp(-\kappa(z - z_0))}{4\pi\epsilon_0} \left[\frac{\exp(-\kappa r_0)}{r_0} - \frac{\exp(-\kappa r'_0)}{r'_0} \right] \quad (3.2)$$

Sendo que:

$$r_0 = \sqrt{(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2}$$

$$r'_0 = \sqrt{(x - x_0)^2 + (y - y_0)^2 + (z + z_0)^2}$$

Em que ϕ é a função do potencial elétrico; x, y, z são as coordenadas de um ponto qualquer; V_0 é a diferença de potencial elétrico entre a eletrosfera e o solo; 2κ está associado à escala de

altura na condutividade elétrica; H é a escala de altura associada à eletrosfera; q_0 é a carga elétrica da fonte; x_0, y_0, z_0 são as coordenadas da carga elétrica da fonte; r_0 é a distância da fonte até uma posição no espaço; r'_0 é a distância da imagem da fonte até a posição no espaço; ϵ_0 é a permissividade elétrica do vácuo.

Foi desenvolvido em 1996, por Mendes e colegas, um modelo similar ao modelo de Takagi; porém considerando condições mais realistas. Utilizou-se esferas finitas de carga para o modelo da nuvem e uma condutividade elétrica exponencialmente crescente com a altura da atmosfera. Considera-se, no modelo atual, a existência de dois centros de carga, um mais alto (positivo) e outro embaixo (negativo).

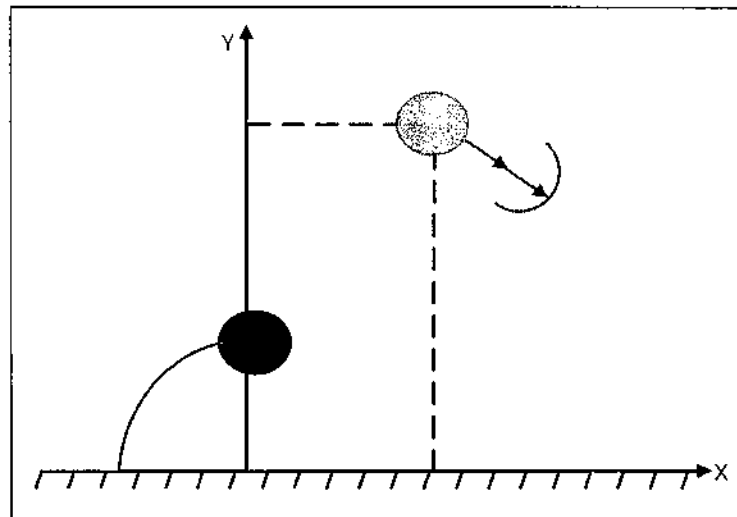


Figura 3.1: Modelo para a simulação

FONTE: Mendes e Domingues (2000)

3.1.1 Exemplo de simulação:

Esses cálculos serão feitos utilizando o modelo hoje existente e que deverá ser implementado.

O relâmpago possuirá 2 centros de carga, um positivo e um negativo. A primeira esfera possuirá carga de -10.0 C, raio de 250 m, distância de 0 m e 2000 m de altura. Já a segunda esfera, possuirá carga de 10.0 C, raio de 250 m, distância de 2000 m e 10000 m de altura. A

condutividade, dada pela constante κ , será de 0.0001. O centro inicial da carga inferior será de 1. O caminho a ser percorrido será de 8000 m, a passos de 20 m. Seu ângulo inicial será 0 graus, o final será 150 graus, com um número de intervalos de 30 graus. A carga na ponta do canal será de 0.1 C e a densidade da carga no canal será de 0.001 C.

Para automatização do processo de compilação do programa está sendo utilizado um makefile, um arquivo com instruções feito pelo programador. As informações do relâmpago, dadas acima, ficam guardadas em arquivos com os nomes `arqneg.in`, que possui as informações da esfera com cargas negativas, e `arqpos.in`, que possui as informações da esfera com cargas positivas. Utilizando uma das opções do programa, cria-se os arquivos `arqneg.cir` e `arqpos.cir`, que definem o contorno da esfera; e cria-se também os arquivos `arqneg.dat` e `arqpos.dat`, que definem os dados do caminho da descarga. Para gerar o gráfico da simulação, utiliza-se opções do `gnuplot` e obtém-se um arquivo com a extensão `eps`. O gráfico gerado é mostrado na figura 3.2.

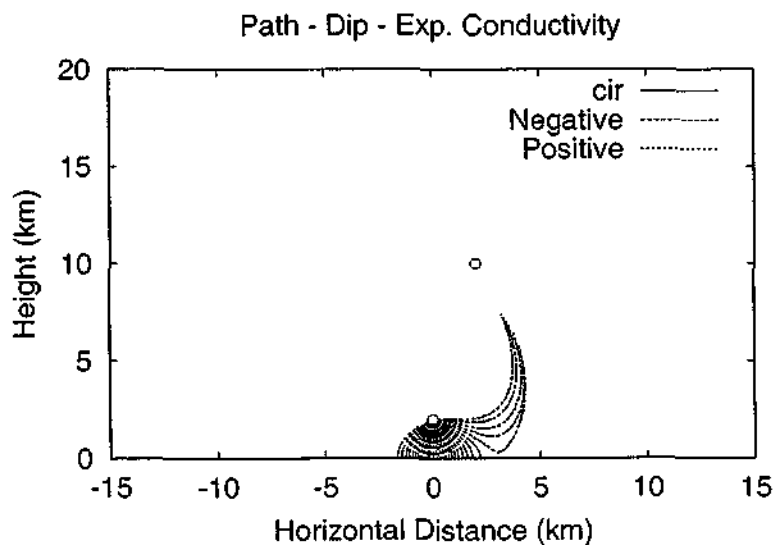


Figura 3.2: Resultado de uma simulação

ambiente Scilab. Apesar de simples, a linguagem do Scilab disponibiliza a maioria das estruturas das linguagens convencionais de programação. A diferença principal é que não há a necessidade de declaração prévia do tipo das variáveis.

O Scilab é um interpretador; os programas escritos nele normalmente são executados em um tempo maior que os mesmos programas utilizados em linguagens compiláveis. Isso acontece, principalmente, com programas utilizados em simulações e otimizações. Nesses casos, pode ser conveniente escrever o código responsável pela lentidão em uma linguagem convencional (C, Maple ou Fortran) e rodar esse código dentro do ambiente Scilab. Deve ser enfatizado, entretanto, que a vantagem na utilização dos programas Scilab vem da facilidade de prototipação e da disponibilidade de uma poderosa biblioteca de funções gráficas. Uma vantagem prática do Scilab é de ensaiar análises desejadas pelo programador de forma rápida, qualificando o próprio desenvolvimento sugerido. Uma explicação mais detalhada sobre o programa Scilab e seu funcionamento e utilização, encontram-se no Apêndice A.

3.3 Programação Orientada a Objeto

A programação orientada a objetos é baseada na escrita de programas em termos de objetos que compõem um sistema. Em um sistema, objetos armazenam tipos específicos de informação e operações específicas são realizadas com essa informação. Um objeto é uma entidade independente que descreve não apenas os dados, mas também as funções que os manipulam (Potts e Monk, 1994). A Programação Orientada a Objetos permite modelar mais proximamente qualquer problema do mundo real. E se os objetos forem bem planejados, pode-se reutilizar o código.

Objeto: Objeto é uma abstração independente de um item que pode ser um registro de dados ou uma aplicação na tela. Um objeto inclui os dados necessários para representar os dados e as funções, ou métodos, que manipulam estes dados.

Classes: Uma classe é um tipo de dado definido pelo usuário para implementar um objeto abstrato. Uma classe inclui membros: dados-membro ou função-membro.

Dados e Funções-Membro: Dados-Membro de uma classe são como variáveis em uma estrutura. Funções-Membro são funções definidas dentro de uma classe que agem sobre os

dados-membro na classe. Membros têm o atributo de visibilidade, ou seja, assim como variáveis eles podem ser vistos ou não por toda parte.

Encapsulamento de Dados: Uma meta da programação orientada a objeto é separar o uso de um objeto de sua implementação. O encapsulamento permite que se esconda um dado-membro de um objeto inteiramente e previne que usuários de um objeto tenham acesso diretamente aos dados-membro da classe.

Capítulo 4

Resultados de Simulação

Neste capítulo, mostra-se os resultados obtidos nesta etapa do projeto, com a criação de gráficos para visualização do cenário elétrico na atmosfera (valor do potencial elétrico em posição arbitrária no espaço)

4.1 Criação das Funções

Para melhor compreensão do comportamento elétrico do ambiente onde ocorrem os relâmpagos e aprofundamento nos conhecimentos sobre a ferramenta computacional Scilab, foram feitos estudos em cima desses dois tópicos . Assim obtiveram-se neste estudo, por meio de funções desenvolvidas no Scilab, resultados que mostram graficamente o cenário elétrico na atmosfera, devido à componente de tempo bom e as cargas acumuladas geradoras de relâmpagos.

Para a visualização do cenário elétrico na atmosfera foi calculado o potencial elétrico em vários pontos da atmosfera, com as posições estabelecidas pelas coordenadas x,y,z . Essas coordenadas determinam três planos ortogonais para visualização. Essa visualização é feita com os planos sendo fixados em um dos três eixos (ver as figuras na secção adiante). Para facilitar e agilizar a atribuição de valores às variáveis foram gerados arquivos, onde estão os valores necessários para fazer todos os cálculos que resultam em ϕ , função do potencial elétrico. Nesses arquivos fixam-se ora o x , ora o y e ora o z , para que assim seja feita a

visualização de dois eixos por vez (que determinam um plano, em um sistema ortogonal). O arquivo que fixa o valor de z , ou seja, o plano xy , foi chamado de `xyPot.sci`; o arquivo que fixa o y (plano xz) foi chamado `xzPot.sci`; e o arquivo que fixa valor de x (plano yz) foi chamado de `yzPot.sci`.

Criou-se também um outro arquivo onde foi desenvolvida a função que recebe os valores (parâmetros e variáveis) $x, y, z, V_0, k, Nq, xq, yq$ e zq , sendo essa função do potencial elétrico denominada `elecpot.sci`. Para que o Scilab interprete o arquivo como uma função, utilizou-se dentro do arquivo a seguinte sintaxe:

```
function[variáveis de saída]=nome da função(variáveis de saída)
```

```
    instruções da função
```

```
endfunction
```

A seguir apresenta-se o conteúdo do arquivo-função criado para retornar o valor de ϕ para o Scilab.

```
// Potencial eletrico atmosferico:
```

```
// Unidades no Sistema MKS
```

```
// Definicao da funcao do potencial eletrico
```

```
// devido ao numero de cargas Nq com os valores
```

```
// de carga q(i) e coordenadas xq(i), yq(i) e zq(i)
```

```
// sob os parametros de modelo
```

```
// v0 potencial ionosfera-solo, H altura de escala de v0,
```

```

// 2k=1/H\condutividade em que H\condutividade eh a escala

//de altura da condutividade electrica atmosferica

function[phi]=elecpot(x,y,z,v0,k,H,Nq,q,xq,yq,zq)

$e0=8.542*10^(-12)$ // Permissividade elétrica do vácuo

phi=v0*[(1-(exp(-2*k*z)))/(1-(exp(-2*k*H)))]

for i=1:Nq

    r=sqrt(((x-xq(i))^2)+((y-yq(i))^2)+((z-zq(i))^2))

    rl=sqrt(((x-xq(i))^2)+((y-yq(i))^2)+((z+zq(i))^2))

    phi=phi+(q(i)/(4*pi*e0))*(exp(-k*(z-zq(i))))*[((exp(-k*r))/r)-((exp(-k*rl))/rl)]

end

endfunction

```

É necessário introduzir uma linha de comando no Scilab, para que este carregue o arquivo que contém a função a ser executada, ou seja, este comando faz a chamada da função do potencial elétrico na atmosfera. A linha de comando utilizada é a seguinte:

```
getf('localização do arquivo/nome do arquivo')
```

Assim que este comando é executado, atribui-se os parâmetros do modelo atmosférico do potencial. Esses parâmetros são V_0 , κ e H .

Faz-se também a atribuição dos valores das variáveis físicas, valores de carga e coordenadas; estas são Nq , $q[]$, $xq[]$, $yq[]$, $zq[]$.

Para definir a função dentro do ambiente Scilab, basta digitar a seguinte linha de comando:

```
deff('w=f(x,y)', 'w=elecpot(x,y,z,v0,k,H,Nq,q,xq,yq,zq)')
```

Após definida a função, define-se os planos ortogonais (x, y, z) de interesse para a análise (x, y, z) .

Para encerrar a criação do gráfico, utiliza-se, por fim, as funções para criação dos gráficos. Duas funções foram utilizadas preliminarmente para este fim, *fplot3d()* e *fcontour()*. A seguir encontram-se as declarações de cada linha de comando:

```
fplot3d(x,y,f,alpha=50,theta=310)
```

```
fcontour(x,y,f)
```

É importante ressaltar que os planos ortogonais presentes na sintaxe do **fplot3d** e do **fcontour** variam de acordo com a variável (x , y ou z) que teve seu valor fixado. Portanto, se o valor de x foi fixado, aparecerão na sintaxe da linha de comando do programa as variáveis y e z ; se o valor fixado for o de y , aparecerão na sintaxe x e z ; e se o valor de z for fixado, aparecerão x e y .

4.2 Conteúdo dos Arquivos Utilizados para Gerar os Gráficos

Apesar de já se ter mostrado acima todas as formas de declaração de variáveis e parâmetros, as utilizações das funções e as chamadas da função; decidiu-se colocar abaixo o conteúdo dos arquivos que contém tudo isso, bastando apenas, na hora de utilizar o programa, selecionar todo o seu conteúdo, copiar e colar na linha de comando do Scilab. Dessa forma, agiliza-se o processo de criação dos gráficos.

Conteúdo do arquivo xyPot.sci

```
// Chamada da funcao do potencial eletrico na atmosfera:
```

```
getf("/home/odim/estudantes/Denise/scilab/elecpot.sci");
```



```
// Atribuicao dos parametros do modelo atmosferico do potencial:
```

```
v0=300.0;
```

```
k=0.0001;
```

```
H=70000.0;
```

```
// Atribuicao dos valores das variaveis fisicas
```

```
// Nq = numero de cargas arbitrado
```

```
Nq=4;
```

```
// Valores de carga q(i) e coordenadas xq(i),yq(i),zq(i)
```

```
q=[-25.0,25.0,25.0,-25.0];
```

```
xq=[0.0,0.0,0.0,0.0];
```

```
yq=[0.0,0.0,10000.0,10000.0];
```

```
zq=[3000.0,6000.0,3000.0,6000.0];
```

```
//q=[-25.0,25.0];
```

```
//xq=[0.0,0.0];
```

```
//yq=[0.0,0.0];
```

```
//zq=[3000.0,6000.0];

// Definicao de funcao dentro do ambiente Scilab

def('w=f(x,y)', 'w=elecpot(x,y,z,v0,k,H,Nq,q,xq,yq,zq)')

// Definicao dos planos ortogonais em analise

// passando pelo ponto arbitrario P(x0,y0,z0)

x=-15000.0:500.0:15000.0;

y=-15000.0:500.0:15000.0;

z= 4500.0;

// Chamada da funcao de criacao do grafico (plot):

xbasc() ;fplot3d(x,y,f,alpha=50,theta=310)

xbasc() ;fcontour(x,y,f)
```

Os resultados do potencial elétrico são apresentados nas figuras 4.1 (gráfico 3D) e 4.2 (isolíneas). Considerou-se o plano xy, distante em z de 4.5 km da posição de referência.

Conteúdo do arquivo xzPot.sci

```
// Chamada da funcao do potencial eletrico na atmosfera:

getf("/home/odim/estudantes/Denise/scilab/elecpot.sci");
```

```
// Atribuicao dos parametros do modelo atmosferico do potencial:

v0=300.0;

k=0.0001;

H=70000.0;

// Atribuicao dos valores das variaveis fisicas

// Nq = numero de cargas arbitrado

Nq=4;

// Valores de carga q(i) e coordenadas xq(i),yq(i),zq(i)

q=[-25.0,25.0,25.0,-25.0];

xq=[0.0,0.0,0.0,0.0];

yq=[0.0,0.0,10000.0,10000.0];

zq=[3000.0,6000.0,3000.0,6000.0];

//q=[-25.0,25.0];

//xq=[0.0,0.0];

//yq=[0.0,0.0];
```

```
//zq=[3000.0,6000.0];

// Definicao de funcao dentro do ambiente Scilab

def('w=f(x,z)', 'w=elecpot(x,y,z,v0,k,H,Nq,q,xq,yq,zq)')

// Definicao dos planos ortogonais em analise

// passando pelo ponto arbitrario P(x0,y0,z0)

x=-15000.0:500.0:15000.0;

y=4500.0;

z=0.0:500.0:10000.0;

// Chamada da funcao de criacao do grafico (plot):

xbasc() ;fplot3d(x,z,f,alpha=50,theta=310)

xbasc() ;fcontour(x,z,f)

Os resultados do potencial elétrico são apresentados nas figuras 4.3 (gráfico 3D) e 4.4 (isolíneas). Considerou-se o plano xz, distante em y de 4.5 km da posição de referência.

Conteúdo do arquivo yzPot.sci

// Chamada da funcao do potencial eletrico na atmosfera:

getf("/home/odim/estudantes/Denise/scilab/elecpot.sci");
```

3.1.2 Fase Atual do Projeto

Já existe um programa feito na linguagem de programação C, para a simulação de relâmpagos. Sua saída gráfica utiliza o GNUPlot. Para a automatização do processo e mais qualidade na simulação, estuda-se a utilização da linguagem orientada a objeto C++ em interação com o Scilab, que gera a saída gráfica. Para tal procedimento, estão sendo feitos estudos em cima do problema físico da simulação e dos programas C++ e Scilab; sendo avaliados e questionados quais seriam as possíveis soluções para esta utilização. Por hora, está sendo testada a criação de fórmulas no Scilab para a geração de gráficos. E está sendo feito também um estudo aprofundado sobre C++ e análise orientada a objetos. Objetiva-se criar o programa orientado a objeto em C++. Após o programa realizar todas as suas operações e armazená-las em arquivos, o Scilab irá utilizar esses arquivos e gerar gráficos. Já estão sendo estudados alguns tipos de gráficos, tanto em duas quanto em três dimensões.

3.2 Scilab

O Scilab é um ambiente utilizado no desenvolvimento de programas para a resolução de problemas numéricos. Criado e mantido por pesquisadores do INRIA (Institut de Recherche en Informatique et en Automatique), através do projeto MÉTALAU (Méthodes, algorithmes et logiciels pour l'automatique) e à ENPC (École Nationale des Ponts et Chaussées), o Scilab é gratuito e é distribuído com o código fonte.

Algumas de suas vantagens são: a última versão do software está sempre disponível, geralmente via internet; o software pode ser legalmente utilizado, copiado, distribuído e modificado; os resultados obtidos podem ser divulgados sem restrições.

No Scilab pode-se trabalhar com vários tipos de dados, como constantes, variáveis booleanas, polinômios, strings, frações envolvendo polinômios, matrizes, entre outros.

3.2.1 Programação no Scilab

Uma característica importante do Scilab é a possibilidade do usuário criar seus próprios programas. Estes programas podem ser executados em qualquer plataforma que possua o

```
// Atribuicao dos parametros do modelo atmosferico do potencial:
```

```
v0=300.0;
```

```
k=0.0001;
```

```
H=70000.0;
```

```
// Atribuicao dos valores das variaveis fisicas
```

```
// Nq = numero de cargas arbitrado
```

```
Nq=4;
```

```
// Valores de carga q(i) e coordenadas xq(i),yq(i),zq(i)
```

```
q=[-25.0,25.0,25.0,-25.0];
```

```
xq=[0.0,0.0,0.0,0.0];
```

```
yq=[0.0,0.0,10000.0,10000.0];
```

```
zq=[3000.0,6000.0,3000.0,6000.0];
```

```
//q=[-25.0,25.0];
```

```
//xq=[0.0,0.0];
```

```
//yq=[0.0,0.0];
```

```
//zq=[3000.0,6000.0];

// Definicao de funcao dentro do ambiente Scilab

deff('w=f(y,z)', 'w=elecpot(x,y,z,v0,k,H,Nq,q,xq,yq,zq)')

// Definicao dos planos ortogonais em analise

// passando pelo ponto arbitrario P(x0,y0,z0)

x=4500.0;

y=-15000.0:500.0:15000.0;

z=0.0:500.0:10000.0;

// Chamada da funcao de criacao do grafico (plot):

xbasc() ;fplot3d(y,z,f,alpha=50,theta=310)

xbasc() ;fcontour(y,z,f)
```

Os resultados do potencial elétrico são apresentados nas figuras 4.5 (gráfico 3D) e 4.6 (isolíneas). Considerou-se o plano yz, distante em x de 4.5 km da posição de referência.

4.3 Gráficos Gerados

Como resultado dos estudos e análises feitos no Scilab para gerar a visualização gráfica do cenário elétrico na atmosfera, obteve-se alguns resultados gráficos preliminares; estes encontram-se a seguir. Esses gráficos ilustram o comportamento do potencial elétrico na atmosfera. Esse resultado reflete o comportamento eletrodinâmico possível de ocorrer. Essa

programação básica permite controlar os parâmetros físicos existentes condicionantes da - eletrodinâmica, como, por exemplo, a condutividade elétrica, a configuração das cargas e o modelo físico adotado. De uma primeira análise dos gráficos, percebe-se que essas informações são importantes para o comportamento eletrodinâmico das descargas elétricas.

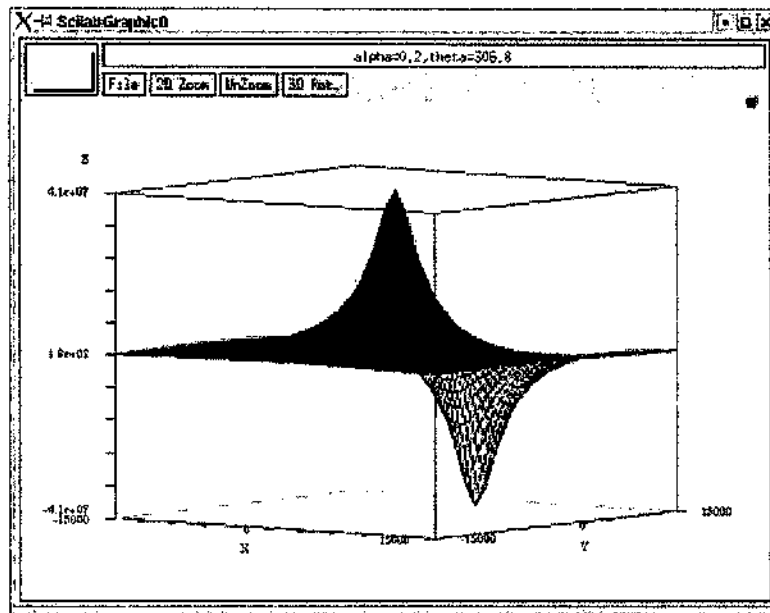


Figura 4.1: Visualização do cenário elétrico na atmosfera em xy utilizando o plot3d

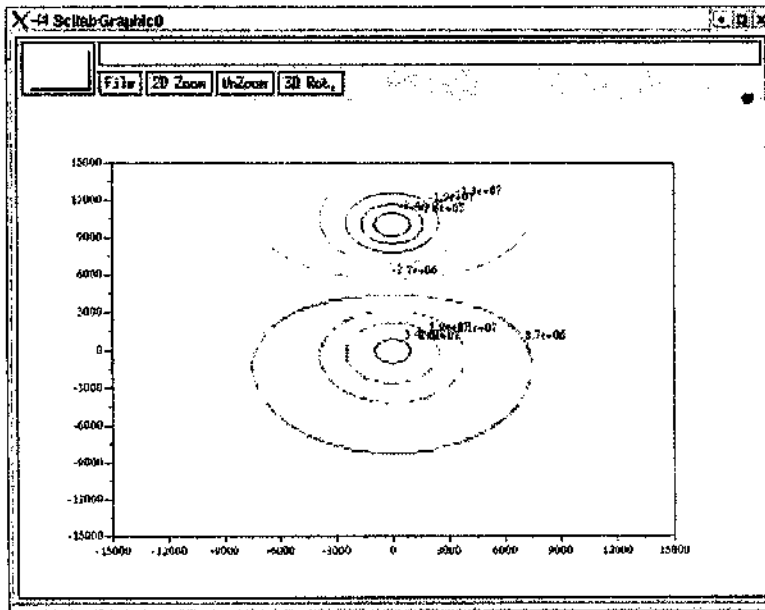


Figura 4.2: Visualização do cenário elétrico na atmosfera em xy utilizando o fcontour

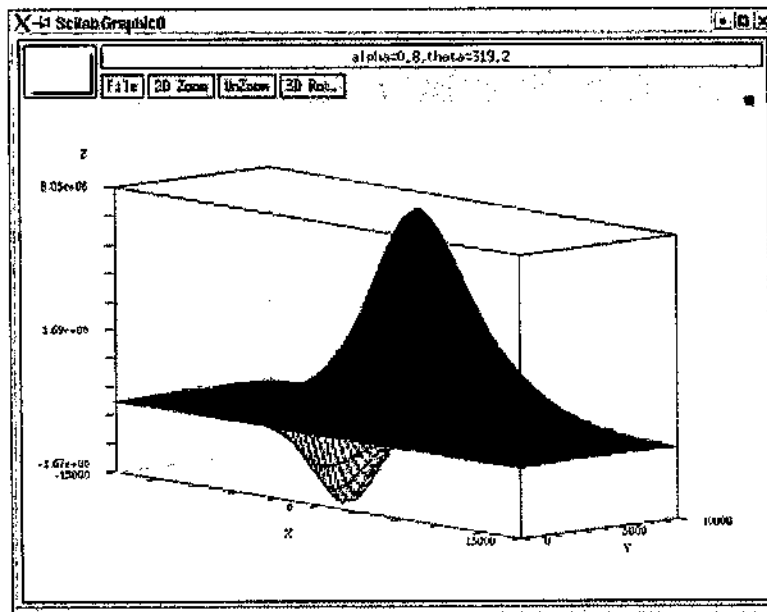


Figura 4.3: Visualização do cenário elétrico na atmosfera em xz utilizando o plot3d

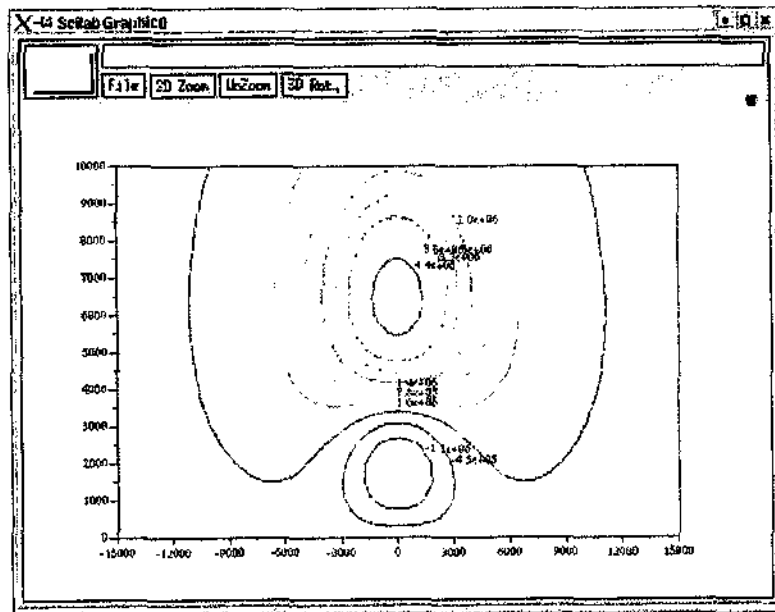


Figura 4.4: Visualização do cenário elétrico na atmosfera em xz utilizando o feontour

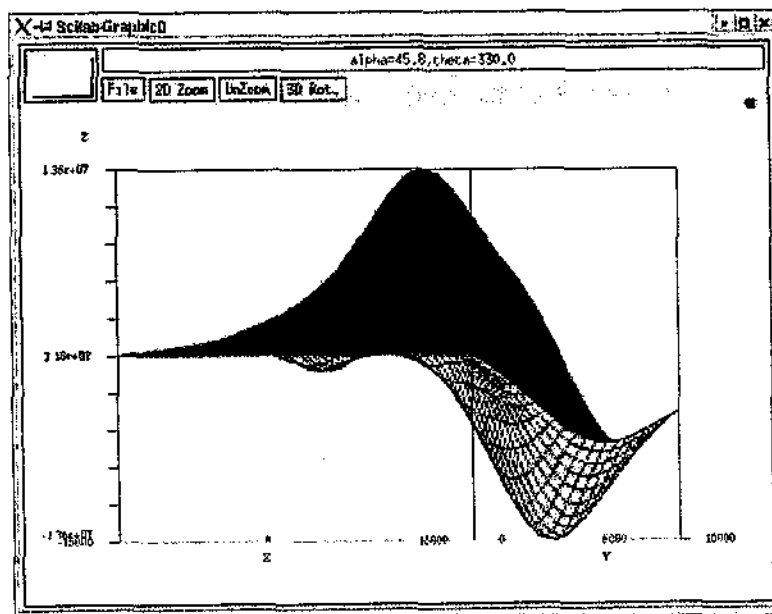


Figura 4.5: Visualização do cenário elétrico na atmosfera em yz utilizando o plot3d

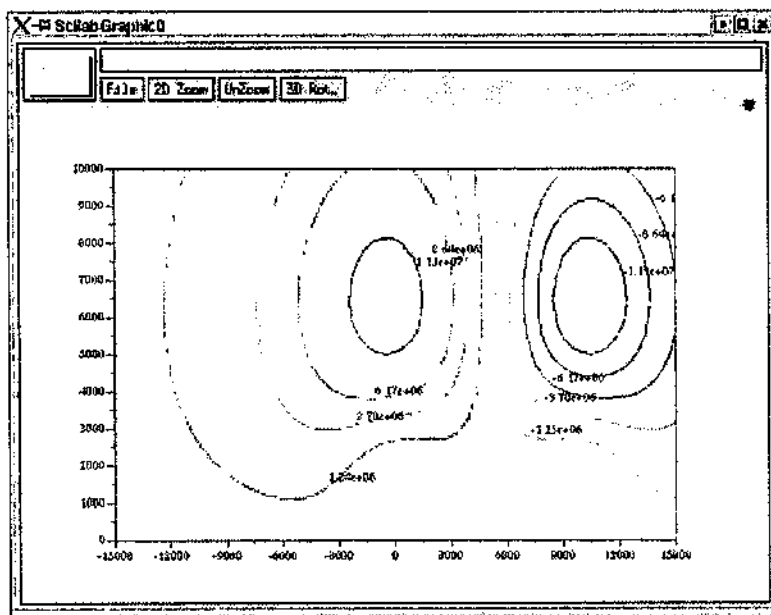


Figura 4.6: Visualização do cenário elétrico na atmosfera em yz utilizando o fcontour

Capítulo 5

MODELO 3D

De forma a aprimorar a análise do caminho das relâmpagos, a proposta é desenvolver o caminho do líder escalonado, que antes era bidimensional, como uma solução tridimensional (utilizando três eixos de coordenadas ortogonais).

5.1 Análise do Problema Físico para Obtenção de um Algoritmo

Os passos para a criação do caminho do relâmpago quanto à idéia do algoritmo são:

- Iniciar a descarga no centro de cargas, obedecendo condições de fronteiras e com o primeiro passo com direção normal à superfície;
- Gerar hemisfério base, que auxiliará no cálculo da direção de propagação;

Para isso define-se λ e θ , que são os ângulos da reta ligando um ponto do hemisfério ao centro do hemisfério em relação a vertical z e em relação a orientação da referência no plano x, y . É possível obter x, y, z dos pontos do hemisfério (V1), montando uma grade. Dessa forma, calcula-se $\Delta x, \Delta y$ e Δz utilizando λ e θ .

- Calcular um hemisfério associado ao deslocamento;

Quando se inicia o caminho do relâmpago, na ponta da descarga ajusta-se um hemisfério cuja direção não é necessariamente a mesma do hemisfério base; portanto será necessário encontrar Δx , Δy e Δz no novo hemisfério. Para isso, utiliza-se uma matriz de transformação, G , que gera o novo hemisfério, e $V1$, que são os pontos x , y , z do hemisfério base, caracterizando um processo de transformada de coordenadas (de um sistema ortogonal estático para um dinâmico).

- Encontrar x , y , z em todos os pontos da grade do novo hemisfério, para definir o potencial elétrico;

Assim que são encontrados Δx , Δy , Δz em todos os pontos do novo hemisfério, é necessário encontrar x , y , z desses pontos. Para tanto, é feita a soma do ponto x , y , z da ponta do canal com cada ponto Δx , Δy , Δz encontrado.

- Achar a diferença de potencial (ddp) entre cada ponto x , y , z e o centro do hemisfério dinâmico;

Obtendo os valores de x , y , z em cada ponto do hemisfério, é necessário calcular a ddp em cada um desses pontos. Isso é feito subtraindo-se o potencial de cada ponto do hemisfério do potencial do centro, ou seja, na ponta do canal. Onde houver a maior diferença de potencial (gradiente), será, em princípio, por onde a descarga elétrica seguirá.

- Encontrar λ e θ da reta que liga o centro do hemisfério à nova ponta do canal, obedecendo ao gradiente;

Dispondo das coordenadas x , y , z de dois pontos no sistema ortogonal escolhido, pode-se encontrar λ e θ da direção de deslocamento.

- Dar um passo ΔL (arbitrado por situação física);

Seguir com um passo na mesma direção da reta feita na direção do gradiente e repetir o processo, sob controle de continuidade devido a parâmetros físicos.

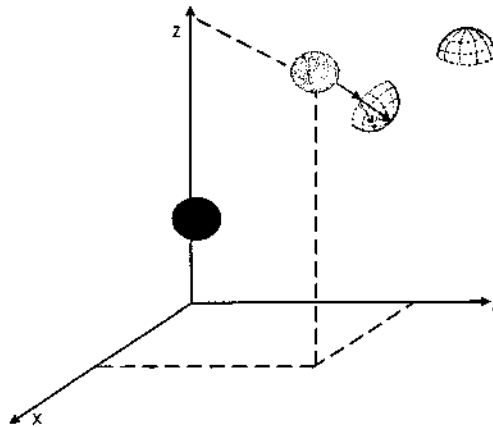


Figura 5.1: Modelo 3D

5.2 Idéias para o Desenvolvimento da Programação Orientada a Objeto

No Main serão lidos a opção de simulação e o nome do arquivo que possui os dados da descarga elétrica; serão feitos os testes para verificar se a opção de simulação é válida e se o nome do arquivo a ser lido é válido. Será ainda criado um objeto do tipo TCharge.

Será chamada, em seguida, a função Read da classe TCharge. Ela receberá como parâmetros: namein, parameter, qc, rrc, rc e zc. Nela será feita a leitura do arquivo que possui as informações necessárias, e os valores lidos serão atribuídos a variáveis-membro da classe TCharge.

Na classe TCharge, devem existir funções para desenhar a carga (Draw), calcular a trajetória (Traject) e desenhar o caminho (CalcPath).

Na função TCharge.Draw(namein, parameter, rrc, rc, zc) são feitos cálculos para gerar as coordenadas do centro da carga, a partir da qual se inicia o relâmpago; nela é gerado o arquivo .CIR, com essas coordenadas do centro da carga.

A função TCharge.Traject(model, namein, parameter, qc, rrc, rc, zc), cria o arquivo .DAT, que contém informações de onde se inicia e termina a descarga, dos valores para que cada passo da descarga seja dado. Nessa função é feito o controle do início e o fim da descarga; e nela é chamada a função Potential, da própria classe TCharge.

A função `TCharge.Potential(model, r, z, rc, zc, qq, kcoefic)` é responsável pelo cálculo do potencial. E a `TCharge.CalcPath(model, ang0, parameter, qc, rrc, rc, zc)` é responsável por encontrar o gradiente (caminho pelo qual a descarga seguirá) e enviá-lo para a função `CalcPath`, para que nela seja criado a caminho da descarga.

A classe `TtreeBin` é utilizada para auxiliar o armazenamento dos valores associados à evolução do caminho e obtidos pela classe `TCharge.CalcPath`.

5.3 Diagrama de Casos de Uso e Diagrama de Classes

Nas figuras 5.2 e 5.3 é possível visualizar as idéias para o desenvolvimento da programação orientada a objeto.

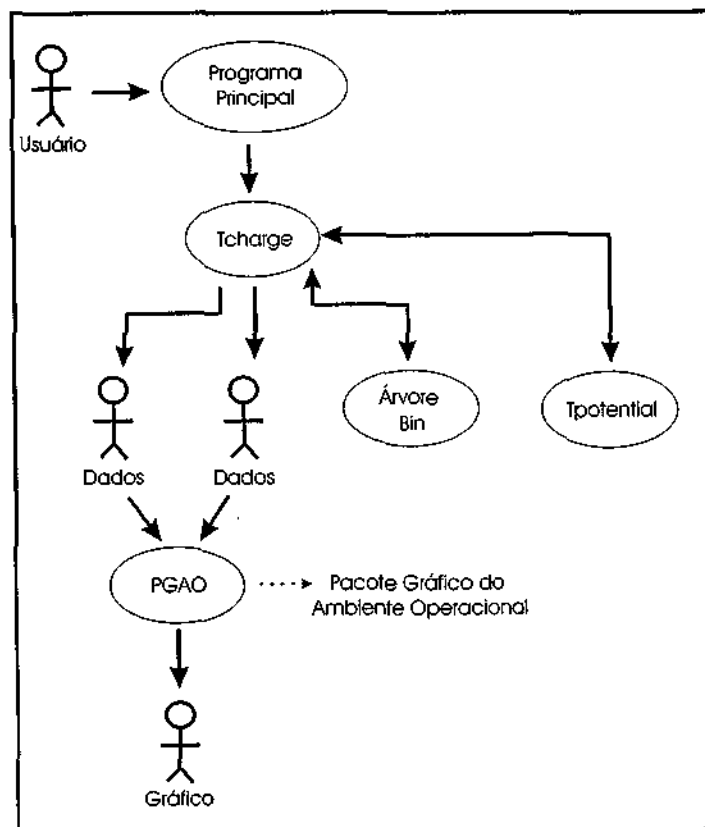


Figura 5.2: Diagrama Casos de Uso

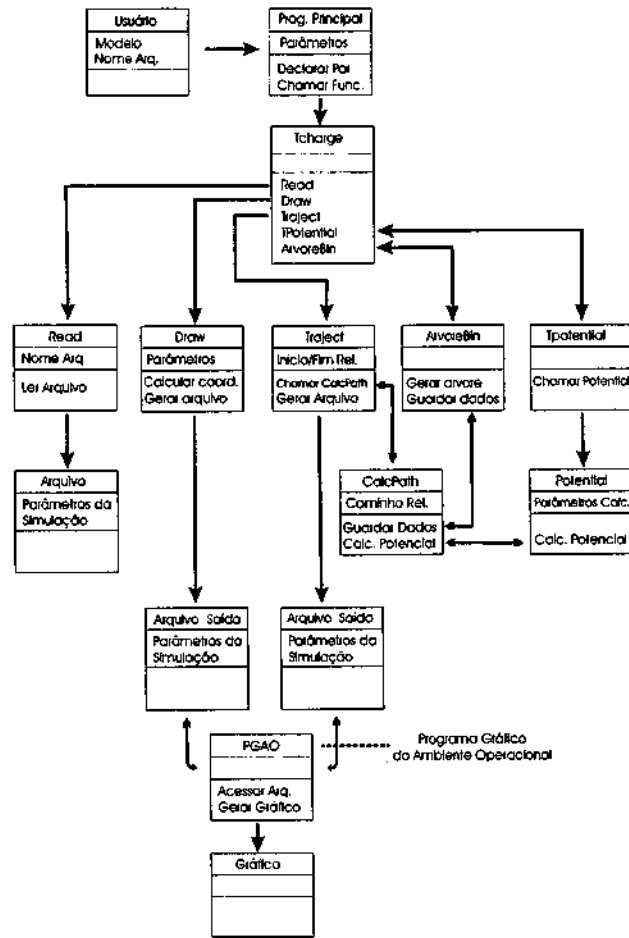


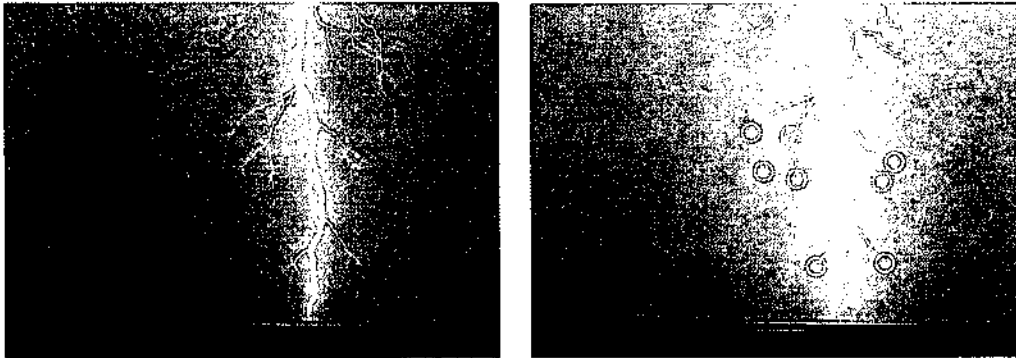
Figura 5.3: Diagrama de Classes

5.4 Código para Implementação de Árvores Binárias

Em decorrência do comportamento das descargas elétricas, particularmente da primeira descarga que é ramificada, observa-se que as ramificações seguem um padrão de bifurcação. Essa é a motivação de essencialmente implementar uma árvore binária.

Na imagem abaixo é possível visualizar as semelhanças entre um relâmpago e uma árvore binária.

Para implementação de árvores binárias dentro do programa já existente, criou-se o código abaixo, que gera árvores binárias. Atualmente é um programa teste, serão necessárias algumas



modificações neste código para que possa ser incorporado no código do programa já existente.

Programa - sugestão de manuseio de árvores binárias

```
#include <iostream>
using std::cout;
using std::cin;
using std::endl;

#include<iomanip>
using std::setw;

struct Noh{
    char valor; //string valor de cada noh
    int nivel; //nivel do noh
    Noh* pFE; //ponteiro do filho esquerdo
    Noh* pFD; //ponteiro do filho direito
};

class ArvoreBin{

public:
    ArvoreBin();
```

```
    ~ArvoreBin();
void Destruir(Noh*);
void Criar(Noh*);
void Construir(Noh*);
bool temRaiz();
    void ImprimeNoh(Noh*);
    void Exibir(int);
void PreOrdem(Noh*);
void Simetrica(Noh*);
void PosOrdem(Noh*);

private:
Noh *ptRaiz;
bool raiz;
};

ArvoreBin::ArvoreBin(){ //construtor
ptRaiz=NULL;
raiz=false;
}

ArvoreBin::~ArvoreBin(){ //destrutor
if(ptRaiz)
Destruir(ptRaiz);
}

bool ArvoreBin::temRaiz(){ //função utilitária, diz se a árvore possui
//uma raiz ou não

return raiz;
}
```

```
void ArvoreBin::Criar(Noh *pt){ //Insere a raiz da árvore e começa o
                                //processo de construção

    ptRaiz=pt;
    raiz=true;
    Construir(pt);
}

void ArvoreBin::Destruir(Noh *pt){ //Destrói a árvore, quando chamada
                                    //pelo destrutor

    if(pt->pFE)
        Destruir(pt->pFE);
    if(pt->pFD)
        Destruir(pt->pFD);
    delete pt;
}

void ArvoreBin::Construir(Noh *pt){ //Constrói recursivamente a árvore
    char resposta;
    cout<<"O no de valor '" <<pt->valor<<"' tem filho esquerdo? (s/n)\n-> ";
    cin>>resposta;
    if((resposta=='s')||(resposta=='S')){
        Noh* ptnovo = new Noh();
        ptnovo->pFE = NULL;
        ptnovo->pFD = NULL;
        cout<<"Digite o valor do filho esquerdo:\n-> ";
        cin>>ptnovo->valor;
        ptnovo->nivel=(pt->nivel)+1;
        pt->pFE = ptnovo;
        Construir(ptnovo); // Recursão
    }
}
```

```

}
cout<<"O no de valor '" <<pt->valor<<"' tem filho direito? (s/n)\n-> ";
cin>>resposta;
if((resposta=='s')||(resposta=='S')){
Noh* ptnovo = new Noh();
ptnovo->pFE = NULL;
ptnovo->pFD = NULL;
cout<<"Digite o valor do filho direito:\n-> ";
cin>>ptnovo->valor;
ptnovo->nivel=(pt->nivel)+1;
pt->pFD = ptnovo;
Construir(ptnovo);          //Recursão
}
}

void ArvoreBin::Exibir(int opcao){ //Recebe a escolha um dos tipos
                                //de percurso e o chama

if(opcao==1)
PreOrdem(ptRaiz);
else if(opcao==2)
Simetrica(ptRaiz);
else if(opcao==3)
PosOrdem(ptRaiz);
else
cout<<"\nErro interno: Opcao de Impressao!\n"<<endl;
}

void ArvoreBin::ImprimeNoh(Noh *pt){ //Imprime cada noh, de acordo com
                                //seu nivel

for(int i = 0; i<pt->nivel; i++){

```

```
cout<<" ";
}
cout<<pt->valor<<"-----\n";
}

void ArvoreBin::PreOrdem(Noh *pt){ //Percurso em pre-ordem
  ImprimeNoh(pt);
  if(pt->pFE)
    PreOrdem(pt->pFE);
  if(pt->pFD)
    PreOrdem(pt->pFD);
}

void ArvoreBin::Simetrica(Noh *pt){ //Percurso em ordem simetrica
  if(pt->pFE)
    Simetrica(pt->pFE);
  ImprimeNoh(pt);
  if(pt->pFD)
    Simetrica(pt->pFD);
}

void ArvoreBin::PosOrdem(Noh *pt){ //Percurso em pos-ordem
  if(pt->pFE)
    PosOrdem(pt->pFE);
  if(pt->pFD)
    PosOrdem(pt->pFD);
  ImprimeNoh(pt);
}

/*
PROGRAMA PARA TESTE DA ARVORE BINARIA
```

```
*/
void main(){

cout<<"** ARVORE BINARIA - Criação e percurso por Pre-Ordem";
cout<<"Simetria e Pos-Ordem **\n\n";
ArvoreBin arvore;
Noh* ptr = new Noh();
int entrada;
do{
cout<<"\n"<<setw(35)<<"- Menu -\n\n";
cout<<"1) Criar Arvore\n";
cout<<"2) Percorrer por Pre-Ordem\n";
cout<<"3) Percorrer por Simetrica\n";
cout<<"4) Percorrer por Pos-Ordem\n";
cout<<"5) Sair\n-> ";
cin>>entrada;

switch(entrada){

case 1: //criar arvore
cout<<"\nDigite o valor da raiz da arvore(letra ou numero):\n-> ";
cin>>ptr->valor;
ptr->nivel = 0;
ptr->pFE = NULL;
ptr->pFD = NULL;
arvore.Criar(ptr);
break;

case 2: //pre ordem
if(arvore.temRaiz())
```

```
arvore.Exibir(1);
break;

case 3:    //simetrica
if(arvore.temRaiz())
arvore.Exibir(2);
break;

case 4:    //pos ordem
if(arvore.temRaiz())
arvore.Exibir(3);
break;

case 5:
cout<<"\n\nEste Programa Terminou!\n"<<endl;
break;

default:
cout<<"\nEntrada invalida!\n";
break;
}
}while(entrada!=5);
}
```

Capítulo 6

Conclusão

Na primeira etapa do trabalho foi feita uma revisão bibliográfica envolvendo embasamentos em matemática, física e informática, para trabalhar com a questão do caminho do relâmpago na atmosfera. Iniciou-se um estudo mais aprofundado sobre o Scilab e a linguagem de programação C++, que permite que sejam feitos programas em uma plataforma orientada a objeto. Verificou-se também a possibilidade de uma integração entre esses dois programas, para uma maior qualidade e rapidez da simulação.

Por meio de gráficos gerados no Scilab, sendo utilizadas funções e linhas de comando, foi possível visualizar e analisar o cenário elétrico na atmosfera em determinados pontos. Como resultados preliminares, mapeou-se o potencial elétrico na atmosfera, constatando a influência que exercem os parâmetros atmosféricos, como, por exemplo, a condutividade elétrica, a configuração das cargas elétricas e o modelo físico adotado.

Na continuidade do trabalho, foi esboçado o desenvolvimento do caminho do líder escalonado, que antes era bidimensional, como uma solução tridimensional (utilizando três eixos de coordenadas ortogonais). Foi feita também uma análise do problema físico para obtenção de um algoritmo que possui os passos para a criação do caminho do relâmpago. Além disso, foram sugeridas idéias para o desenvolvimento da Programação Orientada a Objeto, com os Diagramas de Casos de Uso e de Classes, incorporando o conceito de árvores binárias.

Referências Bibliográficas

- Alonso, M. F.; Edward, J. F. **Física: Um Curso Universitário**. São Paulo: Edgard Blücher Ltda, 1967. v. 2.
- Bent, R.; Lyons, W. A. A theoretical evaluations and initial operational experiences of LPATS (Lightning Position and Tracking System) to monitor lightning ground strikes using a Time-of-Arrival TOA technique. In: American Meteorological Society. **VII International Conference on Atmospheric Electricity**. Albany: American Meteorological Society, 1984.
- Bordinhão, J. Raios. Internet, Jan. 2002. [Http://www.zemoleza.com.br/trabalho.asp?cod=653](http://www.zemoleza.com.br/trabalho.asp?cod=653).
- Festa, M. Os relâmpagos. Internet, Jan. 2001. [Http://www.iagusp.usp.br/~edimilson/-relampago.htm](http://www.iagusp.usp.br/~edimilson/-relampago.htm).
- Gin, R. B. B. **Estudo das características dos relâmpagos nuvem-solo em Minas Gerais no verão de 1993**. São José dos Campos. 173 p. INPE-6393-TDI/609. Doutorado em Ciência Espacial / Geofísica Espacial – Instituto Nacional de Pesquisas Espaciais, Mar. 1997.
- Goody, R. M.; Walker, J. C. G. **Atmosferas planetárias**. São Paulo: Edgard Blücher, 1975. Cap. 3, p. 40–64.
- Iribarne, J. V.; Cho, H. R. **Atmospheric Physics**. Dordrecht: D. Reidel, 1980.
- Krehbiel, P. R.; Thomas, R. J.; Rison, W.; Hamlin, T.; Harlin, J. GPS – based mapping system reveals lightning inside storms. **EOS**, v. 81, p. 21–25, 2000.

- Lutgens, F. K.; Tarbuck, J. E. **The Atmosphere**. 2. ed. Englewood Cliffs, 1982.
- MacGorman, D. R.; Rust, W. D. **The Electrical Nature of Storms**. Oxford University, 1998. 422 p.
- Magno. Simulações. Internet, Jan. 2002. [Http://thunderonline.hpg.ig.com.br/-simulacoes.htm](http://thunderonline.hpg.ig.com.br/-simulacoes.htm).
- Mendes, O. J.; Domingues, M. O. Programa para análise de relâmpagos : ThoR_e - SLA . In: Microserv. **XI Congresso Brasileiro de Meteorologia**. Rio de Janeiro: Anais, 2000. p. 2077-2081. CD ROM.
- Mendes, O. J.; Domingues, M. O.; Gin, R. B. B.; Pinto, O. J.; Pinto, I. R. C. A. Climatological evaluation of parameters for a more realistic lightning modeling proceedings. In: CBMET. **X Congresso Brasileiro de Meteorologia**. Brasília: CBMET, 1998. CD ROM.
- Mendes, O. J.; Pinto, O. J.; Pinto, I. R. C. A.; Chryssafidis, M. Basic elements and model comparisons in electrodynamic lightning trajectory. In: SBGF. **V International Congress of Geophysical Brazilian Society**. São Paulo: SBGF, 1997. v. 2, p. 1149-1151.
- Potts, S.; Monk, T. **Borland C++: 4 técnicas de programação**. Axcel Books, 1994. 710 p.
- Saba, M. M. F. **Estudo de condutividade, campo elétrico e relâmpagos associados a nuvens de tempestade**. São José dos Campos. 151 p. INPE-6675-TDI/625. Doutorado em Ciência Espacial / Geofísica Espacial - Instituto Nacional de Pesquisas Espaciais, 1997.
- Serway, R. A. **Eletricidade, magnetismo e Ótica**. 3. ed. LTC Livros Técnicos e Científicos, 1996. v. 3.
- Uman, M. A. **The Lightning Discharge**. Flórida: Academic Press, 1987. 377 p.
- Volland, H. **Atmospheric eletrodynamics**. New York: Springer Verlag, 1984.
- Williams, E. R. The eletrification of thunderstorms. **Scientific American**, v. 259, n. 5, p. 48-65, Nov. 1988.

Apêndice A

INTRODUÇÃO AO SCILAB

A.1 Interface

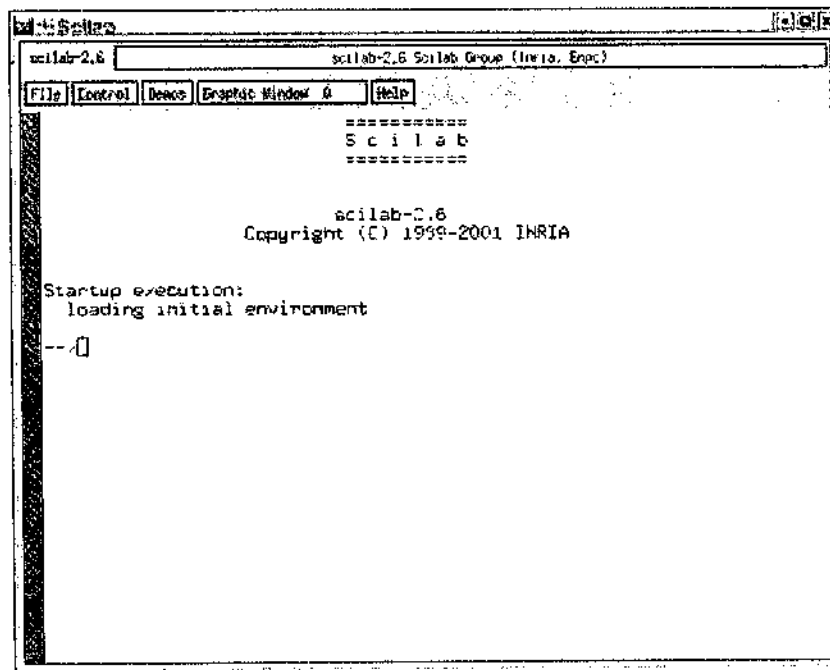


Figura A.1: Ambiente do Scilab

A.2 Características

- Criado pelo INRIA - França;
- Software gratuito;
- Foi desenvolvido para várias plataformas: Linux (a versão mais estável), Solaris, Macintosh, Windows 95/NT;
- Distribuído com o código fonte;
- Programa de manipulação de matrizes, funções matemáticas (faz operações básicas, complexas, trigonométricas, logarítmicas e plotagem) e programação;
- Permite fácil comunicação com programas em C e Fortran;
- Possui funções para converter programas em Matlab, C e Fortran para programas em SCILAB;
- Case Sensitive;

A.3 Operações básicas

A.3.1 Atribuição de valores

```
--> A=1 (ENTER)
A =
1.
```

Para que a resposta não seja mostrada na tela como acima, basta usar ";" no final da atribuição.

```
-->a = 3; b = 5; c = 8;
```

É possível fazer várias atribuições em uma única linha utilizando o ";".

```
--> a + c
    ans =
    11.
```

Pode-se fazer cálculos simples com as variáveis declaradas anteriormente.

A.3.2 Definindo polinômios

Os polinômios são criados através da função `poly`; podendo ser somados, subtraídos, multiplicados e divididos entre si.

```
-->S = POLY (0, 's')

// Foi criada uma variável S para ser utilizada como incógnita do polinômio.
```

```
-->Q = 2 - 3 * S + S ^ 2
```

Criou-se um polinômio do segundo grau (acima) e calculou-se a raiz (abaixo).

```
--> ROOTS (Q)
    ans =
    ! 1 !
    ! 2 !
```

Pode ser feita qualquer operação matemática com polinômios

A.4 Vetores

A.4.1 Declaração de vetores

```
--> Primos = [ 2 3 5 7 11 13 ]
    Primos =
    !2 3 5 7 11 13!
```

```
--> Pares = 2 : 2 : 17
      Pares =
      !2 4 6 8 10 12 14 16!
//Trabalhando desta forma definimos automaticamente um vetor.
```

No caso acima o vetor terá início em 2, irá acrescentar de 2 em 2 elementos e terá fim em 17.

A.4.2 Operações com vetores

```
--> Natural = [ 1 2 3 4 5 6 ];

--> Primos + Natural           //Exemplo 1
      ans =
      3 5 8 11 16 19

--> SUM(Natural)              //Exemplo 2
      ans=
      21
```

Pode-se somar dois vetores, como no primeiro exemplo, ou apenas somar os elementos de um vetor, como no segundo exemplo.

```
--> Natural ^ 2
```

No exemplo acima o vetor está sendo elevado a "2".

```
--> G = 5 .* Natural
      G =
      5 10 15 20 25 30
```

Utiliza-se o "." para multiplicar elemento por elemento do vetor. Pode-se usar este recurso para multiplicar tanto um número por um vetor como um vetor por outro.

Obs: É possível fazer qualquer operação matemática com vetores, mas para multiplicar dois vetores é necessário que o número de colunas do primeiro vetor seja igual ao número de linhas do segundo.

A.5 Operadores:

É possível utilizar os operadores relacionais e lógicos, apresentados nas tabelas A.1 e A.2, para fazer cálculos e verificações:

```
--> A = 1 : 9
      ! 1 2 3 4 5 6 7 8 9 !
--> B = 9 - A
      ! 8 7 6 5 4 3 2 1 !
--> VF = ( A > 4 )
      ! F F F F T T T T !
```

Foram atribuídos valores para os vetores A e B e em seguida foi verificado se cada elemento do vetor A é maior que "4"; o SCILAB retorna então valores F (falso), que é igual a "0", para os elementos que são menores que "4" e T (verdadeiro), que é igual a "1", para os elementos que são maiores que "4".

Tabela A.1: Operadores relacionais

Operador	Significado
<	menor que
<=	menor ou igual
>	maior que
>=	maior ou igual
==	igual a
=	diferente

Tabela A.2: Operadores lógicos

Operador	Significado
&	E
	OU
~	NÃO

A.6 Matrizes

```
--> A = [ 1 2 3; 4 5 6; 7 8 9 ]
```

```
--> B = [ 9 8 7  
        6 5 4  
        3 2 1]
```

// Essas são as duas formas possíveis de declaração de uma matriz.

```
--> DET ( A )
```

// Calcula a determinante da matriz A

```
--> INV ( A )
```

// Calcula a inversa da matriz A

```
--> C = [ 'x', 'y'; 'z', 'w+v' ]
```

É possível construir matrizes simbólicas e posteriormente trabalhar com elas como se fossem matrizes numéricas.

A.7 Gráficos

A.7.1 Plotando gráficos

```
--> x = 1:0.5:100;      // Define-se o eixo X através de um vetor  
--> F = 2 * x .* x;     // Função a ser plotada  
--> PLOT (x , F)       // Será plotado o gráfico utilizando os valores de
```

X e F.

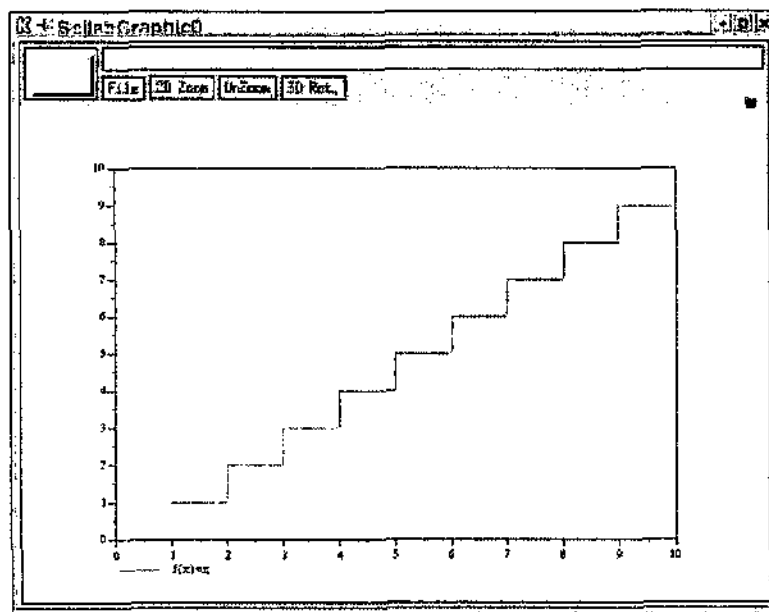


Figura A.2: Exemplo de gráfico plotado

- PLOT2d1, PLOT2d2, PLOT2d3 à permitem a visualização de duas ou mais curvas no mesmo gráfico, oferecendo também várias opções e formas de visualização dos gráficos como pode ser visto nas figuras ??curva e ??curvas.
- Com o PLOT2d3 é possível fazer gráficos de barras

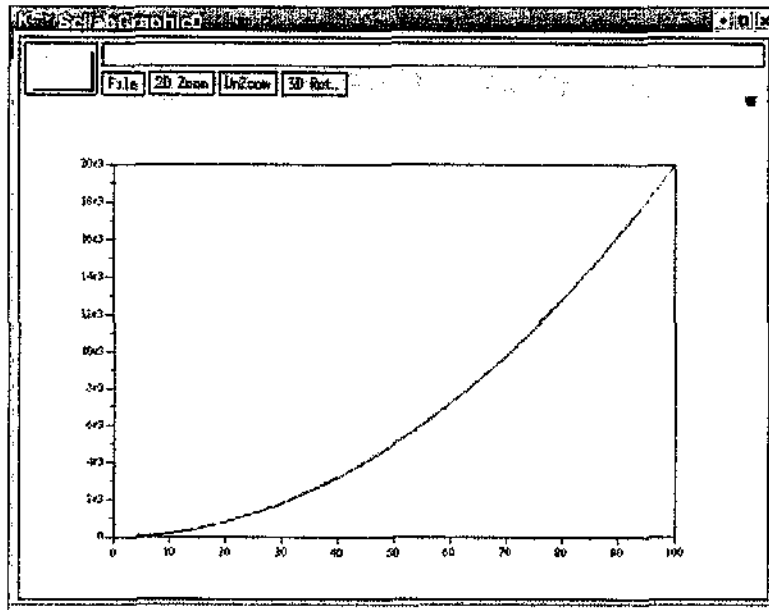


Figura A.3: Curva plotada no Scilab

A.8 Recursos de Programação

A.8.1 Criando funções:

- Deve-se utilizar um arquivo de texto, utilizando qualquer editor de texto, para escrever funções contendo uma sequência de comandos;
- Estão presentes no SCILAB as estruturas de controles clássicas em programação (estruturas de decisão e repetição);
- Na primeira linha do arquivo texto deve haver a definição dos valores de retorno, nome e parâmetros da função.

FUNCTION(valores de retorno) = Nome(valores de entrada)

Exemplo de Função:

Cálculo de fatorial

```
FUNCTION [Y] = FATORIAL(x)
```

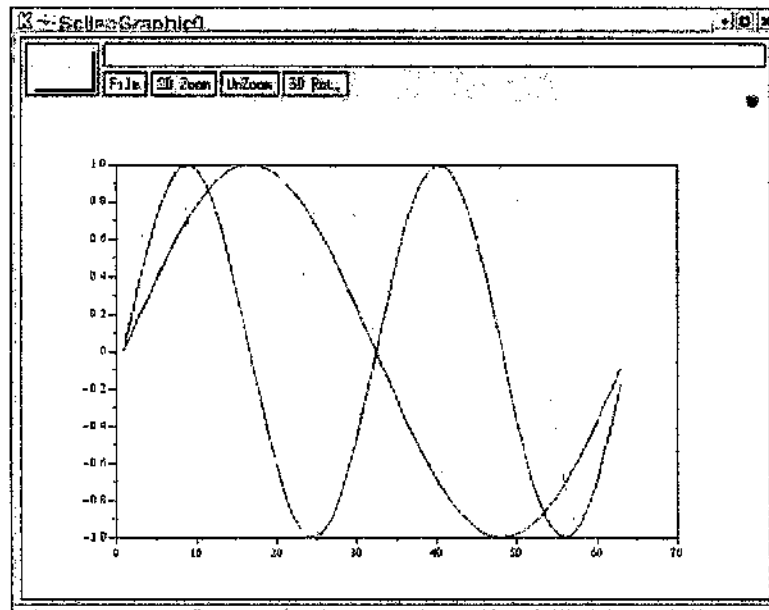


Figura A.4: Múltiplas curvas plotadas ao mesmo tempo

```

IF x < 1 THEN
    x = 1;
END
Y = 1;
FOR i = 1 : x
    Y = Y * i;
END

```

Função que verifica se o número digitado é positivo, se não for retorna-se o mesmo valor para a função; do contrário o cálculo é efetuado através de um *loop* que inicia-se em 1 e chega ao valor do argumento de entrada.

A.8.2 Comandos para Iterações:

Dois comandos permitem a realização de iterações, *loops*: o *loop* implementado com o comando **for** e o *loop* implementado com o comando **while**.

A primeira forma é utilizada quando os programas são escritos em um arquivo, enquanto a Segunda é utilizada quando programa-se em ambiente Scilab.

O *loop* baseado no **for** executa instruções uma quantidade de vezes determinada; enquanto o *loop* baseado no **while** realiza uma seqüência de intruções enquanto uma determinada condição estiver sendo satisfeita.

A.8.3 Comandos Condicionais:

O Scilab implementa dois tipos de comandos condicionais: **if-then-else** e **select-case**.

A primeira forma é utilizada quando os programas são escritos em um arquivo, enquanto a Segunda é utilizada quando programa-se em ambiente Scilab.

O **if-then-else** avalia uma expressão. Se esta expressão for verdadeira, executa as instruções subseqüentes; se for falsa, executa as instruções após o **else**.

O **select-case** compara o valor de uma variável de teste com as várias expressões dos **case**.

A.8.4 Ligação do Scilab com Programas em C:

O Scilab permite que rotinas ou funções escritos em FORTRAN, Maple e C sejam utilizados dentro de seu ambiente.

Uma função escrita na linguagem C pode ser ligada ao Scilab de três formas:

- a) Através do comando *link* , em um processo chamado de ligação dinâmica;
- b) Através de gateways, programas de interface escritos pelo usuário;
- c) Através da adição de uma nova função ao código do Scilab.

a) Ligação Dinâmica

```
link('foo.o', 'foo', 'c')
```

Liga o arquivo objeto escrito na linguagem C, indicado elo argumento *c*, ao Scilab. O segundo argumento é o nome da função executada.

SIMULAÇÃO NUMÉRICA DE RELÂMPAGOS: IMPLEMENTAÇÕES

Denise Fernandes da Silva¹⁶ (UNITAU, Bolsista PIBIC/CNPq)
Dr. Odim Mendes Junior¹⁷ (CEA/INPE)
Margarete Oliveira Domingues¹⁸ (CPTEC/INPE)

RESUMO

O relâmpago, embora um fenômeno muito antigo no conhecimento da humanidade, apresenta inúmeros aspectos básicos a serem melhor conhecidos, como o instante exato da sua ocorrência e o caminho que percorrerá. Este trabalho objetiva contribuir para o estudo da eletrodinâmica das descargas elétricas atmosféricas, por meio da simulação numérica do caminho do líder escalonado, que é a etapa precursora na geração desse caminho. Esse caminho fica visualmente bem identificado quando, por exemplo, em uma descarga descendente que atinge o solo, a descarga de retorno, que vem depois da conclusão do caminho, ocorre produzindo luminosidade intensa ao longo da trajetória seguida. No intuito de aprimorar o código de simulação do caminho do relâmpago, no primeiro ano de trabalho, houve o embasamento na física do relâmpago; o manuseio do código de computador já existente (desenvolvido pelo orientador); a pesquisa de uma ferramenta de programação e visualização integradas (o programa Scilab); e o ensaio de resultados envolvendo o entendimento de efeitos no campo elétrico atmosférico, por meio do controle de alguns parâmetros atmosféricos (como configuração de cargas, condutividade elétrica atmosférica e condições de fronteira). A metodologia de trabalho consistiu da programação neste ambiente integrado, que caracteriza-se por uma ferramenta robusta e de livre utilização. Como resultados, foram gerados no Scilab alguns gráficos para a visualização no ambiente elétrico onde ocorre a formação de relâmpagos. Da análise, conclui-se que a configuração do campo elétrico na atmosfera/estratosfera mostra-se afetada pelos pressupostos físicos do modelo adotado. No segundo ano, a intenção foi continuar o aprimoramento do código de simulação e fazer o desenvolvimento em linguagem C++ sob o paradigma POO. Devido à característica de bifurcação do relâmpago, pretendeu-se introduzir o conceito de árvore binária. A implementação está no estágio de análise e proposição.

¹⁶ Aluna do Curso de Computação Científica, UNITAU. E-mail: defersi@hotmail.com

¹⁷ Pesquisador da Coordenação Geral de Ciências Espaciais e Atmosféricas. E-mail: odim@dge.inpe.br

¹⁸ Pesquisadora do Centro de Previsão de Tempo e Clima. E-mail: margaret@cptec.inpe.br