MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

INPE-1676-TDI/019

APROXIMAÇÕES SUB-ÓTIMAS PARA O CONTROLE EM PROGRAMAS DINÂMICOS DE OTIMIZAÇÃO

Décio Castilho Ceballos

Dissertação de Mestrado do Curso da Pós-Graduação em Ciência Espacial, orientada pelo Dr. Atair Rios Neto, aprovada em 08 de outubro de 1979.

INPE
São José dos Campos
1980

521.6:519.711.3

CEBALLOS, D. C.
Aproximações sub-ótimas para o controle em programas dinâmicos de otimização / D. C. Ceballos. São José dos Campos: INPE, 1980.
198p. – (INPE-1676-TDI-019).

1.Engenharia e tecnologia espacial. 2.Controle sub-ótimos. 3.Otimização de trajetórias. 4.Otimização de parâmetros. 5.Método do gradiente. I. Título.

Aprovada pela Banca Examinadora
em cumprimento dos requisitos exigidos
para a obtenção do título de Mestre em
Ciência Espacial

Dr. Santiago Alves Tavares — Presidente da Banca

Dr. Atair Rios Neto — Orientador

Dr. Octávio Maizza Neto — Co-Orientador

Dr. Jerzy Tadeusz Sielawa — Membro da Banca -convidado-

Dr. Nellore S. Venkataraman — Membro da Banca

Décio Castilho Ceballos — Candidato

São José dos Campos, 08 de outubro de 1979

A Naide e Diego

AGRADECIMENTOS

Ao Instituto de Pesquisas Espaciais, pelas facilidades concedidas à realização deste trabalho.

Ao Dr. Atair Rios Neto, pelo encorajamento em abordar o tema e pela orientação deste trabalho.

À Naide, minha esposa, pela colaboração e incentivo.

Aos professores Dr. Giorgio E. O. Giacaglia e Dr. Otávio Maizza Neto, e a todas as pessoas que direta ou indiretamente colaboraram para a realização deste trabalho.

ABSTRACT

This work aims a method for numerical solution of suboptimal control problems, in which the control is taken as a functional dependent on time and a finite number of parameters. In order to test the method, a computer program is developed with the capacity to solve a highly important group of optimization problems, requiring just two subprograms to supplied. One of them embodies the dynamical equations of motion and the other is the partial differential of the constraints and the performance index equations with respect to final time and state vector. Applying the procedure to an Earth-Mars or orbit transfer problem, tests are made for various approximations of the control. For the previous problem, a comparison is presented, relating the results obtained to the numerical optimal solution and to results from suboptimal procedures found in the literature.

ÍNDICE

LISTA DE SÍMBOLOS

A - Matriz de coeficientes.

A_{ij} - Coeficientes da matriz A no problema de programação linear.

a - Vetor de parâmetros otimizáveis.

B - Matriz de coeficientes.

c - Vetor de parâmetros otimizáveis inerentes ao sistema dinâmico.

c - Vetor dos fatores de custo.

G - Índice de desempenho.

g - Número de parâmetros otimizáveis.

IP - Índice de desempenho.

M - Vetor das equações de restrições.

m - Número de equações de restrições.

m_o - Massa inicial.

$\dot{m}$ - Fluxo de massa.

n - Número de variáveis de estado.

n - Número de variáveis no problema de programação linear.

p - Número de variáveis de controle.

r - Posição radial.

r - Taxa de redução dos incrementos.

r_1, r_2, r_{12} - Variáveis para avaliação do tempo de processamento.

$\hat{s}$ - Variáveis do problema de programação linear.

SV_i - Somatória dos desvios nos vínculos da iteração i.

$SV_i^!$ - Valor limite admissível para SV_i.

T - Empuxo na direção do jato do foguete.

t - Variável tempo.

t_i - Tempo nodal.

t_ω - Tempo normalizado.

u - Vetor de controle.

v_r - Velocidade radial.

v_t - Velocidade tangencial.

x - Vetor de variáveis de estado.

x - Vetor de variáveis do problema de programação linear.

x_{f_a} - Matriz de derivadas parciais das variáveis de estado em relação aos parâmetros a serem otimizados.

w - Índice de desempenho no problema auxiliar.

z - Variável auxiliar nos problemas de programação linear.

z - Índice de desempenho no problema de programação linear.

α - Parâmetro de convergência nas equações de restrições.

β - Parâmetro de convergência no sentido de minimizar o índice de desempenho.

ξ_p - Parâmetro que define a fronteira de soluções próximas.

ξ_b - Incremento mínimo no índice de desempenho e nas restrições.

ω - Parâmetro de peso.

μ - Constante gravitacional.

η_v - Eficiência limite.

$(\ )_o$ - O subscrito indica condição inicial.

$(\ )_f$ - O subscrito indica condição final.

$(\ )_{t_f}$ - O subscrito indica derivada parcial em relação ao tempo final.

$(\ )_{x_f}$ - O subscrito indica derivada parcial em relação às variáveis de estado finais.

$|\ |$ - Valor em módulo.

LISTA DE FIGURAS

LISTA DE TABELAS

CAPÍTULO I

INTRODUÇÃO

A otimização de sistemas dinâmicos, em particular a otimização da trajetória de espaçonaves, tem tido grande interesse desde 1919, quando a agência "Goddard's" procurava maximizar a altura alcançada pelos foguetes de sondagem (Tapley e Lewallen, 1967). Desde essa época, significativo progresso ocorreu nos métodos de resolução de problemas complexos, principalmente após o advento de grandes computadores.

A resolução do problema de controle ótimo consiste, basicamente, na determinação do vetor de controle que minimize um índice de desempenho e leve o sistema dinâmico a obedecer a vínculos inerentes ao problema. Entretanto, a resolução do problema de controle ótimo na forma irrestrita, muitas vezes redundam em complexidades numéricas, que reduzem a possibilidade de aplicação de procedimentos ótimos.

O propósito deste trabalho é desenvolver um procedimento para resolver o problema restrito, no qual a forma de controle é postulada.

Adotado um funcional conveniente para o controle, é de se esperar que, em troca da simplificação feita, exista um procedimento para resolver o novo problema, que apresente vantagens em relação aos procedimentos ótimos, nas características associadas à complexidade numérica do problema original (características de simplicidade de formulação e implementação, memória requerida de computador, tempo de convergência, etc.).

O procedimento apresentado neste trabalho é um procedimento direto, e como tal, em cada iteração típica, faz-se necessário determinar o incremento no vetor de parâmetros a ser otimizado, a partir do objetivo de, caminhar na direção de satisfação dos vínculos e de extremizar o índice de desempenho.

Quando o número de parâmetros a serem otimizados for maior do que o número mínimo necessário para que as equações de restrições possam ser obedecidas, existem infinitos vetores de incrementos que satisfazem o objetivo acima. Esta indeterminação pode ser eliminada, escolhendo se, entre os infinitos vetores de incrementos, por exemplo, aquele de norma mínima e isto leva a um novo problema de otimização associada a cada iteração típica.

O procedimento apresentado neste trabalho utiliza-se de uma norma linear e de uma expansão linear, para a imposição de caminhada na direção de satisfação dos vínculos e de extremizar o índice de desempenho. Portanto, o problema de otimização associado a uma iteração típica pode ser resolvido por programação linear, o que permite ao algoritmo global ótimas características de formulação e implementação.

Elaborou-se um programa para computador, com capacidade para resolver uma classe bastante geral de problemas dinâmicos de otimização. Foram feitos testes para o problema de transferência de órbita Terra-Marte, que é clássico em trabalhos de comparações (Tapley e Lewallen,1967) e desenvolvimentos de procedimentos (Willianson, 1971; Hull e Edgeman, 1975; Rios Neto e Ceballos, 1979).

A escolha adequada do funcional ou vetor de funcionais, para o controle, é de grande importância para o sucesso na aplicação do procedimento proposto. Testes foram feitos para várias aproximações do controle no problema Terra-Marte, com apresentação de resultados e comentários sobre as características de cada aproximação.

No Capítulo II, apresentam-se os fundamentos de controle ótimo, uma visão panorâmica dos procedimentos ótimos mais importantes, dois outros procedimentos semelhantes ao deste trabalho (Willianson, 1971; Hull e Edgeman, 1975), e são delineadas as bases para desenvolvimento do procedimento apresentado.

No Capítulo III, desenvolve-se o método analítico.

No Capítulo IV, desenvolve-se o procedimento numérico.

No Capítulo V, faz-se o teste do procedimento para o problema de transferência de órbita Terra-Marte.

No Capítulo VI, faz-se considerações de ordem qualitativa, visando classificar o procedimento proposto entre os procedimentos mais importantes existentes.

No Capítulo VII, apresentam-se as conclusões e sugestões para desenvolvimentos futuros.

CAPÍTULO II

FUNDAMENTOS

2.1 - INTRODUÇÃO

A menos que as equações dinâmicas, o índice de desempenho e as equações de restrições sejam bastante simples, faz-se necessária a resolução de problemas dinâmicos de otimização por procedimentos numéricos. Mesmo em problemas relativamente simples, a grande quantidade de operações envolvidas torna necessária a utilização do computador e, devido a isso, somente após 1950 o desenvolvimento e a utilização de procedimentos, para determinação do controle ótimo, foram intensificados.

Neste capítulo são apresentados os elementos teóricos fundamentais para o desenvolvimento do procedimento proposto neste trabalho.

2.2 - PROBLEMA DE OTIMIZAÇÃO

Embora outras formulações sejam consideradas neste trabalho, o problema básico a ser tratado é encontrar o vetor de controle u, no intervalo $t_0 \leq t \leq t_f$, de modo que seja solução do problema a seguir.

Minimizar:

$$IP = IP(x_f, t_f)$$

sujeito à:

$$\dot{x} = f(x;u;t)$$

$$M(x_f, t_f) = \bar{0} \qquad \text{(II.1)}$$

dados,

$$x(t_0) = x_0$$

onde,

x é o vetor de estado nx1;

u é o vetor de controle px1;

M é o vetor de condições terminais mx1;

x_f é o vetor de estado final.

2.3 - VETOR DE CONTROLE ÓTIMO

O problema II.1 fica resolvido (Citron,1969) quando determinado o vetor de controle ótimo, que será designado por u*.

Se u* for da forma u*(t), ele é chamado de função ótima de controle. Por outro lado, se u* for da forma u*(x,t), ele é chamado lei ótima de controle.

Na ausência de erros no controle e de perturbações externas, tem-se a mesma trajetória nominal x*(t), utilizando-se a função ou lei ótima de controle.

A princípio, a lei ótima de controle seria indicada quanto às possibilidades de reação a um desvio da trajetória nominal, devido a perturbações externas ou erros no controle, mas, geralmente é inviável a utilização da lei ótima de controle, devido à grande dificuldade de obtenção, ou pela complexidade do "hardware", para isto requerido.

A função de controle, geralmente, utilizada, não é capaz de refazer distúrbios provocados por perturbações externas ou erros no controle. É possível contornar es se problema, nos casos em que o erro no controle e as perturbações externas forem pequenas, através de um controle em malha fechada, sobre o problema linearizado na trajetória prevista por u*(t), chamada trajetória nominal ou trajetória de referência.

2.4 - PROCEDIMENTOS NUMÉRICOS PARA O PROBLEMA ÓTIMO

A função ótima de controle, para problemas do tipo II.1, é normalmente obtida por métodos numéricos, que, de maneira geral, podem ser classificados em métodos diretos e métodos indiretos. Uma descrição mais detalhada é feita a seguir, baseada em Tapley e Lewallen (1967).

Os métodos indiretos usam as condições necessárias de otimização requeridas pela matemática; os métodos diretos usam somente as equações do movimento,orientados pelo objetivo de satisfazer os vínculos e reduzir o índice de desempenho.

Os métodos indiretos podem ser divididos em dois grupos. A diferença essencial entre esses dois grupos está na forma em que a trajetória base é gerada. Os *Métodos de Perturbação*, requerem que a trajetória base seja gerada pela integração das equações diferenciais não lineares do movimento, enquanto que os *Métodos de Quase-linearização* requerem a integração das equações diferenciais linearizadas do movimento.

O grupo dos métodos de quase-linearização inclue métodos propostos por vários autores e popularizados através de vários nomes: variação diferencial (Hestures), quase-linearização (Bellman e Kalaba), Newton-Raphson generalizado (Kenneth e M.C Gill), além de estudos de convergência e extensões realizados por diversos autores.

Os métodos de perturbação são divididos em duas classes: os métodos que usam as equações perturbadas (método da função de perturbação) e os métodos que usam equações adjuntas (método das equações adjuntas).

Um primeiro *procedimento direto* doi sugerido em 1960 por Kelley, denominado método de gradiente; um outro procedimento direto foi sugerido por Bryson e Kelley. Os métodos diretos possuem, em geral, dificuldade de convergência ao redor da solução ótima. Este problema foi atacado por Rosenbaum e Stancil.

Existem, ainda, métodos com características simultâneas de diretos e indiretos.

Enfim, existe uma grande quantidade de métodos. Assim, o método deve ser escolhido a partir de suas características de confiabilidade, simplicidade de formulação e implementação, capacidade requerida de computador, sensitividade de convergência e tempo de convergência, etc.

2.5 - FORMA ESPECIFICADA DO CONTROLE

A aplicação dos métodos para obtenção do controle ótimo, muitas vezes,é limitada pelas dificuldades decorrentes da complexidade numérica desses métodos.

Dependendo do problema e do método, essas dificuldades podem ser:

. realização da solução em tempo real;

. capacidade de computador disponível;

. implementação (trabalho de preparação do programa do computador);

. obtenção de uma boa solução de partida;

. custo de computação (tempo de convergência, capacidade de memória requerida);

. duplicação do resultado obtido (muitas vezes é preferível resultados na forma de um funcional.

Pelos motivos anteriores, pode ser necessária ou conveniente a utilização de métodos com maior simplicidade de realização, que apresentem resultados que deem um desempenho aproximado em relação ao obtido pelo controle ótimo do sistema.

O propósito deste trabalho é desenvolver um procedimento para resolver o problema restrito, restrição esta associada à postulação "a priori" da forma do controle. Adotado um funcional conveniente para o controle, é de se esperar que em troca da simplificação feita, exista um procedimento para resolver o novo problema, que apresente vantagens em relação aos procedimentos ótimos, nas características associadas à complexidade de procedimentos.

A escolha adequada da aproximação para o controle é bastante importante para o sucesso do procedimento e, dependendo da aplicação, uma ou outra formulação pode ser conveniente. Por exemplo, pode ser suposto como aproximação uma série polinomial, uma série de Tchebyshev, interpolações lineares em intervalos de tempo, etc.

A partir do problema II.1, após substituição de u por u(a,t) ou, em uma forma mais geral por u(a,x;t), tem-se o problema a seguir.

Minimizar:

$$IP = IP(x_f, t_f)$$

sujeito a:

$$\dot{x} = f(x,a,t)$$

$$M(x_f,t_f) \qquad \text{(II.2)}$$

dados:

$$x(t_o) = x_o$$

onde:

a é vetor de parâmetros a serem otimizados;

x, M, x_o, x_f são definidos no problema II.1.

Nos últimos anos, têm sido publicados procedimentos para resolver problemas do tipo II.2 , sendo que dois desses procedimentos, encontrados na literatura disponível, foram analisados ao longo deste trabalho. Um primeiro procedimento, apresentado por Willianson (1971), utiliza o problema, associado à iteração típica, a seguir.

Minimizar:

$$G = \omega \frac{\delta IP}{\delta a} \Delta a + \frac{1}{2} \Delta a^T \Delta a$$

sujeito a:

$$\Delta M = \frac{\delta M}{\delta a} \Delta a = \alpha M \qquad \text{(II.3)}$$

onde,

a é o vetor de parâmetros a serem otimizados (problema II.2);

Δa é o incremento do vetor a (incógnita do problema associada à iteração típica);

α é um parâmetro negativo ($|\alpha|<1$);

M é o vetor de restrições (problema II.1);

IP é o índice de desempenho (problema II.1);

$\frac{\delta}{\delta a}$ é o operador de derivada parcial, quando a função a ser derivada é considerada como função somente do vetor a;

ω é um parâmetro positivo.

A função G do problema II.3 pode ser interpretada facilmente. A função do parâmetro ω é ponderar o quanto se quer reduzir em IP, de acordo com o permitido pela hipótese de linearidade. Este controle sobre a manutenção da hipótese de linearidade pode ser inconveniente, dependendo da sensibilidade de IP em relação ao vetor a.

Uma análise crítica do problema II.3 sugere outras alternativas, para o problema associado a uma iteração típica. São apresentadas a seguir três destas alternativas, problemas II.4, II.5 e II.7.

Minimizar:

$$G = \omega_1 |\Delta a_1| + \omega_2 |\Delta a_2| + \dots + \omega_g |\Delta a_g|$$

sujeito a:

$$\Delta M = \frac{\delta M}{\delta a} \Delta a = \alpha M$$

$$IP = \frac{\delta IP}{\delta a} \Delta a = \beta(|IP| + 1) \qquad \text{(II.4)}$$

onde,

$\omega_1, \omega_2, \dots, \omega_g$ são parâmetros positivos;

α é um parâmetro negativo ($|\alpha| \leq 1$), pois está associado à necessidade de reduzir cada componente do vetor M, em uma iteração típica (item 3.4);

β é um parâmetro negativo (positivo), pois está associado à minimização (maximização) do índice de desempenho (item 3.4);

M é o vetor de restrições (problema II.1);

IP é o índice de desempenho (problema II.1);

a é o vetor de parâmetros a serem otimizados (problema II.2);

Δa é o incremento do vetor a, ou seja, incógnita do problema associado à iteração típica.

Minimizar:

$$G = \omega_1 |\Delta a_1| + \omega_2 |\Delta a_2| + \ldots + \omega_g |\Delta a_g| + \bar{\omega} \Delta IP$$

sujeito a:

$$\Delta M = \frac{\delta M}{\delta a} \Delta a = \alpha M$$

$$\Delta IP = \frac{\delta IP}{\delta a} \Delta a \geq \beta (|IP| + 1) \qquad \text{(II.5)}$$

onde,

$\bar{\omega}$ é um parâmetro positivo;

ω_1, α, β, etc. como definidos no problema II.4.

Eliminando-se o sinal de desigualdade que aparece em uma das equações do problema II.5, pela introdução de uma variável de folga (Apêndice A.1.3), tem-se o pro blema equivalente a seguir.

Minimizar:

$$G = \omega_1 |\Delta a_1| + \omega_2 |\Delta a_2| + \ldots + \omega_g |\Delta a_g| + \bar{\omega}\Delta IP$$

sujeito a:

$$\Delta M = \frac{\delta M}{\delta a} \Delta a = \alpha M$$

$$\frac{\delta IP}{\delta a} \Delta a - z = \beta(|IP| + 1),$$

$$z \geq 0 \qquad \text{(II.6)}$$

onde,

$\bar{\omega}$ é um parâmetro positivo;

ω_1, α, β, etc. como definidas no problema II.4;

z é uma variável de folga a ser determinada.

Minimizar:

$$G = \omega_1 |\Delta a_1| + \omega_2 |\Delta a_2| + \ldots + \omega_g |\Delta a_g| + \bar{\omega} z$$

sujeito a:

$$\Delta M = \frac{\delta M}{\delta a} \Delta a = \alpha M$$

$$\frac{\delta IP}{\delta a} \Delta a - z = \beta(|IP| + 1),$$

$$z \geq 0 \qquad \text{(II.7)}$$

onde,

ω_{2g+1}, ω_4, α, etc. estão definidos nos problemas II.4 e II.6;

z é uma variável que atua como a variável de folga do problema II.6 e, além disso, ela aparece como um componente a ser minimizado na função G.

As formas de problemas associadas a uma iteração típica, colocadas anteriormente, apresentam melhores características de controle ao incremento ΔIP, e melhores características no que se refere à simplicidade de formulação e implementação, pois, a escolha de uma norma linear para o vetor Δa, permite um problema associado na forma linear, que pode ser resolvido por programação linear.

Um segundo procedimento, devido a Hull e Edgeman (1975), propõe um esquema indireto para resolver o problema dinâmico de otimização com o controle na forma de um funcional, pois resolve o problema gerado pelas condições necessárias para a existência do extremo do vetor ótimo a. Este procedimento, apresenta os inconvenientes quanto ao tempo, por iteração, e quanto à formulação e implementação. No entanto, pode ser um procedimento bastante útil para fazer refinamento de solução obtida por método direto.

2.6 - OUTRAS APLICAÇÕES DO PROCEDIMENTO

Além de problemas do tipo II.1, outros podem chegar à forma do problema II.2 e, portanto, o procedimento desenvolvido neste trabalho, inclusive o programa desenvolvido para computador, é aplicável. Um primeiro tipo de problema é em sistemas onde existem parâmetros, otimizáveis, inerentes ao sistema, como no problema a seguir.

Minimizar:

$$IP = IP(x_f, t_f)$$

sujeito a:

$$\dot{x} = f(x,u,t,c)$$

$$M(x_f, t_f) = 0 \qquad (II.8)$$

dados:

$$x(t_o) = x_o$$

onde,

c é um vetor de parâmetros otimizáveis, inerentes ao sistema.

Dependendo do problema, alguns componentes de x_o também podem ser parâmetros otimizáveis.

Outra possibilidade é em sistemas de malha fechada, como o problema a seguir.

Minimizar:

$$IP = IP(x_f, t_f)$$

sujeito a:

$$\dot{x} = f(x,t,c)$$

$$M(x_f, t_f) = 0 \qquad (II.9)$$

onde,

c é um vetor de parâmetros otimizáveis, inerentes ao sistema.

2.7 - CONSIDERAÇÕES SOBRE A SOLUÇÃO u(a,t)

Em problemas de otimização (ótimos ou sub-ótimos), com qualquer método que se utilize, não é possível fugir do impasse quanto à identificação da solução obtida se é um mínimo local ou um mínimo global.

A aproximação do controle por um funcional pode alterar as características do problema original quanto aos seus mínimos, e pode ocorrer que:

. seja alterado o número de mínimos locais;

. a solução ótima, obtida no problema com o controle especificado, não corresponda a uma aproximação do controle ótimo, ou seja, não obedeça necessariamenmente, em aproximação, o princípio de Pontryagin para o problema original.

CAPÍTULO III

DESENVOLVIMENTO DO MÉTODO

3.1 - INTRODUÇÃO

A colocação abaixo, além de ser a forma mais comum de problemas dinâmicos, é a forma geral dos problemas de otimização em sistemas dinâmicos, com as condições iniciais dadas ou que possam ser colocadas como parâmetros otimizáveis. Entretanto, o método é totalmente geral e, com pequenas alterações na montagem do problema de otimização, associado a cada iteração típica, adapata-se a qualquer problema de otimização em sistemas dinâmicos.

3.2 - COLOCAÇÃO DO PROBLEMA

Considere-se os problemas II.1,II.8 e II.9. Substituindo-se a função de controle $u(a_{j+1}, a_{j+2}, \ldots, a_{g-1}, t)$ no problema original, tem-se o problema formulado a seguir.

Minimizar:

$$IP = IP(x_f, a_g)$$

sujeito a:

$$\dot{x} = f(x, a_{i+1}, a_{i+2}, \ldots, a_{g-1}, t),$$

$$x(t_0) = (a_1, a_2, \ldots, a_i, x_{0(i+1)}, x_{0(i+2)}, \ldots, x_{0(n)})$$

$$M(x_f, a_g) = 0 \tag{III.1}$$

onde,

IP é o índice de desempenho;

t_o tempo inicial (pode ser incluído como um parâmetro otimizável);

x é um vetor nx1 de variáveis de estado;

a é um vetor de g parâmetros otimizáveis, distribuídos em (i) parâmetros iniciais, (j-i) parâmetros inerentes ao sistema, (g-j-1) parâmetros ligados ao funcional, que represente o controle, e um parâmetro que corresponde ao tempo final (parâmetro a_g);

$x_{o(i+1)}, x_{o(i+2)}, \ldots, x_{o(n)}$ são dados;

$M(x_f, a_g)$ são as m equações de restrições em t_f.

3.3 - DESENVOLVIMENTO TEÓRICO

Existem vários procedimentos possíveis ao cálculo de Δa, numa iteração típica, de modo a se chegar o mais próximo possível à condição de satisfação dos vínculos e da minimização do índice de desempenho. No Capítulo II foram apresentados, além dos procedimentos de Willianson (1971) e de Hull e Edgeman (1975), três outras sugestões para problemas associados a uma iteração, problemas II.4, II.5 e II.7.

O procedimento associado ao problema II.4 impõe o valor de ΔIP, requerendo, do algoritmo global, atenção especial ao parâmetro β, pois a imposição de ΔM restringe os ΔIP possíveis ao problema não linear. Isto pode implicar no encontro de vetores Δa, quando da resolução do problema linearizado, não compatível com a hipótese de valida de da linearização. O autor considera que essas caracterís ticas provavelmente garantirão boa capacidade de convergên cia, quando longe da solução ótima e convergência lenta, quando próxima da solução ótima.

Os procedimentos associados ao problema II.5 e II.7 não impõem o valor de ΔIP, apenas estabelecem um li mite inferior para ΔIP e, pode ser observado pela função G, que esses problemas associados visam minimizar IP, mas em consonância com a minimização da norma do vetor a. Através do controle dos parâmetros β e $\bar{\omega}$ (problemas II.7 e II.5), altera-se IP, com o objetivo final de extremizar IP sem comprometer o objetivo de satisfazer as equações de restri ções. O autor espera com isto uma convergência mais acen tuada, quando próxima da solução ótima. Os procedimentos as sociados aos problemas II.5 e II.7 são praticamente equiva lentes, embora o procedimento associado ao problema II.7 le ve uma pequena vantagem, no que se refere à simplicidade de implementação.

A decisão sobre a adoção de um destes três procedimentos é um problema em aberto e depende, provavelmen te, do problema a ser resolvido. Para este trabalho foi es colhido o procedimento associado ao problema II.7.

Considere-se o problema III.1. A partir de uma perturbação linear das equações de restrições e do índi ce de desempenho, respeitados os vínculos dinâmicos, tem-se o sistema de equações a seguir.

$$\Delta M = M_{x_f} x_{f_a} \Delta a + M_{a_g} \Delta a_g$$

$$\Delta IP = IP_{x_f} x_{f_a} \Delta a + IP_{a_g} \Delta a_g \qquad \text{(III.2)}$$

onde,

os subscritos referem-se a derivadas parciais de M, x_f, IP em *relação* ao vetor x_f, vetor a e a_g.

Substituindo-se o sistema III.2 nas equações correspondentes ao problema II.7, tem-se o problema a seguir.

Minimizar:

$$G = \omega_1 |\Delta a_1| + \omega_2 |\Delta a_2| + \ldots + \omega_g |\Delta a_g| + \bar{\omega} z$$

sujeito a:

$$M_{x_f} x_{f_a} \Delta a + M_{a_g} \Delta a_g = \alpha M$$

$$IP_{x_f} x_{f_a} \Delta a + IP_{a_g} \Delta a_g - z = \beta(|IP| + 1),$$

$$z \geq 0 \qquad \text{(III.3)}$$

onde,

a, α, IP, etc são definidos nos problemas II.7 e III.1.

O problema III.3 pode ser transformado na forma usual de programação linear, através das transformações de variáveis a seguir.

$$\Delta a_i = s_i - s_{g+i} \qquad i = 1, 2, \ldots, g$$

$$z = s_{2g+1}$$

$$\bar{\omega} = \omega_{2g+1} \qquad \text{(III.4)}$$

Tem-se então,

Minimizar:

$$G = \sum_{i=1}^{2g+1} \omega_i s_i \qquad \omega_i > 0$$

sujeito a:

$$\sum_{i=1}^{g} A_{ji} s_i - \sum_{i=1}^{g} A_{ji} s_{g+i} = \alpha M_j \quad , \quad j = 1, 2, \ldots, m$$

$$\sum_{i=1}^{g} B_i s_i - \sum_{i=1}^{g} B_i s_{g+i} - s_{2g+1} = \beta(|IP| + 1) \qquad \text{(III.5)}$$

onde,

$$A_{ji} = \frac{\partial M_j}{\partial x_f} \frac{\partial x_f}{\partial a_i} \quad , \quad \text{para } i = 1, 2, \ldots, g-1$$

$$A_{jg} = \frac{\partial M_j}{\partial x_f} \frac{\partial x_f}{\partial a_g} + \frac{\partial M}{\partial a_g}$$

$$B_i = \frac{\partial IP}{\partial x_f} \frac{\partial x_f}{\partial a_i} \quad , \quad \text{para } i = 1, 2, \ldots, g-1$$

$$B_g = \frac{\partial IP}{\partial x_f} \frac{\partial x_f}{\partial a_g} + \frac{\partial IP}{\partial a_g}$$

α, β são definidos no problema II.7;

ω_i são parâmetros positivos.

Considerando-se o desenvolvimento analítico para obtenção das equações correspondentes ao problema III.5, a partir do problema II.7, ter-se-ia ω_i igual ω_{i+g}. No entanto, esta imposição não é necessária e corresponde a uma pequena modificação da norma do vetor a.

3.4 - CONSIDERAÇÕES SOBRE A CONVERGÊNCIA

Das equações correspondentes ao problema III.5, é fácil ver em uma iteração típica i, que para altos valores de ω_{2g+1}, o vetor de parâmetros, corrigido pelos resultados obtidos da solução do problema III.5,está mais próximo da solução ótima, sendo válida a hipótese de linearidade e adotados α e β negativos, com módulos menores do que a unidade.

$$a_{i+1} = a_i + \Delta a_i \qquad \text{(III.6)}$$

então:

$$|M_{i+1}| = |M_i| - |\alpha M_i| \qquad \text{(III.7)}$$

$$IP_{i+1} = IP_i - |\beta|(|IP| + 1.) + s_{2g+1} \qquad \text{(III.8)}$$

Na equação III.7, o requisito de caminhar em direção aos vínculos está satisfeito se

$$s_{2g+1} < |\beta|(|IP| + 1) \qquad \text{(III.9)}$$

então, o mesmo ocorre com relação à redução de IP, pela equação III.8. Entretanto, a validade da equação III.9 é importante somente quando esta equação é viável, e isto é garantido, mantendo-se valores altos para ω_{2g+1}.

É importante que o problema III.5 possua soluções ao longo do processo de convergência, que resultem na satisfação dos vínculos e na minimização de IP. Para isto, é necessário que os parâmetros da função de controle sejam consistentes, de modo a gerar um espaço de soluções para o controle, que satisfaçam os vínculos de restrição e que, nesse espaço, exista uma solução que redunde no menor valor de IP.

O número mínimo de parâmetros com possibilidades de satisfazer a condição acima é m+1. E isto pode ser demonstrado a seguir.

Suponha que exista uma solução $a_1, a_2, \ldots, a_g$, que satisfaça os vínculos e minimize IP. Então, é possível, partindo-se dessa solução, caminhar no sentido de aumentar IP, mantendo-se as restrições satisfeitas. Assim, do problema III.5 tem-se o sistema de equações a seguir.

$$\sum_{i=1}^{g} A_{ji}\,\Delta a_i = 0 \qquad j = 1, 2, \ldots, m$$

$$\sum_{i=1}^{g} B_i\,\Delta a_i = |\Delta IP| \tag{III.10}$$

logo, como o número de equações do sistema III.10 é m+1, o número de variáveis deve ser maior ou igual a m+1, ou seja:

$$g \geq m+1 \tag{III.11}$$

No Capítulo V sobre teste do procedimento, um caso com g = m+1 foi testado e apresentou bons resulta dos, mostrando, portanto, que um vetor de m+1 parâmetros consistentes pode eventualmente ser utilizado.

CAPÍTULO IV

PROCEDIMENTO NUMÉRICO

4.1 - INTRODUÇÃO

O procedimento numérico deve satisfazer os requisitos de rapidez, compacidade e confiabilidade, de acordo com a aplicação e recursos computacionais disponíveis.

O tratamento numérico envolve etapas típicas em cada iteração. As integrações numéricas, as derivações numéricas e a solução do problema de programação linear são as principais etapas e, dependendo do procedimento adotado para cada uma delas, o programa para computador pode ser mais ou menos compacto, rápido ou confiável. No presente trabalho, por se tratar do desenvolvimento de um método sob o aspécto de teste do seu funcionamento, não existe preocupação com relação aos requisitos de rapidez e de compacidade.

. *Integração numérica*

Para a integração foi utilizado um subprograma bastante elaborado, baseado em um método Runge-Kutta,com passo ajustável (Fehlberg, 1969 e Forsythe, 1977). O subprograma utilizado é completo no sentido de dar informações sobre erros internos, permitir o controle sobre o erro de integração, etc. Entretanto, não é um subprograma compacto e pode ser considerado lento.

. *Programação linear*

Para a resolução do problema de programação linear, foi utilizado um subprograma, também, bastante ela borado, que emprega o algoritmo do *Simplex com Multiplicadores* (Apêndice A.3), que é completo no sentido de informações dos erros internos, etc. Mas, por outro lado, o subprograma não é compacto e, embora o tempo de resolução do problema de programação linear seja pequeno, em relação ao tempo de processamento gasto para as integrações numéricas requeridas em cada iteração, não pode ser considerado rápido.

. *Cálculo da matriz de derivadas parciais*

Os procedimentos aplicáveis para o cálculo das derivadas, das variáveis de estado em relação aos parâmetros otimizáveis, são apresentados a seguir.

Um primeiro procedimento, por derivação numérica direta, requer g (número de parâmetros)integrações adicionais das equações dinâmicas do sistema (problema III.1).

Em cada uma destas integrações, um dos componentes a_i é incrementado de Δa_i, o que resulta em $\Delta x_j(t_f)$ e daí calcula-se o elemento $(x_{f_a})_{ij}$ da matriz de derivadas x_{f_a}.

$$\frac{\delta x_j}{\delta a_i}(t_f) \simeq \frac{\Delta x_j(t_f)}{\Delta a_i} \qquad \text{(IV.1)}$$

Os inconvenientes de utilização deste método são os erros numéricos e a escolha dos Δa_i. Se Δa_i é muito pequeno, de modo que, o erro de truncamento na integração numérica influa bastante, em termos percentuais, em Δx_f, o resultado da derivação não é bom; por outro lado se Δa_i for grande, a hipótese de linearidade não é válida.

Um segundo procedimento, apresentado a seguir, é mais preciso do que o método de derivação numérica direta, mas envolve maiores dificuldades de implementação em programas com características de utilização genérica.

Considere-se o sistema linearizado obtido a partir do problema III.1.

$$\delta\dot{x} = f_x \delta x + f_a \delta a \quad ,$$

$$\delta x(t_0) = (\delta a_1, \delta a_2, \ldots, \delta a_k, 0, 0, \ldots, 0) \qquad (IV.2)$$

onde,

k é o número de parâmetros iniciais otimizáveis.

A variação (δx_f) do estado final depende de duas variações parciais, uma primeira $(\delta x_f')$ associada à variação do controle ao longo do tempo e, uma outra $(\delta x_f'')$ associada às perturbações nos parâmetros correspondentes aos tempos nodais otimizáveis. Tem-se então:

$$\delta x_f = \delta x_f' + \delta x_f''$$

$$\delta x_f' = \Phi' \delta a$$

$$\delta x_f'' = \Phi'' \delta a$$

$$x_{f_a} = \Phi' + \Phi'' \qquad (IV.3)$$

onde,

Φ', Φ'', x_{f_a} são matrizes nxg;

Φ' é obtida a partir de g integrações numéricas de t_0 à t_f do sistema IV.2, tomando-se em cada integração i, $\delta a_i = 1$ e $\delta a_j = 0$ $(j \neq i)$;

Φ'' é obtida através da transição para o tempo final das variações locais em x, resultante de perturbações unitárias nos tempos nodais otimizáveis.

Os dois procedimentos de derivação numérica, citados anteriormente, apresentam-se na forma genérica. No entanto, dependendo das características da aproximação adotada para o controle, podem existir simplificações consideráveis no processo de derivação numérica, como pode ser visto no trabalho de Rios Neto e Ceballos (1979).

Neste trabalho foi utilizado o método direto de derivação.

4.2 - DESCRIÇÃO DAS ETAPAS DO ALGORITMO NUMÉRICO

4.2.1 - INTRODUÇÃO

Pode-se dividir o algoritmo numérico, para cada iteração no processo global, nas seguintes etapas principais:

- cálculo e decisões auxiliares;
- análise dos resultados;
- cálculo dos parâmetros que comandam a convergência;
- atualização eventual de x_{f_a} e cálculo dos coeficientes A_{ij} e B_i;
- preparação do problema de programação linear;
- resolução do problema de programação linear;
- resultados do problema de programação linear.

4.2.2 - CÁLCULOS E DECISÕES AUXILIARES

Neste item são descritos os principais cálculos e decisões auxiliares.

i) *Hipótese de linearidade*

Com o objetivo de manter a hipótese de linearidade, foi definido, "a priori", um valor máximo para os incrementos sobre os parâmetros otimizáveis Δa_{max}.

Quando alguns dos $|\Delta a_i|$ ultrapassa este valor, calcula-se r,

$$r = \frac{\Delta a_{max}}{\max|\Delta a_i|} \qquad \text{(IV.4)}$$

e faz-se a multiplicação de todos Δa_i por r. Devido a este artifício, foi mantido constante o valor de α (problema III. 5).

ii) *Proximidade de satisfação dos vínculos*

O parâmetro que quantifica a proximidade de satisfação dos vínculos é ξ_p. Os vínculos estão próximos de serem satisfeitos quando,

$$\sum_{i=1}^{m} |M_i| < \xi_p \qquad \text{(IV.5)}$$

O parâmetro ξ_p é decrescente ao longo do processamento e, é diminuido cada vez que na iteração típica ocorre, simultâneamente, os eventos a seguir.

. $\xi_p > \xi_{p_{min}}$
. $\Delta IP > 0$;
. $\sum_{i=1}^{m} |M_i| < \xi_p$,

onde,

$\xi_{p_{min}}$ é definido "a priori".

O conceito de próximo dinamiza o processo de convergência, pois, quando os vínculos estiverem próximos de serem satisfeitos, é diminuída a preocupação de caminhar na direção de satisfação dos vínculos e aumentada a preocupação de caminhar na direção de redução de IP e,vice-versa.

iii) *Critérios de parada*

O processo é interrompido quando ocorre um dos seguintes eventos:

. um número especificado de atualizações da matriz de derivadas parciais x_{f_a} é ultrapassado (item 4.2.5);

. o problema de programação linear associado a uma iteração típica não possue solução, e não existe possibilidade de modificações (item 4.2.5) do problema de programação linear, visando a alteração desta situação;

. o maior módulo dos incrementos previstos nas restrições e em IP, é menor do que um pequeno parâmetro pré-especificado, ξ_b.

iv) *Rejeição da iteração*

A iteração pode ser rejeitada em dois casos. O primeiro, quando IP não diminui o previsto (itèm 4.2.3) , não é obtido sucesso (item 4.2.3) no que se refere à caminhada na direção de satisfação dos vínculos, e não são completadas três rejeições consecutivas. O segundo, quando IP diminui o previsto, mas não é obtido sucesso no que se refere à caminhada na direção de satisfação dos vínculos, a solução não está próxima (proximidade, neste item) e, não são completadas três rejeições consecutivas.

Em caso de rejeição da iteração, são feitas as alterações e volta-se ao ponto do algoritmo, onde se inicia a resolução do problema associado.

As alterações associadas à rejeição são:

. redução de β;

. redução de ω_{2g+1};

e redução do quanto se quer reduzir IP.

As últimas alterações, em ω_{2g+1} e em IP, é que possibilitam os melhores resultados na próxima tentativa.

4.2.3 - ANÁLISE DOS RESULTADOS

A análise dos resultados do problema em cada iteração é o que permite direcionar o cálculo dos parâmetros de convergência, acionar os critérios de parada ou rejeitar a iteração.

. *Somatória dos desvios dos vínculos*

Define-se como sendo SV_i a somatória dos desvios dos vínculos na iteração i.

$$SV_i = \sum_{j=1}^{m} |M_j| \qquad (IV.6)$$

. SV_i *previsto admissível*

Define-se como sendo SV_i' o desvio previsto admissível.

$$SV_i' = SV_{i-1}(1. - \alpha\eta_v) \quad , \quad \text{para}$$

$$\max|\Delta a_i| \leq \Delta a_{max}$$

$$SV_i' = SV_{i-1}(1. - \alpha\eta_v r) \quad , \quad \text{para}$$

$$\max|\Delta a_i| > \Delta a_{max} \qquad \text{(IV.7)}$$

onde,

$\max|\Delta a_i|$ é o máximo elemento do vetor Δa_i;

r é definido na equação IV.4;

η_v (eficiência limite) é a razão limite admissível entre os incrementos obtidos sobre SV_i e os previstos pelo problema expandido linearmente, sendo definida "a priori".

. *Variação normal das equações de vínculos*

Diz-se que houve sucesso na caminhada em direção à satisfação dos vínculos, quando SV_i for menor do que SV_i'.

. *Previsões sobre a redução de IP*

Em termos de redução de IP, poder-se-ia fazer um procedimento análogo ao de redução dos vínculos. Entretanto, preferiu-se utilizar simplesmente a observação da variável s_{2g+1} (equação III.8). Quando s_{2g+1} é nula, diz-se que IP reduziu o previsto e quando s_{2g+1} é diferente de zero, diz-se que IP não reduziu o previsto. Obviamente, a se gunda hipótese admite o aumento de IP.

4.2.4 - PARÂMETROS QUE COMANDAM A CONVERGÊNCIA

Como pode ser visto no item 3.3, os parâmetros β e ω_{2g+1} são os parâmetros principais, na dosagem dos objetivos de caminhar no sentido de minimizar IP e satisfazer os vínculos.

Quanto maior for ω_{2g+1}, maior é a imposição de que IP deve ser reduzido, tal como o máximo previsto. Por outro lado, quando ω_{2g+1} tende a zero, a solução tende a caminhar no sentido do gradiente de satisfação dos vínculos, ou seja, não é dada maior atenção à redução do índice de desempenho.

Para o cálculo dos parâmetros ω_{2g+1} e β, tem-se as possíveis situações a seguir.

- Ocorreu sucesso e IP diminuiu o previsto. Então, *aumenta-se o valor do módulo de* β.

- Não ocorreu sucesso e IP não diminuiu o previsto. Então, *rejeita-se a iteração.*

- A solução é próxima, ocorreu sucesso e IP não diminuiu o previsto. Então, *aumenta-se fortemente* ω_{2g+1}.

- A solução não é próxima, ocorreu sucesso e IP não diminuiu o previsto. Então, *aumenta-se* ω_{2g+1} *e diminue-se o valor do módulo de* β.

- IP diminuiu o previsto e a solução é próxima. Então, *aumenta-se o valor do módulo de* β.

Os termos aqui utilizados estão definidos nos itens 4.2.2 e 4.2.3.

4.2.5 - CÁLCULO DAS DERIVADAS

As derivadas A_{ij} e B_i (problema III.5) podem ser calculadas a partir dos valores $(x_{f_a})_{ij}$, calculados através da equação IV.1, e das derivadas parciais, M_{a_g}, M_{x_f}, IP_{a_g}, IP_{x_f}, expressas em um subprograma especial.

As equações correspondentes ao problema III.5, bem como as derivadas $(x_{f_a})_{ij}$, são válidas para variações suficientemente pequenas do estado. Quando se ultrapassa a faixa de validade da matriz de derivadas x_{f_a}, o mesmo ocorre com o problema III.5. No entanto, o inverso não é verdadeiro, ou seja, a validade do problema III.5 não depende somente da validade de x_{f_a}. Portanto, este problema pode ser atualizado, sem que x_{f_a} seja atualizada. Se a isto for acrescentado o fato de que o cálculo das derivadas, mencionadas, é o que envolve maior tempo de computador, justifica-se o procedimento adotado de atualizá-las, somente se ocorrer um dos eventos:

. for a primeira iteração;

. o problema de programação linear sem solução (item 4.2.7) e a matriz de derivadas x_{f_a}, não for atualizada nessa iteração;

. a somatória acumulada (desde a última atualização de x_{f_a}) do máximo elemento em módulo do vetor Δa, em cada iteração, ultrapassar um número pré-especificado (s_n), multiplicado pelo elemento máximo em módulo do vetor Δa nessa iteração;

. a somatória acumulada do elemento máximo em módulo, do vetor Δa, em cada iteração, ultrapassar um número ro pré-especificado (Sa_{max}), multiplicado por Δa_{max} (item 4.2.2).

4.2.6 - PREPARAÇÃO E RESOLUÇÃO DO PROBLEMA ASSOCIADO

Com os valores A_{ij} e B_i (item 4.1.4), M, α, β e IP, a montagem do problema associado torna-se simples (problema III.5) e para sua resolução é utilizado o método do Simplex com multiplicadores (item 4.1).

4.2.7 - RESULTADOS DO PROBLEMA ASSOCIADO

A resolução (Apêndice A) do problema III.5, pode ser:

Normal: quando o problema tem solução limitada e o número admissível de iterações, para a resolução, não é ultrapassado.

Anormal: quando o problema possui solução ilimitada, o número limite de iterações é ultrapassado, ou quando o problema não tem solução por redundância ou inconsistência.

4.3 - ALGORITMO NUMÉRICO

Utilizando-se as considerações apresentadas ao longo do item 4.2, apresenta-se abaixo o algoritmo numérico do método.

1) Início.
2) Se for a primeira iteração, lêr e imprimir os dados de entrada.
3) Eventualmente (item 4.2.5), calcular x_{f_a}.
4) Preparar o problema de programação linear (item 4.2.6).

5) Resolver o problema de programação linear (item 4.2.6).

6) Analisar os resultados do problema de programação linear (item 4.2.7).

   . Se a saída for normal, continuar em (7).

   . Se a saída for anormal, verificar se é necessário terminar (item 4.2.2) ou atualizar x_{f_a} (item 4.2.2). Se for necessário terminar, continuar em (15), se for necessário atualizar x_{f_a} continuar em (3).

7) Verificar e providenciar para que a hipótese de linearidade (item 4.2.2) seja obedecida.

8) Atualizar o vetor a de parâmetros otimizáveis.

9) Calcular os elementos necessários para: analisar os resultados (item 4.2.3), atualizar os parâmetros que comandam a convergência (item 4.2.4), acionar os critérios de rejeição da iteração, acionar os critérios de interrupção do procedimento, etc.

10) Se for necessário, imprimir os resultados desta iteração.

11) Se for necessário interromper o procedimento (item 4.2.2), continuar em (15).

12) Se a iteração for rejeitada, (item 4.2.2), providenciar cálculos necessários e continuar em (5).

13) Atualizar os parâmetros que comandam a convergência (item 4.2.4).

14) Retornar a (1).

15) Imprimir resultados finais.

16) Interromper o procedimento.

4.4 - PROGRAMA PARA COMPUTADOR

O Apêndice B apresenta a listagem completa do programa. Ao longo da listagem existem comentários sobre o que cada trecho do programa faz, bem como os comentários de definição das variáveis importantes para o programa.

O programa para computador é de caráter geral e, com pequenas adaptações feitas nos subprogramas a seguir, resolve qualquer problema do tipo III.1.

- Função F, é o subprograma pertencente ao conjunto de subprogramas que faz a integração numérica. Particularmente, é o subprograma que representa as equações dinâmicas.

- Sub-rotina D$V, é o subprograma que representa as derivadas parciais das equações de restrições e do IP, em relação às variáveis de estado e ao tempo final.

O Apêndice C apresenta o gabarito para a entrada de dados, a sequência de dados de todos os exemplos do Capítulo V, e uma saída característica.

CAPÍTULO V

TESTE DO PROCEDIMENTO

5.1 - INTRODUÇÃO

O problema da trajetória Terra-Marte com baixo empuxo, mínimo tempo, e trajetória plana foi o escolhido, pelo fato de ser, este problema, clássico em trabalhos de implementação (Willianson, 1971 e Hull e Edgeman, 1975) ou comparações de métodos (Tapley e Lewallen, 1967).

Para efeito de ilustração foram adotadas várias formulações para o controle e apresentados os resultados.

5.2 - COLOCAÇÃO DO PROBLEMA ADOTADO

O problema da transferência de órbita Terra-Marte, orbitas circulares, trajetória plana, e tempo a ser minimizado pode em primeira aproximação ser representado pelo problema a seguir.

Minimizar:

$$IP = t_f$$

sujeito a:

Vínculos dinâmicos

$$\dot{x}_1 = \dot{r} = v_r$$

$$\dot{x}_2 = \dot{v}_r = v_t^2/r - \mu/r^2 + T \operatorname{Sen} u/(m_0 - \dot{m}t)$$

$$\dot{x}_3 = \dot{v}_t = - v_t v_r / r + T \cos u / (m_0 - \dot{m} t)$$

Vínculos de extremidade

$$M_1 = v_r(t_f) = 0$$

$$M_2 = v_t(t_f) - \sqrt{\mu / r(t_f)} = 0$$

$$M_3 = r(t_f) - \bar{r}_f = 0 \qquad (V.1)$$

Dados adimensionalizados, conforme Bryson e Ho (1975), em relação à base r_0, μ, m_0:

Condições iniciais

$$t_0 = 0$$

$$r(t_0) = 1.$$

$$v_r(t_0) = 0.$$

$$v_t(t_0) = \sqrt{\mu / r(t_0)} = 1. \qquad (V.2)$$

Parâmetros do sistema

$$\mu = 1.$$

$$\dot{m} = .074800391$$

$$m_0 = 1.$$

$$T = .14012969 \qquad (V.3)$$

Condições finais

$\bar{r}_f = 1.523679$

onde (Figura V.1),

u é o controle;

r é a distância radial da espaçonave ao centro de atração;

v_r é a velocidade radial;

v_t é a velocidade tangencial;

m_0 é a massa inicial total da espaçonave;

$\dot{m}$ é o consumo de combustível;

μ é a constante gravitacional;

T é o empuxo na direção do jato.

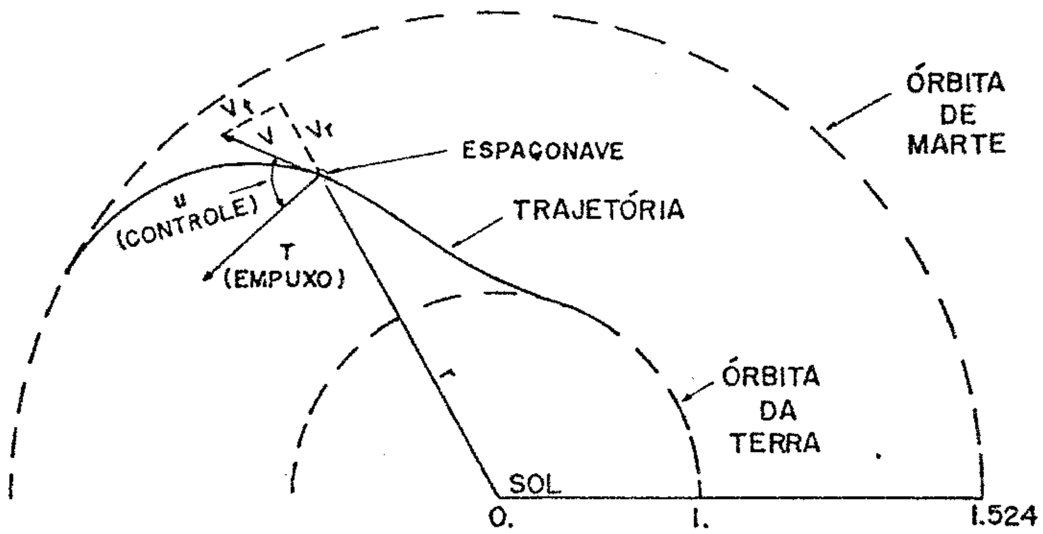

Fig. V.1 - Transferência de órbita Terra-Marte

5.3 - APROXIMAÇÃO PARA O CONTROLE

5.3.1 - INTRODUÇÃO

O funcional para o controle a ser adotado, depende do problema que estiver sendo resolvido, do conhecimento ou não da forma típica do controle para o problema, etc. No entanto, os aspectos a seguir podem servir de base para a escolha do funcional.

i) O número de parâmetros deve ser suficientemente grande, de modo a permitir a satisfação das equações de restrições e minimização de IP, simultaneamente (item 3.4).

ii) Os parâmetros devem ser o mais independentes possíveis, de modo a garantir a independência linear das filas das equações envolvidas no problema III. 5. Por exemplo, quando uma aproximação envolve um espaço de funções base, este deve ser preferivelmente ortogonal.

iii) O número de parâmetros deve ser limitado conforme o valor do tempo limite de convergência, pois o tempo por iteração (item 6.4) depende do número de parâmetros.

iv) O funcional deve ser compatível com a forma ótima do controle, ou seja, deve ser possível satisfazer, através do funcional, as características principais da forma ótima do controle. Este requisito está associado à precisão dos resultados a serem obtidos.

v) A sensibilidade do controle, em relação a cada pa râmetro a ser otimizado, deve, preferivelmente,ser uniforme entre si. Quando não, os parâmetros de peso ω_i (problema III.5) devem ser ponderados, de modo a contornar este problema.

vi) Deve se dar preferência a funcionais, para o controle, que apresentem boa correspondēcia entre os desvios nos valores do controle e o desvio no vetor de parâmetros em relação aos valores finais, ou seja, a um grande desvio no vetor de parâmetros deve corresponder um grande desvio nos valores resultantes e vice-versa. Os funcionais particionados em trechos apresentam boas características no que se refere a esta correspondência.

Para qualquer procedimento numérico ótimo ou sub-ótimo que se utiliza, faz-se necessário adotar uma ou várias condições de partida. Nos casos de desconhecimento das características (número de mínimos locais, regiões de instabilidade de convergência, sensibilidade do problema ao controle na região dos mínimos, etc.) do espaço de soluções do problema, aconselha-se a adoção de várias condições de partida, visando à varredura dos resultados possíveis de saída. Cálculos simples ou resultados de problemas semelhantes podem ser de grande valia na escolha das condições de partida.

Particularmente, para o caso de aproximação do controle por um funcional, a condição de partida corresponde ao vetor de parâmetros de partida. Entretanto, deve ser lembrado que conhecida as características do espaço de soluções para o problema ótimo, não necessariamente existe uma correspondência com as características do espaço de soluções para o problema restrito.

Neste trabalho foram adotados parâmetros correspondendo a funções de partida lineares, com o ângulo correspondente variando dentro do primeiro período (0 à 2π). Os valores numéricos adotados para o tempo final e para o controle nas extremidades do intervalo do tempo são próximos aos valores adotados por Willianson (1971).

5.3.2 - APROXIMAÇÃO POR INTERPOLAÇÕES LINEARES

A aproximação por interpolações lineares possui ótimas características, devido à sua grande flexibilidade. Principalmente nos casos em que a forma típica da solução é desconhecida, ou quando não se possui alguma formulação que se adapte às características do controle, esta aproximação é sugerida.

A aproximação por interpolações lineares, testada neste trabalho, consiste na subdivisão do intervalo de tempo em (g-2) intervalos, e o valor do controle correspondente aos tempos nas extremidades dos intervalos são os parâmetros a serem otimizados. Portanto, a função do controle pode ser representada como a seguir.

Para cada intervalo,

$$t_i < t \leq t_{i+1}$$

Calcula-se,

$$t_\omega = ((t - t_i)/a_g)(g - 2)$$

Substitui-se t_ω em,

$$u = a_i + (a_{i+1} - a_i)\, t_\omega \qquad \text{(V.5)}$$

onde,

t é a variável tempo;

t_i, t_{i+1} são os tempos nodais;

a_g é o tempo final;

a_i é o valor de $u(t_i)$;

a_{i+1} é o valor de $u(t_{i+1})$;

t_ω é o tempo normalizado (0,1);

g é o número de parâmetros otimizáveis;

u é o controle.

i) *Resultados obtidos para* g = 7 (Tabela V.1):

Número total de atualizações de x_{f_a} = 14

TABELA V.1

PARÂMETROS INICIAIS E FINAIS
(Interpolações lineares g=7)

PARÂMETROS	VALORES INICIAIS	VALORES FINAIS
a_1	0.	.3175
a_2	1.	.8421
a_3	2.	1.041
a_4	3.	4.926
a_5	4.	5.145
a_6	5.	5.497
a_7	3.4	3.326

Na Figura V.2, são plotadas a função de controle inicial, a função de controle final e a função de controle ótima.

ii) *Resultados obtidos para* g = 9 (Tabela V.2):

Número total de atualizações de x_{f_a} = 15

TABELA V.2

PARÂMETROS INICIAIS E FINAIS
(Interpolações lineares g=9)

PARÂMETROS	VALORES INICIAIS	VALORES FINAIS
a_1	0.	.3965
a_2	.7	.6008
a_3	1.4	1.o12
a_4	2.1	1.336
a_5	2.8	4.583
a_6	3.5	5.130
a_7	4.2	5.257
a_8	4.9	5.458
a_9	3.4	3.322

Na Figura V.3, são plotadas a função de controle inicial, a função de controle final e a função de controle ótima.

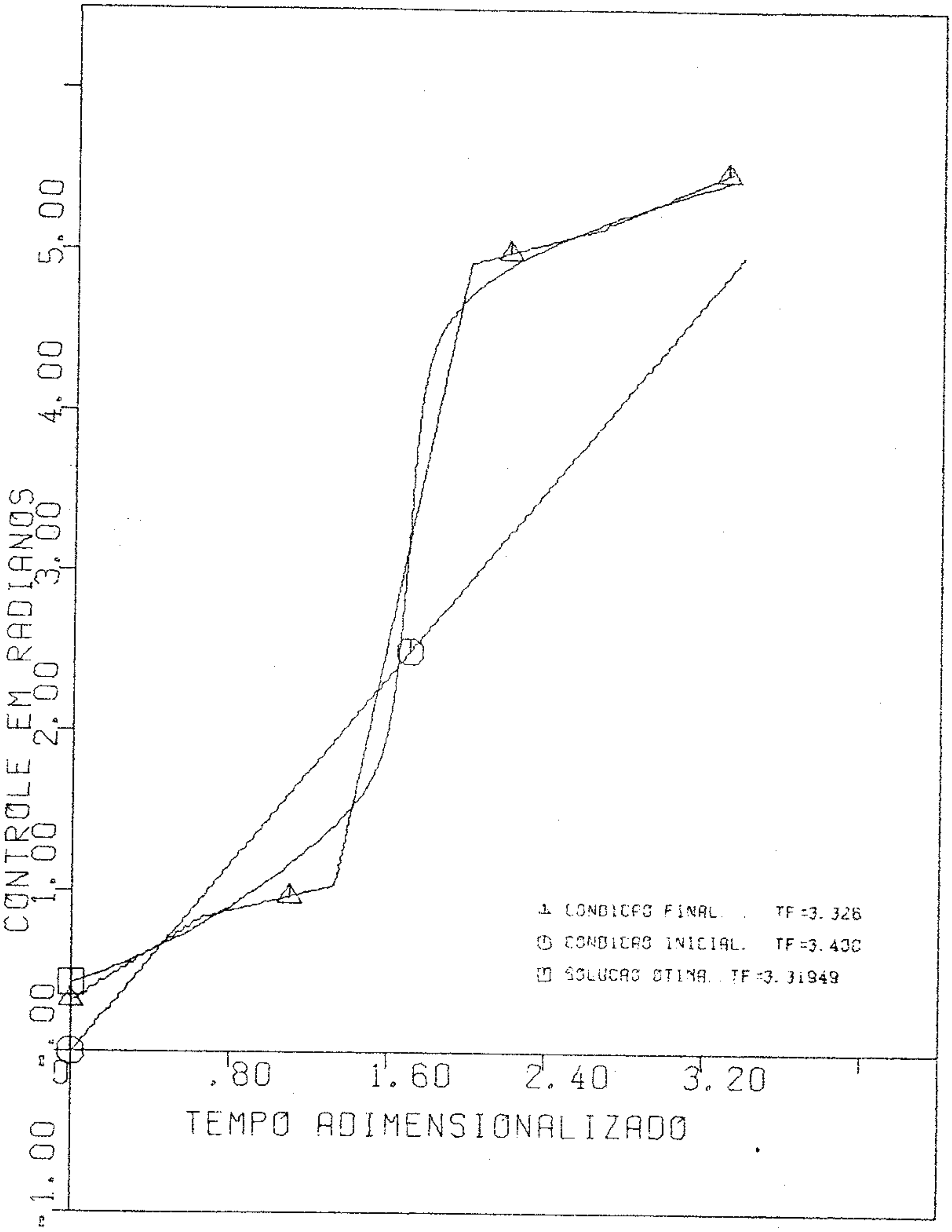

Fig. V.2 - Interpolações lineares (g = 7)

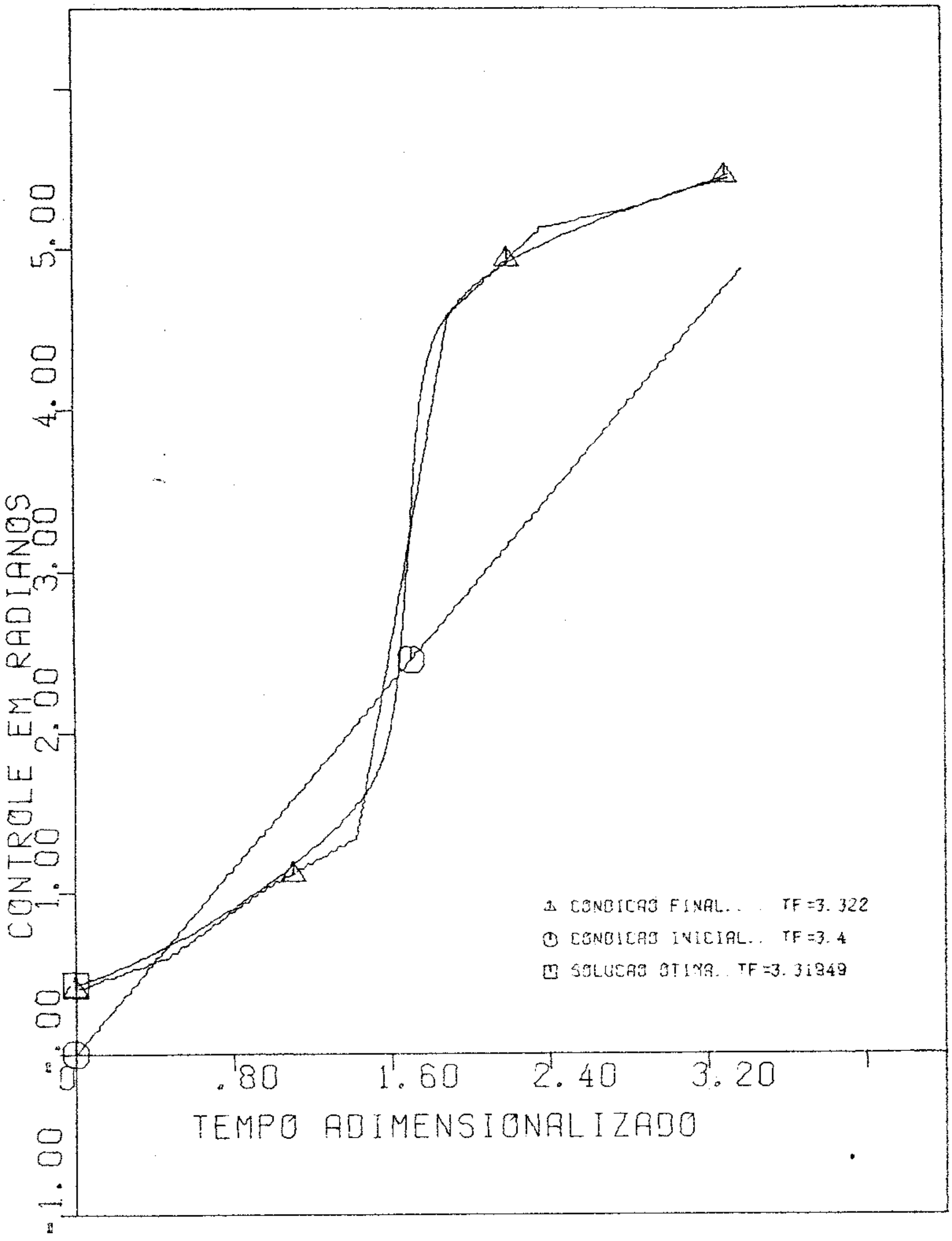

Fig. V.3 - Interpolações lineares (g = 9)

5.3.3 - APROXIMAÇÕES DESCONTÍNUAS

A função de controle ótima, apresenta,em torno de um instante próximo à metade do intervalo de tempo, uma variação brusca na função de controle. Isto sugeriu a adoção de funções de controle com descontinuidade e, para testes, foram adotados os dois casos a seguir.

i) *Descontinuidade com posição fixa*

A descontinuidade foi adotada na metade do intervalo e utilizou-se polinômios de Tchebyshev para as aproximações.

para $t < 0.5a_7$

$t_\omega = 2t/a_7 - 1$

$u = a_1 + a_2 t_\omega + a_3(2t_\omega^2 - 1)$

para $t \geq 0.5a_7$

$t_\omega = 4t/a_7 - 3$

$$u = a_4 + a_5 t_\omega + a_6(2t_\omega^2 - 1) \qquad (V.6)$$

onde,

$a_1, a_2, \ldots, a_6$ são os coeficientes da aproximação;

a_7 é o tempo final;

t_ω é o tempo normalizado (-1, 1).

Resultados (Tabela V.3):

O número total de atualizações de $x_{f_a} = 13$

TABELA V.3

PARÂMETROS INICIAIS E FINAIS
(Descontinuidade com posição fixa)

PARÂMETROS	VALORES INICIAIS	VALORES FINAIS
a_1	1.2	1.000
a_2	1.2	.6500
a_3	0.	.0247
a_4	3.7	4.858
a_5	1.2	.7325
a_6	0.	-.2536
a_7	3.4	3.323

Na Figura V.4, são plotadas a função de controle inicial, a função de controle final e a função de controle ótima.

ii) *Descontinuidade com posição otimizada*

para $t \leq a_7$

$$t_\omega = t/a_7$$

$$u = a_1 + a_2 t_\omega + a_3 t_\omega^2$$

para $t > a_7$

$$t_\omega = (t - a_7) / (a_8 - a_7)$$

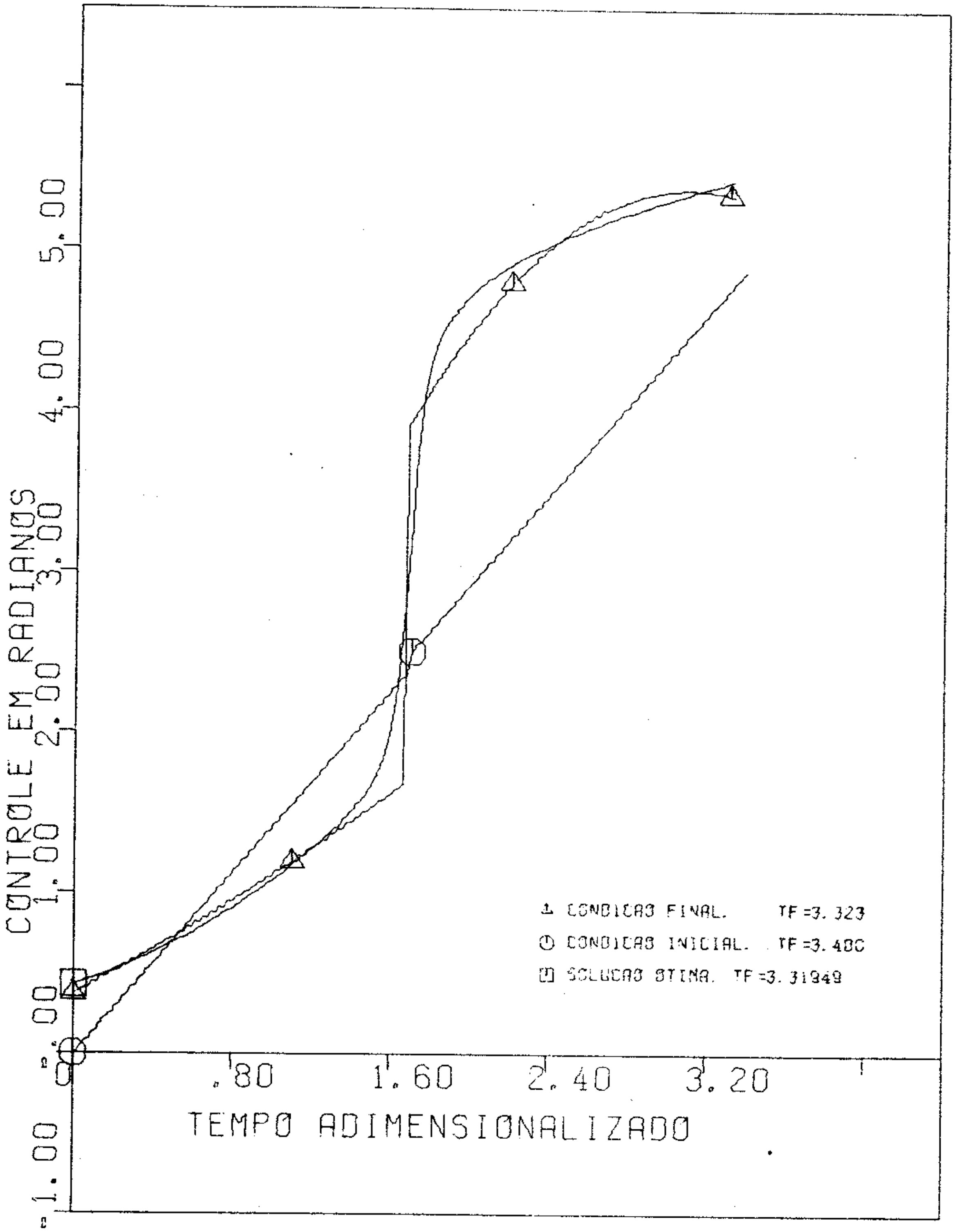

Fig. V.4 - Descontinuidade com posição fixa

$$u = a_4 + a_5 t_\omega + a_6 t_\omega^2 \qquad (V.7)$$

onde,

$a_1, a_2, \ldots, a_6$ são os coeficientes da aproximação;

a_7 é o tempo nodal intermediário;

a_8 é o tempo final;

t_ω é o tempo normalizado (0, 1).

Resultados (Tabela V.4):

Número total de atualizações de x_{f_a} = 10

TABELA V.4

PARÂMETROS INICIAIS E FINAIS

(Descontinuidade com posição otimizada)

PARÂMETROS	VALORES INICIAIS	VALORES FINAIS
a_1	0.	.3169
a_2	2.5	1.483
a_3	0.	0.
a_4	2.5	4.116
a_5	2.5	2.877
a_6	0.	-1.670
a_7	1.7	1.7
a_8	3.4	3.324

Na Figura V.5, são plotadas a função de controle inicial, a função de controle final e a função de controle ótima.

5.3.4 - APROXIMAÇÃO EM DOIS SEGMENTOS PARABÓLICOS

A aproximação é contínua, em dois segmentos parabólicos, com a posição de mudança de segmento definida por um parâmetro otimizável.

para $t < a_4$

$$t_\omega = t/a_4 - 1$$

$$u = a_1 + a_2 t_\omega + a_3 t_\omega^2$$

para $t \geq a_4$

$$t_\omega = (t - a_4) / (a_7 - a_4)$$

$$u = a_1 + a_5 t_\omega + a_6 t_\omega^2 \tag{V.8}$$

onde,

a_1, a_2, a_3, a_5, a_6 são os coeficientes da aproximação;

a_4 é o tempo nodal intermediário;

t_ω é o tempo normalizado (-1, 0; 0,1);

a_7 é o tempo nodal final.

Resultados (Tabela V.5):

O número total de atualizações de x_{f_a} = 24

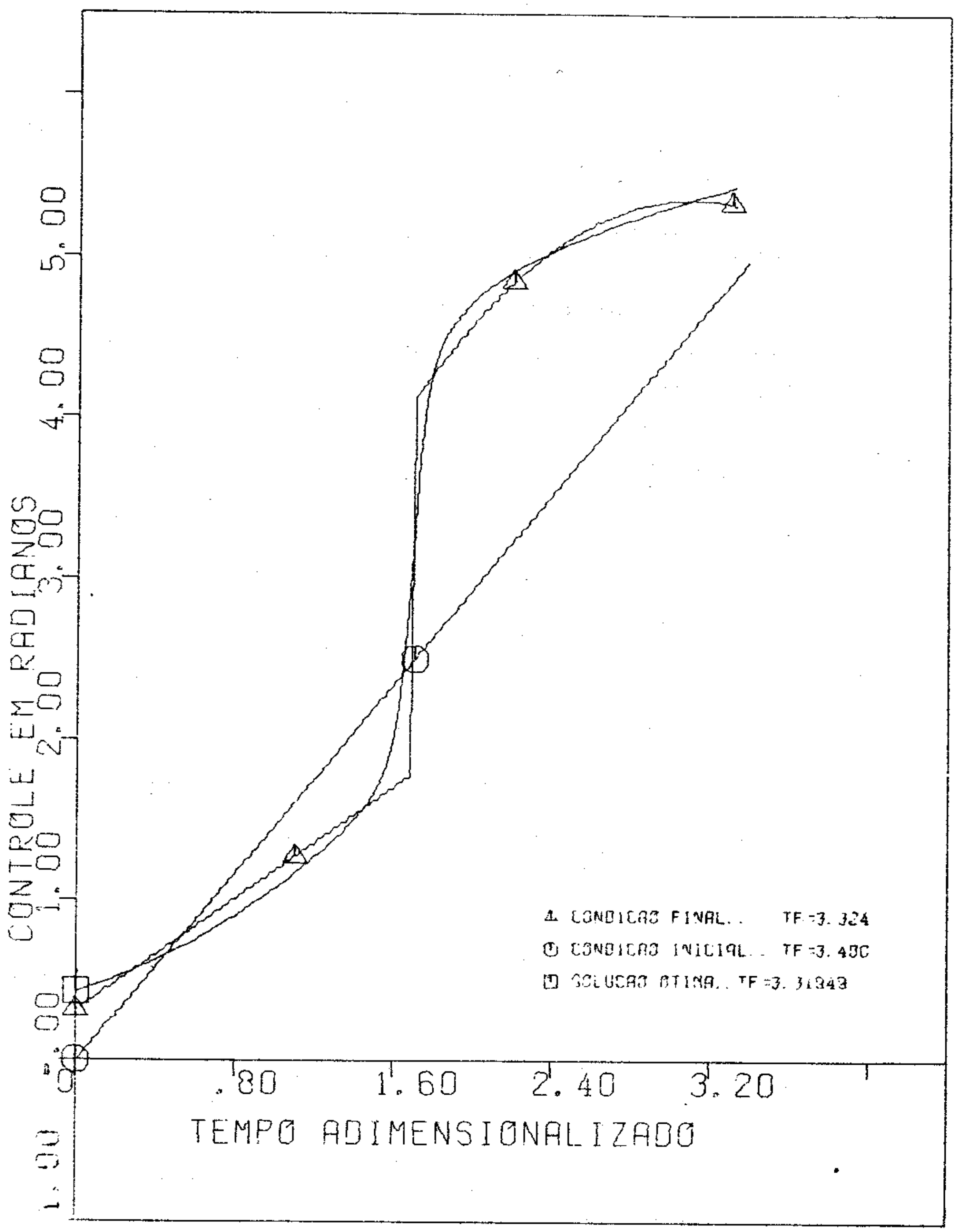

Fig. V.5 - Descontinuidade com posição otimizada

TABELA V.5

VALORES INICIAIS E FINAIS
(Dois segmentos parabólicos)

PARÂMETROS	VALORES INICIAIS	VALORES FINAIS
a_1	2.5	2.606
a_2	2.5	5.802
a_3	0.	3.734
a_4	1.7	1.618
a_5	2.5	8.198
a_6	0.	-5.482
a_7	3.4	3.372

Na Figura V.6, são plotadas a função de controle inicial, a função de controle final e a função de controle ótima.

5.3.5 - APROXIMAÇÃO EM DOIS SEGMENTOS SIMÉTRICOS

Neste caso de aproximação, foi imposta a simetria em relação ao ponto médio da curva resultante. A aproximação adotada é bastante interessante, não só pelo artifício da simetria, mas pelo fato de ter sido utilizado em número mínimo de parâmetros. Para o problema V.1, o número mínimo de parâmetros é quatro (equação III.11).

$$t_\omega = 2t/a_4 - 1$$

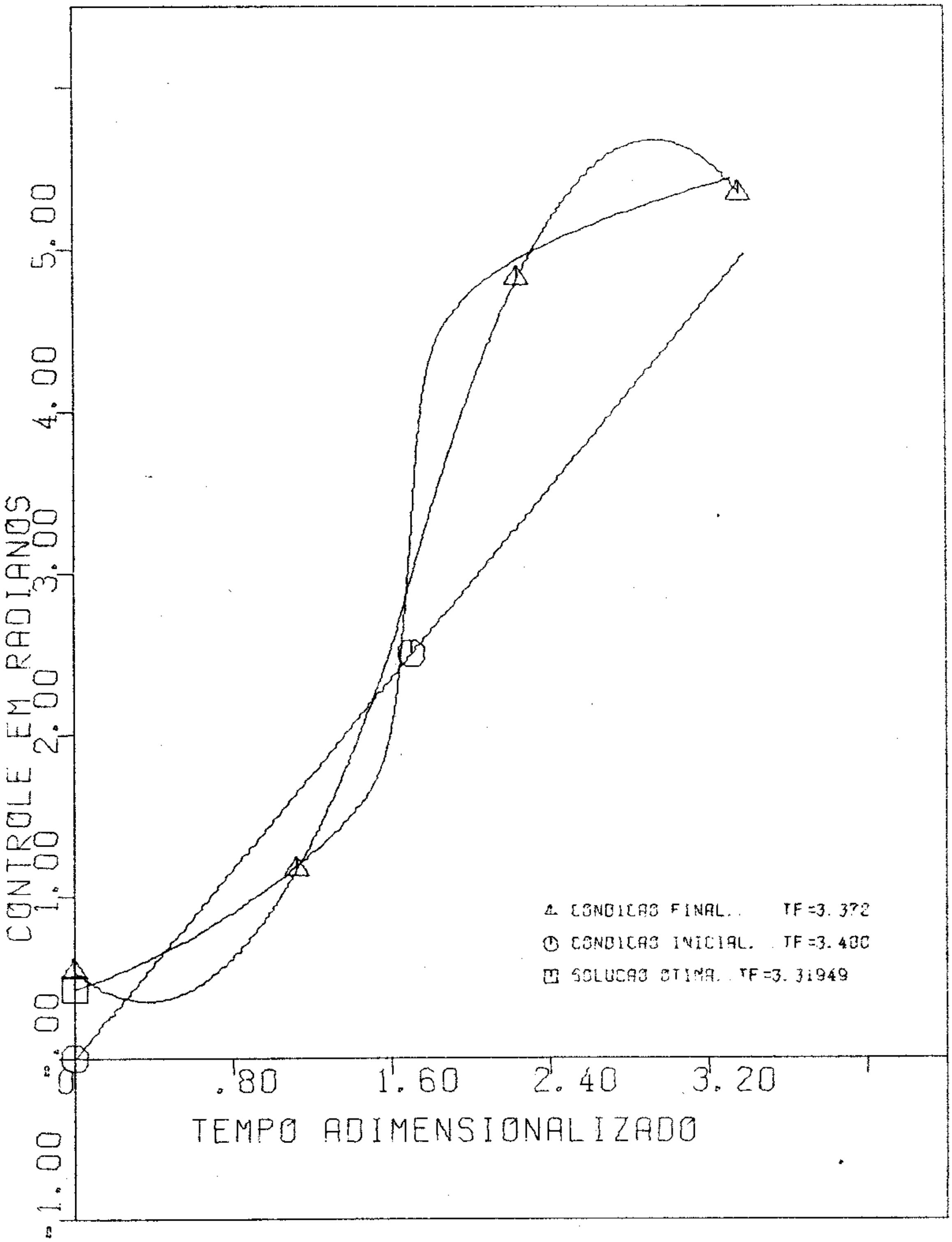

Fig. V.6 - Segmentos parabólicos

para $t_\omega \leq 0$

$$u = a_3 + a_1 t_\omega + a_2 t_\omega^2$$

para $t_\omega > 0$

$$u = a_3 + a_1 t_\omega - a_2 t_\omega^2 \qquad \text{(V.9)}$$

onde,

a_1, a_2, a_3 são os coeficientes da aproximação;

a_4 é o tempo final;

t_ω é o tempo normalizado (-1, 1).

Resultados (Tabela V.6):

O número total de atualizações de x_{f_a} = 11

TABELA V.6

VALORES INICIAIS E FINAIS

(Dois segmentos simétricos)

PARÂMETROS	VALORES INICIAIS	VALORES FINAIS
a_1	2.5	7.222
a_2	0.	4.898
a_3	2.5	2.970
a_4	3.4	3.375

Na Figura V.7, são plotadas a função de controle inicial, a função de controle final e a função de controle ótima.

5.3.6 - APROXIMAÇÕES POLINOMIAIS

Devem ser tomadas precauções ao se decidir pela utilização da formulação polinomial, pois, mesmo um controle com forma típica aparentemente simples, pode não ser bem representado por uma aproximação polinomial.

A escolha adequada do polinômio é bastante importante para rapidez da convergência, precisão dos resultados e, para escolha adequada, deve-se levar em conta os aspectos a seguir:

- . utilizar série de termos, preferivelmente ortonormal;
- . utilizar a série de termos mais adequada. Por exemplo: par ou ímpar, Taylor ou Fourier, etc;
- . adotar família de funcionais que obedeçam a características conhecidas. Por exemplo: passa pela origem, a integral é um valor conhecido, etc.

Neste trabalho, foram feitas duas aproximações, utilizando-se polinômios; aproximação por um polinômio Tchebyshev, de quarta ordem, e aproximação por um polinômio de quarta ordem.

i) *Aproximação de Tchebyshev (Polinômio de quarta ordem)*

$t_{\omega} = 2t/a_6 - 1$

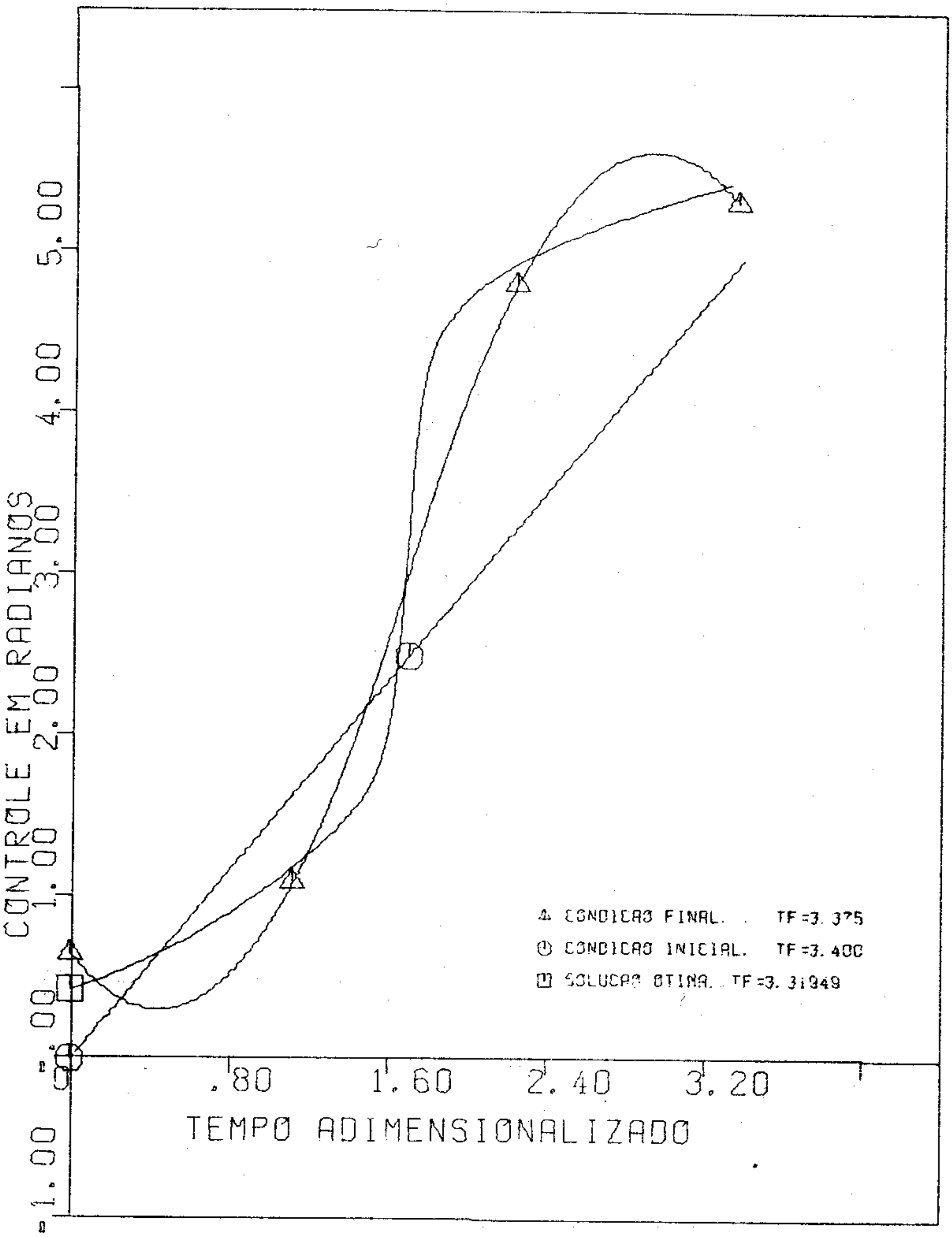

Fig. V.7 - Solução Simétrica

$$u = a_1 + a_2 t_\omega + a_3(2t_\omega^2 - 1) + a_4(4t_\omega^3 - 3t_\omega) +$$

$$+ a_5(8t_\omega^4 - 8t_\omega^2 + 1) \qquad (V.10)$$

onde,

a_1, a_2, ..., a_5 são os coeficientes do polinômio;

a_6 é o tempo final;

t_ω é o tempo normalizado (-1, 1).

Resultados (Tabela V.7):

Número total de atualizações de $x_{f_a} = 8$

TABELA V.7

VALORES INICIAIS E FINAIS
(Polinômio de Tchebyshev)

PARÂMETROS	VALORES INICIAIS	VALORES FINAIS
a_1	2.5	3.018
a_2	2.5	3.142
a_3	0.	0.
a_4	0.	-.8018
a_5	0.	-.854
a_6	3.4	3.423

Na Figura V.8, são plotadas a função de controle inicial, a função de controle final e a função de controle ótima.

ii) *Aproximação polinomial (quarta ordem)*

$$t_\omega = 2t/a_6 - 1$$

$$u = a_1 + a_2 t_\omega + a_3 t_\omega^2 + a_4 t_\omega^3 + a_5 t_\omega^4 \qquad (V.11)$$

onde,

$a_1, a_2, \ldots, a_5$ são os coeficientes do polinômio;

a_6 é o tempo final;

t_ω é o tempo normalizado (-1, 1).

Resultados (Tabela V.8):

O número de atualizações de x_{f_a} = 13

TABELA V.8

VALORES INICIAIS E FINAIS
(Aproximação polinomial)

PARÂMETROS	VALORES INICIAIS	VALORES FINAIS
a_1	2.5	2.946
a_2	2.5	5..715
a_3	0.	.6075
a_4	0.	-3.542
a_5	0.	-.6579
a_6	3.4	3.418

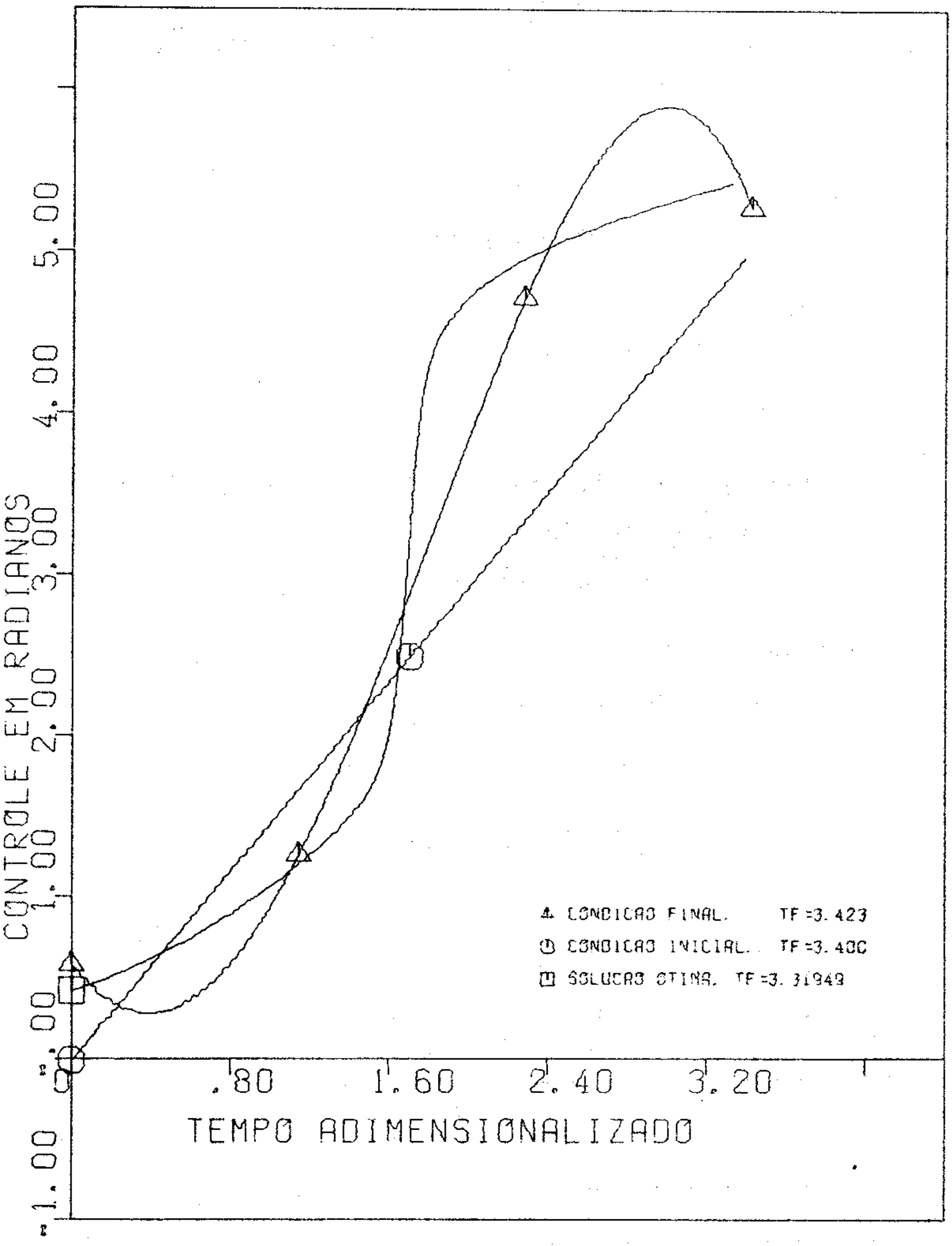

Fig. V.8 - Polinômio de Tchebyshev

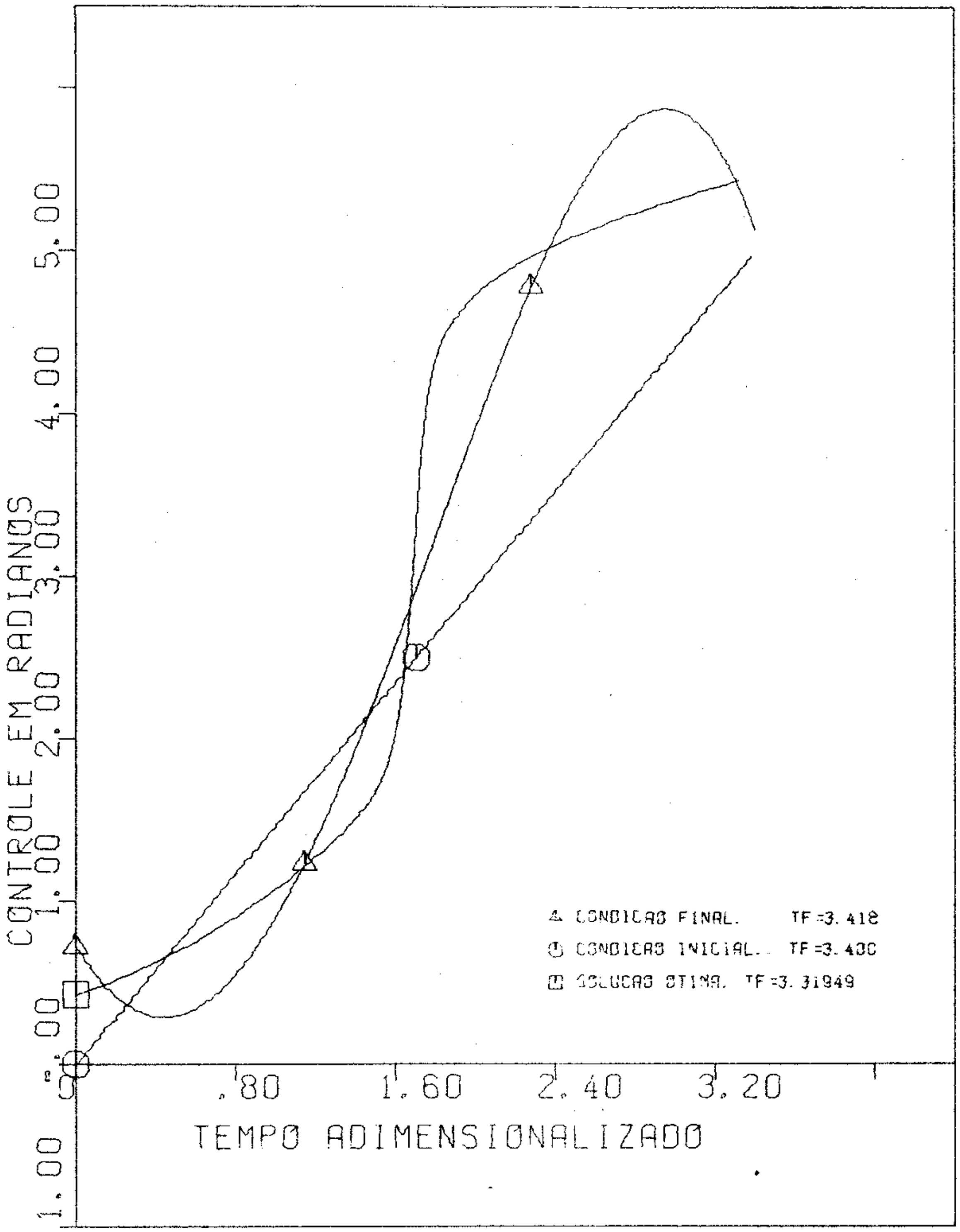

Fig. V.9 - Aproximação polinomial

Na Figura V.9, são plotadas a função de controle inicial, a função de controle final e a função de controle ótima.

5.4 - CONCLUSÕES

Os resultados obtidos nos vários testes apresentados neste capítulo e no Apêndice C, podem ser considerados muito bons. Imposto um erro da ordem de 1.E-04 para as equações de restrições, os desvios apresentados para o índice de desempenho, em muitos testes, foram menores do que (1%), o que, em muitas aplicações na engenharia, podem ser menores do que os erros de realização do controle pelo sistema, perdendo assim o seu significado como erro. Para a maioria dos testes, a convergência foi bastante rápida, vindo a confirmar as expectativas da fase de desenvolvimento .

Um outro aspecto que deve ser salientado, bastante discutido ao longo deste trabalho e confirmado pelos resultados obtidos, é a importância da escolha do funcional adequado para a representação do controle. O funcional adequado, como pode ser visto em considerações feitas neste capítulo, depende das características físicas do problema e de requisitos adicionais, como por exemplo, tempo de convergência, precisão de resultados, etc.

CAPÍTULO VI

CLASSIFICAÇÃO DO MÉTODO PROPOSTO

6.1 - INTRODUÇÃO

A existência de métodos de otimização de vetor de parâmetros, para problemas do tipo IV.1, já é justificada para aplicação na resolução de problemas do tipo problema II.8, problema II.9 e em problemas onde exista o interesse de especificação do controle na forma de um funcional previamente adotado.

Este capítulo visa classificar, quanto às suas características gerais, o método apresentado neste trabalho, entre outros existentes mais conhecidos. Pelo fato de ser muito difícil tirar conclusões definitivas (Tapley e Lewallen, 1967), e mesmo que fosse possível, estas teriam validade para um dado caso (problema, computador disponível, etc), são feitas comparações de ordem qualitativa entre o método apresentado neste trabalho e outros de otimização. São utilizados para comparações os métodos: da função de perturbação; do gradiente; de otimização de parâmetros de primeira ordem (Willianson, 1971); e de otimização de parâmetros de segunda ordem (Hull e Edgeman, 1975). Maiores detalhes sobre as características gerais do método da função de perturbação, do método do gradiente, bem como características gerais sobre uma grande quantidade de outros métodos ótimos, podem ser vistos no trabalho de Tapley e Lewallen, (1967).

As comparações são baseadas nas seguintes características:

. simplicidade de formulação e implementação;

. memória de computador requerida;

. tempo de convergência;

. confiabilidade e precisão da resposta.

6.2 - SIMPLICIDADE DE FORMULAÇÃO E IMPLEMENTAÇÃO

O método apresentado neste trabalho é de grande simplicidade de formulação e implementação, tornando possível inclusive programas para computador com características gerais, como é o caso do programa apresentado neste trabalho, capaz de resolver qualquer problema do tipo III. 1.

O método de otimização de parâmetros de primeira ordem, apresentado por Willianson (1971),é também bastante simples, embora requeira uma série de multiplicações e inversões de matrizes. Um ponto desfavorável no método de Willianson (1971), está na artificialidade da criação do problema associado, enquanto que o método apresentado neste trabalho gera o problema associado diretamente da forma mais natural possível.

Entre os métodos sub-ótimos discutidos neste capítulo, o método de Hull e Edgeman (1975), é o que necessita uma maior elaboração, por requerer derivadas parciais até a segunda ordem.

Os métodos do gradiente e da função de perturbação, também são métodos de formulação e implementação relativamente simples, embora exijam do usuário um maior conhecimento da teoria de otimização.

6.3 - MEMÓRIA DE COMPUTADOR

Entre os métodos ótimos, o método da função de perturbação está entre os que requerem menor memória de computador (principalmente no que se refere a armazenamento de dados), porque as equações do movimento e as equações perturbadas podem ser integradas simultaneamente num sentido único. O método do gradiente, em sua forma mais rápida, requer que a trajetória de referência e o controle de referência sejam armazenados para utilização no passo seguinte. Em uma forma que consome maior tempo de computação,pode ser utilizado o controle de referência armazenado, para tornar a gerar a trajetória de referência. De qualquer forma, o método do gradiente está entre os que requerem maior capacidade de computador.

Os métodos de otimização de parâmetros,em geral, requerem memória da mesma ordem de grandeza que o método de perturbação. Em particular, o método apresentado neste trabalho, além de consumir pouca memória para armazenamento de dados, torna possível programas compactos para computador, desde que se utilize algoritmos compactos para a integração numérica e para a resolução do problema linear.

6.4 - TEMPO DE CONVERGÊNCIA

A análise do tempo de convergência pode ser subdividida em análise do tempo por iteração e análise do número de iterações necessárias para a convergência.

. *Tempo por iteração*

O tempo de processamento por iteração, nos métodos de otimização de parâmetros, é função do número de parâmetros.

Os métodos de primeira ordem (método apresentado neste trabalho e o método apresentado por Willianson), requerem, para cálculo das derivadas e das variáveis de es tado finais, um tempo equivalente à integração de um sistema de r_1 (equação VI.1) equações lineares, enquanto que o método de segunda ordem (Hull e Edgeman, 1975) requer, para os mesmos fins, um tempo equivalente à integração de r_2 (equação VI.2) equações lineares.

Admitindo-se também um aumento proporcional no cálculo restante envolvido, o método de otimização de parâmetros de segunda ordem, requer um tempo, por iteração, da ordem de r_{12} (equação VI.3) vezes maior do que o tempo requerido, por iteração, pelos métodos de primeira ordem.

$$r_1 = gn + m \tag{VI.1}$$

$$r_2 = ng(g + 3)/2 + n \tag{VI.2}$$

$$r_{12} = (g(g + 3)/2 + 1)/(g + 1) \tag{VI.3}$$

onde,

r é o número de variáveis de estado;

g é o número de parâmetros a serem otimizados.

Entre os métodos ótimos estudados por Tapley e Lewallen (1967), o método da função de perturbação é o que apresenta o menor tempo de convergência, por iteração para resolver o problema V.1, enquanto que o método do gradiente, mesmo em sua forma mais rápida, está entre os mais lentos, e requer, por iteração, um tempo da ordem de uma e meia vezes o tempo requerido, por iteração, pelo método da função de perturbação.

Para o problema V.1 o método da função de perturbação, para derivação das variáveis de estado, em relação aos multiplicadores de Lagrange (λ10, λ20, λ30) e ao tempo final (t_f), e para cálculo dos valores finais dos multiplicadores de Lagrange (λ_{1f}, λ_{2f}, λ_{3f}) e das variáveis de estado (x_{1f}, x_{2f}, x_{3f}), requer o equipamento à integração de um sistema com trinta equações de primeira ordem.

Portanto, como as integrações numéricas consomem a principal parcela do tempo de processamento, e, admitindo-se equivalências de tempo nos demais cálculos, pode se concluir que, para resolução do problema V.1,o método da função de perturbação gasta, por iteração,o equivalente aos métodos de otimização de parâmetros de primeira ordem, utilizando uma aproximação com nove parâmetros.

. *Número de iterações*

Segundo Tapley e Lewallen (1967), para condições de partida com erros dados no tempo final e nos multiplicadores de Lagrange, o método da função de perturbação é o que requer menor número de iterações entre os métodos ótimos. Entretanto, a hipótese de erros dados nos multiplicadores de Lagrange é bastante tendenciosa em favor do método da função de perturbação, pois, pequenas alterações na função de controle (que não aumentaria o número de iterações, para um método cuja função de controle inicial fosse dada ponto a ponto), pode provocar grandes alterações nos valores dos multiplicadores de Lagrange (possivelmente aumentando em muito o número de iterações requeridas, quando utilizado o método da função de perturbação).

Para os casos dos métodos de otimização de parâmetros, tem-se algo semelhante. O número de iterações necessárias para a convergência está mais fortemente relacionado ao desvio do vetor de parâmetros iniciais em relação ao vetor de parâmetros ótimos, do que ao desvio na fun ção de controle propriamente dita. No entanto, dependendo da flexibilidade do funcional adotado, pode existir uma boa correspondência entre o desvio no vetor de parâmetros e o desvio nos valores do controle propriamente dito e, neste caso, a convergência é conseguida de forma mais natural e rápida. Como ilustração, podem ser vistos alguns casos (funcionais por trechos, controle com descontinuidade) deste gênero no Capítulo V, com resultados e número de iterações para a convergência bastante bons.

De maneira geral, o número de iterações para resolução de um problema de otimização depende: do método escolhido, da definição dos parâmetros auxiliares e da solução de partida. Acrescente-se a isso, para o caso de otimização de parâmetros, as características do funcional adotado para o controle: sensibilidade do controle em relação aos parâmetros, uniformidade dos parâmetros, capacidade do funcional em representar o controle, etc.

6.5 - PRECISÃO DE RESULTADOS

Com relação aos aspectos da precisão dos resultados, válidas as hipóteses impostas pela teoria dos métodos ótimos à solução do controle, e supondo-se que a solução tenha convergido totalmente, os procedimentos ótimos devem apresentar, a princípio, resultados mais precisos do que os métodos que supõem uma aproximação para o controle.

Entretanto, para escolha adequada de funcionais para o controle, os métodos de otimização de parâmetros podem apresentar resultados suficientemente precisos para a maioria das aplicações em engenharia, haja visto os resultados obtidos pelo método apresentado neste trabalho (Capítulo V).

6.6 - CONFIABILIDADE

Para qualquer método que se utilize, uma decisão final sobre a validade dos resultados - obtidos ou a serem obtidos (existem situações em que é necessário ter a certeza da validade dos resultados a serem obtidos), requer experiência do usuário, bem como um minucioso trabalho de análise, via resultados de problemas próximos resolvidos anteriormente, ou através de resultados obtidos sob várias condições iniciais. Para os métodos sub-ótimos, utilizando a aproximação do controle por um funcional, faz-se necessário acrescentar aos requisitos anteriores a escolha adequada do funcional que aproximará o controle.

6.7 - CONCLUSÕES

Resumindo o que foi discutido neste capítulo, o método apresentado neste trabalho possui as características a seguir:

- necessidade de pouca memória de computador;
- grande simplicidade de formulação e implementação;
- permite programas para computador para resolver problemas gerais;
- escolhido adequadamente o funcional para o controle, é bastante rápido e apresenta resultados de boa precisão.

CAPÍTULO VII

CONCLUSÕES

7.1 - INTRODUÇÃO

Os resultados e as discussões feitas ao longo deste trabalho foram bastante animadores em favor do procedimento proposto. Neste capítulo, sugerem-se aplicações do procedimento proposto, bem como desenvolvimentos futuros.

7.2 - APLICAÇÕES

Existem problemas que requerem necessariamente (por exemplo: problemas II.8 e II.9) a aplicação de métodos do gênero otimização de parâmetros. Entretanto, através da aproximação do controle por um funcional, outras aplicações se apresentam. E, em particular, pelas características do procedimento proposto é possível destacar de imediato as aplicações a seguir.

. *Aplicação em problemas em tempo real*

As características de rapidez e necessidade de pouca memória sugerem o método apresentado, candidato a aplicações em tempo real.

. *Geração de soluções de partida*

A possibilidade de criação de programas para computador, com capacidade de resolver problemas gerais, e as facilidades de implementação de dados de entrada sugerem a utilização do método para a busca de uma solução sub-ótima preliminar, a fim de servir como dados de entrada para um método mais refinado.

. *Resolução de problemas gerais*

Muitas vezes não se utiliza, em engenharia , os recursos de otimização, pela falta de pessoal especializado nesta área, ou falta de tempo na preparação de um programa de computador para resolver este problema.

A facilidade de formulação e de implementação na forma geral, além de capacidade de adaptação aos recursos computacionais, permitirá uma maior utilização do recurso de otimização no dia-dia da engenharia.

7.3 - DESENVOLVIMENTOS FUTUROS

Em várias situações no desenvolvimento do procedimento e do programa para computador, foi necessário optar por uma das alternativas existentes.

Podem ser destacadas as seguintes situações com alternativas:

. escolha (item 3.3) do procedimento associado ao problema II.4 ou escolha do procedimento associado ao problema II.7;

. derivação numérica direta ou derivação através da utilização das equações perturbadas (item 4.1);

. utilização do algoritmo do simplex ou algoritmo do simplex com multiplicadores, para resolver o problema de programação linear (item 4.1);

. associação de procedimentos (um para o início e outro para o final do processo de convergência).

Propõe-se, como próximo passo de desenvolvimento, a análise de conveniência de uma ou outra das alternativas descritas anteriormente. Numa fase seguinte, propõe-se a criação de programas gerais em várias versões, de acordo com as características de aplicação, tais como:

- versão compacta para aplicação em pequenos computadores;
- versão rápida para aplicação em tempo real;
- versão com características de praticidade para utilizações no dia-dia.

BIBLIOGRAFIA

BRYSON, A.E.; HO, Y.C. *Applied optimal control.* New York. John Wiley, 1975.

CITRON, S.J. *Elements of optimal control.* New York, Holt, 1969.

DANTZIG, G.B. *Linear programing and extensions.* 4. ed. Princeton, N.F., The Rand, 1968.

FEHLBERG, E. *Low-order classical Runge-Kutta formulas with Stepsize control and their application to some heat transfer problems.* Washington, D.C., NASA 1969 (NASA TR - R - 315).

FORSYTHE, G.E. *Generation and use orthogonal polynomials for data - fitting with a digital computer. Journal Society for Industrial and Applied Mathematics,* 5 (2): 74 88, Jun., 1957

FORSYTHE, G.E.; MALCOLM, M.A.; MOLER, G.B. *Computer methods for mathematical computations.* Englewood Cliffs, N.J., Prentice - Hall, 1977.

HULL, D.G.; EDGEMAN, L.J. *Suboptimal control using a second - order parameter - optimization method. Journal of Optimization and Aplications,* 17 (5/6): 481 - 491,Dec., 1975.

PENNINGTON, R.H. *Computer methods and numerical analysis.* 2. ed. London, Macmillan, 1971

TAPLEY, B.D.; LEWALLEN, J.M. *Comparison of several numerical optimization methods. Journal of Optimization and Applications*, 1 (1): 1 - 32, 1967.

RIOS NETO, A.; CEBALLOS, D.C. *Approximation by Polynomial arcs to generate suboptimal numerical solutions in control problems.* A ser publicado, 1979.

WILLIANSON, W.E. *Use of polynomial approximations to calculate suboptimal controls.* *AIAA Journal*, 9 (11): 2271-2273, nov. 1971.

APÊNDICE A

TÓPICOS DE PROGRAMAÇÃO LINEAR

1 - PROGRAMAÇÃO LINEAR E ALGORITMO DE SIMPLEX

1.1 - INTRODUÇÃO

O conteúdo deste apêndice foi retirado do livro de programação linear escrito por Dantzig (1968). Tem como objetivo servir de elemento de consulta rápida. Para maiores detalhes, sugere-se a consulta a livros de programação linear.

1.2 - PROBLEMA MATEMÁTICO CENTRAL

O problema de otimização de uma função linear, sujeito a restrições lineares, é chamado problema matemático central de programação linear.

1.3 - FORMA USUAL

Diz-se que, o problema matemático de programação linear está na forma usual, quando possue as características:

- problema de minimização;
- restrições de igualdade;
- variáveis sempre positivas.

É sempre possível, através de transformações convenientes, gerar a forma usual, a partir da forma central. Exemplo:

Minimizar:

$$z = x_1 + 2x_2$$

sujeito a:

$$x_1 + x_2 \geq 6 \qquad (A.1)$$

Passo A:

Introduzindo-se convenientemente a variável de folga $x'_3 (x'_3 \geq 0)$, no problema A.1, tem-se:

Minimizar:

$$z = x_1 + 2x_2$$

sujeito a:

$$x_1 + x_2 - x'_3 = 6 \qquad (A.2)$$

Passo B:

Substituindo-se, x_1 e x_2 por $(x'_1 - x''_1)$ e $(x'_2 - x''_2)$, respectivamente, tem-se o problema a seguir.

Minimizar:

$$z = (x'_1 - x''_1) + 2(x'_2 - x''_2)$$

sujeito a:

$$(x'_1 - x''_1) + (x'_2 - x''_2) - x'_3 = 6 \qquad (A.3)$$

onde,

$$x'_1 \geq 0, \quad x''_1 \geq 0, \quad x'_2 \geq 0, \quad x''_2 \geq 0, \quad x'_3 \geq 0$$

1.4 - FORMA CANÔNICA

Diz-se que, o sistema está na forma canônica se apresentar (m + 1) variáveis, x_1, x_2, ..., x_m e $(-z)$ distribuídas de tal modo que, zerando-se as $(n - m)$ variáveis restantes, o sistema fica diagonalizado.

Considere-se o sistema de equações a seguir.

$$\begin{array}{llllll}
x_1 & & + \bar{a}_{1m+1} x_{m+1} & + \dots + & \bar{a}_{1n} x_n & = \bar{b}_1 \\
\quad x_2 & & + \bar{a}_{2m+1} x_{m+1} & + \dots + & \bar{a}_{2n} x_n & = \bar{b}_2 \\
\quad\quad \ddots & & \quad \vdots & & \quad \vdots & \\
\quad\quad\quad x_m & & + \bar{a}_{mm+1} x_{m+1} & + \dots + & \bar{a}_{mn} x_n & = \bar{b}_m \\
\quad\quad\quad (-z) & & + \bar{c}_{m+1} x_{m+1} & + \dots + & \bar{c}_n x_n & = -\bar{z}_0
\end{array} \qquad (A.4)$$

Tem-se então:

$$z = \bar{z}_0; \quad x_1 = \bar{b}_1; \quad x_2 = \bar{b}_2, \; \dots, \; x_m = \bar{b}_m;$$

$$x_{m+1} = x_{m+2} = \dots = x_n = 0$$

Se,

$$\bar{b}_1 \geq 0, \quad \bar{b}_2 \geq 0, \; \dots, \; \bar{b}_m \geq 0$$

então, diz-se que a solução é uma solução básica viável e que o sistema está na forma canônica viável.

1.5 - TESTE DE OTIMALIDADE

Definição:

Os coeficientes, c_j, do sistema canônico viável, sistema A.4, são chamados fatores relativos de custo.

Teorema 1:

Uma solução básica viável é uma minimal viável com total custo z_0, se todos os fatores relativos de custo são negativos.

$$\bar{c}_j \geq 0 \qquad (j = 1, 2, \ldots, n)$$

Teorema 2:

Dada uma solução básica viável minimal com fatores de custo $\bar{c}_j \geq 0$, qualquer outra solução viável (não necessáriamente básica) com propriedade que $x_j = 0$ para todo $\bar{c}_j > 0$ é, também, uma solução minimal; se existir algum j onde $x_j > 0$ e $c_j > 0$, a solução não é minimal.

Corolário:

Uma solução básica viável é a única solução minimal viável, se $\bar{c}_j > 0$ para todas variáveis não básicas.

1.6 - AS DUAS FASES DO MÉTODO DO SIMPLEX

O algoritmo do simplex propriamente dito parte de uma forma canônica viável, e surge o problema da obtenção de uma forma canônica viável, que geralmente não é imediata.

Um procedimento possível, para obtenção da forma canônica viável, é transformar o sistema usual A.5 no sistema usual artificial A.6, através da introdução das variáveis artificiais x_{n+1}, x_{n+2}, ..., x_{n+m}. Resolve-se, então, o problema em duas fases: numa primeira fase resolve se o problema acrescido das variáveis artificiais, com o intuito de encontrar uma solução básica viável para o problema original, e, numa segunda fase, resolve-se o problema principal para a obtenção dos resultados finais.

$$\begin{aligned}
a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n &= b_1 \\
a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n &= b_2 \\
\dots\dots\dots\dots\dots\dots\dots\dots\dots \\
a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n &= b_m \\
c_1x_1 + c_2x_2 + \dots + c_nx_n &= z
\end{aligned} \qquad (A.5)$$

$$\begin{array}{lllll}
a_{11}x_1 + \dots a_{1n}x_n & + x_{n+1} & & & = b_1 \\
a_{21}x_1 + \dots a_{2n}x_n & & + x_{n+2} & & = b_2 \\
\vdots & & \vdots & & \vdots \\
a_{m1}x_1 + \dots + a_{mn}x_n & & & + x_{n+m} & = b_m \\
c_1x_1 + \dots + c_nx_n & & & -z & = 0 \\
d_1x_1 + \dots + d_nx_n & & & -w & = -w_o
\end{array} \qquad (A.6)$$

onde,

$$\begin{aligned}
d_j &= -(a_{1j} + a_{2j} + \dots + a_{mj}) \\
-w_o &= -(b_1 + b_2 + \dots + b_m)
\end{aligned} \qquad (A.7)$$

observe-se que,

$$w = x_{n+1} + \dots + x_{n+m}$$

Resolve-se a forma canônica, sistema A.6, e tem-se as seguintes conclusões no final da fase I:

a) Min $w > 0$, então, não existe solução viável no problema original;

b) Min $w = 0$, abandona-se as variáveis artificiais e a equação de variável w, e resolve-se, então,o problema principal (fase II), já na forma canônica viável.

1.7 - DETALHAMENTO DO ALGORITMO DE SIMPLEX

No que diz respeito ao algoritmo, a fase I e a fase II são idênticas e, basicamente, procura-se caminhar no sentido de obter $c_j \geq 0$, mantendo-se a forma canônica viável ($b_i \geq 0$).

Procedimento:

Passo I:

i) Se todos $d_j \geq 0$ (fase I) ou $c_j \geq 0$ (fase II), então,

a) *Fase I*

$w_o > 0$: termina-se o problema e, não existe nenhuma base viável.

$w_o = 0$, inicia-se a fase II.

b) *Fase II*

Termina-se o problema e, a solução ótima é:

$x_{ji} = \bar{b}_i$, $x_j = 0$ $z = \bar{z}_o$

$(j \neq j_i, \quad i = 1, 2, \ldots, m)$

ii) Se algum $d_j < 0$ (fase I) ou $\bar{c}_j < 0$ (fase II), escolhe-se x_s para entrar na base, no lugar de x_r (a ser determinado no passo II) a sair da base.

Fase I $\quad \bar{d}_s = \min d_j < 0$

Fase II $\quad \bar{c}_s = \min \bar{c}_j < 0$

Passo II:

i) Se todos $\bar{a}_{is} \leq 0$, termina-se o problema, pois a solução é ilimitada.

ii) Se algum $\bar{a}_{is} > 0$, escolhe-se r, tal que

$$\bar{b}_r/a_{rs} = \min \bar{b}_i/\bar{a}_{is} \qquad a_{is} > 0$$

Passo III:

Multiplica-se a linha r escolhida no (passo II) por $1/\bar{a}_{rs}$.

Soma-se sobre a linha r, a linha que vai sair multiplicada por um fator; sobre as outras linhas, soma-se a linha r multiplicada por fatores, de tal modo que o sistema volte à forma canônica.

Retorna-se ao passo I.

1.8 - EXISTÊNCIA DE FORMA CANÔNICA VIÁVEL

Quando não existe forma canônica viável, o problema pode ser:

a) *Redundante*

Existem linha que são combinações lineares de outra.

b) *Inconsistente*

Uma (ou mais) das equações fica impossível de ser satisfeita, dado os requisitos das demais.
Exemplo:

$x > 6$

$$x + y < 5, \quad y \geq 0 \qquad (A.8)$$

2 - MULTIPLICADORES DE LAGRANGE

Considere-se o problema de programação linear na forma usual ($x_j \geq 0$).

Minimizar:

$$c_1 x_1 + c_2 x_2 + \ldots + c_n x_n = z$$

sujeito a:

$$\begin{aligned} a_{11} x_1 + a_{12} x_2 + \ldots + a_{1n} x_n &= b_1 \\ a_{21} x_1 + a_{22} x_2 + \ldots + a_{2n} x_n &= b_2 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ a_{m1} x_1 + a_{m2} x_2 + \ldots + a_{mn} x_n &= b_m \qquad (A.9) \end{aligned}$$

É possível substituir as relações de não negatividade por:

$$x_j - u_j^2 = 0 \quad (j = 1, 2, \ldots, n) \qquad (A.10)$$

E considere-se a função com mínimo ilimitado a seguir.

$$\bar{z} = \left(\sum_{i=1}^{m} \pi_i b_i\right) + \left(c_1 - \sum_{i=1}^{m} \pi_i a_{i1} - \bar{c}_1\right) x_1 +$$

$$+ \ldots + \left(c_n - \sum_{i=1}^{m} \pi_i a_{in} - \bar{c}_n\right) x_n +$$

$$+ \bar{c}_1 u_1^2 + \bar{c}_2 u_2^2 + \ldots + \bar{c}_n u_n^2 \qquad \text{(A.11)}$$

A função $\bar{z}$ possue um mínimo limitado, somente se:

a) Os coeficientes de x_j se anularem;

b) Os coeficientes de u_j^2 forem não negativos. Ou então,

$$\bar{c}_j = c_j - [\pi_1 a_{1j} + \pi_2 a_{2j} + \ldots + \pi_m a_{mj}] \geq 0 \qquad \text{(A.12)}$$

Além disso, para um mínimo local, as derivadas parciais com respeito a u_j deverão anular-se, ou seja,

$$\bar{c}_j u_j = 0 \quad \text{ou} \quad \bar{c}_j x_j = 0 \quad (j = 1, 2, \ldots, n) \qquad \text{(A.13)}$$

logo,

$$z = \left(\sum_{i=1}^{m} \pi_i b_i\right)$$

$$\bar{c}_j = c_j - [\pi_1 a_{1j} + \pi_2 a_{2j} + \pi_n a_{nj}] \geq 0 \qquad . \qquad \text{(A.14)}$$

É fácil concluir que existem combinações de $(n - m)$ valores, $x_j = 0$ e m valores $\bar{c}_k = 0$, que satisfazem o sistema A.13 e, cada conjunto de m valores c_k nulos, determinam univocamente os valores π_i para os casos não degenerados, restando então determinar o conjunto π_i^* minimizante.

3 - MÉTODO DO SIMPLEX UTILIZANDO-SE MULTIPLICADORES

A idéia do simplex, usando-se multiplicadores, está no uso de uma série de números chamados multiplicadores e no fato de que a inversa da matriz de base é gerada diretamente da equação original (item A.2).

Basicamente, pode-se dizer que este método é vantajoso, quando trabalhamos com números elevados de coeficientes nulos, ou nos casos em que o número de restrições é muito menor do que o número de variáveis.

3.1 - ILUSTRAÇÃO DO MÉTODO

TABELA A.1

TABELA DO CICLO 0

CICLO 0 VARIÁVEIS BÁSICAS

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	-z	constante
1		1		1		1					3
	1		1		1		1				2
1			1					1			2
	1			1					1		1
-8	-9	-7	-6	-8	-9					1	-90

Base B

CICLO k

Base inicial

CICLO 0

TABELA A.2

TABELA DO CICLO k

CICLO k

variáveis básicas	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	$-z$	constante
x_1	1				1	-1		-1	1	1		1
x_2		1			1					1		1
x_3			1			1	1	1	-1	-1		2
x_4				1	-1	1		1		-1		1
$-z$					3	-4	7	5	1	4		-53

Inversa da Base

Multiplicadores $\pi = (-7,-5,-1,-4)$

Das Tabelas A.1 e A.2, definem-se:

$$B^* = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad B^{*^{-1}} = [B_{ij}] \quad \begin{bmatrix} 0 & -1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & -1 & -1 \\ 0 & 1 & 0 & -1 \end{bmatrix} \qquad (A.15)$$

Pode ser visto, através de transformações simples, que os a_{ij} para o ciclo k, podem ser calculados utilizando-se a matriz $[B_{ij}]$, na forma a seguir.

$$\bar{a}_{ij} = B_{i1}a_{1j} + B_{i2}a_{2j} + B_{i3}a_{3j} + B_{i4}a_{4j} \qquad (A.16)$$

onde,

a_{ij} são os coeficientes correspondentes no ciclo o.

E os multiplicadores (π_1, π_2, π_3, π_4) podem ser determinados univocamente, utilizando-se o sistema A.14. Isolando-se convenientemente os valores $\bar{c}_k$ nulos, tem-se o sistema de equações a seguir.

$$\pi_1 + \pi_3 = -8$$

$$\pi_2 + \pi_4 = -9$$

$$\pi_1 = -7$$

$$\pi_2 + \pi_3 = -6 \qquad (A.17)$$

Resolvendo-se o sistema A.17 e utilizando-se novamente o sistema A.14 para determinaros valores $\bar{c}_k$ restantes, têm-se:

Multiplicadores $\pi = (-7, -5, -1, -4)$

$$\bar{c}_j = c_j - (\pi_1 a_{1j} + \pi_2 a_{2j} + \pi_3 a_{3j} + \pi_4 a_{4j}) \qquad (A.18)$$

Por procedimento análogo ao do utilizado no algoritmo do simplex usual, observando-se a tabela correspondente ao ciclo k, conclue-se que a variável x_4 deve sair da base para entrar a variável x_6.

Passa-se, então, para o ciclo k + 1.

Observações:

i) Em um ciclo k, não é necessário calcular todos os coeficientes envolvidos;

ii) as novas constantes b podem ser calculadas como os $\bar{a}_{ij}$;

iii) o valor de z pode ser calculado utilizando-se os multiplicadores sobre as constantes b, tal como $\bar{c}_j$;

iv) cálculos necessários:

- multiplicadores
- $[B_{ij}]$
- $\bar{c}_j$, $\bar{z}$ (última iteração)
- $\bar{a}_{ji}$, quando x_j é o candidato a sair da base
- $\bar{b}_i$ associado a $\bar{a}_{ij}$ positivo.

APÊNDICE B

LISTAGEM DO PROGRAMA

PARA

COMPUTADOR

B6700/B7700 F O R T R A N C O M P I L A T I O N M A R K 2.8.060

```
C*****************************************************************
C   PROGRAMA DE COMPUTADOR PARA RESOLUCAO DE PROBLEMAS DE OTIMIZACAO
C EM SISTEMAS DINAMICOS COM O CONTROLE APROXIMADO POR U(A,T).
C*****************************************************************
C
C              DEFINICAO DAS VARIAVEIS DE ENTRADA
C
C    JOTA      OPCAO SOBRE QUAL TRECHO A CONSIDERAR NA FUNCTION F.
C    M         NUMERO DE EQUACOES DE RESTRICOES.
C    NCO       NUMERO DE PARAMETROS A SEREM OTIMIZADOS(OBS:INCLUSIVE O
C         TEMPO FINAL)
C    NEQN      NUMERO DE EQUACOES DINAMICAS.
C    NTPRT     NUMERO DE INTERVALOS DE INTEGRACAO PARA UMA IMPRESSAO.
C    DT        PASSO DE INTEGRACAO INICIAL.
C    RFINAL    CONSTANTE UTILIZADA NA SUBROUTINE DSV.(EX:RAIO FINAL)
C    AMP       CONSTANTE UTILIZADA NA FUNCTION F.(EX:FLUXO DE MASSA)
C    EMP       CONSTANTE UTILIZADA NA FUNCTION F.(EX:EMPUXO)
C    AMO       CONSTANTE UTILIZADA NA FUNCTION F.(EX:MASSA INICIAL)
C    DMI       CONSTANTE UTILIZADA NA FUNCTION F E NA SUBROUTINE DSV.
C         (EX:CONSTANTE DE FORCA)
C    Y(J)      VETOR DE CONDICOES INICIAIAS.
C    C(J)      VETOR DOS PESOS CJ.
C     ITM(J)    VIDE COMENTARIOS NA SUBROUTINE PRCP.
C    DGMAX      VALOR MAXIMO DOS INCREMENTOS
C    ALFA   TAXA DESEJADA DE CONVERGENCIA NAS RESTRICOES.
C    P(J)      VETOR DOS PARAMETROS A SEREM OTIMIZADOS - SAO LIDOS OS
C         VALORES INICIAIS.
C    RELERR(J),ABSERR(J)---VIDE SUBROUTINE INTEG.
C    DP(J)      INCREMENTO PARA CALCULO DIRETO DAS DERIVADAS.
C    EPS2    PARAMETRO PARA DEFINICAO DE PROXIMIDADE.
C    EPS3      VALOR MINIMO DO MAXIMO INCREMENTO NAS EQUACOES DE RESTRICAO
C       E NO INDICE DE PERFORMANCE
C    SAH,SAI     PARAMETROS ASSOCIADOS A ATUALIZACAO DE DERIVADAS
C
C
C

      EXTERNAL DSV,F
      LOGICAL DTFAIL,DTFIX
      DIMENSION AA(6,5),A(6,31),B(6),C(31),INDVB(4),VMOD(31),ME(4),
     *MF(4),X(3),Y(3),DX(3),RELERR(3),ABSERR(3),WORK(15),FI(3),
     *DV(4,4),DXA(3,15),XK(3)       ,DP(15)
     *,ITM(20),LOCVAR(31),A1(6,31),B1(6),C1(31)
      COMMON/EPX/EPS3
      COMMON/IWORKP/IWORK(14)
      COMMON/SASA/SAH,SAI
      COMMON /PP/P(15)
      COMMON/SERINT/TINIC,TOUT
      COMMON/CTEINT/KPROB,ITER,KTIPO,KATVAL,KVIAVE,KDADOS,KSAIDA,M,
     *IS,NEL,LIMITE,IFLAG,IMETH,NEQN,ITN,INT,NCO,NCO2,NCO2M1,NI,NCO1,
     *KONT1,NTPRT,M1,M2
      COMMON/CTREAL/ERRABS,ERPERC,TOLANG,Z,DT,T,BETA,PCBB,BETAG,
     *TMEDIO,ALFA,PCT,DGMAX,EPS,EPS2
     *,EMP,AMO,AMP ,RFINAL,GIP,DMI
      COMMON/RKCON / NREJ,NREJT,NSTP,DTFAIL,DTFIX,IER
      COMMON/COEF24/A0000
      COMMON/QUINA/JOTA
      READ(5,2)M,NEQN,NTPRT
```

```
      IWORK(1)=1
      DO55I=2,14
   55 IWORK(I)=IWORK(I-1)+NEQN
      IFLAG=1
      IMETH=1
      READ(5,1)DT,RFINAL,AMP,EMP
      EPS=.1E-09
      M2=M*2
      N=NEQN
      N1=N+1
      M1=M+1
      EPS=.1E-08
       ERRABS=-1.
      ERPERC=-1.
      BETA=-.01
      MM1=0
      MM2=6
      READ(5,1)AMQ,DMI,(Y(J),J=1,N)
      READ(5,2)(ITM(J),J=1,20)
    2 FORMAT(20I3)
      READ(5,1)DGMAX,ALFA,EPS3,EPS2,(RELERR(J),J=1,N),(ABSERR(J),J=1,N)
      READ(5,1)SAM,SAI
      READ(5,2)JOTA
      READ(5,2)NCO
      NCO1=NCO+1
      NCO2=NCO+NCO
      NCO2M1=NCO2+1
      READ(5,1)(P(J),J=1,NCO)
      READ(5,1)(DP(J),J=1,NCO)
      READ(5,1)(C(J),J=1,NCO2M1)
    1 FORMAT(6F10.4)
      WRITE(6,3)JOTA,M,NCO,NEQN,NTPRT

    3 FORMAT('1',T35,5('*'),'DADOS DE ENTRADA',5('*'),/,T20,
     *'APROXIMACAO DO TIPO=',I3,/,T20,'NUMERO DE EQUACOES DE RESTRICAO'
     *,I3,/,T20,'NUMERO DE PARAMETROS=',I3,/,T20,
     *'NUMERO DE EQUACOES DINAMICAS=',I3,/,T20,
     *'NUMERO DE PASSOS DE INTEGRACAO PARA CADA IMPRESSAO=',I3)
      WRITE(6,4)RFINAL,AMP,EMP,DT
    4 FORMAT(T20,'CONSTANTE RFINAL UTILIZADA NA SUBROTINA DSV=',F10.4,/,
     *T20,'CONSTANTE AMP UTILIZADA NA SUBROTINA DE DERIVADAS=',F10.4,/,
     *T20,'CONSTANTE EMP UTILIZADA NA SUBROTINA DE DERIVADAS=',F10.4,/,
     *T20,'PASSO INICIAL DE INTEGRACAO=',F10.4)
      WRITE(6,5)DGMAX,ALFA,BETA,EPS3,EPS2,(RELERR(J),J=1,N),
     *(ABSERR(J),J=1,N),(DP(J),J=1,NCO)
    5 FORMAT(T20,'PARAMETRO DE CONTROLE SOBRE O AVANCO=',F10.4,/,T20,
     *'AVANCO NAS RESTRICOES=',F10.4,/,T20,'AVANCO NO MERITO=',F10.4,
     */,T20,'EPS3=',E10.4,/,T20,'EPS2=',E10.4,/,T20,5('*'),
     *'ERROS NA SUBROTINA RUNGE KUTTA E INCREMENTOS PARA DERIVADAS',
     *5('*'),/,T10,5(10E10.4,/,T10))
      CALL PRCP(&77,AA,A,B,C,INDVB,VMOD,ME,MF,MM1,MM2,F,X,Y,DX,
     *RELERR,ABSERR,WORK,DSV,ITM,FI,DXA,DP,P,DV,NM1,XK,LUCVAR,
     *A1,B1,C1)
   77 CONTINUE
      STOP
      END
```

SE

```
      SUBROUTINE PREP(*,/AA/,/A/,/B/,/C/,/INDVB/,/VMOD/,/ME/,/MF/,
     *MM1,MM2,F,/X/,/Y/,/DX/,/RELERR/,/ABSERR/,/WORK/,DSV,/ITM/,/FI/,
     */DXA/,/DP/,/P/,/DV/,N,M7,/XK/,/LOCVAR/,/A1/,/B1/,/C1/)
      LOGICAL TM1,TM2,TM3,DTFIX,DTFAIL
      EXTERNAL DSV,F
      COMMON/RKCON /NREJ,NREJT,NSTP,DTFAIL,DTFIX,IER
      COMMON/SERINT/TINIC,TOUT
      COMMON/CTEINT/KPROB,ITER,KTIPO,KATVAL,KVIA,E,KDADOS,KSAIDA,M,
     *IS,NEL,LIMITE,IFLAG,IMETH,NEQN,ITN,INT,NCO,NCO2,NCO2M1,N1,NCO1,
     *KONT1,NTPRT,M1,M2
      COMMON/EPX/EPS3
      COMMON/CTREAL/ERRABS,ERPERC,TOLANG,Z,DT,T,BETA,PCBB,BETAG,
     *TMEDIO,ALFA,PCT,DGMAX,EPS,EPS2
     *,EMP,AMO,AMP ,RFINAL,GIP,DMI
      COMMON/SASA/SAM,SAI
      COMMON/QUINA/JOTA
      DIMENSION A(MM1,1),AA(MM2,1),B(1),C(1),INDVB(1),LOCVAR(1),ME(1),
     *MF(1),VMOD(1),X(1),Y(1),DX(1),RELERR(1),ABSERR(1),WORK(1),FI(1),
     *DV(MF,1),DXA(N,1),ITM(1),P(1),DP(1),XK(1)
     *,A1(MM1,1),B1(1),C1(1)
C                         DEFINICOES
C         DGMAX    MAXIMO VALOR DO MAIOR INCREMENTO DOS PARAMETROS.
C         DMMAZ    MAXIMO VALOR DA SOMATORIA ACUMULADA DESDE A ULTIMA
C         ATUALIZACAO DAS DERIVADAS DAS VARIAVEIS DE ESTADO EM RLACAO
C         AOS PARAMETROS A SEREM OTIMIZADOS.
C         INT    CONTADOR DAS ATUALIZACOES DAS DERIVADAS DAS VARIAVEIS
C         DE ESTADO EM RELACAO AOS PARAMETROS A SEREM OTIMIZADOS.
C         INTMAX    VALOR MAXIMO DE INT.
C         ITN    CONTADOR DE TRECHOS IMPRESSOS.(INT=CONSTANTE)
C         ITM(J)     VETOR QUE ARMAZENA  OS INT QUE DEVEM SER IMPRESSOS.
C         X(J)     VETOR DE VARIAVEIS DE ESTADO.
C         DV    MATRIZ DAS DERIVADAS DE IP E DAS EQUACOES DE RESTRICOES
C         EM RELACAO AS VARIAVEIS DE ESTADO.
C         DXA     MATRIZ DE DERIVADAS DAS VARIAVEIS DE ESTADO EM RELACAO
C         AOS PARAMETROS A SEREM OTIMIZADOS.
C         FI      VETOR DE DESVIO DAS EQUACOES DE RESTRICOES.
C         KTT      CONTADOR DE INTERACOES REJEITADAS .
C         DSM     SOMATORIA ACUMULADA DOS INCREMENTOS MAXIMOS.
C         FIXX     VALOR DO DESVIO DA RESTRICAO.
C         FIMX     VALOR MAXIMO DE FIXX.
C         FIXZ    SOMATORIA DE FIXX.
C         FIXZA    VALOR LIMITE PREVISTO DE FIXZ.
C         GIP      INDICE DE PERFORMANCE.
C  OS PARAMETROS EM COMMON OU EM ARGUMENTO DE SUBROTINA PODEM ESTAR
C    DEFINIDOS NO SUBPROGRAMA CONECTADO.
      FIXZAP=100.
      DMMAZ=SAM*DGMAX
      NEL=NCO2
      KPROB=1
      EPS4=EPS3/M/30.
      KVIAVE=0
      KATVAL=0
      LIMITE=NCO2*NCO2+NCO2+NCO2
      ITN=1
      INT=0
      INTMAX=40
      FIMX=100.
 1000 CONTINUE
      IF(INT.GT.INTMAX)GO TO 77
```

```
      IF(ABS(G(M1)).LT.EPS3.AND.FIMX.LT.EPS3)GOTO721
C FAZ A INTEGRACAO NUMERICA.
      DO44 JS=1,N
   44 X(JS)=Y(JS)
      IF(INT.NE.ITM(ITN))GO TO 10
      CALL IMPRE(F,X,RELERR,ABSERR,WORK,P)
      GO TO 191
   10 TINIC=0.
      TOUT=P(NCO)
      CALL INTEG(F,X,RELERR,ABSERR,WORK)
C              ***
  191 GIPA=GIP
      CALL DSV(DV,X,FI,M1,P)

C VERIFICA SE E A PRIMEIRA ITERACAO.
   40 IF(INT.EQ.0.AND.DSM.EQ.0.)GO TO 201
C              ***
C CALCULA OS DESVIOS DAS EQUACOES DE RESTRICOES.
      FIMX=0.
      FIXZ=0.
      DO 41 KJ=1,M
      FIXX=ABS(FI(KJ))
      FIXZ=FIXZ+FIXX
   41 FIMX=AMAX1(FIMX,FIXX)
C              ***
      KTT=KTT+1
      IF(KTT.LT.3)GOTO88
  223 KTT=0
      FIXZAP=FIXZ
C         VERIFICA SE DXA DEVE SER ATUALIZADA.
  301 IF(DSM.LT.DMMAX)GOTO222
C              ***
      IF(INT.EQ.ITM(ITN))ITN=ITN+1
       INT=INT+1
C CALCULA AS DERIVADAS DAS VARIAVEIS DE ESTADO EM RELACAO AOS PARAMETROS
  201 DO 19 J=1,NCO
      P(J)=P(J)+DP(J)
      IF(J.NE.NCO.OR.JOTA.NE.1)GOTO55
      TINIC=TOUT
      DO 33 JJ=1,N
   33 XK(JJ)=X(JJ)
      GO TO 333
   55 CONTINUE
      DO 919 JJ=1,N
  919 XK(JJ)=Y(JJ)
      TINIC=0.
  333 CONTINUE
      TOUT=P(NCO)
      CALL INTEG(F,XK,RELERR,ABSERR,WORK)
      DO 23 I=1,N
   23 DXA(I,J)=(XK(I)-X(I))/DP(J)
   19 P(J)=P(J)-DP(J)
      DSM=0.
C              ***
C * * * * * * * * * * * * * * * * * *
C PREPARA A ENTRADA  NA SUBROUTINA YH0218
  222 CONTINUE
      DO 104 I1=1,M1
      DO 103 I2=1,NCO
      A(I1,I2)=0.
      DO 105 JK=1,N
```

```
  105 A(I1,I2)=A(I1,I2)+DA(JK,I2)*DV(I1,JK)
  103 A(I1,I2+NCJ)=-A(I1,I2)
      A(I1,NCJ)=A(I1,NCJ)+DV(I1,N1)
  104 A(I1,NCJ2  )=-A(I1,NCJ)
      A(M1,NCJ2M1)=-1.
C  *    *   *   *   *   *    *   *   *   *   *   *   *
  303 DO 102 KJ=1,M
  102 B(KJ)=-ALFA*FI(KJ)
      B(M1)=BETA*(ABS(GI-)+1.)
      IF( INT.NE.ITV(ITN).OR.DSM.NE.0.)GO TO 6
      WRITE(6,399)INT
  399 FORMAT('1',T10,10('*'),'ATUALIZACAO NUMERO=',I3,
     *' DA MATRIZ DE DERIVADAS(DXA).',10('*'))
      KJADOS=1
      KSAIDA=2
      GO TO 78
    6 KJADOS=3
      KSAIDA=3
   78 DO 415 J=1,NCO2M1
      DO 414 KJ=1,M1
      A1(KJ,J)=A(KJ,J)
  414 B1(KJ)=B(KJ)
  415 C1(J)=C(J)
      CALL YH0216(A1,B1,C1,AA,INDVB,LOCVAR,ME,MF,MM1,MM2,VMOD)
C  *    *   *   *   *   *   *   *   *   *   *   *   *
C   FAZ COMPARACAO E ALTERA ATITUDE
      IF(KTIPO)710,720,730
  710 WRITE(6,71)
   71 FORMAT(1X,30("*"),"O LIMITE DE INTERACOES PARA RESOLUCAO DO PROBLE
     *MA LINEAR FOI EXCEDIDO")
      GO TO 730
  712 WRITE(6,72)ITER,INT
   72 FORMAT(1X,30("*"),"INTERACOES NO PROBLEMA LINEAR =",I4,/,
     *1X,30("*"),"INTERACAO NUMERO=",I3,"DO PROBLEMA")
      GOTO29
  730 IF(DSM.NE.0.)GOTO201
   11 WRITE(6,73)
   73 FORMAT(1X,("*"),"NA RESOLUCAO DO PROBLEMA LINEAR NAO FOI POSSIVEL
     *  TORNAR O PROBLEMA VIAVEL")
      GO TO 712
C  *   *   CONDICAO NORMAL
  720 DMAX=0.
C   CALCULA O INCREMENTO MAXIMO.
      DO 718 KJ=1,M1
      IF(INDVB(KJ).NE.NCO2M1)DMAX=AMAX1(ABS(AA(KJ,M2)),DMAX)
C   ********
  718 CONTINUE
      PCTB=1.
      IF(DMAX.LE.DGMAX)GOTO1101
      PCTB=DGMAX/DMAX
      DO 400 JB=1,M1
  400 AA(JB,M2)=PCTB*AA(JB,M2)
      DMAX=DMAX*PCTB
C  *    *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
 1101 CALLSOME(1., P , AA , INDVB ,MM1)
      DSM=DSM+DMAX
      DMMAX=AMIN1(DMMAZ,SAI*DMAX)
      GO TO 1000
C XXX CALCULA OS PARAMETROS BETA,C(NCO2M1)***OU C2G+1,EPS2***OU EPSON*P.
   88 KP=0
      FIXZA=FIXZAP*(1.-ALFA*PCTR**.5)
```

```
      DO 1 J=1,M1
      IF(INDVB(J).EQ.NCO2M1)KP=KP+NCO2M1
      IF(INDVB(J).EQ.NCO.AND.FIXZ.LT.EPS2.AND.EPS2.GT.EPS4)EPS2=.5*EPS2
      IF(INDVB(J).EQ.NCO2)KP=KP+NCO2
    1 CONTINUE
      IF(KP.NE.NCO2)GO TO 444
C 111 IP DIMINUIU O PREVISTO 111.
      IF(FIXZ.GT.FIXZA.AND.FIXZ.GT.EPS2)GOTO233
      BETA=BETA*1.1
      GO TO223
  233 C(NCO2M1)=.6*C(NCO2M1)
      DSM=DSM-DMAX
      CALL SOME(-1.,P,AA,INDVB,MM1)
      B(M1)=B(M1)*.5
      BETA=BETA*.8
      GOTO8
C              111
C 222 IP NAO DIMINUIU O PREVISTO 222.
  444 IF(FIXZ.GT.FIXZA)GOTO224
      IF(FIXZ.LT.EPS2)GOTO555
      C(NCO2M1)=1.2*C(NCO2M1)
      BETA=BETA*.8
        GOTO223
  555 C(NCO2M1)=2.5*C(NCO2M1)
      GO TO 223
  224 BETA=BETA*.8
      C(NCO2M1)=.9*C(NCO2M1)
      CALL SOME(-1.,P,AA,INDVB,MM1)
      DSM=DSM-DMAX
      B(M1)=B(M1)*.5
      GOTO8
C      222
C        XXXXXXXXXXXXXXXX
   79 CONTINUE
   77 WRITE(6,39)
   39 FORMAT(T10,'FOI ULTRAPASSADO ON. PREVISTO DE ATUALIZACOES DAS DERI
     *VADAS DAS VARIAVEIS DE ESTADO EM RELACAO AOS PARAMETROS')
      GOTO29
  721 WRITE(6,59)
   59 FORMAT('1',T25,'SAIDA NORMAL,A CONVERGENCIA TORNOU-SE LENTA')
   29 WRITE(6,89)INT
   89 FORMAT(T25,'IMPRESSAO DE RESULTADOS FINAIS *** ATUALIZACAO NUMERO
     *=',I3)
      DO 49 JS=1,N
   49 X(JS)=Y(JS)
      CALL IMPRE(F,X,RELERR,ABSERR,WORK,P)
      WRITE(6,577)EPS2,C(NCO2M1),DMAX,BETA
  577 FORMAT(T10,5('*'),'VALORES FINAIS DOS PARAMETROS QUE CONTROLAM A
     *CONVERGENCIA',5('*'),/,T30,'EPS2=',E10.4,/,T30,'C2G+1',E10.4,/,
     * T30,'INCREMENTO MAXIMO NOS PARAMETROS OTIMIZAVEIS NESTA ITERACAO=
     *',E10.4,/,T30,'BETA=',E10.4)
      RETURN
      END
```

SE

```
      SUBROUTINE F(T,X,DX,IER)
      COMMON/CTEINT/KERROR,ITER,KTIPO,KATVAL,KVIAVE,KDADOS,KSAIDA,M,
     *IS,NEL,LIMITE,IFLAG,IMETM,NEQ,ITN,INT,NCO,NCO2,NCO2M1,NI,NCO1,
     *KONTI,NIPRT
      COMMON/CTREAL/ERRABS,ERPERC,TOLANG,Z,DT,TZ,BETA,PC4D,BETAG,
     *IMEDIO,ALFA,PCT,DMMAX,EPS,EPS2
     *,EMP,AMO,AMP ,RFINAL,GIP,OMI
      COMMON /PP/P(15)
      COMMON/QUINA/JOTA
C  UTILIZE U COMO SENDO O CONTROLE PRINCIPAL
      COMMON/CONTRL/U
      DIMENSION X(1),DX(1)
      GOTO(1,3,5,7,15,25,27)JOTA
    1 CONTINUE
C      CONDICAO NORMAL NENHUMA QUINA
      TW=2.*T/P(NCO)-1.
      U=0.
      TU0=1.
      TU1=TW
      U=P(1)+TW*P(2)
      DO2J=3,NCO-1
      TU2=-TU0+2.*TW*TU1
      U=U+P(J)*TU2
      TU0=TU1
    2 TU1=TU2
    9 DX(1)=X(2)
      X3X=X(3)/X(1)
      AMTP=EMP/(AMO-AMP*T)
      DX(2)=X(3)*X3X-OMI/(X(1)*X(1))+AMTP*SIN(U)
      DX(3)=-X(2)*X3X+AMTP*COS(U)
      RETURN
    3 CONTINUE
C    UMA QUINA NA METADE    DO INTERVALO  DE TEMPO  REQUER NCO=7
      TW=4.*T/P(NCO)-1.
      IF(TW.GT.1.)GO TO8
      U=P(1)+P(2)*TW+P(3)*(TW*TW*2.-1.)
      GOTO9
    8 CONTINUE
      TW=TW-2.
      U=P(4)+P(5)*TW+P(6)*(TW*TW*2.-1.)
      GOTO9
    5 CONTINUE
C     UMA QUINA NO TRECHO        CONTROLE CONTINUO
C REQUER NCO=7
      IF(T.GT.P(4))GOTO10
      TW=T/P(4)-1.
      U=P(1)+P(2)*TW+P(3)*TW*TW
      GOTO9
   10 TW=(T-P(4))/(P(NCO)-P(4))
      U=P(1)+P(5)*TW+P(6)*TW*TW
      GOTO9
    7 CONTINUE
C    FAZ IMPOSICAO DE SIMETRIA REQUER NCO=4
      TW=T/P(NCO)*2.
      TW5=TW-1.
      IF(TW.GT.1.)GOTO11
      U=P(3)+TW5*TW5*P(2)+TW5*P(1)
      GOTO9
   11 U=P(3)-TW5*TW5*P(2)+TW5*P(1)
      GOTO9
```

```
C   CASO COM ENE SEGMENTOS LINEARES     REQUER NCO=ENE+2
    15 DINT=P(NCO)/(NCO-2)
       TDINT=T/DINT
       JI=(TDINT+1.)
       IF(JI.EQ.(NCO-1))JI=NCO-2
       TW=TDINT-(JI-1)
       U=P(JI)+(P(JI+1)-P(JI))*TW
       GOTOY
    25 CONTINUE
C    CONTROLE DESCONTINUO  - POSICAO DA DESCONTINUIDADE OTIMIZADA.
C    REQUER  NCO=8
       IF(T.GT.P(7))GOTO35
       TW=T/P(7)
       U=P(1)+P(2)*TW+P(3)*TW*TW
       GOTOY
    35 TW=(T-P(7))/(P(NCO)-P(7))
       U=P(4)+P(5)*TW+P(6)*TW*TW
       GOTOY
    27 TW=2.*T/P(NCO)-1.
       U=P(NCO-1)
       DO37IJ=2,NCO-1
       U=U*TW+P(NCO-IJ)
    37 CONTINUE
       GOTOY
       END
```

SE

```
      SUBROUTINE SOME(AL,/P/,/AA/,/INDVB/,MM2)
      DIMENSION AA(MM2,1),P(1),INDVB(1)
      COMMON/CTEINT/KPROB,ITER,KTIPO,KATVAL,KVIAVE,KDADOS,KSAIDA,M,
     *IS,NEL,LIMITE,IFLAG,IMETH,NEQN,ITN,INT,NCO,NCO2,NCO2M1,N1,NCO1,
     *KONT1,NTPRT,M1,M2
      COMMON/CTREAL/ERRABS,ERPERC,TOLANG,Z,DT,T,BETA,PCBB,BETAG,
     *TMEDIO,ALFA,PCT,DMMAX,EPS,EPS2
     *,EMP,AMO,AMP ,RFINAL,GIP,DMI
      DO 5 KJ=1,M1
      IND=INDVB(KJ)
      IF(IND.GT.NCO)GO TO 3
      P(IND)=P(IND)+AL*AA(KJ,M2)
      GO TO 5
    3 IND=IND-NCO
      IF(IND.GE.NCO1)GO TO 5
      P(IND)=P(IND)-AL*AA(KJ,M2)
    5 CONTINUE
      RETURN
      END
```

SE

```
      SUBROUTINE IMPRE(F,/X/,/RELERR/,/ABSERR/,/WORK/,/P/)
      EXTERNAL F
      DIMENSIUN X(1),RELERR(1),ABSERR(1),WORK(1),P(1)
      COMMON/CTEINT/KPROB,ITER,KTIPO,KATVAL,KVIAVE,KDADOS,KSAIDA,M,
     *IS,NEL,LIMITE,IFLAG,IMETH,NEQN,ITN,INT,NCO,NCO2,NCO2M1,N1,NCO1,
     *KONT1,NTPRT,M1
      COMMON/SERINT/TINIC,TOUT
      COMMON/CTREAL/ERRABS,ERPERC,TOLANG,Z,DT,T,BETA,PCBB,BETAG,
     *TMEDIO,ALFA,PCT,OMMAX,EPS,EPS2
     *,EMP,AMO,AMP ,RFINAL,GIP,OMI
C SOMENTE IMPRIME O CONTROLE PRINCIPAL
      COMMON/CONTRL/U
      N=N1-1
      TFIN=P(NCO)
      TOUT=0.
      WRITE(6,1)(("X",I),I=1,N)
    1 FORMAT(T5,'TEMPO',T12,'CONTROLE',T22,20(7X,A1,I2))
      DTOUT=TFIN/NTPRT
      WRITE(6,6)TOUT,  (X(J),J=1,N)
    6 FORMAT(T5,E10.4,10('*'),20F10.4)
    3 FORMAT(T5,20E10.4)
    2 TINIC=TOUT
      TOUT=TOUT+DTOUT
      CALL INTEG(F,X,RELERR,ABSERR,WORK)
      WRITE(6,3)TOUT,U,(X(J),J=1,N)
      IF(ABS(TOUT-TFIN).LE.EPS)GO TO 5
      GOTO2
      RETURN
    5 WRITE(6,4)((J,P(J)),J=1,NCO)
    4 FORMAT(3(T10,8("P(",I2,")",E10.4),/))
      END
```

```
      SUBROUTINE DSV(/DV/,/X/,/FI/,M1,/P/)
      DIMENSION DV(M1,1),X(1),FI(1),P(1)
      COMMON/CTEINT/KPROB,ITER,KTIPO,KATVAL,KVIAVE,KDADOS,KSAIDA,M,
     *IS,NEL,LIMITE,IFLAG,IMETH,NEQN,ITN,INT,NCO,NCO2,NCO2M1,N1,NCO1,
     *KONT1,NIPRT
      COMMON/CTREAL/ERRABS,ERPERC,TOLANG,Z,DT,T,BETA,HCBH,BETAG,
     *THEDIO,ALFA,PCT,DMMAX,EPS,EPS2
     *,EMP,AMG,AMP ,RFINAL,GIP,DMI
      GIP=P(NCO)
      FI(1)=X(2)
      FI(2)=X(3)-SQRT(DMI/X(1))
      FI(3)=X(1)-RFINAL
      DV(1,2)=1.
      DV(2,1)= .5*SQRT(DMI/X(1))/X(1)
      DV(2,3)=1.
      DV(4,4)=1.
      DV(3,1)=1.
      RETURN
      END
```

SE

```
C---------------------------------------------------------------------
C
C     SUB-ROTINA YH0218
C
C     FINALIDADE -
C                    MAXIMIZAR OU MINIZAR
C                                             C * X
C                    SUJEITO AS RESTRICOES
C                                             X .GE. ZERO
C                                          E  A * X .EQ. B
C
C     OBSERVACAO -
C        ADMITE-SE QUE O PROBLEMA JA ESTEJA NA FORMA CANONICA, OU SEJA,
C        AS RESTRICOES EM FORMA DE IGUALDADE E VARIAVEIS NAO-NEGATIVAS.
C           CASO CONTRARIO TRANSFORME-O PRIMEIRAMENTE PARA A FORMA
C        ACIMA MENCIONADO ANTES DE CHAMAR ESTA SUB-ROTINA. ISTO E CONSE-
C        GUIDO CHAMANDO A "SUB-ROTINA YH0217".
C           CASO HAJA INTERESSE NOS VALORES DAS VARIAVEIS ORIGINAIS NO
C        PONTO DE OTIMO, DEVEMOS CHAMAR A "SUB-ROTINA YH0219" LOGO APOS
C        A CHAMADA DA "YH0218".
C
C     REFERENCIA -
C                    METODO - O SIMPLEX USANDO MULTIPLICADORES . GEORGE B.
C        DANTZIG . CAPITULOS 5 E 9. PARAGRAFOS 6.5 E 9.5 .
C
C     USO: CALL YH0218(A,B,C,AA,INDVB,LOCVAR,ME,MF,KPROB,ERRABS,ERPERC,
C    *  TOLANG,Z,ITER,KTIPO,MM1,MM2,KTVAL,KVIAVE,KDADOS,KSAIDA,
C    *  M,N,S,NEL,VMOD,LIMITE)
C
C     DESCRICAO DOS PARAMETROS
C        A       - MATRIZ DAS RESTRICOES - DIM(M,N)  - VEJA **
C        B       - VETOR COM TERMOS CONTANTES -DIM(N)  - VEJA **
C        C       - VETOR OBJETIVO -DIM(M)  - VEJA **
C        AA      - MATRIZ DE DIM(M+2,M+1) . CONTEM NO FINAL DO CICLO K -
C                  O INVERSO DA MATRIZ FORMADA PELOS VETORES COLUNAS
C                  DE "A" CORESPONDENDO AS VAR. QUE ESTAO NA BASE NAS
C                  "M" PRIMEIRAS LINHAS E COLUNAS
C                  MULTIPLICADOR "-SIGMA" E "-W" - LINHA M+2
C                  MULTIPLICADOR "-PI" E "-Z" - LINHA M+1
C        INDVB   - VETOR INDICADOR DAS VARIAVEIS BASICAS - DIM(M)-
C        LOCVAR  - VETOR COM A LOCALIZACAO DAS VARIAVEIS - DIM(N)
C                          = 0 - NA BASE
C                          = 1 - FORA DA BASE
C        ME      - VETOR DE TRABALHO - DIM(M)
C        MF      - VETOR DE TRABALHO - DIM(M)
C        P       - VETOR DE TRABALHO
C        KPROB   - TIPO DE OTIMZACAO
C                          =1  PROBLEMA DE MINIMO
C                          =2  PROBLEMA DE MAXIMO
C        ERRABS  - REFERENCIA DE ZERO PARA VALORES NAO RELATIVADOS.
C                  SE  "ERRABS.LE.0"  ENTAO  "ERRABS=1.E-7" .
C        ERPERC  - REFERENCIA DE ZERO PARA VALORES RELATIVADOS.
C                  SE "ERPERC.LE.0." ENTAO  "ERPERC=1.E-05" .
C        TOLANG  - TOLENCIA PARA O ANGULO DE DOIS VETORES MEDIDO PELO
C                  COSSENO DA TOLERANCIA PERMITIDA.
C                  SE  "TOLANG.LE.0."  ENTAO  "TOLANG"=1.E-05" .
C        Z       - VALOR DA FUNCAO OBJETIVO
C        KTIPO   - VIABILIDADE DO PROBLEMA
C                          =-1 O N. LIMITE DE ITER. FOI ULTRAPASSADO.
C                          =0 VIAVEL LIMITADO
C                          =1 ILIMITADO
```

```
C                           =2 INVIAVEL
C       ITER   - NUMERO DE ITERACOES
C       MM1    - N. DE LINHAS DA MATRIZ ASSOC. A MAT "A" - DIM. NO PP.
C       MM2    - N. DE LINHAS DA MATRIZ ASSOC. A MAT "AA"- DIM. NO PP
C       KATVAL - INDICA DE QUANTAS EM QUANTAS ITERACOES SAO EFETUADAS
C                ATUALIZACOES DA INVERSA DA MATRIZ BASICA
C               SE "KATVAL.LE.0" ENTAO KATVAL=2*M .
C       KVIAVE - INDICADA SE DESEJAMOS APENAS FASE 1 OU AMBAS
C                           =1 DESEJA UMA SOLUCAO BASICA VIAVEL
C                        .NE.1 DESEJA  A  SOLUCAO BASICA OTIMA
C       KDADOS - OPCAO DE SIADA DOS DADOS DO PROBLEMA
C                     =1 - OS DADOS SAO LISTADOS
C       KSAIDA - OPCAO DE SAIDAS DA SUBROTINA "REGMAG" .
C                           .LE.0 TABLEAU COMPLETO A CADA ITERACAO
C                           =1 SOLUCAO VIAVEL E CUSTOS DOS FATORES POR
C                              ITERACAO
C                           =2 SOLUCAO OTIMA
C                           .GE.3 APENAS TRANSFERENCIA POR MEIO DOS
C                              PARAMETROS FRMAIS.
C       M      - N. DE RESTRICOES DO PROBLEMA
C       N      - N. DE VARIAVEIS DO PROBLEMA
C       S      - INDICA A ULTIMA VARIAVEL QUE ENTROU OU QUE E CANDIDA-
C                TA A ENTRAR NA BASE
C       NEL    - NUMERO DE ELEMENTOS DAS MATRIZES IMPRESSOS POR LINHA.
C       VMOD   - VETOR DE TRABALHO. DIM(N)
C                SE  "LIMITE.LE.0" E ESTIPULADO O VALOR
C       LIMITE - NUMERO LIMITE PARA O NUMERO DE ITERACOES
C                     MINIMO(M,N)**2+M+N LIMITADO SUPERIORMENTE EM 500
C
C       **   -   NA REALIDADE, COMO A, B E C SAO USADOS COMO AREA DE
C                TRABALHO NA SUBROTINA "REGMAG", SUAS DIMENSOES DEVEM
C                SER         A - DIM(M+2,N+1)
C                            B - DIM(M+2)
C                            C - DIM(N+1)
C
C    *** IMPORTANTE ***
C             ESTA SUB-ROTINA CHAMA AS SUB-ROTINAS
C    ** YF1001 ** ** YF1008 ** ** YF1220 **
C
C    REFERENCIA
C       VEJA SUB-ROTINA REGMAG, ALPROG, VOLUME 2, P.185-190 210-217.
C
C-------------------------------------------------------------------------
                                                                        FO
                                                                        FO
      SUBROUTINE YH0218(/A/,/B/,/C/,/AA/,/INDVB/,/LOCVAR/,  /ME/,/MF/,
     *MM1,MM2,/VMOD/)
      INTEGER R,S
      COMMON/CTEINT/KPROB,ITER,KTIPO,KATVAL,KVIAVE,KDADOS,KSAIDA,MX,
     *IS,NEL,LIMITE,IFLAG,IMETH,NEQN,ITN,INT,NCO,NCO2,NCO2M1,N7,NCO1,
     *KONT1,NTPRT,MR
      COMMON/CTREAL/ERRABS,ERPERC,TOLANG,Z,DT,T,BETA,PCBB,BETAG,
     *TMEDIO,ALFA,PCT,DMMAX,EPS,EPS2
      EQUIVALENCE(N,NCO2M1),(M,MR),(IS,S)
      DIMENSION A(MM1,1),AA(MM2,1),B(1),C(1),INDVB(1),LOCVAR(1),ME(1),
     *MF(1),            FINLAO(4),VMOD(1)
      DATA FINLAO/6HMINIMI,6HZAR    ,6HMAXIMI,6HZAR    /
      MODPI=0
      IF(KATVAL.LE.0)KATVAL=2*M
      IF(ERPERC.LE.0.)ERPERC=1.E-05
```

```
      IF(ERRABS.LE.0.)ERRABS=1.E-07
      IF(TOLANG.LE.0.)TOLANG=1.E-05
      IF(LIMITE.GT.0) GO TO 10
      LIMITE=MINO(M,N)**2+M+N
      IF(LIMITE.GT.500) LIMITE=500
   10 IF(KSAIDA.GE.3) GO TO 33
      KI=1
      IF(KPROB.EQ.2) KI=3
      KF=KI+1
      WRITE(6,2010)(FINLAO(I),I=KI,KF)
 2010 FORMAT(2X,2A6,/2X,9(1H-), /12X, 5HC * X /2X,10HSUJEITOS A,
     * /2X,10(1H-),/12X,9HA * X = B,/
     *16X,1HE,/12X,11HX .GE. ZERO,/)
   20 IF(KDADOS.EQ.1)GO TO 21
      IF(KSAIDA.EQ.2)GO TO 33
   21 WRITE(6,2020)
 2020 FORMAT( 2X,"MATRIZ A",/2X,8(1H-))
      CALL YF1220(A,M,1,N,B,ME,5,MM1,NEL)
      WRITE(6,2030)
 2030 FORMAT(  /2X, 7HVETOR C,/2X,7(1H-))
 2050 FORMAT(5X,8E14.6)
      WRITE(6,2050) (C(I),I=1,N)
      WRITE(6,2040)
 2040 FORMAT(  /2X, 7HVETOR B,/2X,7(1H-))
      WRITE(6,2050) (B(I),I=1,M)
   33 IF(KPROB.EQ.1) GO TO 38
      IF(KSAIDA.LE.1) WRITE(6,2043)
 2043 FORMAT( /2X,20HPROBLEMA EQUIVALENTE,/2X,20(1H-), /2X,9HMINIMIZAR,/
     1 2X,9(1H-), /12X, 7H- C * X, /2X,10HSUJEITOS A,/2X,10(1H-),/12X,
     2 9HA * X = B,/16X,1HE,/12X,11HX .GE. ZERO,//2X,
     3 31HMAXIMO C * X = - MINIMO - C * X,/2X,31(1H-),)
      DO 35 I=1,N
   35 C(I)=-C(I)
   38 N1=N+1
      M1=M+1
      M2=M+2
C     VERIFICACAO DA DISPERSAO DOS DADOS DA MATRIZ DAS RESTRICOES.
C
C       - VERIFICACAO DAS MAGNITUDES DOS COEFICIENTES DAS RESTRICOES.
C
   37 DO 40 I=1,M
      XMIN=1.E+60
      XMAX=1.E-40
      DO 39 J=1,N
      ESC=ABS(A(I,J))
      IF(ESC.LT.XMIN.AND.ESC.GT.0.)XMIN=ESC
      IF(ESC.GT.XMAX) XMAX=ESC
   39 CONTINUE
      ESC=XMAX/XMIN
      IF(ESC.GT.1.E05) WRITE(6,2047)I ,XMAX,XMIN
 2047 FORMAT(/5X,96(1H*),//5X,28HOS COEFICIENTES DA RESTRICAO,1X,I3,3X,
     1 58HSAO MUITOS DISPERSOS.PODE RESULTAR EM PROBLEMAS NUMERICOS.,
     A /5X,"COEFICIENTE MAXIMO=",E14.6,10X,"COEFICIENTE MINIMO=",E14.6,
     2 //5X,96(1H*),/)
   40 CONTINUE
C
C       - VERIFICACAO DA MAGNITUDE DOS COEFICIENTES DAS VARIAVEIS.
C
      DO 43 J=1,N
      SOMA=0
      XMIN=1.E+60
```

```
      XMAX=1.E-40
      DO 42 I=1,M
      ESC=ABS(A(I,J))
      IF(ESC.LT.XMIN.AND.ESC.GT.0.)XMIN=ESC
      IF(ESC.GT.XMAX)XMAX=ESC
      SOMA=SOMA + A(I,J)**2
   42 CONTINUE
      SOMA=SQRT(SOMA)
      VMOD(J)=SOMA
      ESC=XMAX/XMIN
      IF(ESC.GT.1.E05)WRITE(6,2044)J ,XMAX,XMIN
 2044 FORMAT(/5X,96(1H*),//5X,27HOS COEFICIENTES DA VARIAVEL,1X,I3,3X,
     1 58HSAO MUITOS DISPERSOS.PODE RESULTAR EM PROBLEMAS NUMERICOS.,
     A /5X,"COEFICIENTE MAXIMO=",E14.6,10X,"COEFICIENTE MINIMO=",E14.6,
     2 //5X,96(1H*),/)
   25 CONTINUE
   43 CONTINUE
      DO 45 J=1,N1
      A(M2,J)=0.
   45 A(M1,J)=C(J)
      A(M1,N1)=0.
      DO 60 I=1,M
      A(I,N1)=B(I)
      IF(B(I).GE. 0.)GO TO 55
      DO 50 J=1,N1
   50 A(I,J)=-A(I,J)
   55 DO 60 J=1,N1
   60 A(M2,J)=A(M2,J)-A(I,J)
      HO=A(M2,N1)
      IF(KSAIDA.GE.2) GO TO 65
      WRITE(6,2045)
 2045 FORMAT(1H1,4X,8HMATRIZ A,/5X,8(1H-))
      CALL YF1220(A,M2,1,N1,B,ME,5,MM1,NEL)
   65 NN=M*N
      DO 70 I=1,M2
      DO 70 J=1,M
      IF(I.EQ.J)GO TO 75
      AA(I,J)=0.
      GO TO 70
   75 AA(I,J)=1.
   70 CONTINUE
      DO 80 I=1,M
      J=N+I
      INDVB(I)=J
   80 AA(I,M1)=A(I,N1)
      AA(M1,M1)=0.
      AA(M2,M1)=A(M2,N1)
      DO 90 I=1,N
   90 LOCVAR(I)=I
      ITER=0
      KTIPO=0
      ICONTI=0
      MIM2=M2
      IF (KSAIDA.EQ.3) GO TO 100
      WRITE(6,2052)
 2052 FORMAT(  /5X,9HMATRIZ AA,/5X,9(1H-))
 3333 CALL YF1220(AA,M2,1,M1,B,INDVB,3,MM2,NEL)
      WRITE(6,2065)(INDVB(I),I=1,M)
 2065 FORMAT(  /5X,11HVETOR INDVB,/5X,11(1H-),/(5X,40I3))
      WRITE(6,2070)(LOCVAR(I),I=1,N)
 2070 FORMAT(  /5X,12HVETOR LOCVAR,/5X,12(1H-),/(5X,40I3))
```

```
      IF(KSAIDA.EQ.1) WRITE(6,2100)
 2100 FORMAT(1H1)
C
C
  100 IF(KSAIDA.EQ.1)WRITE(6,2110)
 2110 FORMAT(////5X,10HITERACAO =,I3,/5X,13(1H-))
      IF(ITER.LE.LIMITE) GO TO 105
      WRITE(6,2120)  LIMITE
 2120 FORMAT(/5X,80(1H*)/5X,"O NUMERO DE ITERACOES ULTRAPASSOU O N. LIMI
     *TE DE ITERACOES =",I3/5X,80(1H*))
      KTIPO=-1
      RETURN
C
C      - MONTAGEM DO VETOR C COM O CUSTO DOS FATORES RELATIVOS AO CONJUN
C          TO DE VARIAVEIS BASICAS.
C
  105 DO 130 J=1,N
      IF(LOCVAR(J).EQ.0)GO TO 120
      SOMA=A(M1M2,J)
      DO 110 I=1,M
  110 SOMA=SOMA + AA(M1M2,I)*A(I,J)
      C(J)=SOMA
      GO TO 130
  120 C(J)=0
  130 CONTINUE
C
C
      IF(KSAIDA.LE.1) WRITE(6,2130) (C(L),L=1,N)
 2130 FORMAT(     /5X, 6HCUSTOS,/5X,6(1H-),/(5X,8E14.6))
C
C      - DETERMINACAO DO CUSTO DO FATOR RELATIVO MINIMO.
C
      XMIN=1.E+60
      DO 145 J=1,N
      IF(LOCVAR(J).EQ.0)GO TO 145
      IF(A(M1M2,J).EQ.0) GO TO 135
      IF(C(J)/ABS(A(M1M2,J)).GE.-ERPERC) GO TO 145
      GO TO 140
  135 IF(MODPI.EQ.0.OR.VMOD(J).EQ.0) GO TO 145
      IF(C(J)/(MODPI*VMOD(J)).GE.-TOLANG) GO TO 145
  140 IF(XMIN.LE.C(J))GO TO 145
      XMIN=C(J)
      S=J
  145 CONTINUE
C
C
      IF(XMIN.EQ.1.E+60 )GO TO 270
      ITER=ITER+1
C
C      - DETERMINACAO DO VETOR P=B**-1 P(J) .
C
      B(M1M2)=C(S)
      IF(M1M2.EQ.M1) GO TO 155
C     CALCULO DE C(J)
      SOMA=A(M1,S)
      DO 150 I=1,M
  150 SOMA=SOMA+AA(M1,I)*A(I,S)
C     CALCULO DE P(J)
      B(M1)=SOMA
  155 DO 162 I=1,M
      SOMA=0
```

```
      DO 160 J=1,M
  160 SOMA=SOMA+AA(I,J)*A(J,S)
  162 B(I)=SOMA
C
C
      IF(KSAIDA.LE.1) WRITE(6,2140) (B(L),L=1,M1M2)
 2140 FORMAT( /11X,1H-,/5X,7HVETOR P,/5X,7(1H-),/(5X,8E14.6))
C
C       - DETERMINACAO DA VARIAVEL CANDIDATA A SAIR DA BASE.
C           NOS CASOS DEGENERADOS APLICA-SE A REGRA DO LEXUGRAFICO.
C
      XMIN=1.E+60
      NDEG=0
      NPNEG=0
      DO 180 I=1,M
      IF(B(I).GT.ERRABS)GO TO 165
      NPNEG=NPNEG+1
      GO TO 180
  165 QBP=AA(I,M1)/B(I)
      IF(QBP-XMIN)170,175,180
  170 XMIN=QBP
      NDEG=1
      ME(NDEG)=I
      GO TO 180
  175 NDEG=NDEG+1
      ME(NDEG)=I
  180 CONTINUE
      IF(NPNEG.LT.M)GO TO 185
      KTIPO=1
      WRITE(6,2145) S,ERRABS,("X",INDVB(I),I=1,M)
 2145 FORMAT(/5X,80(1H*)/37X,"PROBLEMA ILIMITADO"/5X,"A VARIAVEL CANDIDA
     1TA A  ENTRAR NA BASE E",3X,J3,10X,"ERRABS=ZERO=",E14.7/5X,"VARIAVE
     2IS BASICAS"/(10(5X,A1,J3)))
      WRITE(6,2146) (B(I),I=1,M)
 2146 FORMAT(11X,"-",/5X,"VETOR P",/(5X,8E15.7))
      WRITE(6,2148)
 2148 FORMAT(5X,80(1H*))
      RETURN
  185 IF(NDEG.GT.1)GO TO 190
      R=ME(1)
      GO TO 230
  190 DO 220 J=1,M
      IF(J.EQ.1)GO TO 200
      NDEG=NLEX
      DO 195 K=1,NDEG
  195 ME(K)=MF(K)
  200 XMIN=1.E60
      NLEX=0
      DO 215 I=1,NDEG
      QBP=AA(ME(I),J)/B(ME(I))
      IF(QBP-XMIN)205,210,215
  205 XMIN=QBP
      NLEX=1
      MF(NLEX)=ME(I)
      GO TO 215
  210 NLEX=NLEX+1
      MF(NLEX)=ME(I)
  215 CONTINUE
      IF(NLEX.EQ.1)GO TO 225
  220 CONTINUE
  225 R=MF(1)
```

```
C
C
  230 ICONTI=ICONTI+1
C
C      - EFETUAR O PIVOTAMENTO: COLOCAR A VARIAVEL X(S) NA BASE E RETI-
C          RAR A VARIAVEL X(R).
C
      LOCVAR(S)=0
      IF(INDVB(R).LE.N) LOCVAR(INDVB(R))=1
      INDVB(R)=S
      IF(ICONTI.GE.KATVAL)GO TO 238
      DO 232 J=1,M1
  232 AA(R,J)=AA(R,J)/B(R)
      DO 236 I=1,M1M2
      IF(I.EQ.R)GO TO 236
      DO 234 J=1,M1
  234 AA(I,J)=AA(I,J)-B(I)*AA(R,J)
  236 CONTINUE
      GO TO 258
  238 ICONTI=0
      DO 246 K=1,M
      IF(INDVB(K).GT.N)GO TO 242
      DO 240 I=1,M
  240 AA(I,K)=A(I,INDVB(K))
      GO TO 246
  242 DO 244 I=1,M
  244 AA(I,K)=0.
      L=INDVB(K)-N
      AA(L,K)=1.
  246 CONTINUE
      CALL YF1001(0,0,M,M,MM2,AA,AA)
      CALL YF1008(AA,M,DET,B,C,ERRABS=1.0E-2)
      CALL YF1001(1,0,M,M,MM2,AA,AA)
      DO 249 I=1,M
      SOMA=0.
      DO 248 J=1,M
  248 SOMA=SOMA+AA(I,J)*A(J,N1)
  249 AA(I,M1)=SOMA
      SOMAA=0.
      DO 252 J=1,M
      SOMA=0.
      DO 250 I=1,M
      IF(INDVB(I).GT.N)GO TO 250
      SOMA=SOMA-A(M1,INDVB(I))*AA(I,J)
  250 CONTINUE
      AA(M1,J)=SOMA
  252 SOMAA=SOMAA+AA(M1,J)*A(J,N1)
      AA(M1,M1)=SOMAA
      IF(M1M2.EQ.M1)GO TO 258
      SOMAA=A(M2,N1)
      DO 256 J=1,M
      SOMA=0.
      DO 254 I=1,M
      IF(INDVB(I).GT.N)GO TO 254
      SOMA=SOMA-A(M2,INDVB(I))*AA(I,J)
  254 CONTINUE
      AA(M2,J)=SOMA
  256 SOMAA=SOMAA+AA(M2,J)*A(J,N1)
C
C
  258 MODPI=0
```

```
      DO 265 J=1,M
  265 MODPI=MODPI+AA(M1M2,J)**2
      MODPI=SQRT(MODPI)
C
C
      IF(KSAIDA.GE.2)GO TO 100
      IF(KSAIDA.LE.0)GO TO 260
      WRITE(6,2150)(INDVB(L),AA(L,M1),L=1,M)
 2150 FORMAT(/5X,"VAR. BASICAS",/5X,13(1H-)/(5X,6("X",I3,"=",E14.6,2X)))
      Z=-AA(M1M2,M1)
      WRITE(6,2155)Z
 2155 FORMAT( /5X,17HFUNCAO OBJETIVO =,E14.8,/5X,31(1H-))
      GO TO 100
  260 WRITE(6,2160) ITER
 2160 FORMAT(1H1,20X,41HT     A     B     L     E     A     U     N.     ,I3,/
     *21X,41H-     -     -     -     -     -     --     ,3H---,)
      CALL YF1220(AA,M1M2,1,M1,M,INDVB,3,MM2,NEL)
      GO TO 100
  270 IF(M1M2.EQ.M1) GO TO 340
      IF(W0.EQ.0.) GO TO 275
      SOMA=ABS(AA(M2,M1)/W0)
      IF(SOMA.LT.ERPERC) GO TO 280
  273 KTIPO=2
      WRITE(6,2175) AA(M2,M1), W0,ERRABS,ERPERC,TOLANG,(C(J),J=1,N)
 2175 FORMAT(/5X,80(1H*)/36X,"PROBLEMA INVIAVEL"/5X,"-W=",E12.5,2X,"-W0=
     1",E14.5,/5X,"ERRABS=ZERO=",E12.5,2X,"ERPERC=",F12.5,2X,"TOLANG="
     *E12.5,/11X,"-"/5X,"VETOR D"/(5X,5E15.7))
      WRITE(6,2176) (A(M2,J),J=1,N)
 2176 FORMAT(5X,"VETOR D",/(5X,5F15.7))
      WRITE(6,2140)
      RETURN
  275 IF(AA(M2,M1) .GT.-ERRABS)  GO TO 280
      GO TO 273
  280 M1M2=M1
      IF(KSAIDA.LT.2)WRITE(6,2180)
 2180 FORMAT(///5X,35HFINAL DA FASE 1 ** INICIO DA FASE 2,/5X,35(1H-)/)
      IF(KVIAVE.EQ.1) RETURN
      GO TO 100
  340 Z=-AA(M1,M1)
      IF(KPROB.EQ.2) Z=-Z
      IF(KSAIDA.EQ.3) GO TO 999
      WRITE(6,2260) Z ,ITER
 2260 FORMAT((1H1,5X,120(1H*),///5X,19HSOLUCAO OTIMA   Z =,E14.8,/5X,
     *33(1H-),//5X,17HN. DE ITERACOES =,I3,/5X,20(1H-)//))
      WRITE(6,2270)(INDVB(I),AA(I,M1),I=1,M)
 2270 FORMAT(45X,33HVARIAVEIS NA BASE     -     VALORES,/45X,37(1H-),/(
     *52X,I3,11X,1H-,1X,E14.8))
      WRITE(6,2280)
 2280 FORMAT(//5X,120(1H*))
  999 RETURN
      END
```

SE

```
C
C     SUB-ROTINA YF1220
C
C     OBJETIVO
C          IMPRESSAO DA I-ESIMA ATE A J-ESIMA COLUNA DE UMA MATRIZ GERAL
C          SAO IMPRESSOS TAMBEM, OPCIONALMENTE, NUMEROS INDICATIVOS DE
C          VARIAVEIS PARA AS LINHAS E/OU COLUNAS.
C
C     USO: CALL YF1220(A,M,N1,N2,FOBA,INDVNB,KMATRI,MA,NEL)
C
C     DESCRICAO DOS PARAMETROS
C
C     REFERENCIA
C        VEJA SUB-ROTINA ESCMAT, ALPROG, VOLUME 2, P.190-191 218.
C
C-----------------------------------------------------------------------
                                                                      FO
      SUBROUTINE YF1220(A,M,N1,N2,FOBA,INDVNB,KMATRI,MA,NEL)
      DIMENSION A(MA,1),FOBA(1),INDVNB(1)
      N=N2-N1+1
      KK=N/NEL
      K=KK+1
      IF(KK*NEL.EQ.N) K=KK
      DO 100 J=1,K
      WRITE(6,2025)
      KI=N1+(J-1)*NEL
      KF=KI+NEL-1
      IF(J.EQ.K)KF=N2
      GO TO (10,20,50,40,50),KMATRI
   10 IF(J.EQ.1) WRITE(6,1900) "1",(FOBA(LL)  ,INDVNB(LL)  ,LL=KI+1,KF)
      IF(J.NE.1) WRITE(6,2100)(FOBA(LL),INDVNB(LL),LL=KI,KF)
      GO TO 50
   20 IF(J.EQ.1) WRITE(6,1900) "1",("X",LL,LL=KI,KF-1)
      IF(J.NE.1) WRITE(6,2100) ("X",LL,LL=KI,KF)
      GO TO 50
   40 WRITE(6,2100) ("X",LL,LL=KI,KF)
   50 DO 100 I=1,M
      IF(J.NE.1)GO TO 80
      GO TO (60,70,75,80,60),KMATRI
      IF(I.EQ.1)WRITE(6,3000) (A(1,LL),LL=KI,KF)
      IF(I.NE.1) WRITE(6,3010)FOBA(I),INDVNB(I),(A(I,LL),LL=KI,KF)
      GO TO 100
   60 WRITE(6,3010) "X",I,(A(I,LL),LL=KI,KF)
      GO TO 100
   70 IF(I.EQ.1) WRITE(6,3000)        (A(1,LL),LL=KI,KF)
      IF(I.NE.1) WRITE(6,3010) "X",I-1,(A(I,LL),LL=KI,KF)
      GO TO 100
   75 IF(I-N2) 77,78,79
   77 WRITE(6,3010) "X",INDVNB(I),(A(I,LL),LL=KI,KF)
      GO TO 100
   78 WRITE(6,3015) "-Z",(A(I,LL),LL=KI,KF)
      GO TO  100
   79 WRITE(6,3015) "-W",(A(I,LL),LL=KI,KF)
      GO TO 100
   80 WRITE(6,3020)(A(I,LL),LL=KI,KF)
  100 CONTINUE
 2025 FORMAT(1X/)
 1900 FORMAT(16X,A1,6X,7(5X,A1,I3,5X))
 2100 FORMAT (9X,8(8X,A1,I3,2X))
 3000 FORMAT(8X,1H1,8E14.6)
```

```
3010 FORMAT(5X,A1,I3,8E14.6)
3015 FORMAT(7X,A2,8E14.6)
3020 FORMAT(9X,8E14.6)
     RETURN
     END
                                                              SE
```

```
C      SUBROTINA YF1008
C
C      OBJETIVO
C          INVERTE MATRIZ ARMAZENADA EM FORMATO VETORIAL
C
C      USO: CALL YF1008(UCOT,N,D,L,M,TOL)
C
C      DESCRICAO DE PARAMETROS
C       UCOT  VETOR COM OS ELEMENTOS DA MATRIZ GERAL A SER INVERTIDA, DES
C             TRUIDO NA COMPUTACAO E SUBSTITUIDO PELA INVERSA
C          N  ORDEM DA MATRIZ A
C          D  DETERMINANTE
C          L  VETOR DE TRABALHO DE DIMENSAO N
C          M  VETOR DE TRABALHO DE DIMENSAO N
C        TOL  TOLERANCIA PARA ELEMENTO NULO - RECOMENDA-SE 1.0E-9
C
C      OBSERVACAO
C          A MATRIZ A DEVE SER GERAL
C
C      METODO
C          E USADO O METODO PADRAO DE GAUSS-JORDAN.
C          E CALCULADO TAMBEM O DETERMINANTE.
C          A MATRIZ E SINGULAR SE  D=0.
C
C      REFERENCIA
C          VEJA SUBROTINA MINV, SSP-IBM, V-3, PG. 118
C
C-----------------------------------------------------------------------

      SUBROUTINE YF1008(UCOT,N,D,L,M,TOL)
      DIMENSION UCOT(1),L(1),M(1)
      D=1.0
      NK=-N
      DO 180 K=1,N
      NK=NK+N
      L(K)=K
      M(K)=K
      KK=NK+K
      BIGA=UCOT(KK)
      DO 120 J=K,N
      IZ=N*(J-1)
      DO 120 I=K,N
      IJ=IZ+I
  110 IF(ABS(BIGA)-ABS(UCOT(IJ))) 115,120,120
  115 BIGA=UCOT(IJ)
      L(K)=I
      M(K)=J
  120 CONTINUE
      J=L(K)
      IF(J-K) 135,135,125
  125 KI=K-N
      DO 130 I=1,N
      KI=KI+N
      HOLD=-UCOT(KI)
      JI=KI-K+J
      UCOT(KI)=UCOT(JI)
  130 UCOT(JI)=HOLD
  135 I=M(K)
      IF(I-K) 145,145,138
  138 JP=N*(I-1)
      DO 140 J=1,N
```

```
      JK=NK+J
      JI=JP+J
      HOLD=-UCOT(JK)
      UCOT(JK)=UCOT(JI)
140   UCOT(JI)=HOLD
145   IF(ABS(BIGA)-TOL)146,146,148
146   D=0.0
      RETURN
148   DO 155 I=1,N
      IF(I-K) 150,155,150
150   IK=NK+I
      UCOT(IK)=UCOT(IK)/(-BIGA)
155   CONTINUE
      DO 165 I=1,N
      IK=NK+I
      HOLD = UCOT (IK)
      IJ=I-N
      DO 165 J=1,N
      IJ=IJ+N
      IF(I-K) 160,165,160
160   IF(J-K) 162,165,162
162   KJ=IJ-I+K
      UCOT(IJ)=HOLD     *UCOT(KJ)+UCOT(IJ)
165   CONTINUE
      KJ=K-N
      DO 175 J=1,N
      KJ=KJ+N
      IF(J-K) 170,175,170
170   UCOT(KJ)=UCOT(KJ)/BIGA
175   CONTINUE
      D=D*BIGA
      UCOT(KK)=1.0/BIGA
180   CONTINUE
      K=N
200   K=K-1
      IF(K) 250,250,205
205   I=L(K)
      IF(I-K) 220,220,208
208   JQ=N*(K-1)
      JR=N*(I-1)
      DO 210 J=1,N
      JK=JQ+J
      HOLD=UCOT(JK)
      JI=JR+J
      UCOT(JK)=-UCOT(JI)
210   UCOT(JI)=HOLD
220   J=M(K)
      IF(J-K) 200,200,225
225   KI=K-N
      DO 230 I=1,N
      KI=KI+N
      HOLD=UCOT(KI)
      JI=KI-K+J
      UCOT(KI)=-UCOT(JI)
230   UCOT(JI)=HOLD
      GO TO 200
250   RETURN
      END
```

SE

```
C------------------------------------------------------------------------
C
C       SUBROTINA YF1001
C
C       OBJETIVO.
C           CONVERTER CONJUNTO DE DADOS DE DUPLA-DIMENSAO PARA DIMENSAO
C           SIMPLES E VICE-VERSA
C
C       USO: CALL YF1001(CV,MS,M,N,MA,A,B)
C
C       DESCRICAO DOS PARAMETROS.
C           CV - CODIGO QUE INDICA O TIPO DE CONVERSAO
C                =1  DE DIMENSAO SIMPLES PARA DUPLA
C                ≠1  DE DIMENSAO DUPLA PARA SIMPLES
C           MS - MODO DE ARMAZENAMENTO DA MATRIZ
C                <1  MODO GERAL
C                =1  MODO SIMETRICO
C                >1  MODO DIAGONAL
C           M  - NUMERO DE LINHAS ATUAIS DA MATRIZ A
C           N  - NUMERO DE COLUNAS ATUAIS DA MATRIZ A
C           MA - NUMERO DE LINHAS ESPECIFICADAS PARA A MATRIZ A NO COMANDO
C                DIMENSION
C           A  - MATRIZ DE SAIDA SE CV=1  OU
C                       DE ENTRADA SE CV≠1 CONTENDO SEUS ELEMENTOS NAS PRI
C                       MEIRAS I LINHAS E PRIMEIRAS J COLUNAS
C           B  - VETOR DE ENTRADA SE CV=1 OU
C                      DE SAIDA SE CV≠1 CONTENDO OS ELEMENTOS DA MATRIZ A
C                      ARMAZENADOS CONTIGUAMENTE
C
C------------------------------------------------------------------------
      SUBROUTINE YF1001(CV,MS,M,N,MA,A,B)
      INTEGER CV
      DIMENSION A(1),B(1)
      MI=MA-M
      IF(CV-1)200,10,200
C
C  CONVERSAO DE DIMENSAO SIMPLES PARA DIMENSAO DUPLA
C
   10 MMN=MA*N+1
      IF(MS-1)20,40,70
C
C  MODO DE ARMAZENAMENTO GERAL
C
   20 MN=M*N+1
      DO 30 J=1,N
      MMN=MMN-MI
      DO 30 I=1,M
      MN=MN-1
      MMN=MMN-1
   30 A(MMN)=B(MN)
      RETURN
C
C  MODO DE ARMAZENAMENTO SIMETRICO
C
   40 MN=(N+1)*N/2+1
      II=N+1
      DO 50 J=1,N
      MMN=MMN-MI
      MI=MI+1
      II=II-1
```

```
      DO 50 I=1,II
      MN=MN-1
      MMN=MMN-1
   50 A(MMN)=B(MN)
      N1=N-1
      DO 60 J=1,N1
      JJ=J-1
      J1=J+1
      DO 60 I=J1,N
   60 A(JJ*MA+I)=A((I-1)*MA+J)
      RETURN
C
C  MODO DE ARMAZENAMENTO DIAGONAL
C
   70 MMN=MMN+N
      MN=N+1
      MM=MA+1
      DO 80 J=1,N
      MMN=MMN-MM
      MN=MN-1
   80 A(MMN)=B(MN)
      MMN=-MI
      DO 100 J=1,N
      MMN=MMN+MI
      DO 100 I=1,M
      MMN=MMN+1
      IF(I-J)90,100,90
   90 A(MMN)=0.
  100 CONTINUE
      RETURN
C
C  CONVERSAO DE DIMENSAO DUPLA PARA DIMENSAO SIMPLES
C
  200 MN=0
      MMN=0
      IF(MS-1)210,240,270
C
C  MODO DE ARMAZENAMENTO GERAL
C
  210 DO 230 J=1,N
      DO 220 I=1,M
      MN=MN+1
      MMN=MMN+1
  220 B(MN)=A(MMN)
  230 MMN=MMN+MI
      RETURN
C
C  MODO DE ARMAZENAMENTO SIMERICO
C
  240 MI=MA-1
      DO 260 J=1,N
      DO 250 I=1,J
      MN=MN+1
      MMN=MMN+1
  250 B(MN)=A(MMN)
      MMN=MMN+MI
  260 MI=MI-1
      RETURN
C
C  MODO DE ARMAZENAMENTO DIAGONAL
C
```

```
270 DO 280 I=1,M
    MMN=MMN+1
    B(I)=A(MMN)
280 MMN=MMN+MA
    RETURN
    END
                                                  SE
```

```
      SUBROUTINE INTEG(F,/X/,/RELERR/,/ABSERR/,/WORK/)
      COMMON/IWORKP/IXO,IFO,IF1,IF2,IF3,IF4,IF5,IF6,IF7,IF8,IF9,
     *IF10,IF11,IF12
      COMMON/SERINT/T    ,TOUT
      COMMON/CTEINT/KPROB,ITER,KTIPO,KATVAL,KVIAVE,KDADOS,KSAIDA,M,
     *IS,NEL,LIMITE,IFLAG,IMETH,NEON,ITN,IXX,NCO,NCO2,NCO2M1,N1,NCO1,
     *KONT1,NTPRT
      COMMON/CTREAL/ERRABS,ERPERC,TOLANG,Z,DT,TY,BETA,PCBB,BETAG,
     *TMEDIO,ALFA,PCT,DMMAX,EPS,EPS2
C
C     PURPOSE : INTEGRATION OF NEON FIRST ORDER O.D.E.
C
C     METHOD     RUNGE-KUTTA  OR  ODE
C
C     INPUTS: F           - DERIVATIVE SUBROUTINE RETURNING THE VALUE
C                             OF THE DERIVATIVES, GIVEN THE STATE AND
C                             THE VALUE OF THE INDEPENDANT VARIABLE
C                                 EXAMPLE CALL: CALL F (T,X,DX,IER)
C             NEON        - NUMBER OF EQUATIONS TO BE INTEGRATED
C             X           - DEPENDANT VARIABLE ARRAY
C             T           - CURRENT VALUE OF INDEPENDANT VARIABLE
C             TOUT        - VALUE OF T FOR WHICH OUTPUT IS DESIRED
C             RELERR      - RELATIVE LOCAL TRUNCATION ERROR
C                             ARRAY
C             ABSERR      - ABSOLUTE LOCAL TRUNCATION ERROR TOLERANCE
C                             ARRAY
C             IFLAG        - CONTROL PARAMETER , MUST BE SET TO 1
C                              ON INITIAL ENTRY FOR INITIALIZATION. ON
C                              OUTPUT THIS FLAG MAY CHANGE VALUE , SO IT
C                              MUST BE A VARIABLE IN THE CALLING ROUTINE.
C             WORK        - WORK STORAGE ARRAY
C                             NOTE: IF RK24 IS USED, DIMENSION TO 5*NEWN
C                                   IF RK45 IS USED, DIMENSION TO 7*NEWN
C                                   IF RK78 IS USED, DIMENSION TO 14*NEON
C                                   IF ODE IS USED DIMENSION TJ 100+21*NEON
C             DT          - INITIAL STEP SIZE ( NOT REQUIRED FOR ODE)
C             IMETH       - FLAG FOR TYPE OF INTEGRATOR USED
C                         = 1 RUNGE-KUTTA (2)4
C                         = 2 RUNGE-KUTTA (4)5
C                         = 3 RUNGE-KUTTA (7)8
C                         = 4 SHAMPINE-GORDON    (ODE)
C
C      OUTPUT: IFLAG        - FLAG INDICATING STATUS OF THE INTEGRATION
C                           = 2    NORMAL COMPLETION , T = TOUT
C                           # 2 ABNORMAL TERMINATION , T DID NOT REACH TOUT
C
C      NOTE:     WHEN USING ODE, THE USER MUST SET IFLAG = 1 ON FIRST
C                ENTRY, IF THE USER DESIRES  ODE TO RESTART, HE CAN RESET
C                IFLAG = 1 , OTHERWISE IT SHOULD REMAIN UNTOJCHED
C
C
       EXTERNAL F
C
       DIMENSION X(1),WORK(1),IWORK(14),ABSERR(1),RELERR(1)
       DIMENSION INT(4) , NMUL(4) , NADD(4) , MSG(2)
       DATA INT /4HRK24 , 4HRK45 , 4HRK78 , 4H ODE/ ,
     :      NMUL/5,7,14,21/ , NADD /0,0,0,100/ ,
     :      MSG /6H ERROR , 6H NOTE  /
       ITMER = 0
```

```
      ITIMES = 0
      IF ( T .EQ. TOUT ) GO TO 999
C
C     PREFORM INITIALIZATION IF NECESSARY
C
      IF ( IFLAG .NE. 1) GO TO 6
      IF ( IMETH .NE. 4) GO TO 4
C
C     CHECK FOR ANY ABSERR = 0.0 AND X = 0.0
C
      IER = 0
      DO 3 I=1,NEQN
         IF (X(I) .NE. 0. .OR. ABSERR(I) .NE.0.) GO TO 3
      ABSERR(I)=3.0E-10
         IER = 1
    3 CONTINUE
      IF ( IER .EQ. 0 ) GO TO 1
C
C     ABSERR = 0.0 FOR A VARIABLE WITH VALUE 0.0
C
      PRINT 2000
      PRINT 2001
      GO TO 1
C
C     PARTITION WORK ARRAY FOR RK S
C
    4 CONTINUE
      IF ( IMETH .EQ. 1 ) CALL RK24CO
      IF ( IMETH .EQ. 2 ) CALL RK45CO
      IF ( IMETH .EQ. 3 ) CALL RK78CO
C
    6  DT = SIGN(DT,TOUT-T)
C
C     BRANCH ACCORDING TO INTEGRATION MODE
C
    1 GO TO (10 , 20 , 30 , 40 ) , IMETH
C
C     RUNGE-KUTTA (2)4
C
   10 CALL RK24(F,NEQN,DT,X,T,TOUT,RELERR,ABSERR,IFLAG,
     *WORK(IF0),WORK(IF1),WORK(IF2),WORK(IF3),WORK(IX0))
      GO TO 100
C
C     RUNGE-KUTTA (4)5
C
   20 CALL RK45(F,NEQN,DT,X,T,TOUT,RELERR,ABSERR,IFLAG,
     1          WORK(IF0),WORK(IF1),WORK(IF2),WORK(IF3),
     2          WORK(IF4),WORK(IF5),WORK(IX0))
      GO TO 100
C
C
C     RUNGE-KUTTA (7)8
C
   30 CALL RK78(F,NEQN,DT,X,T,TOUT,RELERR,ABSERR,IFLAG,
     1         WORK(IF0),WORK(IF1),WORK(IF2),WORK(IF3),
     2          WORK(IF4),WORK(IF5),WORK(IF3),WORK(IF7),
     3          WORK(IF8),WORK(IF9),WORK(IF10),WORK(IF11),
     4          WORK(IF12),WORK(IX0))
      GO TO 100
C
C     SHAMPINE-GORDON  (ODE)
```

```
C
   40 CALL ODE(F,NEQN,X,T,TOUT,RELERR,ABSERR,IFLAG,WORK,IWORK)
      DT = WORK(90)
      GO TO 100
C
C     CHECK ERROR FLAGS
C
  100 IFL = IABS(IFLAG)
      GO TO (901,999,903,904,905,906,907,908,909) , IFL
C
C
C   ***************
C   *             *
C   * FATAL ERRORS *
C   *             *
C   ***************
C
C             IFLAG = 1
C
  901 PRINT 1000,MSG(1) , INT(IMETH) , IFLAG
      GO TO 950
C
C             IFLAG = 5
C
  905 PRINT 1000,MSG(1) , INT(IMETH) , IFLAG
      PRINT 1005
      GO TO 950
C
C             IFLAG = 6
C
  906 PRINT 1000,MSG(1) , INT(IMETH) , IFLAG
      PRINT 1006
      GO TO 950
C
C             IFLAG = 7
C
  907 PRINT 1000,MSG(1) , INT(IMETH) , IFLAG
      PRINT 1007
      GO TO 950
C
C             IFLAG = 8
C
  908 PRINT 1000,MSG(1) , INT(IMETH) , IFLAG
      PRINT 1008
      GO TO 950
  909 PRINT 1000,MSG(1),INT(IMETH),IFLAG
      PRINT 1050 , NEGN ,DT , T , TOUT , NWK
      PRINT 1009
      GO TO 950
C
C     FATAL ERROR EXIT
C
  950 NWK =  NEQN*NMUL(IMETH) + NADD(IMETH)
      PRINT 1081, ABSERR
       PRINT 1082, RELERR
      PRINT 1060,(X(I),I=1,NEQN)
      CALL F ( T , X , WORK(1),TEM)

      PRINT 1070 , (WORK(I),I=1,NEQN)
      PRINT 1080
      RETURN
```

```
C
C
C    *******************
C    *                 *
C    * NON-FATAL ERRORS *
C    *                 *
C    *******************
C
C             IFLAG = 3
C
  903 ITMER = ITMER + 1
      GO TO 40
C
C             IFLAG = 4
C
  904 ITIMES = ITIMES + 1
      GO TO 40
C
C
C     NORMAL RETURN , IFLAG = 2
C
C
  999 IF ( ITIMES .EQ. 0) GO TO 960
      PRINT 1000,MSG(2) , INT(IMETH) , IFLAG
      PRINT 1004,ITIMES
  960 IF (ITMER .EQ. 0) GO TO 970
      PRINT 1000,MSG(2) , INT(IMETH) , IFLAG
      PRINT 1003 , ITMER
      PRINT 1081, ABSERR
      PRINT 1082, RELERR
      PRINT 1080
  970 CONTINUE
      RETURN
C
C
C     FORMATS
C
 1000 FORMAT (/,1X,5(1H*),A6,13H FROM INTEG: , A4,
     * 21H RETURNED A VALUE OF ,I2,11H FOR IFLAG ,68(1H*),/)
 1003 FORMAT(30X,34HERROR TOLERANCES INCEASED BY ODE ,I4,7H TIMES )
 1004 FORMAT( 30X,28HODE TOOK MORE THEN 500 STEPS , I4 , 8H TIME(S),
     *          /,1X,130(1H*))
 1005 FORMAT(30X,32HTHE EQUATIONS APPEAR TO BE STIFF)
 1006 FORMAT(30X,20HINVALID INPUT TO ODE)
 1007 FORMAT (30X,46HRK REJECTED MORE THAN MAXREJ STEPS IN A ROW,   /
     *          30X,46HTHE VALUE OF MAXREJ IS INITIALY 10 FOR ALL     /
     *          30X,46HRKS                                            /
     *          30X,46HIF THIS OCCURED ON THE FIRST STEP, THE USER    /
     *          30X,46HSHOULD TRY REDUCING THE INITIAL VALUE OF DT    )
 1008 FORMAT(30X,46HTHE RK ATTEMPED TO USE TOO SMALL A STEP           )
 1009 FORMAT (30X,27HERROR IN DERIVATIVE ROUTINE)
 1050 FORMAT(/,15X,4HNEQN,I10,17X,6HDT     ,G22.14,/,
     *           15X,6HT      ,G22.14, 3X, 6HTOUT  ,G22.14, 3X ,/,15X,
     *           15X,44HWORK ARRAY SHOULD BE DIMENSIONED TO AT LEAST ,
     *           I10)
 1060 FORMAT(/,10X,3HX  ,6(G16.10,2X))
 1070 FORMAT(/,10X,3HDX ,6(G16.10,2X))
 1080 FORMAT(/,1X,130(1H*))
 1081 FORMAT(/,10X,9HABSERR   ,6(G16.10,2X))
 1082 FORMAT(/,10X,9HRELERR   ,6(G16.10,2X))
 2000 FORMAT(/,1X,5(1H*),19H NOTE FROM INTEG: ,106(1H*))
```

```
2001 FORMAT(/,30X,46HINTEGRATION WITH ODE IS DESIRED , BUT
     1          30X,46HABSERR = 0.0 AND AT LEAST ONE COMPONENT OF     /
     2          30X,46HX = 0.0 , THIS WILL RESULT IN AN INFINITE IN   /
     3          30X,46HROUTINE STEP. ABSERR INCREASED TO 1.0E-12      /
     4           1X,130(1H*))
      END
                                                                     SE
```

```
      SUBROUTINE RK24(F,NEQN,DT,X,T,TOUT,RELERR,ABSERR,IFLAG,
     */F0/,/F1/,/F2/,/F3/,/X0/).
C
      LOGICAL DTFAIL,DTFIX
C
C     PURPOSE : INTEGRATION OF NEQN FIRST ORDER ORDINARY
C               DIFFERENTIAL EQUATIONS
C
C     METHOD :  RUNGE-KUTTA (2)4 WITH AUTOMATIC SIZE CONTROL
C
C     REFERANCE :
C
C     CALLING SEQUENCE :
C               CALL RK24(F,NEQN,DT,X,T,TOUT,RELERR,ABSERR,IFLAG,
C                                   F0,F1,F2,F3,X0)
C
C.    INPUT :   NEQN     - NUMBER OF EQUATIONS TO BE INTEGRATED
C               DT       - INITIAL STEP-SIZE
C               X        - INITIAL DEPENDANT VARIABLE ARRAT
C               T        - INITIAL VALUE OF INDEPENDANT VARIABLE
C               TOUT     - NEXT VALUE OF T FOR WHICH OUTPUT DESIRED
C               RELERR - ARRAY OF RELATIVE ERROR TOLERANCES
C               ABSERR   - ARRAY OF ABSOLUTE ERROR TOLERANCES
C                        - THE ESTIMATED LOCAL TRUNCATION ERROR
C                                IS KEPT LESS THAN
C                          RELERR(I)*(X(I)+X0(I))/2 + ABSERR(I)
C                          AT EACH STEP OF THE INTEGRATION
C                        - IF ABSERR = RELL
C                        - IF ABSERR = RELERR = 0.0, NO STEP SIZE
C                          ADJUSTMENTS ARE MADE
C
C    ROUTINES REQUIRED :
C               F        - SUBROUTINE RETURNING THE DERIVATIVES
C                          OF THE FUNCTIONS FOR VALUES OF T AND X
C                          EXEMPLE CALL :  CALL F(T,X,F0)
C            F0....X0    - REAL ARRAYS, EACH DIMENSIONED TO AT LEAST
C                          NEQN WORDS. ANY ARRAYS WHOSE CONTENTS
C                          ARE EXPENDABLE MAY BE USED
C
C    OUTPUT :   T        - TOUT IF IFLAG=1
C                        - MOST RECENT T IF IFLAG=2
C               DT       - MOST RECENT STEP SIZE
C               X        - VALUE OF X AT TOUT
C               IFLAG    - FLAG FOR THE TYPE OF RETURN
C                        = 2 NORMAL RETURN , T REACHED TOUT
C                        = 7 MORF THAN MAXREJ REJECTED STEPS IN A ROW
C                        = 8 THE RK ATTEMPTED TO USE TOO SMALL A STEP
C                        = 9 THE DERIVATIVE ROUTINE ENCOUNTERED A
C                            FATAL ERROR
C
C    MODIFICATIONS :
C           . NONE
     DIMENSION X(1)
    *,RELERR(1),ABSERR(1)
    *,F0(1),F1(1),F2(1),F3(1),X0(1)
     COMMON/RKCON / NREJ,NREJT,NSTP,DTFAIL,DTFIX,IER
     COMMON/COEF24/ A0,A1,A2,A3,B10,B20,B21,B30,B31,B32,C0,C1,C2,
    1              C3,E0,E1,E2,E3,B,BLO,BUP,REMIN,DTINC,DTDEC,MAXREJ
C
C    SET FLAG IF FIXED STEP MODE DESIRED
```

```
C
      NREJT=0
      NSTP = 0
      DTFIX=.FALSE.
      DO 5 NEQ = 1,NEQN
    5 IF (ABSERR(NEQ) .EQ. 0. .AND. RELERR(NEQ) .EQ. 0.) DTFIX = .TRUE.
C
  666 CONTINUE
      DTOLD=DT
    1 DTFAIL = .FALSE.
        NREJ = 0
C
C     RESET STEP SIZE IF THIS WILL PUT T GREATER THAN TOUT
C
      DELT = TOUT - T
      IF ( ABS(DT) .LT.  ABS(DELT)) GO TO 2
      DT = DELT
      GO TO 3
    2 IF ( ABS(DT+DT) .LT.  ABS(DELT)) GO TO 3
      DT = DELT/2.
    3 CONTINUE
C
C     IF REQUIRED STEP IS TOO SMALL, EXTRAPOLATE AND RETURN
C
      IF( ABS(DT) .LT. 2.80E-11* ABS(T)) GO TO 130
C
C     FIRST EVALUATION
C
      TO=T
      DO 10 NEQ=1,NEQN
      XO(NEQ)=X(NEQ)
   10 CONTINUE
      CALL F (T,X,FO,IER)

      IF (IER .NE. 0) GO TO 150
C
C     SECOND EVALUATION
C
   20 CONTINUE
      T=TO+A1*DT
      DO=B10*DT
      DO 30 NEQ=1,NEQN
      X(NEQ)=DO*FO(NEQ)+XO(NEQ)
   30 CONTINUE
      CALL F (T,X,F1,IER)

      IF (IER .NE. 0) GO TO 150
C
C     THIRD EVALUATION
C
      T=TO+A2*DT
      DO=B20*DT
      D1=B21*DT
      DO 40 NEQ=1,NEQN
      X(NEQ)=DO*FO(NEQ)+D1*F1(NEQ)+XO(NEQ)
   40 CONTINUE
      CALL F (T,X,F2,IER)

      IF (IER .NE. 0) GO TO 150
C
C     FOURTH EVALUATION
```

```
C
      T=T0+A3*DT
      D0=B30*DT
      D1=B31*DT
      D2=B32*DT
      DO 50 NEQ=1,NEQN
      X(NEQ)=D0*F0(NEQ)+D1*F1(NEQ)+D2*F2(NEQ)+X0(NEQ)
 50   CONTINUE
      CALL F (T,X,F3,IER)

      IF (IER .NE. 0) GO TO 150
C
C     COMPUTE STATE AT T + DT
C
      D0 = C0 * DT
      D1 = C1 * DT
      D2 = C2 * DT
      D3 = C3 * DT
      DO 110 NEQ = 1,NEQN
  110 X(NEQ) = D1*F1(NEQ) + D0*F0(NEQ) + D3*F3(NEQ) + D2*F2(NEQ) +
     1              X0(NEQ)
C
C     IF FIXED STEPSIZE DESIRED, GO TO 100
C
      IF(DTFIX) GO TO 100
C
C     COMPUTE MAX LOCAL TRUNCATION ERROR
C
      RTE=0.
      D0=E0*DT
      D1=E1*DT
      D2=E2*DT
      D3=E3*DT
      DO 80 NEQ = 1,NEQN
      RER = AMAX1(RELERR(NEQ),2.272E-11+REMIN)
      SCALE = RER/2.
      TE= ABS(D3*F3(NEQ) + D2*F2(NEQ) + D0*F0(NEQ) + D1*F1(NEQ))
      XMAG = ( ABS(X(NEQ)) +  ABS(X0(NEQ)))*SCALE +  ABSERR(NEQ) + 1.0
     *E-11
   80 RTE = AMAX1(RTE,TE/XMAG)
      IF(RTE .LT. 1.) GO TO 100
C
C     REJECT THIS STEP
C
      DTFAIL =.TRUE.
      NREJT=NREJT+1
       NREJ=NREJ+1
       IF (NREJ .LT. MAXREJ) GO TO 95
      DO 90 NEQ=1,NEQN
   90 X(NEQ)=X0(NEQ)
      T=T0
      IFLAG = 7
      RETURN
   95 CONTINUE
      PCT=DTDEC
      IF(RTE .LT. BUP) PCT=B/RTE**(1./3.)
      DT=PCT*DT
      DTOLD=DT
      GO TO 20
C
C     TAKE A STEP
```

```
C
 100  CONTINUE
      T=TO+DT
      IF ( ABS(TOUT-T) .GT. 1.0E-14) GO TO 115
      DT=DTOLD
      IFLAG = 2
      RETURN
C
  115 IF(DTFIX) GO TO 1
      PCT=DTINC
      IF(RTE .GT. BLO) PCT = B/RTE**(1./3.)
      IF(DTFAIL) PCT=AMIN1(PCT,1.E+00)
      DT=DT*PCT
      DTOLD=DT
      GO TO 1
C
C     CHECK FOR TOO SMALL A STEP SIZE
C
  130 CONTINUE
      IF ( ABS(DELT) .GT.  ABS(DT)) GO TO 145
C
C     EXTRAPOLATION
C
      CALL F (T,X,FO,IER)

      IF (IER .NE. 0) GO TO 150
      DO 140 NEQ=1,NEQN
  140 X(NEQ)=DT*FO(NEQ)+X(NEQ)
      T=T+DT
      DT=DTOLD
      IFLAG = 2
      RETURN
C
C     ATTEMPTED TO USE TOO SMALL A STEP SIZE
C
  145 IFLAG = 8
      RETURN
C
C     FATAL ERROR ENCOUNTERED IN DERIVATIVE ROUTINE
C
  150 IFLAG = 9
      RETURN
      END
                                                                        SE
```

```
      SUBROUTINE RK24CO
C
C     PURPOSE : TO SET UP THE RUNGE-KUTTA 24 CONSTANTS
C
C     DR. HULL S CONSTANTS
C
      LOGICAL SETFLG
      COMMON/COEF24/ A0,A1,A2,A3,B10,B20,B21,B30,B31,B32,C0,C1,C2,
     1               C3,E0,E1,E2,E3,B,BLO,BUP,REMIN,DTINC,DTDEC,MAXREJ
      MAXREJ=10
      SETFLG=.FALSE.
      DTINC=20
      DTDEC=.025
      REMIN=3.E-09
      B=.9
C
C     CHECK IF CONSTANTS SET
C
      IF(SETFLG) GO TO 999
   10 SETFLG = .TRUE.
C
C     SET CONSTANTS
C
      A0=0.
      A2=1./2.
      A1=1./4.
      A3=1.
      B10=1./4.
      B20=0.
      B21=1./2.
      B31=-2.
      B32=2.
      B30=1.
      C0=1./6.
      C1=0.
      C2=2./3.
      C3=1./6.
      E1=-2.
      E2=2./3.
      E0=7./6.
      E3=1./6.
C
C
C     SET STEP-SIZE CONTROL FACTORS
      BUP=(B/DTDEC)**3
      BLO=(B/DTINC)**3
  999 RETURN
C
      END
```

SE

```
                                                                      FO
WARNING:THE SUBROUTINE "RK45CO" WAS NOT FOUND

WARNING:THE SUBROUTINE "RK78CO" WAS NOT FOUND                         SE

WARNING:THE SUBROUTINE "RK45" WAS NOT FOUND                           SE

WARNING:THE SUBROUTINE "RK78" WAS NOT FOUND                           SE

WARNING:THE SUBROUTINE "ODE" WAS NOT FOUND                            SE

                                                                      SE
    NO ERRORS DETECTED.  NUMBER OF CARDS = 1801.                      SE
    COMPILATION TIME = 76 SECONDS ELAPSED.  14.36 SECONDS PROCESSING(7515 CPM).
    02 STACK SIZE = 60 WORDS.  FILESIZE = 140 WORDS.  ESTIMATED CORE STORAGE REQU
    TOTAL PROGRAM CODE = 4007 WORDS.  ARRAY STORAGE = 827 WORDS.
    NUMBER OF PROGRAM SEGMENTS = 42.  NUMBER OF DISK SEGMENTS = 303.
    PROGRAM CODE FILE = (ORBITAL)TORBP, COMPILER COMPILED ON 08/27/76
```

APÊNDICE C

DADOS DE ENTRADA E SAÍDA

1. INTRODUÇÃO

Neste apêndice são apresentados e discutidos os dados de entrada do programa. São apresentados também, os resultados mais importantes e uma saída normal típica do programa.

2. DADOS DE ENTRADA

2.1 - INTRODUÇÃO

A boa escolha dos dados de entrada é bastante importante para o sucesso do processamento. Aqui são descritos os dados de entrada, suas características, sequência de entradas e valores adotados para os testes realizados neste trabalho.

2.2 - CLASSIFICAÇÃO POR FUNÇÃO

Neste item é feita a classificação dos dados de entrada, conforme sua aplicação no procedimento numérico. A nomenclatura utilizada será a mesma do desenvolvimento analítico. Quando não houver coincidência, será apresentada entre parentêses a nomenclatura utilizada no programa para computador.

i) *Dados auxiliares*

NP (NTPRT) - Número de subdivisões da integração numérica para efeito de impressão de resultados.

Vetor ITM - Indica em quais atualizações de x_{f_a} se deseja impressões de resultados.

ξ_b (EPS3) - Valor associado ao critério normal de parada (item 4.2.2).

Para efeito de compatibilidade numérica, esta variável deve ser maior do que os erros numéricos envolvidos no processo e menor do que a precisão desejada.

ii) *Dados associados à integração numérica*

DT - Passo inicial de integração.

Vetor ERRABS e RELERR - Cada componente corresponde, respectivamente, ao valor absoluto e ao valor relativo admissível, para o erro de arredondamento em cada passo de integração, na respectiva variável.

iii) *Dados associados à derivação numérica*

Vetor DP - É o vetor dos valores Δa_i (equação IV.1). Como foi visto no item 4.1, os valores DP_i não podem ser muito grandes, para não fugir da hipótese de linearidade, e nem muito pequeno a ponto de, erros de truncamento na integração, introduzirem erros significativos no cálculo das derivadas.

iv) *Dados associados ao problema dinâmico*

n (NEQN) - Número de variáveis de estado do problema dinâmico.

m (M) - Número de equações de restrições do problema dinâmico.

Vetor x_0 (Vetor Y) - Correspondem às condições iniciais do vetor de variáveis de estado.

$\bar{r}_f$, μ, m, m_0, T (RFINAL, DMI, AMP, AMO, EMP) - Corresponde aos parâmetros associados ao problema dinâmico.

v) *Dados associados à hipótese de linearidade*

Sa_{max}, Sn (SAH, SAI) - Os valores Sa_{max}, Sn (item 4.2.5), quando pequenos, tendem a aumentar o tempo de processamento; por outro lado, estes valores devem ser suficientemente pequenos para não comprometer a validade de x_{f_a}.

Δa_{max} (DGMAX) - Máximo valor admissível para os incrementos nos parâmetros a serem otimizados (item 4.2.2).

vi) *Dados associados à convergência*

ω_{2g+1} (Vetor C) - O valor inicial de ω_{2g+1} (problema III.5) normalmente deve ser adotado alto, para impor desde o início a caminhada no sentido de minimização de IP.

ω_i, para $i \neq 2g+1$ - Os valores de ω_i ponderam a importância relativa dos incrementos dos parâmetros. Normalmente, devem ser definidos de modo a uniformizar a convergência, mas, quando se possue alguma outra estratégia de convergência, os parâmetros ω_i devem ser preparados para executá-la. Por exemplo: em casos onde as posições de descontinuidade devam ser otimizadas, é aconselhável adotar valores maiores para os custos correspondentes aos incrementos nas posições de descontinuidade, pois isso fará com que as posições de descontinuidade permaneçam inalteradas no início do processo de convergência.

α (ALFA) - Parâmetro que define a taxa de avanço nas restrições (problema III.5).

ξ_p (EPS2) - Parâmetro associado às condições de proxidade (item 4.2.2).

O valor inicial de ξ_p deve ser escolhido, de modo que a solução inicial não seja próxima, para permitir o ajuste inicial dos valores ω_{2g+1} e β. Por outro lado ξ_p não deve ser muito pequeno.

vii) *Valores que dependem do funcional adotado*

Vetor a (Vetor P) - Os valores iniciais representam a forma inicial do controle.

g (NCO) - Número de parâmetros a serem otimizados.

2.3 - CLASSIFICAÇÃO POR SEQUÊNCIA DE ENTRADA

A sequência de dados será apresentada na forma de conjuntos, que correspondem a comandos na leitura do programa. Podem corresponder a um ou mais cartões de entrada, dependendo do número de elementos desse conjunto.

Algumas variáveis possuem nomenclatura baseada no problema utilizado para teste do procedimento. Naturalmente, para outros problemas estas mesmas variáveis correspondem a outros significados físicos.

Sequência de dados:

1º CONJUNTO . . . . FORMATO 20I3

M - Número de equações de restrições.

NEQN - Número de variáveis de estado.

NTPRT - Número de subdivisões da integração para efeito de impressão.

2º CONJUNTO . . . . FORMATO 8F10.4

DT - Passo inicial de integrações.

RFINAL - Parâmetro utilizado na sub-rotina D$V (por exemplo: raio final).

AMO - Parâmetro utilizado no subprograma função F (Por exemplo: fluxo de massa).

EMP - Constante utilizada no subprograma função F (Por exemplo: empuxo).

3º CONJUNTO . . . . FORMATO 8F10.4

AMO - Parâmetro utilizado na função F. (Por exemplo: massa inicial).

DMI - Parâmetro utilizado na função F. (Por exemplo: constante gravitacional).

y(1), y(2), ..., y(n) - Variáveis de estados iniciais.

4º CONJUNTO . . . . FORMATO 20I3

ITM(1), ITM(2), ..., ITM(20) - Atualizações onde serão impressos resultados.

5º CONJUNTO FORMATO 8F10.4

DGMAX - Máximo valor admitido para incremento nos parâmetros.

ALFA - Taxa de convergência nas restrições.

EPS3 - Valor mínimo do máximo dos incrementos, entre os incrementos previstos em IP e nas restrições.

EPS2 - Parâmetro que define a fronteira de proximidade. É lido o valor inicial.

RELERR(1), RELERR(2), ..., RELERR(n) - Valores dos erros relativos de truncamento admissíveis na integração numérica.

ABSERR(1), ABSERR(2), ..., ABSERR(n) - Valores dos erros absolutos de truncamento admissíveis na integração numérica.

6º CONJUNTO . . . . FORMATO 8F10.4

SAH - Parâmetro associado à hipótese de linearidade (parâmetro Sa_{max} na equação IV.7).

SAI - Parâmetro associado à hipótese de linearidade (parâmetro Sn na equação IV.7).

7º CONJUNTO . . . . FORMATO 2013

JOTA - Define o tipo de aproximação.

8º CONJUNTO . . . . FORMATO 2013

NCO - Número de parâmetros a serem otimizados.

9º CONJUNTO . . . . FORMATO 8F10.4

P(1), P(2), ..., P(NCO) - Valores iniciais dos parâmetros.

10º CONJUNTO . . . . FORMATO 8F10.4

DP(1), DP(2), ..., DP(NCO) - Incrementos a serem dados aos parâmetros, para efeito de cálculo das derivadas.

11º CONJUNTO . . . . FORMATO 8F10.4

C(1), C(2), ..., C(2NCO+1) - Fatores de custo para montagem do índice de desempenho do problema associado. O valor C(2NCO+1) é inicial.

2.4 - DADOS DE ENTRADA PARA OS TESTES

Neste item serão transcritos os dados de entrada, de todos os testes do Capítulo V. Os valores a seguir foram mantidos para todos os testes.

M = 3 NEQN = 3

NTPRT = 15 DT = .01
RFINAL = 1.523679 AMP = .74800391
EMP = .14012969 AMO = 1.
DMI = 1. Y(1) = 1.
Y(2) = 0. Y(3) = 1.
Vetor ITM = 0,1,3,8,12,16,20,25,30,35,39,40
EPS3 = .1E-03 EPS2 = .1
Vetor RELERR = .1E-04, .1E-04, .1E-04
Vetor ABSERR = .1E-04, .1E-04, .1E-04
SAH = 125 SAI = 5.
DP(i) = .02 para todo i

Aproximação por interpolações lineares (NCO = 7)

DGMAX = .8 ALFA = .7
JOTA = 5 NCO = 7
P(1) = 0 P(2) = 1.
P(3) = 2. P(4) = 3.
P(5) = 4. P(6) = 5.
P(7) = 3.4 C(1) = 1.
C(2) = 1. C(3) = 1.
C(4) = 1. C(5) = 1.
C(6) = 1. C(7) = 1.
C(8) = 1. C(9) = 1.
C(10) = 1. C(11) = 1.
C(12) = 1. C(13) = 1.
C(14) = 1. C(15) = 50.

Aproximação por interpolações lineares (NCO = 9)

DGMAX = 1.12 ALFA = .7
JOTA = 5 NCO = 9
P(1) = 0. P(2) = .7
P(3) = 1.4 P(4) = 2.1
P(5) = 2.8 P(6) = 3.5
P(7) = 4.2 P(8) = 4.9

P(9) = 3.4	C(1) = 1.
C(2) = 1.	C(3) = 1.
C(4) = 1.	C(5) = 1.
C(6) = 1.	C(7) = 1.
C(8) = 1.	C(9) = 1.
C(10) = 1.	C(11) = 1.
C(12) = 1.	C(13) = 1.
C(14) = 1.	C(15) = 1.
C(16) = 1.	C(17) = 1.
C(18) = 1.	C(19) = 50.

Aproximação com uma descontinuidade no controle, com posição fixa

DGMAX = .4	ALFA = .7
JOTA = 2	NCO = 7
P(1) = 1.2	P(2) = 1.2
P(3) = 0.	P(4) = 3.7
P(5) = 1.2	P(6) = 0.
P(7) = 3.4	C(1) = 1.
C(2) = 1.	C(3) = 1.
C(4) = 1.	C(5) = 1.
C(6) = 1.	C(7) = 1.
C(8) = 1.	C(9) = 1.
C(10) = 1.	C(11) = 1.
C(12) = 1.	C(13) = 1.
C(14) = 1.	C(15) = 50.

Aproximação com uma descontinuidade no controlè, com posição otimizada

DGMAX = .6	ALFA = .7
JOTA = 6	NCO = 8
P(1) = 0.	P(2) = 2.5
P(3) = 0.	P(4) = 2.5

P(5) = 2.5	P(6) = 0.
P(7) = 1.7	P(8) = 3.4
C(1) = 1.	C(2) = 1.
C(3) = 1.	C(4) = 1.
C(5) = 1.	C(6) = 1.
C(7) = 5.	C(8) = 1.
C(9) = 1.	C(10) = 1.
C(11) = 1.	C(12) = 1.
C(13) = 1.	C(14) = 1.
C(15) = 5.	C(16) = 1.
C(17) = 10.	

Aproximação em dois trechos parabólicos, com posição de subdivisão otimizada

DGMAX = .6	ALFA = .7
JOTA = 3	NCO = 7
P(1) = 2.5	P(2) = 2.5
P(3) = 0.	P(4) = 1.7
P(5) = 2.5	P(6) = 0.
P(7) = 3.4	C(1) = 1.
C(2) = 1.	C(3) = 1.
C(4) = 5.	C(5) = 1.
C(6) = 1.	C(7) = 1.
C(8) = 1.	C(9) = 1.
C(10) = 1.	C(11) = 5.
C(12) = 1.	C(13) = 1.
C(14) = 1.	C(15) = 20.

Aproximação com parábolas simétricas

DGMAX = 1.	ALFA = 1.
JOTA = 4	NCO = 4
P(1) = 2.5	P(2) = 0.
P(3) = 2.5	P(4) = 3.4

C(1) = 1.	C(2) = 1.
C(3) = 1.	C(4) = 1.
C(5) = 1.	C(6) = 1.
C(7) = 1.	C(8) = 1.
C(9) = 50.	

Aproximação por um polinômio de Tchebyshev (4º ordem)

DGMAX = .6	ALFA = .7
JOTA = 1	NCO = 6
P(1) = 2.5	P(2) = 2.5
P(3) = 0.	P(4) = 0.
P(5) = 0.	P(6) = 3.4
C(1) = 1.	C(2) = 1.
C(3) = 1.	C(4) = 1.
C(5) = 1.	C(6) = 1.
C(7) = 1.	C(8) = 1.
C(9) = 1.	C(10) = 1.
C(11) = 1.	C(12) = 1.
C(13) = 20.	

Aproximação polinomial (4º ordem)

DGMAX = .6	ALFA = .7
JOTA = 7	NCO = 6
P(1) = 2.5	P(2) = 2.5
P(3) = 0.	P(4) = 0.
P(5) = 0.	P(6) = 3.4
C(1) = 1.	C(2) = 1.
C(3) = 1.	C(4) = 1.
C(5) = 1.	C(6) = 1.
C(7) = 1.	C(8) = 1.
C(9) = 1.	C(10) = 1.
C(11) = 1.	C(12) = 1.
C(13) = 20.	

3 - RESULTADOS

3.1 - INTRODUÇÃO

Neste item são apresentados os resultados mais importantes de cada teste, gráficos mostrando as funções de controle parciais e uma saída típica.

Para maior simplicidade de referência e para efeito de preenchimento de tabela, define-se:

Teste 1 - Aproximação por interpolações lineares (NCO = 7).

Teste 2 - Aproximação por interpolações lineares (NCO = 9).

Teste 3 - Aproximação com uma descontinuidade, com posição fixa.

Teste 4 - Aproximação com uma descontinuidade, com posição otimizada.

Teste 5 - Aproximação em dois trechos parabólicos, com posição da subdivisão otimizada.

Teste 6 - Aproximação com parábolas simétricas.

Teste 7 - Aproximação com um polinômio de Tchebyshev (quarta ordem).

Teste 8 - Aproximação polinomial (quarta ordem).

NA_t - Número total de atualizações da matriz de derivadas x_{f_a} ($NA_t = NA_f + 1$).

NA_p - Número parcial de atualizações da matriz de derivadas x_{f_a}.

NA_f - Número da última atualização da matriz de derivadas x_{f_a}.

IP_p - Índice de desempenho parcial.

IP_f - Índice de desempenho final.

IP_o - Índice de desempenho ótimo.

DIP% - Erro em porcentagem no índice de desempenho $(100(IP_f - IP_o) / IP_o)$.

ξ_{p_f} - Valor final de ξ_p.

β_f - Valor final de β.

$c_{f_{2g+1}}$ - Valor final de c_{2g+1}.

PTIME - Tempo de processamento (Burroughs 6700/400k).

r_{f_p} - Valor parcial do raio final.

$v_{r_p}(t_f)$ - Valor parcial de velocidade radial final.

$v_{t_p}(t_f)$ - Valor parcial da velocidade tangencial final.

t_{f_p} - Tempo final parcial.

3.2 - RESULTADOS POR TESTES

A seguir são apresentados os resultados para cada teste, associados a um erro nos valores das equações de restrições da ordem de 1.E-04.

TESTE 1

$NA_t = 14$

$DIP_f\% = 0.20\%$

$c_{f2g+1} = 73.$

$\xi_{p_f} = 1.E-04$

$IP_f = 3.326$

PTIME = 1min 19seg

$\beta_f = .16E-04$

TABELA C.1

RESULTADOS PARCIAIS DO TESTE 1

NA_p	INICIAL	0	1	3	8	FINAL
IP_p	3.400	3.389	3.363	3.240	3.333	3.326
r_{f_p}	1.460	1.292	1.496	1.504	1.523	1.5237
$V_{r_p}(t_f)$	-.1911	-.2245	.0086	.0091	-.8E-04	.0000
$V_{t_p}(t_f)$	.5459	.6809	.7687	.7949	.8102	.8101
P(1)	0.	-1.600	-1.274	.1035	.2571	.3175
P(2)	1.	1.071	1.071	.5002	1.014	.8421
P(3)	2.	2.0	1.468	1.468	.9392	1.041
P(4)	3.	3.0	3.0	4.553	4.810	4.926
P(5)	4.	4.0	5.600	5.753	5.366	5.145
P(6)	5.	6.048	6.048	4.591	5.249	5.497
P(7)	3.4	3.389	3.363	3.240	3.333	3,326

TESTE 2

$NA_t = 15$

$DIP_f\% = .08\%$

$c_{f2g+1} = 143.$

$\xi_{p_f} = 1.E-04$

$IP_f = 3.322$

PTIME = 2min

$\beta_f = .2E-04$

TABELA C.2

RESULTADOS PARCIAIS DO TESTE 2

NA_p	INICIAL	0	1	3	8	FINAL
IP_p	3.4	3.394	3.386	3.511	3.319	3.322
r_{f_p}	1.491	1.347	1.475	1.528	1.521	1.5237
$v_{r_p}(t_f)$	-.1661	-.1975	-.0253	.0047	.4E-03	.0000
$v_{t_p}(t_f)$	.5258	.6361	.7188	.8061	.8087	.8101
P(1)	0.	-1.680	-1.573	-1.101	.3269	.3965
P(2)	.7	.70	.70	.70	.70	.6008
P(3)	1.4	1.4	1.298	1.164	1.012	1.012
P(4)	2.1	2.067	1.688	1.688	1.436	1.336
P(5)	2.8	2.8	2.8	3.370	4.443	4.588
P(6)	3.5	3.5	3.5	5.468	5.412	5.130
P(7)	4.2	4.237	5.917	5.487	5.323	5.257
P(8)	4.9	6.048	6,117	6.093	5.083	5.458
P(9)	3.4	3.394	3.386	3.511	3.319	3.322

TESTE 3

$NA_t = 13$

$DIP_f\% = .11\%$

$c_{f2g+1} = 106.$

$\xi_{p_f} = 4.E-05$

$IP_f = 3.323$

PTIME = 1min 25seg

$\beta_f = .2E-04$

TABELA C.3

RESULTADOS PARCIAIS DO TESTE 3

NA_p	INICIAL	0	1	3	8	FINAL
IP_p	3.4	3.392	3.370	3.394	3.325	3.323
r_{f_p}	1.499	1.414	1.478	1.508	1.525	1.5237
$v_{r_p}(t_f)$	-.1602	-.1606	-.0058	-.0034	.0018	.0000
$v_{t_p}(t_f)$	.5275	.6235	.7483	.8123	.8092	.8101
P(1)	1.2	.9308	.5633	.6670	.9676	1.000
P(2)	1.2	1.2	1.2	.2645	.6500	.6500
P(3)	0.	-.8000	-.7816	-.7816	-.0800	.0247
P(4)	3.7	3.7	4.500	4.564	4.802	4.858
P(5)	1.2	1.916	2.260	1.569	.7846	.7325
P(6)	0.	0.	0.	-.5747	-.4039	-.2536
P(7)	3.4	3.392	3.370	3.394	3.325	3.323

TESTE 4

$NA_t = 10$

$DIP_f\% = .14\%$

$c_{f_{2g+1}} = 256.$

$\xi_{p_f} = 1.E-03$

$IP_f = 3.324$

PTIME = 1min

$\beta_f = .6E-03$

TABELA C.4

RESULTADOS PARCIAIS DO TESTE 4

NA_p	INICIAL	0	1	3	8	FINAL
IP_p	3.4	3.394	3.343	3.252	3.323	3.324
r_{f_p}	1.460	1.467	1.510	1.523	1.524	1.5237
$v_{r_p}(t_f)$	-.1911	-.1598	-.0329	.0116	-.3E-04	.0000
$v_{t_p}(t_f)$	.5459	.5758	.6588	.7611	.8102	.8101
P(1)	0.	0.	-.0988	-.3840	.3173	.3169
P(2)	2.5	2.708	2.708	2.537	1.483	1.483
P(3)	0.	0.	0.	0.	0.	0.
P(4)	2.5	2.036	2.023	3.345	4.116	4.116
P(5)	2.5	2.5	2.877	2.877	2.877	2.877
P(6)	0.	1.2	2.400	.0109	-1.670	-1.670
P(7)	1.7	1.7	1.7	1.7	1.700	1.7
P(8)	3.4	3.394	3.343	3.252	3.323	3.324

TESTE 5

NA_t = 24

$DIP_f\%$ = 1.6%

$c_{f_{2g+1}}$ = 114.

ξ_{p_f} = 1.E-04

IP_f = 3.372

PTIME = 2min 34seg

β_f = .3E-04

TABELA C.5

RESULTADOS PARCIAIS DO TESTE 5

NA_p	INICIAL	1	8	12	20	FINAL
IP_p	3.4	3.381	3.841	3.517	3.377	3.372
r_{f_p}	1.460	1.432	1.524	1.505	1.523	1.5237
$v_{r_p}(t_f)$	-.1911	-.0964	-.1E-04	-.0078	-.1E-03	.0000
$v_{tp}(t_f)$	.5459	.7038	.8100	.8161	.8102	.8101
P(1)	2.5	2.410	2.410	2.566	2.603	2.606
P(2)	2.5	2.518	3.864	4.263	5.802	5.802
P(3)	0.	-1.168	.8139	1.866	3.734	3.734
P(4)	1.7	1.7	1.667	1.658	1.618	1.618
P(5)	2.5	3.643	5.937	6.821	7.907	8.198
P(6)	0.	0.	-1.312	-2.977	-5.050	-5.482
P(7)	3.4	3.381	3.841	3.517	3.377	3.372

TESTE 6

$NA_t = 11$

$DIP_f\% = 1.7\%$

$c_{f2g+1} = 441.$

$\xi_{p_f} = .2E-03$

$IP_f = 3.375$

$PTIME = 43seg$

$\beta_f = .4E-04$

TABELA C.6

RESULTADOS PARCIAIS DO TESTE 6

NA_p	INICIAL	0	1	3	8	FINAL
IP_p	3.4	3.396	3.392	3.356	3.375	3.375
r_{f_p}	1.460	1.410	1.418	1.499	1.524	1.5237
$v_{r_p}(t_f)$	-.1911	-.1869	-.0872	-.0029	-.1E-09	.0000
$v_{t_p}(t_f)$	.5459	.5995	.7287	.8156	.8101	.8101
P(1)	2.5	1.305	3.305	6.328	7.222	7.222
P(2)	0.	-2.000	-.4807	3.519	4.898	4.898
P(3)	2.5	2.480	2.534	2.977	2.970	2.970
P(4)	3.4	3.396	3.392	3.356	3.375	3.375

TESTE 7

$NA_t = 8$ $\qquad$ $IP_f = 3.423$

$DIP_f\% = 3.1\%$ $\qquad$ $PTIME = 48seg$

$c_{f2g+1} = 27.$ $\qquad$ $\beta_f = .2E-04$

$\xi_{p_f} = 1.E-04$

TABELA C.7

RESULTADOS PARCIAIS DO TESTE 7

NA_p	INICIAL	0	1	3	FINAL
IP_p	3.4	3.384	3.353	3.422	3.423
r_{f_p}	1.460	1.290	1.477	1.528	1.5237
$v_{r_p}(t_f)$	-.1911	-.1923	.0503	.0011	.0000
$v_{t_p}(t_f)$	.5459	.6983	.8318	.8066	.8101
P(1)	2.5	2.480	2.835	3.003	3.018
P(2)	2.5	3.700	3.824	3.180	3.142
P(3)	0.	0.	0.	0.	0.
P(4)	0.	.6203	-.4905	-.7654	-.8018
P(5)	0.	0.	0.	-.0902	-.0854
P(6)	3.4	3.384	3.353	3.422	3.423

TESTE 8

$NA_t = 13$

$DIP_f\% = 3.0\%$

$c_{f_2g+1} = 202.$

$\xi_{p_f} = .5E-04$

$IP_f = 3.418$

PTIME = 1min 16seg

$\beta_f = .2E-04$

TABELA C.8

RESULTADOS PARCIAIS DO TESTE 8

NA_p	INICIAL	0	1	3	8	FINAL
IP_p	3.4	3.390	3.371	3.339	3.419	3.418
r_{f_p}	1.460	1.247	1.401	1.464	1.524	1.5237
$v_{r_p}(t_f)$	-.1911	-.2620	-.0891	-.0337	.6E-06	.0000
$v_{t_p}(t_f)$	.5459	.7010	.7325	.7874	.8101	.8101
P(1)	2.5	2.591	2.514	2.757	2.928	2.946
P(2)	2.5	2.5	2.877	4.340	5.702	5.715
P(3)	0.	0.	1.200	1.200	.7831	.6075
P(4)	0.	1.200	1.200	-1.200	-3.542	-3.542
P(5)	0.	-.8017	-.8017	-.8017	-.8017	-.6579
P(6)	3.4	3.390	3.371	3.339	3.419	3.418

As várias funções de controle, mostradas em cada uma das tabelas anteriores, são apresentadas na sequência de gráficos no término deste apêndice (Figuras C.1 a C.8).

3.3 - RESULTADOS COMBINADOS

Para melhor visualização dos resultados, são apresentadas tabelas com os resultados principais para os vários testes. As Figuras C.9 e C.10, do final deste apêndice, apresentam plotadas, no mesmo gráfico, funções de controle finais para vários testes.

TABELA C.9

PRINCIPAIS VALORES DE PARTIDA

TESTE	IP_p	r_{f_p}	$v_{r_p}(t_f)$	$v_{t_p}(t_f)$	c_{2g+1}	β
1	3.4	1.460	-.1911	.5459	50.	.01
2	3.4	1.491	-.1661	.5258	50.	.01
3	3.4	1.499	-.1602	.5275	50.	.01
4	3.4	1.460	-.1911	.5459	10.	.01
5	3.4	1.460	-.1911	.5459	20.	.01
6	3.4	1.460	-.1911	.5459	50.	.01
7	3.4	1.460	-.1911	.5459	20.	.01
8	3.4	1.460	-.1911	.5459	20.	.01

TABELA C.10

PRINCIPAIS VALORES FINAIS

TESTE	IP_f	c_{f2g+1}	NA_t	PTIME	$DIP_f\%$
1	3.326	73.	14	1'19"	.20%
2	3.322	143.	15	2'00"	.08%
3	3.323	106.	13	1'25"	.11%
4	3.324	256.	10	1'00"	.14%
5	3.372	114.	24	2'34"	1.6%
6	3.375	441.	11	0'43"	1.7%
7	3.423	27.	8	0'48"	3.1%
8	3.418	202.	13	1'16"	3.0%

3.4 - GRÁFICOS E UMA SAÍDA TÍPICA DO PROGRAMA

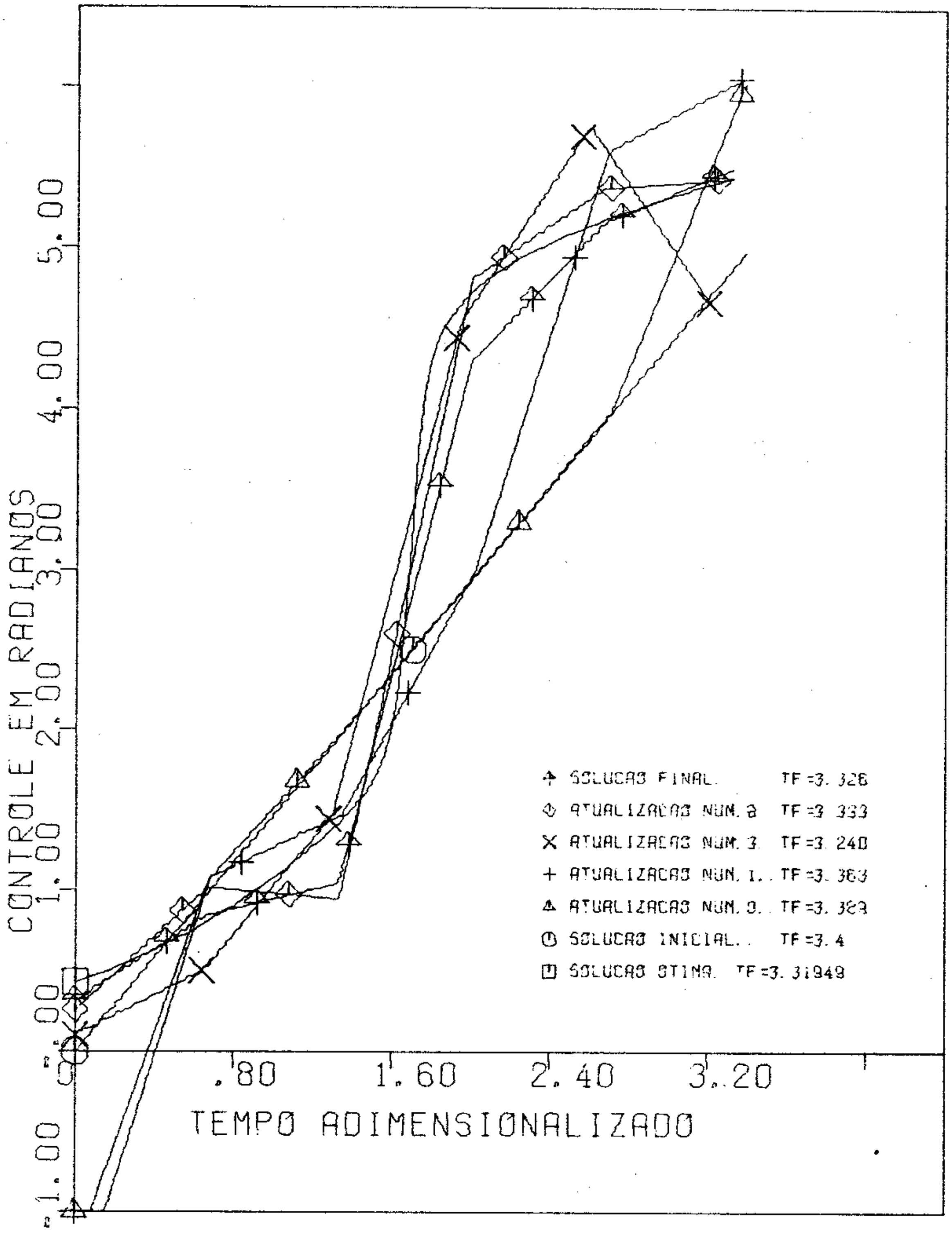

Fig. C.1 - Interpolações lineares

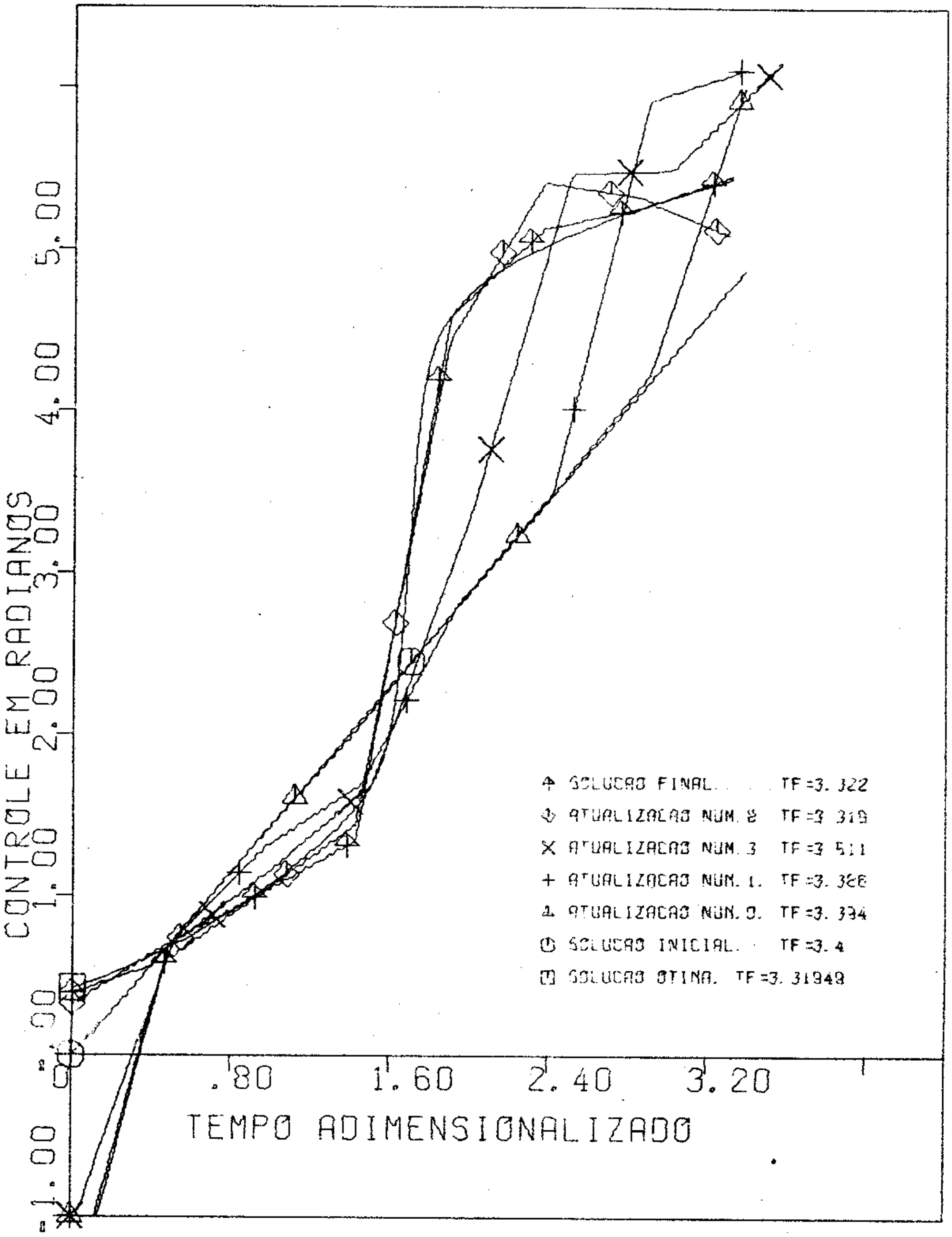

Fig. C.2 - Interpolações lineares

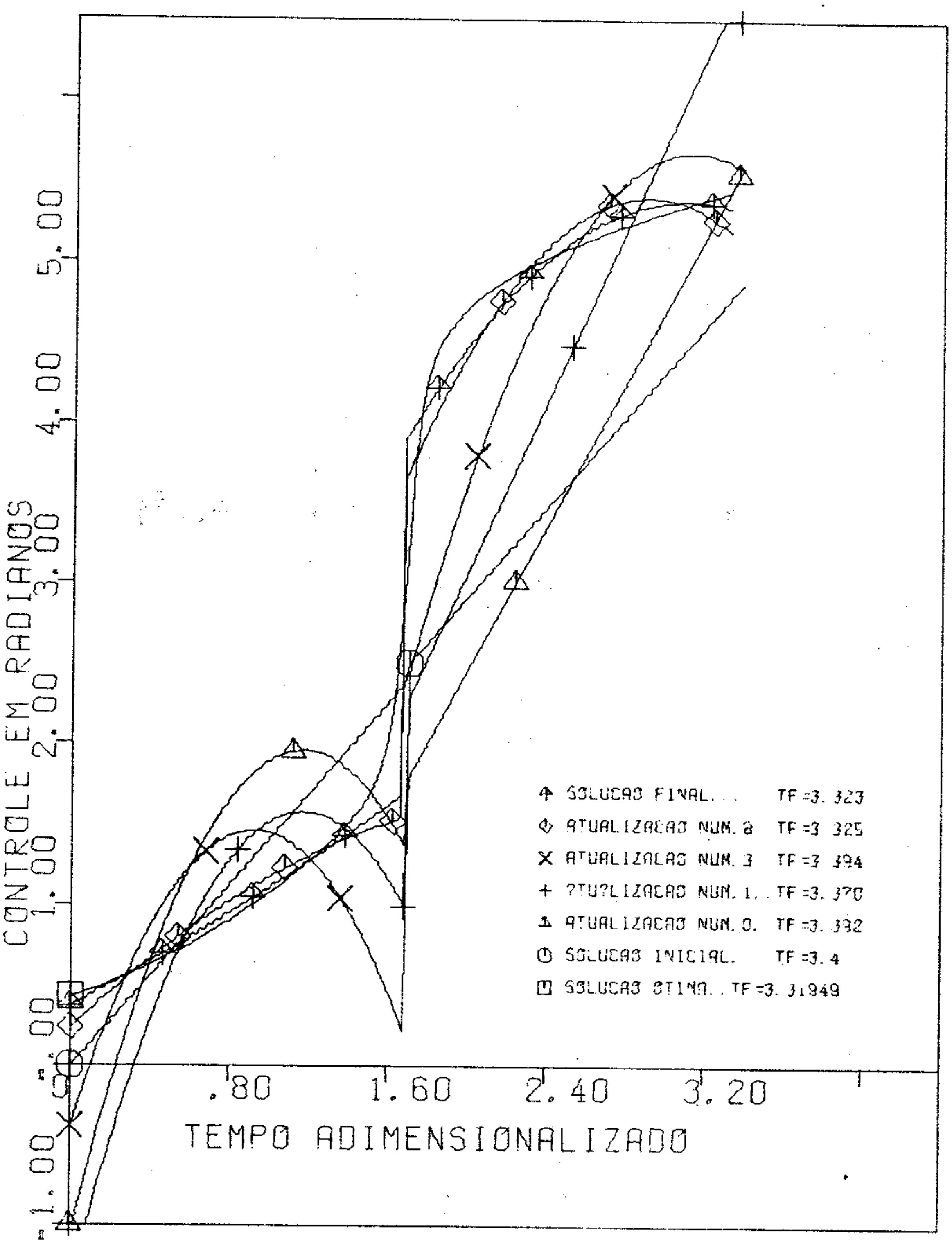

Fig. C.3 - Descontinuidade com posição fixa

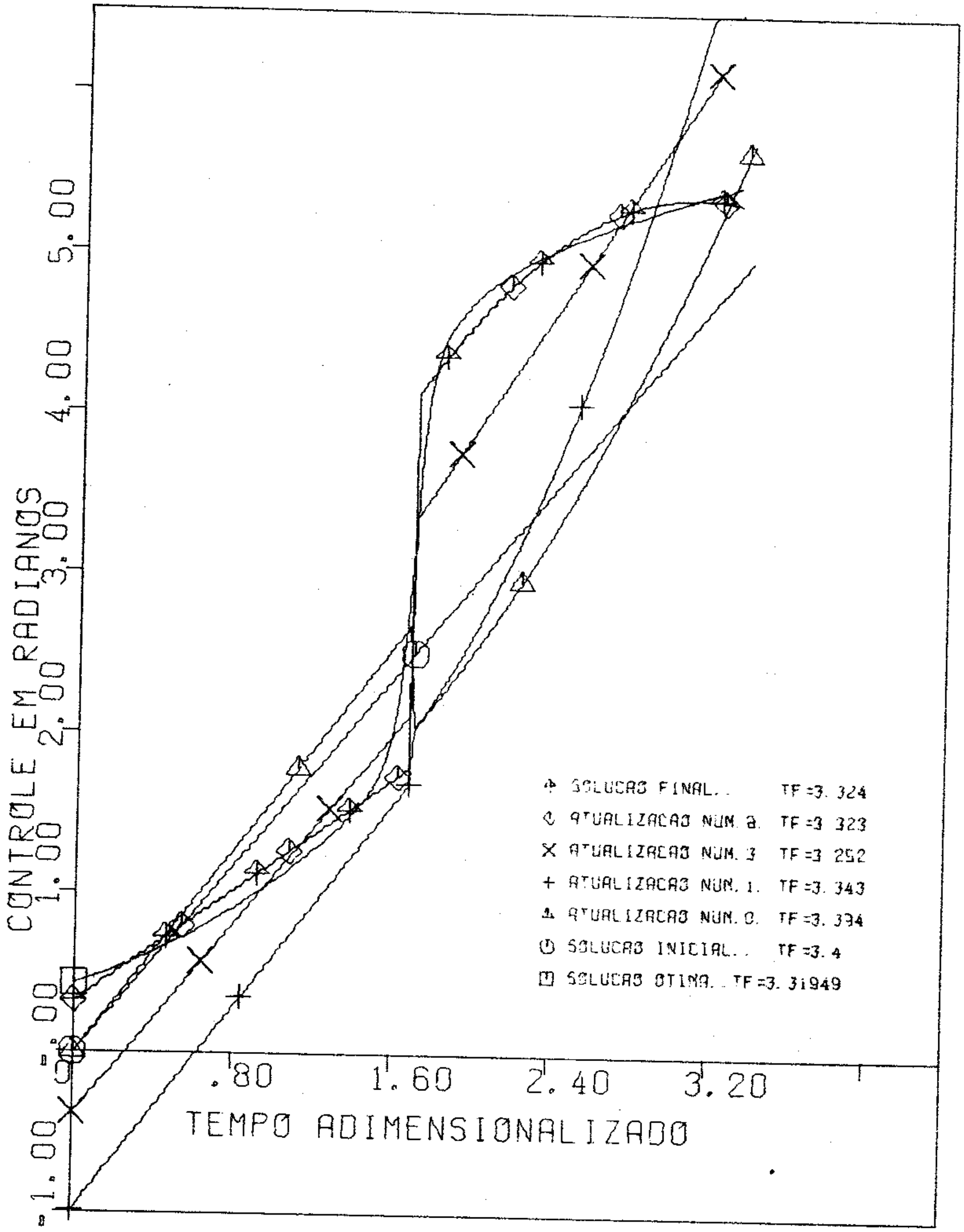

Fig. C.4 - Descontinuidade com posição otimizada

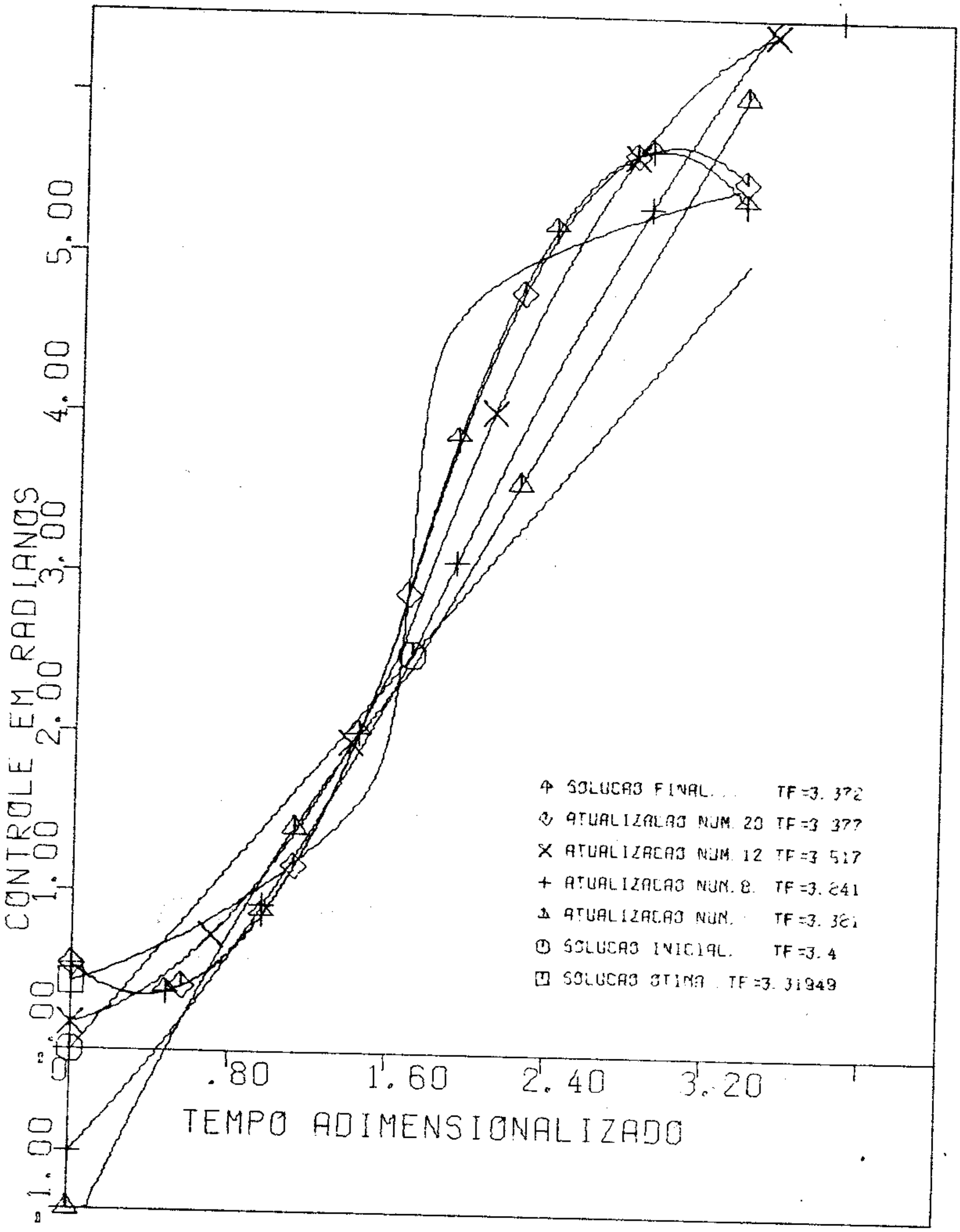

Fig. C.5 - Dois segmentos parabólicos

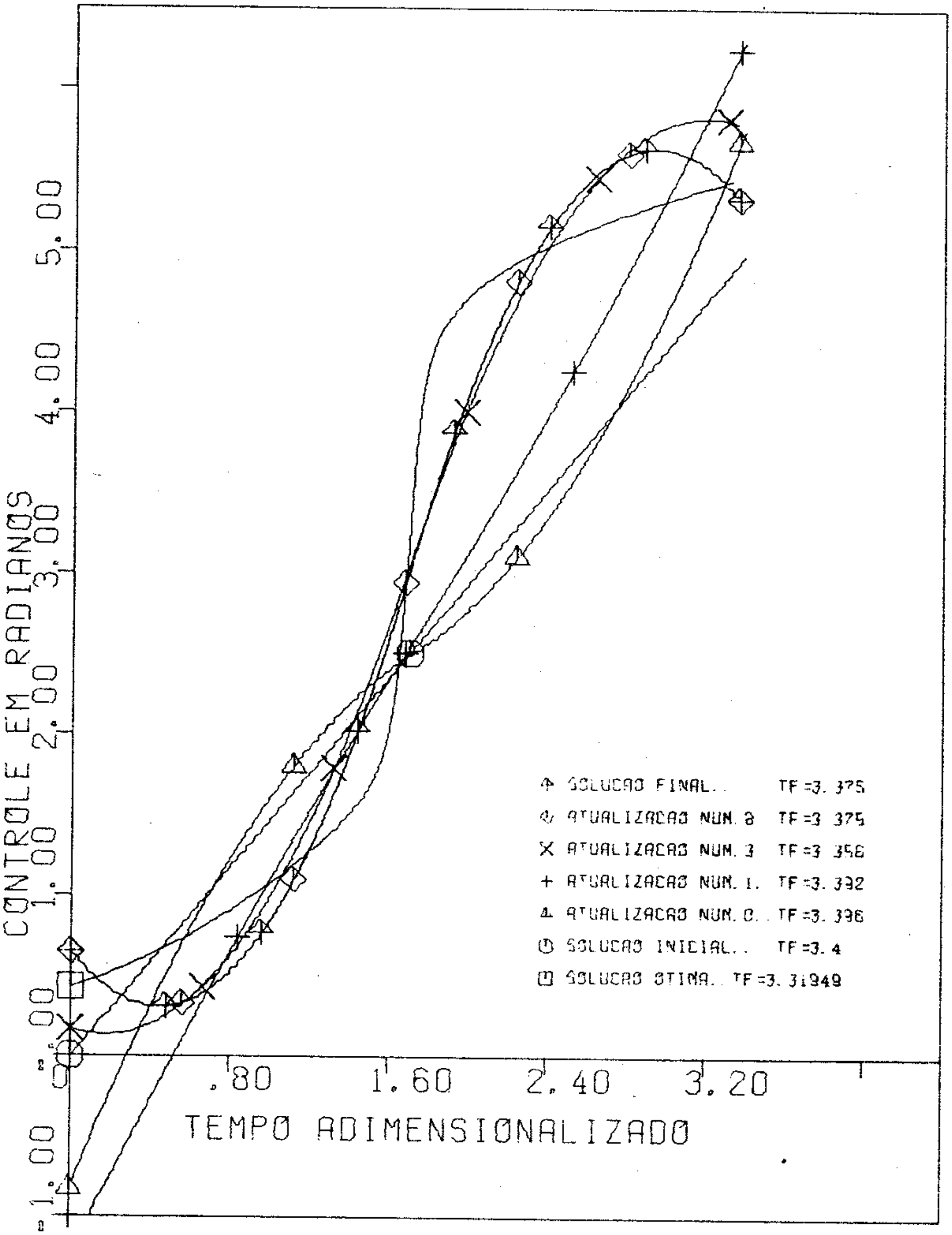

Fig. C.6 - Solução simétrica

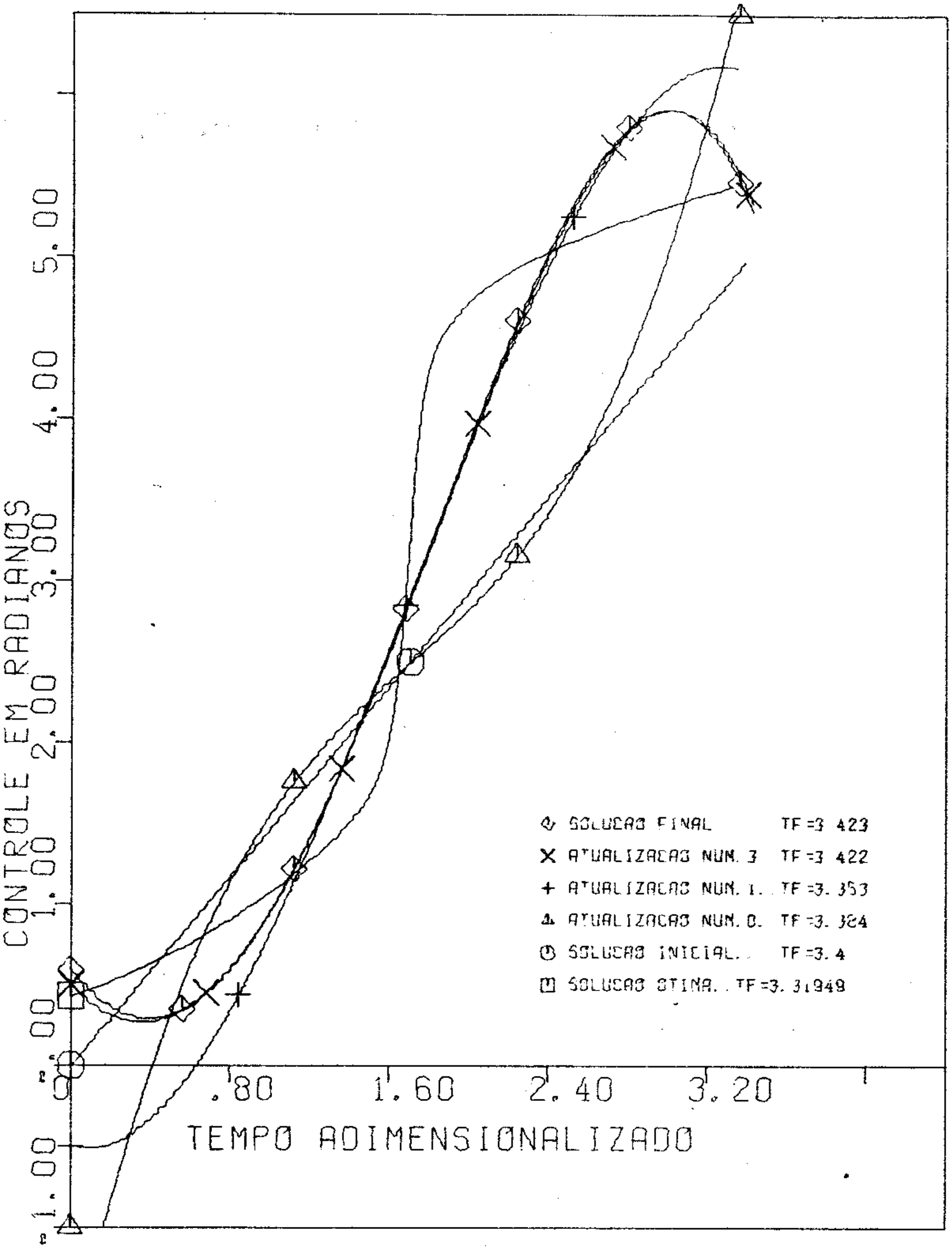

Fig. C.7 - Aproximação de Tchebyshev

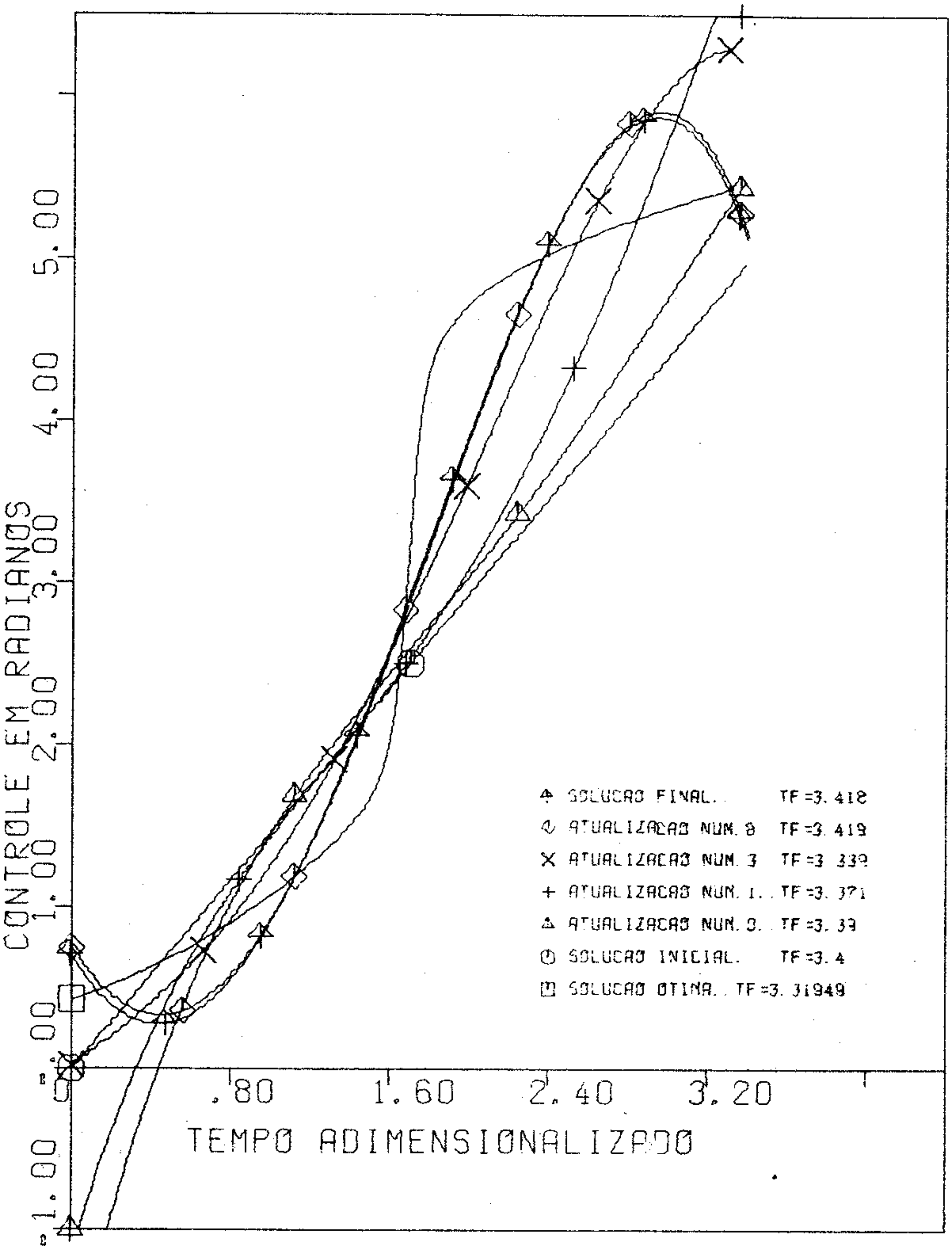

Fig. C.8 - Aproximação polinomial

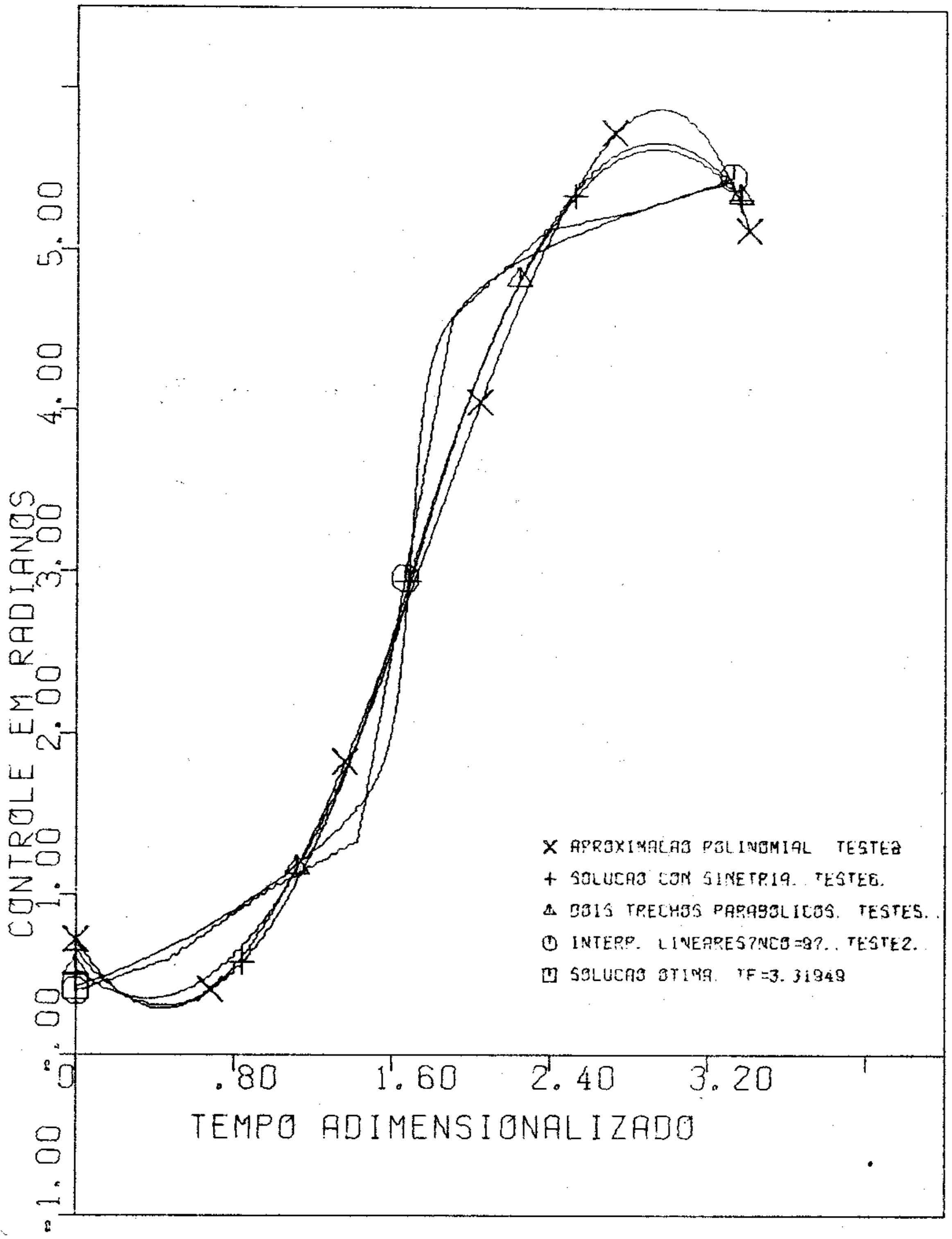

Fig. C.9 - Vārias aproximações do controle

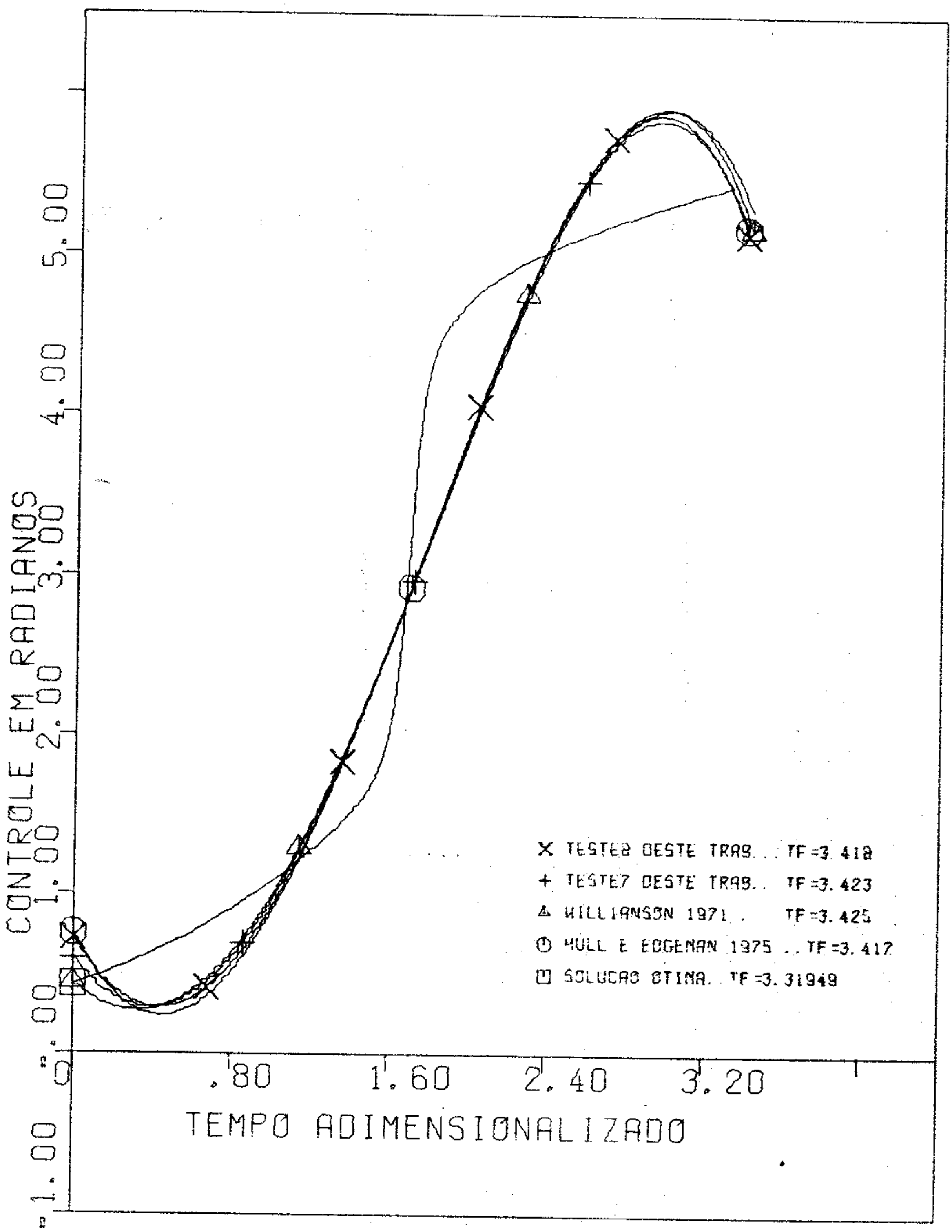

Fig. C.10 - Aproximações polinomiais de quarta ordem

```
                    *****DADOS DE ENTRADA*****
            APROXIMACAO DO TIPO=  2
            NUMERO DE EQUACOES DE RESTRICAO  3
            NUMERO DE PARAMETROS=  7
            NUMERO DE EQUACOES DINAMICAS=  3
            NUMERO DE PASSOS DE INTEGRACAO PARA CADA IMPRESSAO= 15
            CONSTANTE RFINAL UTILIZADA NA SUBROTINA DSV=     1.5237
            CONSTANTE AMP UTILIZADA NA SUBROTINA DE DERIVADAS=     0.0746
            CONSTANTE EMP UTILIZADA NA SUBROTINA DE DERIOADAS=     0.1401
            PASSO INICIAL DE INTEGRACAO=     0.0100
            PARAMETRO DE CONTROLE SOBRE O AVANCO=     0.4000
            AVANCO NAS RESTRICOES=     0.7000
            AVANCO NO MERITO=   -0.0100
            EPS3= .1000E-03
            EPS2= .1000E+00
            *****ERROS NA SUBROTINA RUNGE KUTTA E INCREMENTOS PARA DERIVADAS*****
     .1000E-04 .1000E-04 .1000E-04 .1000E-04 .1000E-04 .1000E-04 .2000E-01 .2000E-01 .2000E-01 .2000E-01
     .2000E-01 .2000E-01 .2000E-01
TEMPO  CONTROLE         X 1        X 2        X 3
0.         ********** .1000E+010.            .1000E+01
 .2267E+00 .3200E+00 .1001E+01 .1229E-01 .1031E+01
 .4533E+00 .6400E+00 .1007E+01 .4776E-01 .1053E+01
 .6800E+00 .9600E+00 .1024E+01 .1009E+00 .1058E+01
 .9067E+00 .1280E+01 .1054E+01 .1626E+00 .1043E+01
 .1133E+01 .1600E+01 .1098E+01 .2223E+00 .1006E+01
 .1360E+01 .1920E+01 .1154E+01 .2707E+00 .9502E+00
 .1587E+01 .2240E+01 .1219E+01 .3013E+00 .8824E+00
 .1813E+01 .2660E+01 .1289E+01 .3094E+00 .8076E+00
 .2040E+01 .2980E+01 .1357E+01 .2932E+00 .7325E+00
 .2267E+01 .3300E+01 .1420E+01 .2552E+00 .6633E+00
 .2493E+01 .3620E+01 .1472E+01 .1977E+00 .6041E+00
 .2720E+01 .3940E+01 .1508E+01 .1235E+00 .5583E+00
 .2947E+01 .4260E+01 .1527E+01 .3621E-01 .5286E+00
 .3173E+01 .4580E+01 .1524E+01-.5990E-01 .5177E+00
 .3400E+01 .4900E+01 .1499E+01-.1602E+00 .5275E+00
     P( 1) .1200E+01P( 2) .1200E+01P( 3)0.        P( 4) .3700E+01P( 5) .1200E+01P( 6)0.        P( 7) .3400E+01P(
```

```
          **********ATUALIZACAO NUMERO=  0 DA MATRIZ DE DERIVADAS(DXA).**********
MINIMIZAR
---------
          C * X
SUJEITOS A
----------
          A * X = B
              E
          X .GE. ZERO

MATRIZ A
--------

          -.445268E+00  -.863958E-01   .198128E+00  -.188552E+00   .960932E-01   .568311E-01  -.751207E-01   .445268E+00
           .863958E-01  -.198128E+00   .188552E+00  -.960932E-01  -.568311E-01   .751207E-01
          -.711514E-01  -.126588E-01   .261607E-01   .157036E+00   .758064E-01  -.589585E-01  -.110511E+00   .711514E-01
           .126588E-01  -.261607E-01  -.157036E+00  -.758064E-01   .589585E-01   .110511E+00
          -.540611E+00  -.127314E+00   .268065E+00  -.218431E+00   .109265E+00   .513926E-01   .283788E+00   .540611E+00
           .127314E+00  -.268065E+00   .218431E+00  -.109265E+00  -.513926E-01  -.283788E+00
          0.            0.            0.            0.            0.            0.             .100000E+01  0.
          0.            0.            0.            0.            0.           -.100000E+01

          0.
          0.
          0.
          -.100000E+01
VETOR C
-------
        .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01
        .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .500000E+02
VETOR B
-------
        .112116E+00   .202493E+00   .172415E-01  -.440000E-01
   MATRIZ AA
   ---------

   X 16   .100000E+01  0.            0.            0.             .112116E+00
   X 17  0.             .100000E+01  0.            0.             .202493E+00
   X 18  0.            0.             .100000E+01  0.             .172415E-01
   X 19  0.            0.            0.             .100000E+01   .440000E-01
     -Z  0.            0.            0.            0.            0.
     -W  0.            0.            0.            0.            -.375850E+00
   VETOR INOVB
   -----------
    16 17 18 19
   VETOR LOCVAR
   ------------
     1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
```

```
**********************************************************************************************************

SOLUCAO OTIMA   Z = .67366138E+01
---------------------------------

N. DE ITERACOES =  5
--------------------

                              VARIAVEIS NA BASE   -    VALORES
                              ------------------------------------
                                      10          -   .29921342E+01
                                       5          -   .27094415E+01
                                       8          -   .99103810E+00
                                      14          -   .44000000E-01

**********************************************************************************************************
TEMPO  CONTROLE           X 1         X 2         X 3
0.          ********** .1000E+010.            .1000E+01
 .2263E+00 .1573E+00 .9995E+00 .1433E-02 .1031E+01
 .4525E+00 .7333E+00 .1003E+01 .3660E-01 .1056E+01
 .6788E+00 .1196E+01 .1018E+01 .9355E-01 .1060E+01
 .9051E+00 .1544E+01 .1046E+01 .1557E+00 .1037E+01
 .1131E+01 .1779E+01 .1088E+01 .2102E+00 .9941E+00
 .1358E+01 .1900E+01 .1140E+01 .2513E+00 .9389E+00
 .1584E+01 .1907E+01 .1200E+01 .2796E+00 .8801E+00
 .1810E+01 .2346E+01 .1266E+01 .2949E+00 .8181E+00
 .2036E+01 .2763E+01 .1332E+01 .2881E+00 .7474E+00
 .2263E+01 .3179E+01 .1394E+01 .2570E+00 .6779E+00
 .2489E+01 .3596E+01 .1447E+01 .2022E+00 .6169E+00
 .2715E+01 .4012E+01 .1484E+01 .1272E+00 .5708E+00
 .2942E+01 .4429E+01 .1503E+01 .3756E-01 .5450E+00
 .3168E+01 .4846E+01 .1501E+01-.5903E-01 .5428E+00
 .3394E+01 .5262E+01 .1476E+01-.1539E+00 .5658E+00
     P( 1) .1068E+01P( 2) .1200E+01P( 3)-.4000E+00P( 4) .3700E+01P( 5) .1562E+01P( 6)0.        P( 7) .3394E+01P(
TEMPO  CONTROLE           X 1         X 2         X 3
0.          ********** .1000E+010.            .1000E+01
 .2257E+00 .1036E-01 .9963E+00-.8558E-02 .1029E+01
 .4515E+00 .8254E+00 .9994E+00 .2473E-01 .1056E+01
 .6772E+00 .1420E+01 .1011E+01 .8268E-01 .1057E+01
 .9030E+00 .1795E+01 .1037E+01 .1418E+00 .1029E+01
 .1129E+01 .1950E+01 .1074E+01 .1881E+00 .9828E+00
 .1354E+01 .1885E+01 .1121E+01 .2213E+00 .9299E+00
 .1580E+01 .1600E+01 .1173E+01 .2456E+00 .8815E+00
 .1806E+01 .1999E+01 .1231E+01 .2637E+00 .8363E+00
 .2032E+01 .2523E+01 .1291E+01 .2664E+00 .7745E+00
 .2257E+01 .3046E+01 .1349E+01 .2439E+00 .7070E+00
 .2483E+01 .3569E+01 .1399E+01 .1931E+00 .6450E+00
 .2709E+01 .4092E+01 .1435E+01 .1175E+00 .5995E+00
 .2935E+01 .4616E+01 .1451E+01 .2598E-01 .5789E+00
 .3160E+01 .5139E+01 .1446E+01-.6878E-01 .5876E+00
 .3386E+01 .5662E+01 .1421E+01-.1530E+00 .6246E+00
     P( 1) .9408E+00P( 2) .1200E+01P( 3)-.7735E+00P( 4) .3700E+01P( 5) .1962E+01P( 6)0.        P( 7) .3386E+01P(
TEMPO  CONTROLE           X 1         X 2         X 3
0.          ********** .1000E+010.            .1000E+01
 .2260E+00-.6843E-02 .9982E+00-.9446E-02 .1028E+01
 .4520E+00 .8252E+00 .9990E+00 .2352E-01 .1056E+01
 .6781E+00 .1430E+01 .1011E+01 .8156E-01 .1057E+01
 .9041E+00 .1806E+01 .1036E+01 .1403E+00 .1029E+01
```

```
.1130E+01 .1956E+01 .1073E+01 .1867E+00 .9827E+00
.1356E+01 .1878E+01 .1119E+01 .2197E+00 .9300E+00
.1582E+01 .1572E+01 .1172E+01 .2441E+00 .8824E+00
.1808E+01 .2021E+01 .1229E+01 .2622E+00 .8375E+00
.2034E+01 .2537E+01 .1289E+01 .2644E+00 .7753E+00
.2260E+01 .3054E+01 .1347E+01 .2413E+00 .7076E+00
.2486E+01 .3571E+01 .1396E+01 .1901E+00 .6458E+00
.2712E+01 .4088E+01 .1431E+01 .1142E+00 .6004E+00
.2938E+01 .4604E+01 .1446E+01 .2240E-01 .5798E+00
.3164E+01 .5121E+01 .1441E+01-.7302E-01 .5883E+00
.3390E+01 .5638E+01 .1414E+01-.1585E+00 .6252E+00
     P( 1) .9336E+00P( 2) .1200E+01P( 3)-.8000E+00P( 4) .3700E+01P( 5) .1938E+01P( 6)0.        P( 7) .3390E+01P(
TEMPO  CONTROLE          X 1         X 2         X 3
0.         *********** .1000E+010.              .1000E+01
 .2262E+00-.9620E-02 .9981E+00-.9534E-02 .1028E+01
 .4523E+00 .8224E+00 .9990E+00 .2338E-01 .1056E+01
 .6785E+00 .1427E+01 .1011E+01 .8146E-01 .1057E+01
 .9046E+00 .1804E+01 .1036E+01 .1403E+00 .1030E+01
 .1131E+01 .1953E+01 .1073E+01 .1869E+00 .9829E+00
 .1357E+01 .1875E+01 .1119E+01 .2201E+00 .9302E+00
 .1583E+01 .1569E+01 .1172E+01 .2446E+00 .8826E+00
 .1809E+01 .2039E+01 .1230E+01 .2627E+00 .8373E+00
 .2035E+01 .2550E+01 .1290E+01 .2645E+00 .7746E+00
 .2262E+01 .3061E+01 .1347E+01 .2409E+00 .7069E+00
 .2488E+01 .3572E+01 .1396E+01 .1894E+00 .6451E+00
 .2714E+01 .4083E+01 .1431E+01 .1134E+00 .5997E+00
 .2940E+01 .4594E+01 .1447E+01 .2146E-01 .5789E+00
 .3166E+01 .5105E+01 .1440E+01-.7431E-01 .5870E+00
 .3392E+01 .5616E+01 .1414E+01-.1606E+00 .6235E+00
     P( 1) .9308E+00P( 2) .1200E+01P( 3)-.8000E+00P( 4) .3700E+01P( 5) .1916E+01P( 6)0.        P( 7) .3392E+01P(
```

```
     *********ATUALIZACAO NUMERO=  1 DA MATRIZ DE DERIVADAS(DXA).**********
MINIMIZAR
---------
          C * X
SUJEITOS A
----------
          A * X = B
              E
          X .GE. ZERO

MATRIZ A
--------

          -.440658E+00  -.133130E+00   .309565E+00  -.148869E+00   .145958E+00   .571893E-01  -.911883E-01   .440658E+00
           .133130E+00  -.309565E+00   .148869E+00  -.145958E+00  -.571893E-01   .911883E-01
          -.543857E-01  -.110175E-01  -.883442E-02   .123850E+00   .880144E-01  -.678704E-01  -.813246E-01   .543857E-01
           .110175E-01   .883442E-02  -.123850E+00  -.880144E-01   .678704E-01   .813246E-01
          -.529412E+00  -.206572E+00   .483281E+00  -.203538E+00   .114428E+00   .540156E-01   .222255E+00   .529412E+00
           .206572E+00  -.483281E+00   .203538E+00  -.114428E+00  -.540156E-01  -.222255E+00
          0.            0.            0.            0.            0.            0.             .100000E+01  0.
          0.            0.            0.            0.            0.           -.100000E+01


          0.
          0.
          0.
          -.100000E+01

VETOR C
-------
       .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01
       .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .180000E+02

VETOR B
-------
       .112435E+00   .152317E+00   .770359E-01  -.173937E-01

   MATRIZ AA
   ---------

   X 16   .100000E+01  0.            0.            0.             .112435E+00
   X 17  0.             .100000E+01  0.            0.             .152317E+00
   X 18  0.            0.             .100000E+01  0.             .770359E-01
   X 19  0.            0.            0.             .100000E+01   .173937E-01
     -Z  0.            0.            0.            0.            0.
     -W  0.            0.            0.            0.           -.359182E+00

   VETOR INDVB
   -----------
    16 17 18 19

   VETOR LOCVAR
   ------------
    1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
```

```
*************************************************************************************************************

SOLUCAO OTIMA   Z = .15657505E+01
-----------------------------------

N. DE ITERACOES =  5
--------------------

                                        VARIAVEIS NA BASE   -   VALORES
                                        ---------------------------------
                                                5           -  .58338461E+00
                                                4           -  .67770172E+00
                                                8           -  .28727045E+00
                                               14           -  .17393714E-01

*************************************************************************************************************
TEMPO   CONTROLE          X 1        X 2        X 3
0.       ********** .1000E+010.              .1000E+01
 .2255E+00-.1792E+00 .9977E+00-.1457E-01 .1026E+01
 .4509E+00 .6528E+00 .9967E+00 .1240E-01 .1057E+01
 .6764E+00 .1257E+01 .1006E+01 .6917E-01 .1066E+01
 .9019E+00 .1634E+01 .1028E+01 .1324E+00 .1046E+01
 .1127E+01 .1763E+01 .1065E+01 .1872E+00 .1005E+01
 .1353E+01 .1705E+01 .1112E+01 .2302E+00 .9558E+00
 .1578E+01 .1394E+01 .1168E+01 .2638E+00 .9102E+00
 .1804E+01 .2141E+01 .1230E+01 .2893E+00 .8615E+00
 .2029E+01 .2744E+01 .1296E+01 .2934E+00 .7903E+00
 .2255E+01 .3347E+01 .1360E+01 .2656E+00 .7172E+00
 .2480E+01 .3949E+01 .1414E+01 .2067E+00 .6575E+00
 .2706E+01 .4552E+01 .1452E+01 .1265E+00 .6235E+00
 .2931E+01 .5155E+01 .1471E+01 .4127E-01 .6211E+00
 .3157E+01 .5758E+01 .1471E+01-.3109E-01 .6481E+00
 .3382E+01 .6360E+01 .1459E+01-.7603E-01 .6941E+00
    P( 1) .7613E+00P( 2) .1200E+01P( 3)-.8000E+00P( 4) .4100E+01P( 5) .2260E+01P( 6)0.        P( 7) .3382E+01P(
TEMPO   CONTROLE          X 1        X 2        X 3
0.       ********** .1000E+010.              .1000E+01
 .2246E+00-.3758E+00 .9972E+00-.1964E-01 .1022E+01
 .4493E+00 .4445E+00 .9942E+00-.5971E-03 .1057E+01
 .6739E+00 .1042E+01 .9996E+00 .5242E-01 .1074E+01
 .8986E+00 .1418E+01 .1019E+01 .1188E+00 .1065E+01
 .1123E+01 .1571E+01 .1053E+01 .1822E+00 .1032E+01
 .1348E+01 .1502E+01 .1100E+01 .2360E+00 .9884E+00
 .1573E+01 .1211E+01 .1158E+01 .2800E+00 .9455E+00
 .1797E+01 .2541E+01 .1225E+01 .3104E+00 .8887E+00
 .2022E+01 .3144E+01 .1295E+01 .3078E+00 .8069E+00
 .2246E+01 .3747E+01 .1360E+01 .2695E+00 .7337E+00
 .2471E+01 .4349E+01 .1414E+01 .2040E+00 .6832E+00
 .2696E+01 .4952E+01 .1451E+01 .1268E+00 .6634E+00
 .2920E+01 .5555E+01 .1471E+01 .5643E-01 .6745E+00
 .3145E+01 .6158E+01 .1476E+01 .9149E-02 .7078E+00
 .3370E+01 .6760E+01 .1476E+01-.5625E-02 .7483E+00
    P( 1) .5633E+00P( 2) .1200E+01P( 3)-.7816E+00P( 4) .4500E+01P( 5) .2260E+01P( 6)0.        P( 7) .3370E+01P(
```

```
          **********ATUALIZACAO NUMERO=  3 DA MATRIZ DE DERIVADAS(DXA).**********
MINIMIZAR
---------
          C * X
SUJEITOS A
----------
          A * X = B
              E
          X .GE. ZERO

MATRIZ A
--------

     -.282075E+00  -.531711E-01   .209064E+00   .159085E+00   .996455E-01  -.130741E+00  -.445013E-01   .282075E+00
      .531711E-01  -.209064E+00  -.159085E+00  -.996455E-01   .130741E+00   .445013E-01
     -.500207E-01  -.171578E-01  -.547500E-02   .170030E+00   .176249E-01  -.813543E-01  -.120439E-01   .500207E-01
      .171578E-01   .547500E-02  -.170030E+00  -.176249E-01   .813543E-01   .120439E-01
     -.303320E+00  -.976203E-01   .356922E+00  -.583570E-02   .831219E-01  -.863992E-01   .294013E+00   .303320E+00
      .976203E-01  -.356922E+00   .583570E-02  -.831219E-01   .863992E-01  -.294013E+00
     0.            0.            0.            0.            0.            0.             .100000E+01  0.
     0.            0.            0.            0.            0.           -.100000E+01

     0.
     0.
     0.
    -.100000E+01

VETOR C
-------
      .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01
      .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .180000E+02
VETOR B
-------
     -.732858E-02   .118580E-01   .764392E-02  -.250987E-01

  MATRIZ AA
  ---------

  X 16   .100000E+01  0.            0.            0.             .732858E-02
  X 17  0.             .100000E+01  0.            0.             .118580E-01
  X 18  0.            0.             .100000E+01  0.             .764392E-02
  X 19  0.            0.            0.             .100000E+01   .250987E-01
    -Z  0.            0.            0.            0.            0.
    -W  0.            0.            0.            0.           -.519292E-01

  VETOR INDVB
  -----------
   16 17 18 19

  VETOR LOCVAR
  ------------
   1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
```

```
****************************************************************************************************

SOLUCAO OTIMA   Z = .10886337E+01
---------------------------------

N. DE ITERACOES =  7
--------------------

                              VARIAVEIS NA BASE    -    VALORES
                              --------------------------------------
                                      1            -   .20906705E+00
                                      4            -   .48096346E-01
                                      9            -   .80637161E+00
                                     14            -   .25098704E-01

****************************************************************************************************
TEMPO     CONTROLE        X 1        X 2        X 3
0.        **********  .1000E+010.             .1000E+01
 .2211E+00 .3949E+00 .1000E+01 .7250E-02 .1030E+01
 .4422E+00 .9726E+00 .1006E+01 .4649E-01 .1048E+01
 .6633E+00 .1328E+01 .1022E+01 .1004E+00 .1044E+01
 .8844E+00 .1461E+01 .1050E+01 .1549E+00 .1021E+01
 .1106E+01 .1372E+01 .1090E+01 .2036E+00 .9865E+00
 .1327E+01 .1060E+01 .1139E+01 .2449E+00 .9564E+00
 .1548E+01 .5262E+00 .1197E+01 .2763E+00 .9332E+00
 .1769E+01 .2876E+01 .1261E+01 .2962E+00 .8869E+00
 .1990E+01 .3744E+01 .1325E+01 .2806E+00 .8102E+00
 .2211E+01 .4448E+01 .1382E+01 .2301E+00 .7570E+00
 .2432E+01 .4989E+01 .1426E+01 .1680E+00 .7344E+00
 .2653E+01 .5366E+01 .1457E+01 .1105E+00 .7364E+00
 .2874E+01 .5580E+01 .1476E+01 .6342E-01 .7540E+00
 .3095E+01 .5631E+01 .1486E+01 .2583E-01 .7804E+00
 .3317E+01 .5518E+01 .1488E+01-.6179E-02 .8105E+00
     P( 1) .6670E+00P( 2) .2905E+00P( 3)-.7816E+00P( 4) .4524E+01P( 5) .1569E+01P( 6)-.5747E+00P( 7) .3317E+01P(
TEMPO     CONTROLE        X 1        X 2        X 3
0.        **********  .1000E+010.             .1000E+01
 .2201E+00 .7592E+00 .1002E+01 .1915E-01 .1026E+01
 .4402E+00 .1229E+01 .1010E+01 .6098E-01 .1034E+01
 .6603E+00 .1477E+01 .1029E+01 .1091E+00 .1021E+01
 .8804E+00 .1503E+01 .1058E+01 .1543E+00 .9953E+00
 .1100E+01 .1308E+01 .1096E+01 .19? E+00 .9652E+00
 .1321E+01 .8904E+00 .1143E+01 .2264E+00 .9405E+00
 .1541E+01 .2513E+00 .1195E+01 .2479E+00 .9269E+00
 .1761E+01 .2876E+01 .1251E+01 .2610E+00 .8872E+00
 .1981E+01 .3744E+01 .1307E+01 .2447E+00 .8155E+00
 .2201E+01 .4448E+01 .1356E+01 .1940E+00 .7663E+00
 .2421E+01 .4989E+01 .1392E+01 .1319E+00 .7474E+00
 .2641E+01 .5366E+01 .1415E+01 .7450E-01 .7528E+00
 .2861E+01 .5580E+01 .1426E+01 .2784E-01 .7740E+00
 .3081E+01 .5631E+01 .1428E+01-.8742E-02 .8042E+00
 .3301E+01 .5518E+01 .1422E+01-.3889E-01 .8383E+00
     P( 1) .7378E+00P( 2)-.1095E+00P( 3)-.7797E+00P( 4) .4524E+01P( 5) .1569E+01P( 6)-.5747E+00P( 7) .3301E+01P(
TEMPO     CONTROLE        X 1        X 2        X 3
0.        **********  .1000E+010.             .1000E+01
 .2205E+00 .7569E+00 .1002E+01 .1911E-01 .1026E+01
 .4411E+00 .1228E+01 .1010E+01 .6106E-01 .1034E+01
 .6616E+00 .1477E+01 .1029E+01 .1894E+00 .1021E+01
 .8821E+00 .1503E+01 .1058E+01 .1547E+00 .9953E+00
```

```
 .1103E+01 .1307E+01 .1097E+01 .1941E+00 .9651E+00
 .1323E+01 .8888E+00 .1143E+01 .2270E+00 .9403E+00
 .1544E+01 .2482E+00 .1196E+01 .2485E+00 .9266E+00
 .1764E+01 .2878E+01 .1252E+01 .2615E+00 .8860E+00
 .1985E+01 .3746E+01 .1309E+01 .2451E+00 .8149E+00
 .2205E+01 .4451E+01 .1358E+01 .1942E+00 .7658E+00
 .2426E+01 .4991E+01 .1394E+01 .1320E+00 .7469E+00
 .2646E+01 .5369E+01 .1416E+01 .7457E-01 .7524E+00
 .2867E+01 .5583E+01 .1427E+01 .2794E-01 .7738E+00
 .3087E+01 .5633E+01 .1429E+01-.6554E-02 .8040E+00
 .3308E+01 .5520E+01 .1424E+01-.3858E-01 .8383E+00
     P( 1) .7356E+00P( 2)-.1095E+00P( 3)-.7816E+00P( 4) .4526E+01P( 5) .1569E+01P( 6)-.5747E+00P( 7) .3308E+01P(
TEMPO  CONTROLE          X 1        X.2        X 3
0.         *********** .1000E+010.             .1000E+01
 .2263E+00 .4140E+00 .1000E+01 .8300E-02 .1031E+01
 .4525E+00 .9848E+00 .1006E+01 .4917E-01 .1049E+01
 .6788E+00 .1333E+01 .1024E+01 .1046E+00 .1044E+01
 .9051E+00 .1459E+01 .1054E+01 .1602E+00 .1019E+01
 .1131E+01 .1363E+01 .1096E+01 .2095E+00 .9846E+00
 .1358E+01 .1045E+01 .1148E+01 .2511E+00 .9515E+00
 .1584E+01 .5037E+00 .1209E+01 .2823E+00 .9279E+00
 .1810E+01 .2916E+01 .1275E+01 .3011E+00 .8802E+00
 .2036E+01 .3784E+01 .1342E+01 .2831E+00 .8021E+00
 .2263E+01 .4458E+01 .1400E+01 .2302E+00 .7494E+00
 .2489E+01 .5029E+01 .1445E+01 .1667E+00 .7284E+00
 .2715E+01 .5406E+01 .1476E+01 .1090E+00 .7324E+00
 .2942E+01 .5620E+01 .1495E+01 .6272E-01 .7520E+00
 .3168E+01 .5671E+01 .1505E+01 .2666E-01 .7803E+00
 .3394E+01 .5558E+01 .1508E+01-.3363E-02 .8123E+00
     P( 1) .6670E+00P( 2) .2645E+00P( 3)-.7816E+00P( 4) .4564E+01P( 5) .1569E+01P( 6)-.5747E+00P( 7) .3394E+01P(
```

```
MINIMIZAR**********ATUALIZACAO NUMERO=  8 DA MATRIZ DE DERIVADAS(DXA).**********
---------
           C * X
SUJEITOS A
----------
           A * X = B
               E
           X .GE. ZERO

MATRIZ A
--------

      -.337539E+00  -.353816E-01   .129078E+00   .270469E+00  -.174945E-01  -.130445E+00  -.500940E-01   .337539E+00
       .353816E-01  -.129078F+00  -.270469E+00   .174945E-01   .130445E+00   .500940E-01
      -.395901E-01  -.241118E-01   .709995E-02   .207407E+00   .190974E-01  -.607758E-01  -.646334E-02   .395901E-01
       .241118E-01  -.709995E-02  -.207407E+00  -.190974E-01   .607758E-01   .646334E-02
      -.433399E+00  -.245624E-01   .169597E+00   .139621E+00  -.214848E-01  -.825536E-01   .293051E+00   .433399E+00
       .245624E-01  -.169597E+00  -.139621E+00   .214848E-01   .825536E-01  -.293051E+00
      0.            0.            0.            0.            0.            0.             .100000E+01  0.
      0.            0.            0.            0.            0.           -.100000E+01

      0.
      0.
      0.
     -.100000E+01
VETOR C
-------
     .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01
     .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .118098E+02
VETOR B
-------
    -.672765E-02   .172731E-02  -.592076E-02  -.175801E-02

  MATRIZ AA
  ---------

  X 16   .100000E+01  0.            0.            0.             .672765E-02
  X 17  0.            .100000E+01  0.            0.             .172731E-02
  X 18  0.           0.             .100000E+01  0.             .592076E-02
  X 19  0.           0.            0.             .100000E+01   .175801E-02
    -Z  0.           0.            0.            0.            0.
    -W  0.           0.            0.            0.           -.161337E-01

  VETOR INDVB
  -----------
   16 17 18 19

  VETOR LOCVAR
  ------------
   1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
```

```
***************************************************************************************************************

SOLUCAO OTIMA   Z = .23383057E+00
------------------------------------

N. DE ITERACOES =  7
--------------------

                                              VARIAVEIS NA BASE   -    VALORES
                                              ---------------------------------
                                                      7           -   .58852132E-03
                                                      5           -   .17806171E+00
                                                      6           -   .27468223E-01
                                                     15           -   .23465350E-02

***************************************************************************************************************
TEMPO  CONTROLE        X 1        X 2        X 3
0.         ********** .1000E+010.            .1000E+01
 .2215E+00 .4813E+00 .1002E+01 .1750E-01 .1028E+01
 .4430E+00 .7059E+00 .1009E+01 .5335E-01 .1046E+01
 .6645E+00 .9077E+00 .1026E+01 .1025E+00 .1051E+01
 .8859E+00 .1067E+01 .1055E+01 .1586E+00 .1040E+01
 .1107E+01 .1243E+01 .1096E+01 .2152E+00 .1013E+01
 .1329E+01 .1376E+01 .1150E+01 .2677E+00 .9745E+00
 .1550E+01 .1487E+01 .1215E+01 .3132E+00 .9274E+00
 .1772E+01 .4015E+01 .1287E+01 .3204E+00 .8634E+00
 .1993E+01 .4518E+01 .1353E+01 .2773E+00 .8063E+00
 .2215E+01 .4902E+01 .1409E+01 .2244E+00 .7746E+00
 .2436E+01 .5169E+01 .1453E+01 .1721E+00 .7635E+00
 .2658E+01 .5317E+01 .1485E+01 .1247E+00 .7662E+00
 .2879E+01 .5347E+01 .1508E+01 .8226E-01 .7775E+00
 .3101E+01 .5259E+01 .1522E+01 .4294E-01 .7929E+00
 .3322E+01 .5053E+01 .1527E+01 .4184E-02 .8081E+00
     P( 1) .9641E+00P( 2) .6500E+00P( 3)-.8001E-01P( 4) .4802E+01P( 5) .6670E+00P( 6)-.4155E+00P( 7) .3322E+01P(
TEMPO  CONTROLE        X 1        X 2        X 3
0.         ********** .1000E+010.            .1000E+01
 .2215E+00 .4802E+00 .1002E+01 .1747E-01 .1028E+01
 .4430E+00 .7047E+00 .1009E+01 .5330E-01 .1046E+01
 .6645E+00 .9065E+00 .1026E+01 .1025E+00 .1051E+01
 .8859E+00 .1086E+01 .1055E+01 .1586E+00 .1040E+01
 .1107E+01 .1242E+01 .1096E+01 .2153E+00 .1013E+01
 .1329E+01 .1375E+01 .1150E+01 .2677E+00 .9747E+00
 .1550E+01 .1486E+01 .1215E+01 .3133E+00 .9276E+00
 .1772E+01 .3949E+01 .1287E+01 .3214E+00 .8628E+00
 .1993E+01 .4463E+01 .1354E+01 .2789E+00 .8036E+00
 .2215E+01 .4863E+01 .1409E+01 .2252E+00 .7704E+00
 .2436E+01 .5149E+01 .1453E+01 .1715E+00 .7582E+00
 .2658E+01 .5321E+01 .1486E+01 .1227E+00 .7609E+00
 .2879E+01 .5379E+01 .1508E+01 .7961E-01 .7731E+00
 .3101E+01 .5322E+01 .1521E+01 .4053E-01 .7904E+00
 .3322E+01 .5152E+01 .1526E+01 .2994E-02 .8086E+00
     P( 1) .9629E+00P( 2) .6500E+00P( 3)-.8001E-01P( 4) .4802E+01P( 5) .7514E+00P( 6)-.4014E+00P( 7) .3322E+01P(
TEMPO  CONTROLE        X 1        X 2        X 3
0.         ********** .1000E+010.            .1000E+01
 .2215E+00 .4813E+00 .1002E+01 .1750E-01 .1028E+01
 .4431E+00 .7059E+00 .1009E+01 .5337E-01 .1046E+01
 .6646E+00 .9077E+00 .1026E+01 .1026E+00 .1051E+01
 .8861E+00 .1087E+01 .1055E+01 .1586E+00 .1040E+01
```

```
 .1108E+01 .1243E+01 .1097E+01 .2153E+00 .1013E+01
 .1329E+01 .1376E+01 .1150E+01 .2677E+00 .9745E+00
 .1551E+01 .1487E+01 .1215E+01 .3133E+00 .9273E+00
 .1772E+01 .3952E+01 .1287E+01 .3213E+00 .8626E+00
 .1994E+01 .4467E+01 .1354E+01 .2787E+00 .8036E+00
 .2215E+01 .4867E+01 .1409E+01 .2249E+00 .7705E+00
 .2437E+01 .5152E+01 .1453E+01 .1713E+00 .7585E+00
 .2658E+01 .5322E+01 .1486E+01 .1226E+00 .7612E+00
 .2880E+01 .5378E+01 .1508E+01 .7956E-01 .7734E+00
 .3101E+01 .5318E+01 .1521E+01 .4049E-01 .7906E+00
 .3323E+01 .5144E+01 .1526E+01 .2853E-02 .8087E+00
     P( 1) .9641E+00P( 2) .6500E+00P( 3)-.8001E-01P( 4) .4802E+01P( 5) .7460E+00P( 6)-.4039E+00P( 7) .3323E+01P(
TEMPO  CONTROLE         X 1        X 2        X 3
0.         ********** .1000E+010.             .1000E+01
 .2215E+00 .4813E+00 .1002E+01 .1750E-01 .1028E+01
 .4431E+00 .7059E+00 .1009E+01 .5337E-01 .1046E+01
 .6646E+00 .9077E+00 .1026E+01 .1026E+00 .1051E+01
 .8861E+00 .1087E+01 .1055E+01 .1566E+00 .1040E+01
 .1108E+01 .1243E+01 .1097E+01 .2153E+00 .1013E+01
 .1329E+01 .1376E+01 .1150E+01 .2677E+00 .9745E+00
 .1551E+01 .1487E+01 .1215E+01 .3133E+00 .9273E+00
 .1772E+01 .3952E+01 .1287E+01 .3213E+00 .8626E+00
 .1994E+01 .4467E+01 .1354E+01 .2787E+00 .8036E+00
 .2215E+01 .4867E+01 .1409E+01 .2249E+00 .7705E+00
 .2437E+01 .5152E+01 .1453E+01 .1713E+00 .7585E+00
 .2658E+01 .5322E+01 .1486E+01 .1226E+00 .7612E+00
 .2880E+01 .5378E+01 .1508E+01 .7956E-01 .7734E+00
 .3101E+01 .5318E+01 .1521E+01 .4049E-01 .7906E+00
 .3323E+01 .5144E+01 .1526E+01 .2853E-02 .8087E+00
     P( 1) .9641E+00P( 2) .6500E+00P( 3)-.8001E-01P( 4) .4802E+01P( 5) .7460E+00P( 6)-.4039E+00P( 7) .3323E+01P(
TEMPO  CONTROLE         X 1        X 2        X 3
0.         ********** .1000E+010.             .1000E+01
 .2216E+00 .4813E+00 .1002E+01 .1751E-01 .1028E+01
 .4431E+00 .7059E+00 .1009E+01 .5338E-01 .1046E+01
 .6647E+00 .9077E+00 .1026E+01 .1026E+00 .1051E+01
 .8863E+00 .1087E+01 .1055E+01 .1587E+00 .1040E+01
 .1108E+01 .1243E+01 .1097E+01 .2153E+00 .1013E+01
 .1329E+01 .1376E+01 .1150E+01 .2678E+00 .9744E+00
 .1551E+01 .1487E+01 .1215E+01 .3133E+00 .9273E+00
 .1773E+01 .3908E+01 .1287E+01 .3219E+00 .8620E+00
 .1994E+01 .4432E+01 .1354E+01 .2797E+00 .8017E+00
 .2216E+01 .4843E+01 .1410E+01 .2255E+00 .7675E+00
 .2437E+01 .5141E+01 .1454E+01 .1709E+00 .7550E+00
 .2659E+01 .5326E+01 .1486E+01 .1214E+00 .7576E+00
 .2880E+01 .5399E+01 .1508E+01 .7793E-01 .7706E+00
 .3102E+01 .5359E+01 .1521E+01 .3907E-01 .7890E+00
 .3324E+01 .5206E+01 .1526E+01 .2321E-02 .8089E+00
     P( 1) .9641E+00P( 2) .6500E+00P( 3)-.8001E-01P( 4) .4802E+01P( 5) .8011E+00P( 6)-.3962E+00P( 7) .3324E+01P(
TEMPO  CONTROLE         X 1        X 2        X 3
0.         ********** .1000E+010.             .1000E+01
 .2216E+00 .4813E+00 .1002E+01 .1751E-01 .1028E+01
 .4431E+00 .7059E+00 .1009E+01 .5338E-01 .1046E+01
 .6647E+00 .9077E+00 .1026E+01 .1026E+00 .1051E+01
 .8863E+00 .1087E+01 .1055E+01 .1587E+00 .1040E+01
 .1108E+01 .1243E+01 .1097E+01 .2153E+00 .1013E+01
 .1329E+01 .1376E+01 .1150E+01 .2678E+00 .9744E+00
 .1551E+01 .1487E+01 .1215E+01 .3133E+00 .9273E+00
 .1773E+01 .3908E+01 .1287E+01 .3219E+00 .8620E+00
 .1994E+01 .4432E+01 .1354E+01 .2797E+00 .8017E+00
 .2216E+01 .4843E+01 .1410E+01 .2255E+00 .7675E+00
 .2437E+01 .5141E+01 .1454E+01 .1709E+00 .7550E+00
```

```
.2659E+01 .5326E+01 .1486E+01 .1214E+00 .7578E+00
.2880E+01 .5399E+01 .1508E+01 .7793E-01 .7706E+00
.3102E+01 .5359E+01 .1521E+01 .3907E-01 .7890E+00
.3324E+01 .5206E+01 .1526E+01 .2321E-02 .8089E+00
   P( 1) .9641E+00P( 2) .6500E+00P( 3)-.8001E-01P( 4) .4802E+01P( 5) .8011E+00P( 6)-.3962E+00P( 7) .3324E+01P(
TEMPO  CONTROLE         X 1        X 2        X 3
0.          ********** .1000E+010.             .1000E+01
 .2217E+00 .4849E+00 .1002E+01 .1762E-01 .1028E+01
 .4434E+00 .7094E+00 .1009E+01 .5357E-01 .1046E+01
 .6651E+00 .9112E+00 .1026E+01 .1028E+00 .1051E+01
 .8868E+00 .1090E+01 .1055E+01 .1589E+00 .1039E+01
 .1108E+01 .1247E+01 .1097E+01 .2155E+00 .1013E+01
 .1330E+01 .1380E+01 .1150E+01 .2678E+00 .9738E+00
 .1552E+01 .1491E+01 .1215E+01 .3132E+00 .9265E+00
 .1774E+01 .3919E+01 .1287E+01 .3215E+00 .8614E+00
 .1995E+01 .4444E+01 .1354E+01 .2789E+00 .8015E+00
 .2217E+01 .4854E+01 .1410E+01 .2247E+00 .7679E+00
 .2439E+01 .5150E+01 .1454E+01 .1703E+00 .7557E+00
 .2660E+01 .5330E+01 .1486E+01 .1211E+00 .7588E+00
 .2882E+01 .5396E+01 .1508E+01 .7780E-01 .7716E+00
 .3104E+01 .5346E+01 .1521E+01 .3892E-01 .7898E+00
 .3325E+01 .5182E+01 .1525E+01 .1859E-02 .8092E+00
   P( 1) .9676E+00P( 2) .6500E+00P( 3)-.8001E-01P( 4) .4802E+01P( 5) .7846E+00P( 6)-.4039E+00P( 7) .3325E+01P(
```

```
          **********ATUALIZACAD NUMERO= 12 DA MATRIZ DE DERIVADAS(DXA).**********
MINIMIZAR
---------
          C * X
SUJEITOS A
----------
          A * X = B
              E
          X .GE. ZERO

MATRIZ A
--------

         -.341521E+00  -.325320E-01   .119555E+00   .270021E+00   .569064E-02  -.117017E+00  -.523634E-01   .341521E+00
          .325320E-01  -.119555E+00  -.270021E+00  -.569064E-02   .117017E+00   .523634E-01
         -.411795E-01  -.248392E-01   .110068E-01   .206466E+00   .844700E-02  -.712444E-01  -.510766E-02   .411795E-01
          .248392E-01  -.110068E-01  -.206466E+00  -.844700E-02   .712444E-01   .510766E-02
         -.436733E+00  -.153993E-01   .161130E+00   .119217E+00  -.689992E-02  -.734695E-01   .287467E+00   .436733E+00
          .153993E-01  -.161130E+00  -.119217E+00   .689992E-02   .734695E-01  -.287467E+00
         0.            0.            0.            0.            0.            0.             .100000E+01  0.
         0.            0.            0.            0.            0.           -.100000E+01

         0.
         0.
         0.
         -.100000E+01

VETOR C
-------
      .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01
      .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .100000E+01   .117679E+03

VETOR B
-------
      .601474E-05  -.268998E-05   .249454E-04  -.257827E-03

   MATRIZ AA
   ---------

   X 16   .100000E+01  0.            0.            0.             .601474E-05
   X 17  0.             .100000E+01  0.            0.             .268998E-05
   X 18  0.            0.             .100000E+01  0.             .249454E-04
   X 19  0.            0.            0.             .100000E+01   .257827E-03
     -Z  0.            0.            0.            0.            0.
     -W  0.            0.            0.            0.           -.291477E-03

   VETOR INDVB
   -----------
    16 17 18 19

   VETOR LOCVAR
   ------------
    1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
```

```
***************************************************************************************************

SOLUCAO OTIMA   Z = .27453851E-01
---------------------------------

N. DE ITERACOES =  9
--------------------

                                   VARIAVEIS NA BASE   =    VALORES
                                   ----------------------------------
                                           4           =   .11240298E-02
                                          12           =   .25587729E-01
                                           1           =   .48426540E-03
                                          14           =   .25782707E-03

***************************************************************************************************
TEMPO  CONTROLE       X 1        X 2        X 3
0.     ********** .1000E+010.            .1000E+01
 .2215E+00 .5257E+00 .1002E+01 .1982E-01 .1026E+01
 .4431E+00 .6833E+00 .1010E+01 .5521E-01 .1044E+01
 .6646E+00 .8479E+00 .1027E+01 .1028E+00 .1050E+01
 .8861E+00 .1019E+01 .1056E+01 .1577E+00 .1040E+01
 .1108E+01 .1198E+01 .1097E+01 .2144E+00 .1016E+01
 .1329E+01 .1384E+01 .1151E+01 .2680E+00 .9776E+00
 .1551E+01 .1576E+01 .1215E+01 .3147E+00 .9288E+00
 .1772E+01 .4112E+01 .1288E+01 .3204E+00 .8643E+00
 .1994E+01 .4501E+01 .1354E+01 .2772E+00 .8083E+00
 .2215E+01 .4818E+01 .1410E+01 .2247E+00 .7748E+00
 .2437E+01 .5063E+01 .1454E+01 .1710E+00 .7600E+00
 .2658E+01 .5235E+01 .1486E+01 .1208E+00 .7598E+00
 .2880E+01 .5336E+01 .1508E+01 .7563E-01 .7702E+00
 .3101E+01 .5364E+01 .1520E+01 .3571E-01 .7879E+00
 .3323E+01 .5320E+01 .1524E+01 .8022E-04 .8101E+00
    P( 1) .1001E+01P( 2) .6500E+00P( 3) .2460E-01P( 4) .4859E+01P( 5) .7146E+00P( 6)-.2536E+00P( 7) .3323E+01P(
TEMPO  CONTROLE       X 1        X 2        X 3
0.     ********** .1000E+010.            .1000E+01
 .2215E+00 .5252E+00 .1002E+01 .1981E-01 .1026E+01
 .4431E+00 .6827E+00 .1010E+01 .5520E-01 .1044E+01
 .6646E+00 .8473E+00 .1027E+01 .1028E+00 .1050E+01
 .8862E+00 .1019E+01 .1056E+01 .1577E+00 .1040E+01
 .1108E+01 .1197E+01 .1097E+01 .2144E+00 .1016E+01
 .1329E+01 .1383E+01 .1151E+01 .2680E+00 .9779E+00
 .1551E+01 .1576E+01 .1216E+01 .3148E+00 .9289E+00
 .1772E+01 .4089E+01 .1288E+01 .3208E+00 .8640E+00
 .1994E+01 .4485E+01 .1354E+01 .2777E+00 .8073E+00
 .2215E+01 .4808E+01 .1410E+01 .2249E+00 .7733E+00
 .2437E+01 .5060E+01 .1454E+01 .1709E+00 .7584E+00
 .2659E+01 .5239E+01 .1486E+01 .1203E+00 .7583E+00
 .2880E+01 .5346E+01 .1508E+01 .7499E-01 .7690E+00
 .3102E+01 .5381E+01 .1520E+01 .3520E-01 .7872E+00
 .3323E+01 .5344E+01 .1524E+01 .7410E-06 .8101E+00
    P( 1) .1000E+01P( 2) .6500E+00P( 3) .2470E-01P( 4) .4858E+01P( 5) .7402E+00P( 6)-.2536E+00P( 7) .3323E+01P(
TEMPO  CONTROLE       X 1        X 2        X 3
0.     ********** .1000E+010.            .1000E+01
 .2215E+00 .5255E+00 .1002E+01 .1981E-01 .1026E+01
 .4431E+00 .6830E+00 .1010E+01 .5521E-01 .1044E+01
 .6646E+00 .8476E+00 .1027E+01 .1028E+00 .1050E+01
 .8861E+00 .1019E+01 .1056E+01 .1577E+00 .1040E+01
```

.1108E+01	.1198E+01	.1097E+01	.2144E+00	.1016E+01
.1329E+01	.1383E+01	.1151E+01	.2680E+00	.9778E+00
.1551E+01	.1576E+01	.1215E+01	.3147E+00	.9288E+00
.1772E+01	.4102E+01	.1288E+01	.3206E+00	.8642E+00
.1994E+01	.4494E+01	.1354E+01	.2775E+00	.8079E+00
.2215E+01	.4814E+01	.1410E+01	.2248E+00	.7741E+00
.2437E+01	.5061E+01	.1454E+01	.1710E+00	.7593E+00
.2658E+01	.5237E+01	.1486E+01	.1206E+00	.7591E+00
.2880E+01	.5340E+01	.1508E+01	.7536E-01	.7696E+00
.3102E+01	.5372E+01	.1520E+01	.3549E-01	.7876E+00
.3323E+01	.5331E+01	.1524E+01	.4999E-04	.8101E+00

P(1) .1000E+01P(2) .6500E+00P(3) .2470E-01P(4) .4858E+01P(5) .7258E+00P(6)-.2536E+00P(7) .3323E+01P(

```
                    SAIDA NORMAL,A CONVERGENCIA TORNOU-SE LENTA
                    IMPRESSAO DE RESULTADOS FINAIS *** ATUALIZACAO NUMERO = 12
TEMPO  CONTROLE          X 1         X 2          X 3
0.         ********** .1000E+010.              .1000E+01
 .2215E+00 .5254E+00 .1002E+01 .1981E-01 .1026E+01
 .4431E+00 .6829E+00 .1010E+01 .5520E-01 .1044E+01
 .6646E+00 .8475E+00 .1027E+01 .1028E+00 .1050E+01
 .8862E+00 .1019E+01 .1056E+01 .1577E+00 .1040E+01
 .1108E+01 .1198E+01 .1097E+01 .2144E+00 .1016E+01
 .1329E+01 .1383E+01 .1151E+01 .2680E+00 .9778E+00
 .1551E+01 .1576E+01 .1215E+01 .3147E+00 .9288E+00
 .1772E+01 .4096E+01 .1288E+01 .3207E+00 .8641E+00
 .1994E+01 .4490E+01 .1354E+01 .2776E+00 .8076E+00
 .2215E+01 .4811E+01 .1410E+01 .2249E+00 .7737E+00
 .2437E+01 .5061E+01 .1454E+01 .1710E+00 .7589E+00
 .2659E+01 .5238E+01 .1486E+01 .1204E+00 .7587E+00
 .2880E+01 .5343E+01 .1508E+01 .7520E-01 .7693E+00
 .3102E+01 .5376E+01 .1520E+01 .3537E-01 .7874E+00
 .3323E+01 .5337E+01 .1524E+01 .4237E-04 .8101E+00
    P( 1) .1000E+01P( 2) .6500E+00P( 3) .2470E-01P( 4) .4858E+01P( 5) .7325E+00P( 6)-.2536E+00P( 7) .3323E+01P(
    *****VALORES FINAIS DOS PARAMETROS QUE CONTROLAM A  CONVERGENCIA*****
                      EPS2= .4863E-04
                      C2G+1 .1059E+03
                      INCREMENTO MAXIMO NOS PARAMETROS OTIMIZAVEIS NESTA ITERACAO= .7679E-02
                      BETA=-.2147E-04
```