

INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Divisão de Mecânica Espacial e Controle - DMC

Relatório Final da Bolsa de Iniciação Científica PIBIC / CNPq

***Efeitos dos Desalinhamentos
do Vetor Empuxo em
Transferências Orbitais***

Bolsista: Nilton Xavier Ribeiro

4.o Ano de Engenharia Mecânica-Aeronáutica -ITA

29 de Junho de 1999

INTRODUÇÃO

A corrida espacial é assunto de grande interesse para qualquer país que queira assegurar o seu lugar num domínio que ainda tem muito a ser explorado que é o Espaço Sideral. É dentro deste contexto que se inserem os crescentes investimentos em pesquisa espacial, envolvendo satélites e seus respectivos veículos lançadores. E é visando obter resultados significativos que se destina este trabalho de pesquisa, o qual se propõe a tratar o problema da transferência orbital sob um ponto de vista mais geral, ou seja, levando em conta fatores que, na maioria das vezes, são desconsiderados ao se simplificar os modelos físicos dos veículos espaciais.

OBJETIVOS

Este trabalho objetivou adaptar e continuar o trabalho iniciado por Rodrigues(1991). Na fase inicial da pesquisa o objetivo consistiu em se entender os conceitos e equações envolvidos na transferência orbital de satélites artificiais, segundo Kuga e Rao(1995). A próxima etapa consistiu em simular modelos onde o empuxo do veículo não está perfeitamente alinhado com o seu centro de massa, o que de fato acontece na realidade e que, quase sempre, é desconsiderado na maioria dos cálculos iniciais realizados atualmente. O que se pretendeu foi mostrar resultados substanciais que provassem que considerar estes desalinhamentos ainda na fase de concepção e nos cálculos do satélite pode revelar-se muito mais realista do que simplesmente ignorá-los numa simplificação que levará a uma série de correções futuras.

TRANSFERÊNCIA ORBITAL

- *Fundamentos Estudados*

- Elementos Keplerianos
- Geometria Analítica da Elipse
- Movimento Orbital
- Leis de Kepler

- *Caracterização do Problema da Transferência Orbital*

Colocar um satélite em órbita é cada vez mais um objeto de intensa pesquisa nos tempos atuais. Cada vez mais se dedica tempo, trabalho e dinheiro no mundo para se alcançar melhorias nos diversos processos e estratégias que fazem um veículo espacial encontrar a sua trajetória da maneira mais precisa possível.

Para isto constantemente se depara com o problema da transferência de um veículo espacial de uma órbita para outra. Isto é possível graças à presença de sistemas de atuadores que permitem que o satélite seja capaz de gerar forças e torques para alterar a sua própria trajetória e o seu movimento de atitude.

A transferência orbital é sempre importante pois constitui uma das principais etapas no lançamento de todos os satélites, uma vez que, na quase totalidade das vezes, o veículo lançador não o injeta diretamente na órbita final (órbita nominal). Por motivos de minimização do uso de combustível a estratégia mais comum é:

Fase 0 – injetar o veículo em uma órbita preliminar, chamada de órbita de estacionamento, para então se proceder à transferência orbital para a órbita nominal.

O processo seguinte é normalmente dividido em três partes (Bertachini, 1989)

Fase 1 – colocar o veículo numa órbita e atitude quase nominal (estado nominal);

Fase 2 – aplicar pequenos incrementos de força e torque até que o estado nominal seja atingido com um certo grau de precisão;

Fase 3 – corrigir periodicamente o estado nominal do veículo, uma vez que perturbações podem alterá-lo constantemente.

Este estudo se voltará basicamente para os fenômenos envolvidos na fase 1, por ser ela a que apresenta modificações mais acentuadas na trajetória do veículo, sendo por isso a fase mais sensível aos efeitos dos desalinhamento do vetor empuxo.

EFEITOS DO DESALINHAMENTO DO EMPUXO

Um satélite é equipado com atuadores que lhe possibilitam alterar a sua própria trajetória. Isto se faz necessário devido à fase de transferência orbital já mencionada acima. Além disso estes atuadores também permitem que se façam pequenas correções na trajetória do satélite em eventuais situações em que elas se façam necessárias. Fenômenos externos de difícil modelamento matemático tais como forças e torques de superfície - como o arrasto atmosférico e a pressão de radiação - acabam perturbando a trajetória desejada para o satélite, que deve, portanto ser corrigida.

Estes propulsores, na maioria dos casos, são projetados para gerar forças cuja linha de ação está orientada na direção do centro de massa-CM do veículo. Entretanto, na prática, não é isto que acontece. Geralmente ocorre o desalinhamento propulsivo, que nada mais é que o desvio do vetor empuxo em relação ao CM, o que faz com que apareçam torques não previstos que giram o veículo e seus propulsores numa realimentação positiva e indesejável. Desta forma é necessário fazer-se um controle corretivo de atitude.

Tais desalinhamentos usualmente apresentam baixas magnitudes e são causados principalmente por fatores como a arquitetura e montagem do veículo, deslocamento do centro de massa originado pelo consumo de combustível, partes móveis (especialmente partes líquidas), etc.

ESTUDO ANALÍTICO SIMPLIFICADO

O que se segue foi obtido originalmente por Rodrigues (1991), mas será resumido aqui para facilitar as referências e a compreensão.

Será feita aqui a análise de um modelo constituído por um veículo não deformável e com massa constante, sujeito a somente a uma única força com orientação constante em relação aos eixos principais de inércia do veículo e com eixo de aplicação não interceptando o CM. Desta maneira a trajetória descrita pelo CM permanece no plano que contém esta força e o CM. Então, pode-se escolher um sistema de referência OXYZ, cujo plano XY contém a trajetória referida (Fig. 1).

As equações de movimento do CM e do movimento de atitude com respeito ao sistema de referência podem ser facilmente obtidas como:

$$I \ddot{\theta}(t) = F \cdot \epsilon \quad (1)$$

$$m \cdot \ddot{\mathbf{R}}(t) = \mathbf{F}(t) \quad (2)$$

onde m é a massa do corpo (suposta constante), $\mathbf{R}(t)$ é o vetor posição do CM com respeito ao sistema de referência OXYZ, $\mathbf{F}(t)$ é a força resultante escrita em termos dos versores de OXYZ, h é a distância longitudinal da posição ideal de aplicação do empuxo até o CM, I é o momento de inércia com respeito ao eixo de rotação Z , θ é o ângulo de atitude, ν e δ são os desalinhamentos linear e angular do vetor empuxo, respectivamente, e t é o tempo. Da equação (1) tem-se:

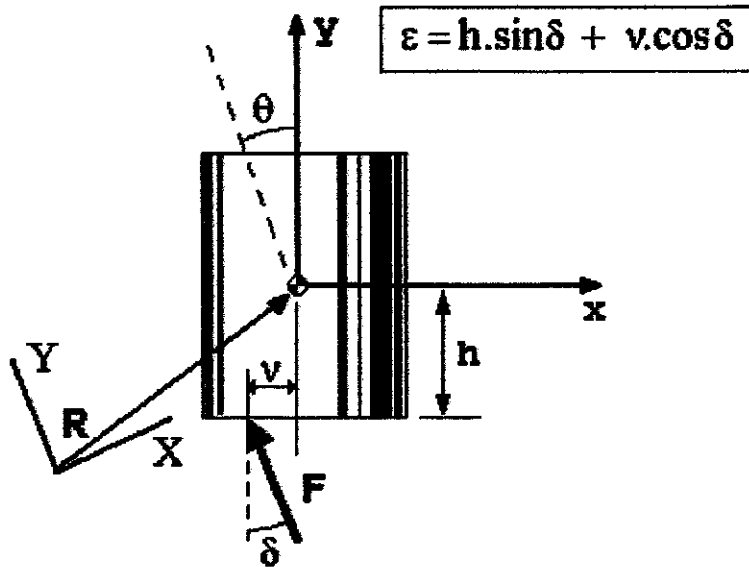


Figura 1 : Desalinhamentos linear e angular em um sistema simplificado

$$\dot{\theta}(t) = \frac{F \cdot \epsilon}{I} t + \dot{\theta}(0) \quad (3)$$

$$\theta(t) = \frac{F \cdot \epsilon}{2I} t^2 + \dot{\theta}(0) \cdot t + \theta(0) \quad (4)$$

Algumas dificuldades matemáticas podem ser evitadas se nós considerarmos condições iniciais nulas: $R(0) = 0$, $\dot{R}(0) = 0$, $\theta(0) = 0$ e $\dot{\theta}(0) = 0$. Isto será válido daqui em diante.

A equação (2) pode ser escrita em termos de componentes do vetor posição do centro de massa $X(t)$ e $Y(t)$:

$$m \cdot \ddot{X}(t) = F \cdot \sin\left(\frac{F \cdot \epsilon}{2I} t^2 - \delta\right) \quad (5)$$

$$m \cdot \ddot{Y}(t) = F \cdot \cos\left(\frac{F \cdot \epsilon}{2I} t^2 - \delta\right) \quad (6)$$

SOLUÇÃO ANALÍTICA APROXIMADA

É possível obter soluções aproximadas das equações (5) e (6) para t perto de zero e para t tendendo ao infinito. Considere-se inicialmente o caso em que $t \rightarrow 0$. Neste caso são válidas as relações $\cos\theta(t) = 1$ e $\sin\theta(t) \cong \theta(t)$. Normalmente δ é bem pequeno; então pode-se escrever $\cos\delta = 1$ e $\sin\delta \cong \delta$. Então as equações (5) e (6) podem ser escritas como

$$m \cdot \ddot{X}(t) \cong \frac{F^2 \cdot \epsilon}{2I} t^2 - F \cdot \delta \quad (7)$$

$$m \cdot \ddot{Y}(t) \cong F \quad (8)$$

Integrando-se (7) e (8) tem-se:

$$\dot{X}_o(t) \cong \frac{F^2 \cdot \varepsilon}{6 \cdot m \cdot I} t^3 - \frac{F \cdot \delta}{m} \cdot t \quad (9)$$

$$\dot{Y}_o(t) \cong \frac{F}{m} t \quad (10)$$

$$X_o(t) \cong \frac{F^2 \cdot \varepsilon}{24 \cdot m \cdot I} t^4 - \frac{F \cdot \delta}{2 \cdot m} \cdot t^2 \quad (11)$$

$$Y_o(t) \cong \frac{F}{2 \cdot m} t^2 \quad (12)$$

Isolando o tempo nas equações (9) e (10) tem-se

$$\dot{X}_o(t) \cong \frac{m^2 \cdot \varepsilon}{6 \cdot F \cdot I} Y_o^3(t) - \delta \cdot Y_o(t) \quad (13)$$

$$X_o(t) \cong \frac{m \cdot \varepsilon}{6 \cdot I} Y_o^2(t) - \delta \cdot Y_o(t) \quad (14)$$

Este modelo é o caso simplificado de um veículo que tem um sistema propulsivo que gera um torque de desalinhamento $F \cdot \varepsilon$. O caso ideal seria sem desalinhamentos $\varepsilon = 0$ e, conseqüentemente, o CM do veículo estaria inteiramente no eixo Y. Devido à presença do desalinhamento $\varepsilon \neq 0$, esta trajetória descreve inicialmente uma parábola conforme a eq.(14).

As equações de (9) a (14) descrevem o comportamento de centro de massa do corpo perto do zero. Nos primeiros instantes da propulsão, os efeitos do desalinhamento são bem pequenos, como mostrado pelas equações (10) e (12), que se refere à componente Y do movimento do centro de massa, seguindo a lei de Newton do movimento linear. Porém, gradualmente o movimento do CM dá lugar ao movimento parabólico descrito pela equação (14), caracterizando a degeneração do movimento.

Para ter o comportamento do CM para $t \rightarrow \infty$, pode-se integrar as eqs. (5)-(6) de 0 a ∞ :

$$m \cdot \int_{t=0}^{t \rightarrow \infty} \dot{X}(t) \cdot dt = F \cdot \int_{t=0}^{t \rightarrow \infty} \sin\left(\frac{F \cdot \varepsilon}{2 \cdot I} t^2 - \delta\right) \cdot dt \quad (15)$$

$$m \cdot \int_{t=0}^{t \rightarrow \infty} \dot{Y}(t) \cdot dt = F \cdot \int_{t=0}^{t \rightarrow \infty} \cos\left(\frac{F \cdot \varepsilon}{2 \cdot I} t^2 - \delta\right) \cdot dt \quad (16)$$

$$\dot{X}_\infty = \frac{(\cos\delta - \sin\delta)}{2 \cdot m} \sqrt{\frac{\pi \cdot F \cdot I}{\varepsilon}} \quad (17)$$

$$\dot{Y}_\infty = \frac{(\cos\delta + \sin\delta)}{2 \cdot m} \sqrt{\frac{\pi \cdot F \cdot I}{\varepsilon}} \quad (18)$$

$$X_\infty = \frac{(\cos\delta - \sin\delta)}{2 \cdot m} \sqrt{\frac{\pi \cdot F \cdot I}{\varepsilon}} \cdot t + k_x \quad (19)$$

$$Y_\infty = \frac{(\cos\delta + \sin\delta)}{2 \cdot m} \sqrt{\frac{\pi \cdot F \cdot I}{\varepsilon}} \cdot t + k_y \quad (20)$$

As equações (17) e (18) mostram que para $t \rightarrow +\infty$, a velocidade do CM se torna constante apesar de haver uma força não nula atuando instantaneamente no corpo. Isto é explicado pelo fato de que quando t se torna muito grande sua velocidade angular é tão alta que ela quase instantaneamente ocupa posições angulares opostas, cancelando os efeitos lineares da força atuante, produzindo, entretanto, uma grande aceleração angular.

Isolando t nas equações (19) e (20) tem-se:

$$\dot{Y}_\infty = \frac{(\cos\delta + \sin\delta)}{(\cos\delta - \sin\delta)} \cdot \dot{X}_\infty \quad (21)$$

$$Y_\infty = \frac{(\cos\delta + \sin\delta)}{(\cos\delta - \sin\delta)} \cdot X_\infty + k \quad (22)$$

No caso ideal ($v = 0$ e $\delta = 0$), a trajetória deveria descrever uma linha reta ao longo do eixo X com velocidade crescendo linearmente com o tempo. No caso real ($v \neq 0$ e $\delta \neq 0$), a trajetória se torna reta mas não necessariamente na direção Y e com velocidade constante, como mostram as equações (17) e (18). É interessante perceber que se $\delta = \pi/4$, então a trajetória do CM retorna a ter somente a componente Y.

A trajetória do CM foi determinada somente para as situações descritas ($t \rightarrow 0$ e $t \rightarrow \infty$). Porém seu comportamento intermediário permanece desconhecido. Entretanto, pode-se ter alguma idéia deste comportamento examinando-se as equações (5) e (6). Estas equações mostram que $X(t)$ e $Y(t)$ apresentam um comportamento oscilatório, com pontos máximos e mínimos dados por $X(t) = 0$ e $Y(t) = 0$. As equações (17) e (18) mostram que as componentes da velocidade convergem para valores constantes. Entretanto, a velocidade do CM deve ser oscilatória. Conseqüentemente, a trajetória para a situação intermediária deverá ser, da mesma maneira, oscilatória amortecida também.

É interessante determinar o valor de t para o qual $Y(t)$ alcança o máximo valor. Tal valor – aqui chamado de t^* – é importante na medida em que ele indica que, se a propulsão continua após este instante, a degradação da trajetória será mais e mais acentuada. Assim, t^* pode ser entendido como o tempo máximo de atuação do sistema de propulsão para uma trajetória levemente degenerada. Da equação (6), para $Y(t)$ ser máximo é necessário que:

$$t^* = \sqrt{\frac{I \cdot (k \cdot \pi + 2 \cdot \delta)}{F \cdot \epsilon}} \quad (23)$$

com $k = 1, 3, 5, \dots$ e o primeiro máximo ocorrendo em $k = 1$.

Aqui termina o resumo da solução analítica obtida por Rodrigues (1991).

SIMULAÇÕES NUMÉRICAS

EXEMPLO 1: (MOV.DO CM SOB EMPUXO FINITO PRÓX.AO PERIGEU COM $\epsilon = 0$)

Esta parte do estudo consistiu de estudar e adaptar os mecanismos de funcionamento do programa MODELO1.FOR originalmente escrito em FORTRAN para o VAX 780 / VMS 2.0 da DMC de modo a se ter condições de implementá-lo e assim obter os resultados numéricos e gráficos da simulação. As subrotinas envolvidas no projeto são a RKF78, e a ORB_POS_PROP. Esta simulação trata de um modelo de satélite simplificado sujeito a uma série de arcos propulsivos do tipo não impulsivo.

Boa parte do trabalho desta pesquisa se voltou à adaptação do programa acima referido para que ele pudesse “rodar” na versão do FORTRAN para PC's, fazendo-se

alterações de modo a facilitar a coleta de seus resultados através de manipulação de arquivos. Também foram feitas alterações no código fonte para se conseguir um melhor *link* das subrotinas. Foram usados dois tipos diferentes de leis de orientação do vetor empuxo no veículo: empuxo perpendicular ao raio vetor e empuxo tangente à trajetória. E tendo como:

Hipóteses: Campo central e newtoniano; modelo propulsivo simplificado; veículo modelado como massa puntual; transferência coplanar.

Dados de entrada:

Massa inicial do veículo antes da propulsão :	1860 Kg
Magnitude do raio vetor do perigeu da órbita de estacionamento:	6578 Km
Excentricidade da órbita de estacionamento:	0,73008904
Duração da fase propulsiva	884,85 s
Ponto de início de propulsão	- 0,65 rad

Dados de saída:

Como dados de saída o programa MODELO1.FOR fornece a rotação da linha dos ápsides da elipse, as alterações da magnitude do semi-eixo maior e da excentricidade da elipse (Figuras 2a, b, c), em função da anomalia verdadeira do início da propulsão do veículo.

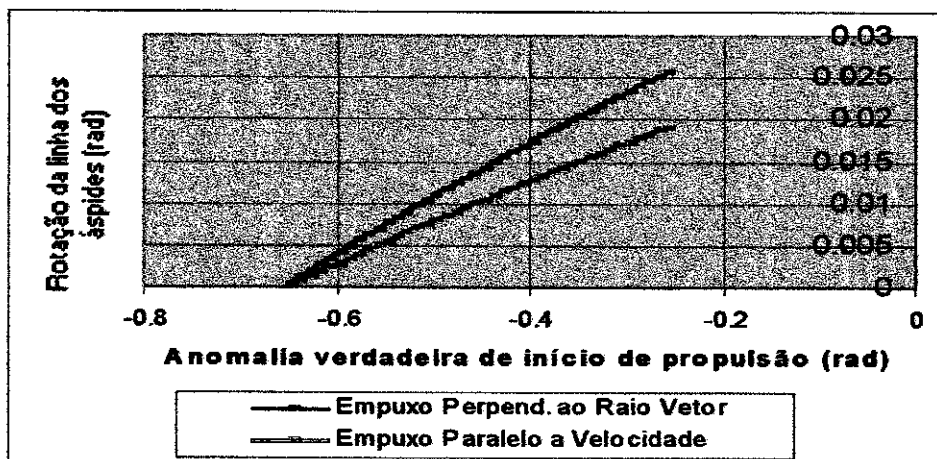


Figura 2a: Rotação da linha dos ápsides em função da anomalia verdadeira de início de propulsão

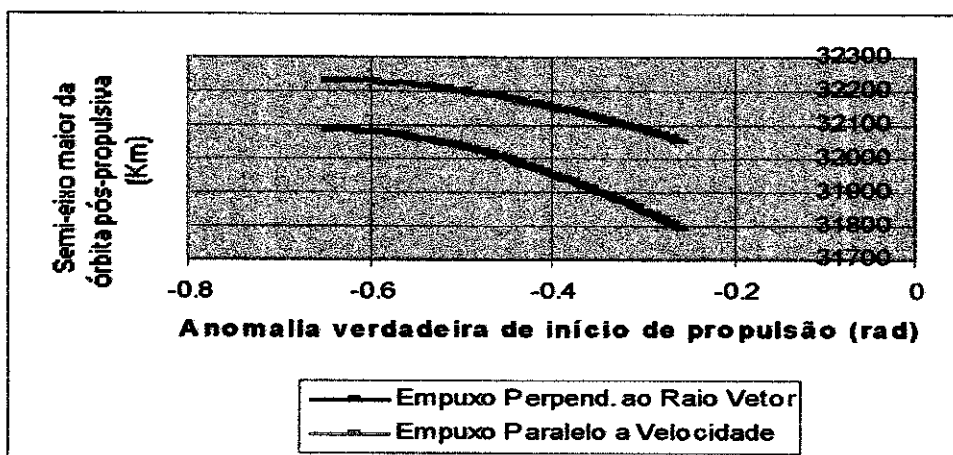


Figura 2b: Semi-eixo maior da órbita pós-propulsiva em função da anom.verd.de início de propulsão

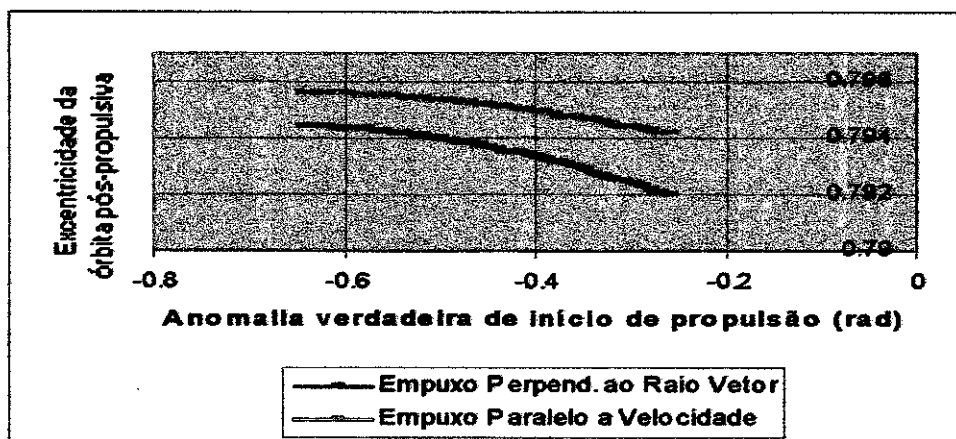


Figura 2c: Excentricidade da órbita pós-propulsiva em função da anomalia verdadeira de início de propulsão

EXEMPLO 2 (MOV. EM TORNO DO CM SOB EMPUXO FINITO COM $\varepsilon \neq 0$)

Seja a situação hipotética onde se tem um corpo cilíndrico reto com raio da base r , altura H , com uma força F aplicada perpendicularmente a base; e: $F = 10 \text{ N}$; $v = 0,01 \text{ m}$; $\delta = 0 \text{ rad}$; $\varepsilon = 0,01 \text{ m}$; $m = 5 \text{ kg}$; $r = 1 \text{ m}$; $H = 3 \text{ m}$; $I = 5 \text{ kg.m}^2$.

As Figuras de 3 a 6 obtidas originalmente por Rodrigues(1991) mostram os resultados para o exemplo em questão. Da Figura 3 observa-se que a solução analítica aproximada está de acordo com a solução real, para pequenos valores de t . Nesta mesma Figura 3 fica claro o comportamento parabólico do centro de massa citado anteriormente.

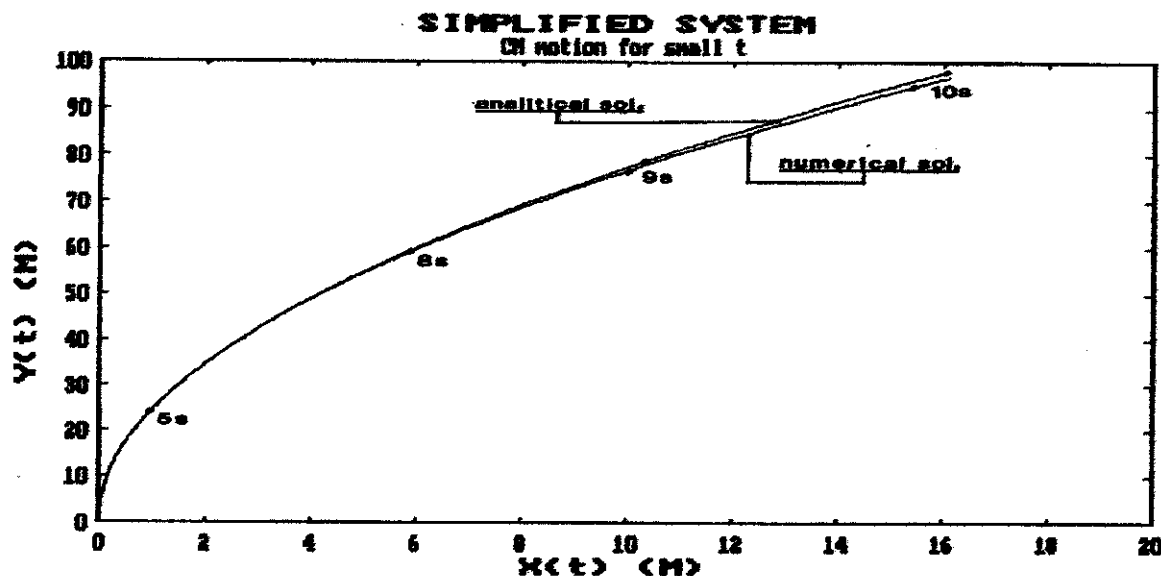


Figura 3: Movimento do CM para pequenos valores de t

Da Figura 4 pode-se observar que as componentes da velocidade do CM tem comportamento oscilatório cujos valores convergem para o intervalo de 12 – 14 m/s. Das equações (17) e (18) consegue-se os valores de convergência $X = Y = 12,53 \text{ m/s}$. Os valores t^* e Y^* neste exemplo, obtidos das equações (23) e (25), são 12,53s e 19,59 m/s, respectivamente. O real valor para Y^* é 19,55 m/s, indicando um erro relativo no modelo analítico menor do que 0,5% (neste exemplo).

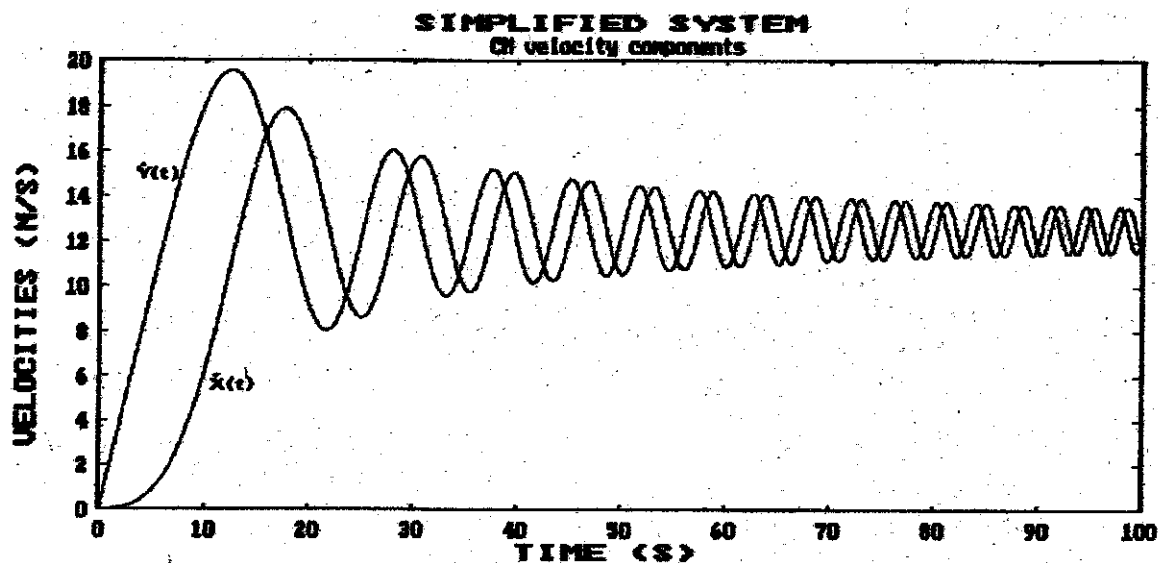


Figura 4: Componentes da velocidade do CM em função do tempo

Das Figuras 5a e 5b percebe-se que para um valor muito grande de t o CM da trajetória se aproxima da linha reta descrita pelas equações (19), (20) e (22). No caso de $\varepsilon = 0$ o CM descreve uma trajetória reta na direção Y, o que de fato ocorre nos primeiros instantes. O desalinhamento causa um desvio da trajetória (quanto maior é o valor de ε mais rápido isto acontece) o que é capaz de alterá-la substancialmente e cancelar os efeitos da força aplicada.

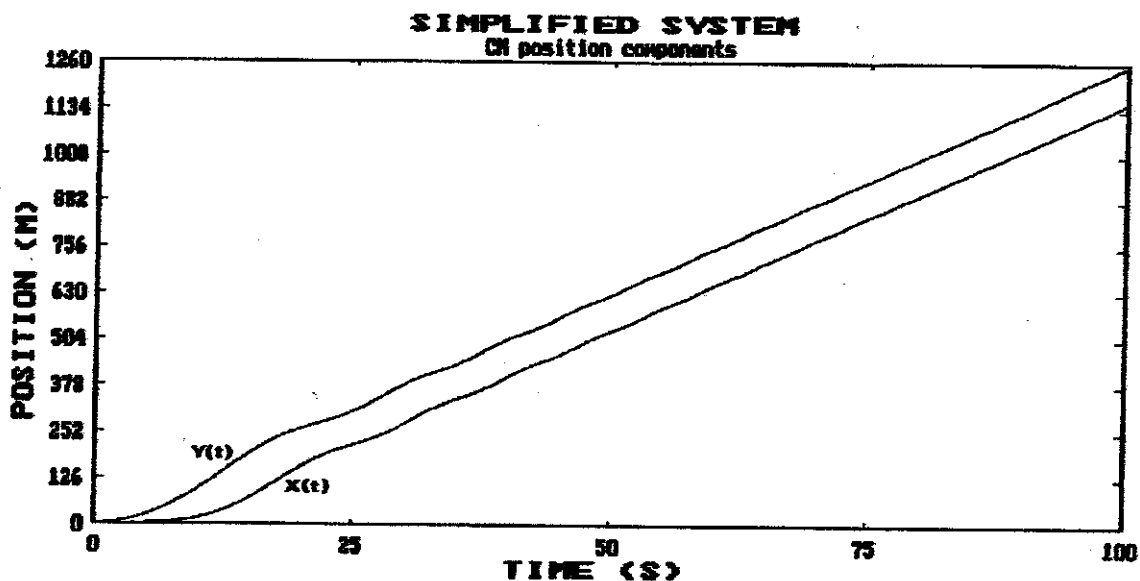


Figura 5a : Componentes da posição do CM em função do tempo

Da Figura 6 percebe-se que a eficiência propulsiva – a relação de Y^* e $(F/m)t^*$ – decresce monotonicamente com o tempo até $Y(t)$ alcançar seu primeiro mínimo relativo.

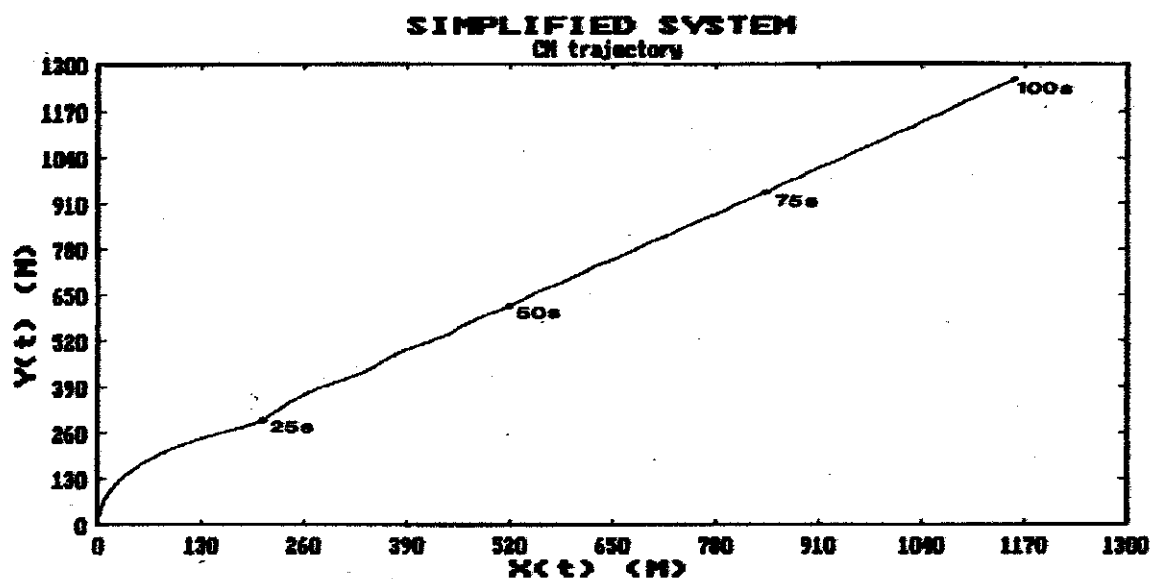


Figura 5b : Trajetória do CM

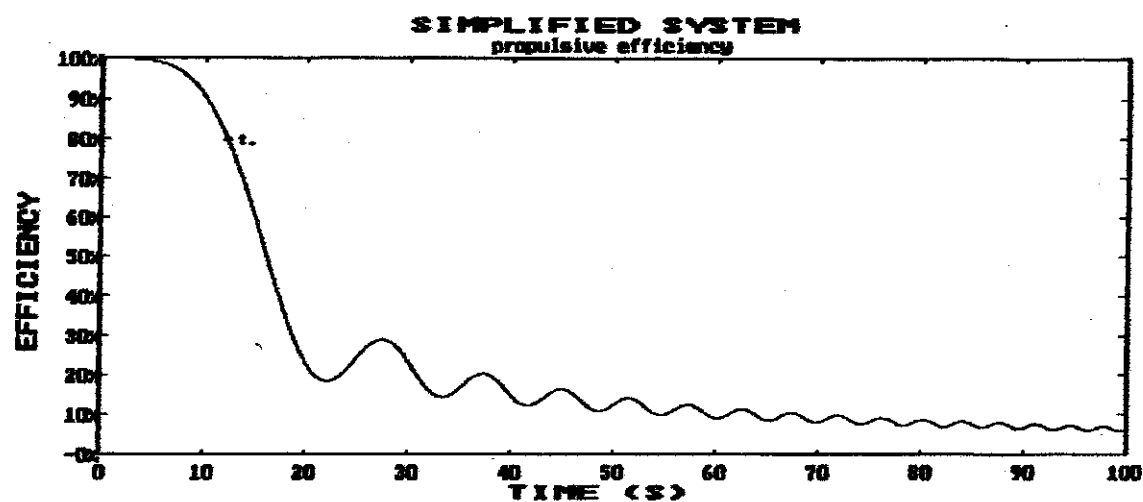


Figura 6 : Eficiência propulsiva em função do tempo

Apesar do exemplo numérico ser específico, as Figuras de 3 até 6 mostram o comportamento geral das perturbações do CM de um corpo sujeito ao torque de desalinhamento F_e na direção perpendicular ao plano do movimento (plano xy).

CONCLUSÕES

De posse dos resultados obtidos pela resolução numérica dos problemas acima citados foi possível verificar que a pesquisa tem apresentado resultados coerentes com os de Rodrigues(1991). O estudo analítico apresentado, bem como a simulações numéricas se apresentaram bastante satisfatórias, permitindo se analisar a trajetória do CM do veículo espacial após a fase propulsiva. E permitiram mostrar a importância dos efeitos dos desalinhamentos do vetor empuxo sobre as transferências orbitais de satélites artificiais.

SUGESTÕES PARA O PROSSEGUIMENTO DESTES TRABALHOS

O trabalho estudado e executado até aqui é parte de um projeto que visa implementar programas que possam ser utilizados na análise de missões reais e de extremo interesse para o INPE. Dessa maneira, mesmo após o término da Bolsa de Iniciação Científica, pretendemos prosseguir estudando e adaptando os programas e subrotinas disponíveis do trabalho de Rodrigues (1991) e aplicando-os a satélites reais com graus cada vez maiores de realismo.

REFERÊNCIAS BIBLIOGRÁFICAS:

- [1] Rodrigues, D. L. F., *Análise Dinâmica da Transferência Orbital*, INPE, S. José dos Campos, SP, Brasil, 1991 (Master Thesis, INPE – 5352 – TDI/461).
- [2] Rodrigues, D. L. F., *Effects of Thrust Misalignments on Orbit Transfers*, INPE, S. José dos Campos, SP, Brasil, 1997.
- [3] Kuga, H. K.; Rao, K. R., *Introdução à Mecânica Orbital*, INPE, S. José dos Campos, SP, Brasil, 1995.
- [4] Prado, A. F. B. A., *Análise, Seleção e Implementação de Procedimentos que Visem Manobras Ótimas de Satélites Artificiais*, INPE, S. José dos Campos, SP, Brasil, 1989 (Master Thesis, INPE – 5003 – TDL/397).

APÊNDICE - CÓDIGO FONTE

Programa principal (modelo1.for)

```

IMPLICIT REAL*8(A-H,O-Z)
EXTERNAL FPERP,FPARA,XNEW
COMMON/GRARE/RE(50)
COMMON/GRANI/NI(50)
COMMON/GRANS/NS(50)
REAL*8 X(5),XP(5),D(200)
REAL*8 R(3),TETA(3),AMAJ1(100),EX1(100)
REAL*8 PI,MI,M0,R0,E0,TB,AO,PO,F,RP,TETAB(100)
REAL*8 RPOS(50),TETPOS(50),DELT(3,100),EX(3,100),AMAJ(3,100)
INTEGER WORK(5000)
CHARACTER*72 TX, TY(3)

C
C NOME DO PROGRAMA : MODELO1.FOR.
C ESTE PROGRAMA DETERMINA O COMPORTAMENTO DA LINHA DOS
C APSIDES NUMA TRANSFERENCIA NAO IMPULSIVA.
C HIPOTHESES :
C
C 1-CAMPO CENTRAL E NEWTONIANO;
C 2-MODELO PROPULSIVO SIMPLIFICADO;
C 3-VEICULO MODELADO COMO MASSA PONTUAL;
C 4-TRANSF. COPLANAR.
C
C ENTRADAS : CARACTERISTICAS ORBITAIS DE ESTACIONAMENTO,
C CARACTERISTICAS PROPULSIVAS, MASSA INICIAL, TIPO DE LEI
C DE CONTROLE DA FORCA DE EMPUXO (NESTE PROGRAMA SAO SIMU-
C LADAS DUAS LEIS : PERPENDICULAR AO RAO VETOR OU TANGENTE
C A ORBITA). AS CARACTERISTICAS PROPULSIVAS DEVEM SER
C FORNECIDAS NO PROG. PRINCIPAL E NAS RESPECTIVAS SUB-ROTINAS.
C
C SAIDAS : CURVAS DE ROTACAO DA LINHA DOS APSIDES
C VERSUS ANOM. VERDADEIRA DE INICIO DE PROPULSAO.
C
C DECLARACAO DAS CONSTANTES E DOS PARAMETROS
C ORBITAIS (DE ESTACIONAMENTO) E PROPULSIVOS.
C
C PI=3.14159265359
C MI=398600.
C
C M0=MASSA INICIAL DO VEICULO, ANTES DA PROPULSAO (Kg);
C R0=MAGNITUDE DO RAO VETOR DO PERIGEU DA ORBITA
C DE ESTACIONAMENTO (Km);

```

```

C      E0=EXCENTRICIDADE DA ORBITA DA ESTACIONAMENTO;
C      TB=DURACAO DA FASE PROPULSIVA (s).
C
C      M0=1860.
C      R0=6578
C      E0=0.73008904
C      TB=884.85
C
C      OBTENCAO DA ORBITA DE ESTACIONAMENTO.
C
C      SEMI-EIXO MAIOR DA ORBITA DE ESTACIONAMENTO.
C
C      A0=R0/(1-E0)
C
C      SEMI-LATUS RETUM DA ORBITA DE ESTACIONAMENTO.
C
C      P0=A0*(1-E0*E0)
C
C      DECLARACAO DO PONTO DE INICIO DE PROPULSAO.
C
C      TETAB(1)=-0.65
C
C      INICIO DO PROCESSO DE OBTENCAO DO COMPORTAMENTO
C      DA LINHA DOS APSIDES COM VETOR DE EMPUXO PERPENDI-
C      CULAR AO RAO VETOR (NLEI=1) OU PARALELA `A
C      VELOCIDADE (NLEI=2).
C
C      DO 400 NLEI=2,2
C
C      INICIO DO PROCESSO DE CALCULO DE DIVERSAS FASES
C      PROPULSIVAS SE INICIANDO EM TETAB.
C
C      DO 400 J=1,40
C
C      VETOR DE ESTADO DAS CONDICoes INICIAIS.
C
C      X(1)=MAGNITUDE DO RAO VETOR (Km);
C      X(2)=POSICAO ANGULAR (rad);
C      X(3)=MAGNITUDE DA VELOCIDADE RADIAL (Km/s);
C      X(4)=MAGNITUDE DA VELOCIDADE NORMAL (Km/s);
C      X(5)=MASSA DO VEICULO (Kg).
C
C      X(1)=P0/(1+E0*COS(TETAB(J)))
C      X(2)=TETAB(J)
C      X(3)=SQRT(MI/P0)*E0*SIN(X(2))
C      X(4)=SQRT(MI/P0)*(1+E0*COS(X(2)))
C      X(5)=M0
C
C      TESTE PARA VERIFICAR O TIPO DE PROPULSAO.
C
C      IF (NLEI.EQ.1) THEN
C
C      CASO EM QUE O VETOR DE EMPUXO E' PERPENDICULAR AO RAO VETOR.
C
C      ENTRADA DE PARAMETROS DA SUBROTINA INTEGRADORA.
C
C      NEQN=5
C      T=0.0
C      TOUT=0.0
C      DT=TB/200.
C      RELERR=1D-9
C      ABSERR=1D-8
C      IFLAG=1
C
C      CHAMADA DA SUBROTINA INTEGRADORA COM VETOR DE EMPUXO
C      PERPENDICULAR AO RAO VETOR.
C
C      DO 100 H=0,200
C      CALL RKF78(FPERP,NEQN,X,T,TOUT,RELERR,ABSERR,IFLAG,WORK,DT)
100 TOUT=TOUT+TB/200.
C
C      TOUT=TOUT-TB/200.
C
C      CASO EM QUE O VETOR DE EMPUXO E' TANGENTE `A VELOCIDADE.
C
C      ELSE
C

```

```

C   ENTRADA DE PARAMETROS DA SUBROTINA INTEGRADORA.
C
NEQN=5
T=0.0
TOUT=0.0
DT=TB/200.
RELERR=1D-9
ABSERR=1D-11
IFLAG=1
C
C   CHAMADA DA SUBROTINA INTEGRADORA COM VETOR DE EMPUXO
C   PARALELO `A VELOCIDADE.
C
DO 200 H=0,200
CALL RKF78(FPARA,NEQN,X,T,TOUT,RELERR,ABSERR,IFLAG,WORK,DT)
200 TOUT=TOUT+TB/200.
C
TOUT=TOUT-TB/200.
C
C   TERMINO DA FASE PROPULSIVA.
C
ENDIF
C
C   PROPAGACAO DA ORBITA POS-PROPULSIVA.
C
N=0
C
C   ENTRADA DE PARAMETROS DA SUBROTINA INTEGRADORA.
C
NEQN=5
T=0.0
TOUT=0.0
DT=20.0
RELERR=1D-9
ABSERR=1D-11
IFLAG=1
C
DO 300 TOUT=0.0,900.0,20.0
C
C   CHAMADA DA SUBROTINA INTEGRADORA SEM PROPULSAO.
C
CALL RKF78(XNEW,NEQN,X,T,TOUT,RELERR,ABSERR,IFLAG,WORK,DT)
C
N=N+1
RPOS(N)=X(1)
TETPOS(N)=X(2)
C
300 CONTINUE
C
C   OBTENCAO DOS PARAMETROS ORBITAIS POS-PROPULSIVOS.
C
R(1)=RPOS(5)
TETA(1)=TETPOS(5)
R(2)=RPOS(30)
TETA(2)=TETPOS(30)
R(3)=RPOS(44)
TETA(3)=TETPOS(44)
C
C   CHAMADA DA SUBROTINA QUE DETERMINA ELEMENTOS DA ORBITA POS-
C   PROPULSIVA.
C
CALL ORB_POS_PROP(R,TETA,ROTAP,EXC,SMEIXO,NR)
C
C   ROTACAO DA LINHA DOS APSIDES,EXCENTRICIDADE E SEMI-EIXO
C   MAIOR DA ORBITA POS-PROPULSIVA.
C
DELT(NLEI,J)=ROTAP
EX(NLEI,J)=EXC
AMAJ(NLEI,J)=SMEIXO
C
C   NOVO VALOR DE TETAB E RETORNO PARA NOVAS INTEGRACOES.
C
write(*,*)'j',j
write(*,*)'tetab',tetab(j)
write(*,*)'rotap',delt(nlei,j)
write(*,*)'exc',exc

```

```

write(*,*)'smeixo',smeixo
HP=SMEIXO*(1-EXC)
HA=SMEIXO*(1+EXC)
WRITE(*,*)'PERIGEU = ',HP,'APOGEU = ',HA

C    WRITE(*,*)'NR',NR
write(*,*)' '
TETAB(J+1)=TETAB(J)+1/100.

C
400 CONTINUE
C
C    PLOTAGEM DOS RESULTADOS.
C
L=0
DO 450 K=1,2
DO 450 J=1,40
L=L+1
D(L)=DELT(K,J)
EX1(L)=EX(K,J)
AMAJ1(L)=AMAJ(K,J)
450 CONTINUE
NUMP=J-1
RE(2)=2
RE(21)=2
RE(23)=1
NS(1)=0
NS(2)=9
TX='Anomalia Verdadeira do Inicio de Propulsao (rad)'  

TY(1)='Rotacao da Linha dos Apsides (rad)'  

TY(2)='Empuxo Perpend. ao Raio Vetor'  

TY(3)='Empuxo Paralelo a Velocidade'  

c    CALL GRAFI(NUMP,TETAB,D,TX,TY)
c#####
open(11,file='res2.dat')
write(11,*) 'Empuxo Paralelo a Velocidade'  

write(11,*)  

write(11,*) 'Anomalia Verdadeira do Inicio de Propulsao (rad)'  

DO 500 i=1,100
write(11,127) TETAB(i)
500 continue
write(11,*) 'Rotacao da Linha dos Apsides (rad)'  

DO 501 i=1,100
write(11,127) D(i)
501 continue
write(11,*) 'Semi-eixo maior da Orbita Pos-propulsiva (Km)'  

DO 502 i=1,100
write(11,128) AMAJ1(i)
502 continue
write(11,*) 'Excentricidade da Orbita Pos-propulsiva'  

DO 503 i=1,100
write(11,127) EX1(i)
503 continue
write(11,*)
close(11,status='keep')
c#####
C
TY(1)='Semi-eixo maior da Orbita Pos-propulsiva (Km)'  

c    CALL GRAFI(NUMP,TETAB,AMAJ1,TX,TY)
C
TY(1)='Excentricidade da Orbita Pos-propulsiva'  

c    CALL GRAFI(NUMP,TETAB,EX1,TX,TY)
C
STOP
127 FORMAT (1X25f9.6,' ')
128 FORMAT (1X25f9.2,' ')
END
C
C    SUBROTINA QUE FORNECE EQUACOES DA FASE NAO PROPULSIVA.
C
SUBROUTINE XNEW(T,X,XP)
IMPLICIT REAL*8(A-H,O-Z)
REAL*8 X(5),XP(5),MI
C
MI=398600.0
XP(1)=X(3)
XP(2)=X(4)/X(1)
XP(3)=X(4)*X(4)/X(1)-MI/X(1)/X(1)

```

```

XP(4)=-X(3)*X(4)/X(1)
XP(5)=0.0
RETURN
END
C
C   SUBROTINA QUE FORNECE AS EQUACOES DA FASE PROPULSIVA.
C   COM LEI DE CONTROLE DE EMPUXO TANGENTE.
C
SUBROUTINE FPARA(T,X,XP)
IMPLICIT REAL*8(A-H,O-Z)
REAL*8 X(5),XP(5),V,F0,MI,M,MO,MP,ISP,GO
C
MI=398600.0
GO=10
F0=400
ISP=300.0
MP=-F0/ISP/GO
MO=1860.-115.3-112.8
C
XP(1)=X(3)
XP(2)=X(4)/X(1)
V=SQRT(X(3)*X(3)+X(4)*X(4))
XP(3)=X(4)*X(4)/X(1)-MI/X(1)/X(1)+F0*X(3)/V/X(5)/1000
XP(4)=-X(3)*X(4)/X(1)+F0*X(4)/V/X(5)/1000
XP(5)=MP
C
RETURN
END
C
C   SUBROTINA QUE FORNECE AS EQUACOES DA FASE PROPULSIVA
C   UTILIZANDO EMPUXO PERPENDICULAR AO RAI0 VETOR.
C
SUBROUTINE FPERP(T,X,XP)
IMPLICIT REAL*8(A-H,O-Z)
REAL*8 X(5),XP(5),MI,V,F0,M,MO,MP,ISP,GO
C
MI=398600.0
GO=10.0
F0=400
ISP=300.0
MP=-F0/ISP/GO
MO=1860.0-115.3-112.8
C
XP(1)=X(3)
XP(2)=X(4)/X(1)
V=SQRT(X(3)*X(3)+X(4)*X(4))
XP(3)=X(4)*X(4)/X(1)-MI/X(1)/X(1)
XP(4)=-X(3)*X(4)/X(1)+F0/X(5)/1000
XP(5)=MP
C
RETURN
END

```

Subrotina ORB_POS_PROP

```

SUBROUTINE ORB_POS_PROP(R,TETA,ROTAP,EXC,SMEIXO,NR)
IMPLICIT REAL*8(A-H,O-Z)
REAL*8 R(3),TETA(3),RTAP1,RTAP2,RAIZ(10)
REAL*8 PI,RTAPMIN,RTAPMAX,PASSO,A1,A2,B1,B2,F1,F2
REAL*8 ROTAP,EXC,SLATUS,SMEIXO
C
C   ESTA SUBROTINA CALCULA OS ELEMENTOS ORBITAIS
C   POS-PROPULSIVOS, SUPONDO O MODELO SIMPLIFICADO
C   DO VEICULO E DAS INFLUENCIAS EXTERNAS.
C   ENTRADAS : TRES VETORES POSICAO (R,TETA rad) DA
C   ORBITA POS-PROPULSIVA.
C   SAIDA:ROTACAO DA LINHA DOS APSIDES (ROTAP),EXCEN-
C   TRICIDADE (EXC) E SEMI-EIXO MAIOR (SMEIXO) DA
C   ORBITA POS-PROPULSIVA, E O PARAMETRO DE CONTROLE
C   NR=NUMERO DE RAIZES ENCONTRADAS.
C   NOME DO PROGRAMA : POSPROP.FOR.
C
C   INTERVALO DE PROCURA DA ROTACAO DA LINHA DOS APSIDES
C
PI=3.14159265359

```

```

RTAPMIN=-PI
RTAPMAX=PI
C
PASSO=(RTAPMAX-RTAPMIN)/100
RTAP1=RTAPMIN
RTAP2=RTAP1+PASSO
C
NR=0
C
C INICIO DO PROCESSO DE PROCURA DA ROT. LINHA APSIDES
C
5 IF (RTAP2.LE.RTAPMAX) THEN
A1=(R(3)-R(2))*(R(1)*COS(TETA(1)-RTAP1)-R(2)*COS(TETA(2)-RTAP1))
B1=(R(2)-R(1))*(R(2)*COS(TETA(2)-RTAP1)-R(3)*COS(TETA(3)-RTAP1))
A2=(R(3)-R(2))*(R(1)*COS(TETA(1)-RTAP2)-R(2)*COS(TETA(2)-RTAP2))
B2=(R(2)-R(1))*(R(2)*COS(TETA(2)-RTAP2)-R(3)*COS(TETA(3)-RTAP2))
F1=A1-B1
F2=A2-B2
IF ((F1*F2).LE.0) THEN
IF (PASSO.GT.1D-8) THEN
PASSO=PASSO/10
RTAP2=RTAP1+PASSO
ELSE
C
C CALCULO DA ROTACAO DA LINHA DOS APSIDES.
C PARA CADA SOLUCAO ENCONTRADA PARA ROTAP, EXISTIRA' A
C SOLUCAO PI+ROTAP. DEVE-SE SER ESCOLHIDA A SOLUCAO DE
C MENOR VALOR ABSOLUTO.
C
NR=NR+1
ROTAP=(RTAP1+RTAP2)/2
RAIZ(NR)=ROTAP
C
PASSO=(RTAPMAX-RTAPMIN)/100
RTAP1=RTAP2+PASSO
RTAP2=RTAP1+PASSO
C
C BUSCA DE UMA NOVA SOLUCAO NO INTERVALO
C
ENDIF
ELSE
RTAP1=RTAP2
RTAP2=RTAP1+PASSO
ENDIF
GO TO 5
ENDIF
C
C ESCOLHA DA RAIZ DE MENOR MAGNITUDE
C
ROTAP=RAIZ(1)
DO 10 I=1,NR
IF (ABS(ROTAP).GT.ABS(RAIZ(I))) THEN
ROTAP=RAIZ(I)
ENDIF
10 CONTINUE
C
C CALCULO DA EXCENTRICIDADE E DO SEMI-EIXO MAIOR.
C
EXC=(R(2)-R(1))/(R(1)*COS(TETA(1)-ROTAP)-R(2)*COS(TETA(2)-ROTAP))
SLATUS=R(1)*(1+EXC*COS(TETA(1)-ROTAP))
SMEIXO=SLATUS/(1-EXC*EXC)
C
RETURN
END

```



```

SUBROUTINE RKF78 (F, NEQN, X, T, TOUT, RELERR, ABSERR, IFLAG,
*                WORK, DT)

```

```

C
C-----
C

```

```

C PURPOSE:

```

```

C THE SUBROUTINE RKF78 INTEGRATES A SYSTEM OF NEQN
C FIRST ORDER ORDINARY DIFFERENTIAL EQUATIONS USING
C THE RUNGE-KUTTA METHOD OF ORDER 7(8) WITH AUTO-
C MATIC STEPSIZE CONTROL, AND USING THE FEHLBERG
C COEFFICIENTS.

```

```

C INPUT:

```

```

C F SUBROUTINE WHICH PROVIDES THE DERIVATIVE
C VALUES DX, GIVEN THE VALUES OF THE DEPENDENT
C VARIABLE X AND THE INDEPENDENT
C VARIABLE T. THE SUBROUTINE F SHOULD BE
C SUPPLIED BY THE USER IN THE FORM GIVEN
C BELOW:

```

```

SUBROUTINE F(T,X,DX).

```

```

SINCE F IS AN ARGUMENT IN THE CALL STATEMENT
OF RKF78, THE DECLARATION:

```

```

"EXTERNAL F"

```

```

IS COMPULSORY IN THE MAIN PROGRAM.

```

```

C NEQN NUMBER OF EQUATIONS TO BE INTEGRATED.

```

```

C X INITIAL DEPENDENT VARIABLE ARRAY
C DIMENSIONED TO NEQN WORDS.

```

```

C T INITIAL VALUE OF THE INDEPENDENT
C VARIABLE.

```

```

C TOUT NEXT VALUE OF T FOR WHICH THE OUTPUT IS
C DESIRED.

```

```

C RELERR ARRAY OF RELATIVE ERROR TOLERANCES DIMEN-
C SIONED TO NEQN WORDS.

```

```

C ABSERR ARRAY OF ABSOLUTE ERROR TOLERANCES DIMEN-
C SIONED TO NEQN WORDS.

```

```

THE ESTIMATED LOCAL TRUNCATION ERROR IS
KEPT LESS THAN

```

```

(RELERR(I)*(X(I)+WORK(I))/2 + ABSERR(I))

```

```

AT EACH STEP OF THE INTEGRATION

```

```

(I = 1, NEQN). IF ABSERR = RELERR = 0.0,

```

```

NO STEPSIZE ADJUSTMENTS ARE MADE AND THE

```

```

SOLUTION IS OBTAINED BY A FIXED STEPSIZE

```

```

EIGHTH ORDER METHOD.

```

```

C IFLAG CONTROL PARAMETER TO INITIALIZE THE
C ROUTINE: TO BE SET TO 1 ON THE FIRST CALL
C OF EACH NEW PROBLEM.

```

```

C WORK REAL ARRAY DIMENSIONED TO AT LEAST
C 14*NEQN WORDS. ANY ARRAY WHOSE CONTENTS
C ARE EXPENDABLE MAY BE USED.

```

```

C DT INITIAL STEP SIZE (LESS THAN OR EQUAL TO
C THE DIFFERENCE BETWEEN T AND THE NEXT
C VALUE OF T, I.E. TOUT, FOR WHICH OUTPUT
C IS DESIRED).

```

```

COMMON/COEF78/

```

```

C A0,... FEHLBERG COEFFICIENTS A0,...,A12, B10,...
C B1211, CH0,...,CH12, E0,...,E12.

```

C B, ... STEPSIZE CONTROL FACTORS, AND NECESSARY
 C CONSTANTS USEFUL TO CALCULATE THEM.
 C
 C OUTPUT:
 C X DEPENDENT VARIABLE ARRAY AT T.
 C T TOUT IF IFLAG = 2. NORMAL RETURN.
 C LAST VALUE (DIFFERENT FROM TOUT) ATTAINED
 C BY THE INDEPENDENT VARIABLE IF IFLAG # 2.
 C ABNORMAL RETURN.
 C IFLAG FLAG FOR THE TYPE OF RETURN:
 C = 2: NORMAL RETURN , T REACHED TOUT. FOR
 C CONTINUATION, DEFINE A NEW TOUT AND
 C CALL THE ROUTINE AGAIN.
 C = 7: MORE THAN MAXREJ REJECTED STEPS IN A
 C ROW. IF THIS OCCURS ON THE FIRST
 C STEP, REDUCE DT AND CALL THE ROUTINE
 C AGAIN. IF THIS OCCURS IN THE MIDDLE
 C OF THE INTEGRATION, INCREASE RELERR
 C AND ABSERR TO CONTINUE THE
 C INTEGRATION.
 C = 8: THE PROGRAM ATTEMPTED TO USE TOO
 C SMALL A STEP. INCREASE DT FOR
 C CONTINUATION.
 C DT MOST RECENT STEP SIZE.
 C
 C COMMON/CTRK78/
 C N... SOME CONSTANT VALUES USEFUL IN SUBSEQUENT
 C CALLS OF THE SUBROUTINE.
 C
 C SUBCALLS:
 C RK78CO.
 C
 C AUTHORS:
 C RICHARD E.MCKENZIE(UNIV. OF TEXAS, U.S.A)
 C 1976 VERSION 1.0
 C KONDAPALLI R. RAO; HELIO K. KUGA
 C OCTOBER/85 VERSION 2.0
 C REF. :
 C KONDAPALLI R. RAO : A REVIEW ON NUMERICAL METHODS
 C FOR INITIAL VALUE PROBLEMS (INPE-3011-RPI/088).
 C KONDAPALLI R. RAO ; HELIO K. KUGA : MANUAL DE USO
 C DE UM CONJUNTO DE INTEGRADORES NUMERICOS PARA
 C PROBLEMAS DE CONDICoes INICIAIS.
 C (INPE-3830-RPI/154).
 C
 C REMARKS:
 C THE 1976 VERSION OF RICHARD E. MCKENZIE WAS MODI-
 C FIED IN TERMS OF NUMBER OF PARAMETERS AND DIMEN-
 C SIONS TO FACILITATE THE USER, BY KONDAPALLI R.
 C RAO, INPE, IN SEPTEMBER 1984. ALSO AN ERROR IN
 C THE VERSION OF MCKENZIE WAS CORRECTED BY HELIO K.
 C KUGA AND KONDAPALLI R. RAO IN OCTOBER 1985. THE
 C SINGLE PRECISION VERSION OF THE PROGRAM WAS
 C TRANSFORMED INTO DOUBLE PRECISION BY HELIO K.
 C KUGA IN JULY 1987.
 C
 C
 C

```
IMPLICIT REAL*8 (A-H,O-Z)
```

```
DIMENSION X(*),RELERR(*),ABSERR(*),WORK(*)
```

```
COMMON/COEF78/A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,
*      A12,B10,B20,B21,B30,B31,B32,B40,B41,B42,
1      B43,B50,B51,B52,B53,B54,B60,B61,B62,B63,
2      B64,B65,B70,B71,B72,B73,B74,B75,B76,B80,
3      B81,B82,B83,B84,B85,B86,B87,B90,B91,B92,
4      B93,B94,B95,B96,B97,B98,B100,B101,B102,
5      B103,B104,B105,B106,B107,B108,B109,B110,
6      B111,B112,B113,B114,B115,B116,B117,B118,
7      B119,B110,B120,B121,B122,B123,B124,
8      B125,B126,B127,B128,B129,B1210,B1211,
9      CH0,CH1,CH2,CH3,CH4,CH5,CH6,CH7,CH8,CH9,
A      CH10,CH11,CH12,E0,E1,E2,E3,E4,E5,E6,E7,
B      E8,E9,E10,E11,E12,B,BLG,BUP,REMIN,DTINC,
C      DTDEC,MAXREJ
COMMON/CTRK78/N,N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,
1      N12,N13
```

```
LOGICAL DTFAIL,DTFIX
```

```
C
C-----
C
```

```
C
C      SET FEHLBERG COEFFICIENTS AND THE VALUES OF SOME
C      CONSTANTS ON THE FIRST CALL
C
```

```
IF (IFLAG.NE.1) GO TO 5
CALL RK78CO
N = NEQN
N1 = N
N2 = 2*N
N3 = 3*N
N4 = 4*N
N5 = 5*N
N6 = 6*N
N7 = 7*N
N8 = 8*N
N9 = 9*N
N10 = 10*N
N11 = 11*N
N12 = 12*N
N13 = 13*N
5 NREJT = 0
NSTP = 0
```

```
C
C      SET FLAG IF FIXED STEP MODE IS DESIRED
C
```

```
DTFIX = .FALSE.
DO 10 NEQ = 1,N
  IF (ABSERR(NEQ) .EQ. 0. D0 .AND. RELERR(NEQ) .EQ. 0. D0)
```

```

*          DTFIX = .TRUE.
10 CONTINUE
   DTOLD = DT
20 DTFAIL = .FALSE.
   NREJ = 0

C
C   RESET STEP SIZE IF THIS WILL PUT T GREATER THAN TOUT
C

   DELT = TOUT-T
   IF (DABS(DT).LT.DABS(DELT)) GO TO 25
   DT = DELT
   GO TO 30
25 IF (DABS(DT+DT).LT.DABS(DELT)) GO TO 30
   DT = DELT/2.D0

C
C   IF REQUIRED STEP IS TOO SMALL, EXTRAPOLATE AND RETURN
C

30 IF(DABS(DT).LT.1.D-15*DABS(T)) GO TO 160

C
C   FIRST EVALUATION
C

   T0 = T
   DO 35 NEQ = 1,N
     WORK(NEQ) = X(NEQ)
35 CONTINUE
   CALL F(T,X,WORK(N1+1))

C
C   SECOND EVALUATION
C

40 T = T0+A1*DT
   D0 = B10*DT
   DO 45 NEQ = 1,N
     X(NEQ) = D0*WORK(N1+NEQ)+WORK(NEQ)
45 CONTINUE
   CALL F(T,X,WORK(N2+1))

C
C   THIRD EVALUATION
C

   T = T0+A2*DT
   D0 = B20*DT
   D1 = B21*DT
   DO 50 NEQ = 1,N
     X(NEQ) = D0*WORK(N1+NEQ)+D1*WORK(N2+NEQ)+WORK(NEQ)
50 CONTINUE
   CALL F(T,X,WORK(N3+1))

C

```

C FOURTH EVALUATION
C

```

T = T0+A3*DT
D0 = B30*DT
D1 = B31*DT
D2 = B32*DT
DO 55 NEQ = 1, N
  X (NEQ) = D0*WORK (N1+NEQ) +D1*WORK (N2+NEQ) +D2*WORK (N3+NEQ) +
1      WORK (NEQ)
55 CONTINUE
CALL F(T,X,WORK(N4+1))

```

C FIFTH EVALUATION
C
C

```

T = T0+A4*DT
D0 = B40*DT
D1 = B41*DT
D2 = B42*DT
D3 = B43*DT
DO 60 NEQ = 1, N
  X (NEQ) = D0*WORK (N1+NEQ) +D1*WORK (N2+NEQ) +D2*WORK (N3+NEQ) +
1      D3*WORK (N4+NEQ) +WORK (NEQ)
60 CONTINUE
CALL F(T,X,WORK(N5+1))

```

C SIXTH EVALUATION
C
C

```

T = T0+A5*DT
D0 = B50*DT
D1 = B51*DT
D2 = B52*DT
D3 = B53*DT
D4 = B54*DT
DO 65 NEQ = 1, N
  X (NEQ) = D0*WORK (N1+NEQ) +D1*WORK (N2+NEQ) +D2*WORK (N3+NEQ) +
1      D3*WORK (N4+NEQ) +D4*WORK (N5+NEQ) +WORK (NEQ)
65 CONTINUE
CALL F(T,X,WORK(N6+1))

```

C SEVENTH EVALUATION
C
C

```

T = T0+A6*DT
D0 = B60*DT
D1 = B61*DT
D2 = B62*DT
D3 = B63*DT
D4 = B64*DT
D5 = B65*DT
DO 70 NEQ = 1, N
  X (NEQ) = D0*WORK (N1+NEQ) +D1*WORK (N2+NEQ) +D2*WORK (N3+NEQ) +

```

```

1          D3*WORK(N4+NEQ)+D4*WORK(N5+NEQ)+D5*WORK(N6+NEQ)+
2          WORK(NEQ)
70 CONTINUE
   CALL F(T,X,WORK(N7+1))

```

C
C
C

EIGHTH EVALUATION

```

T = T0+A7*DT
D0 = B70*DT
D1 = B71*DT
D2 = B72*DT
D3 = B73*DT
D4 = B74*DT
D5 = B75*DT
D6 = B76*DT
DO 75 NEQ = 1,N
   X(NEQ) = D0*WORK(N1+NEQ)+D1*WORK(N2+NEQ)+D2*WORK(N3+NEQ)+
1          D3*WORK(N4+NEQ)+D4*WORK(N5+NEQ)+D5*WORK(N6+NEQ)+
2          D6*WORK(N7+NEQ)+WORK(NEQ)
75 CONTINUE
   CALL F(T,X,WORK(N8+1))

```

C
C
C

NINTH EVALUATION

```

T = T0+A8*DT
D0 = B80*DT
D1 = B81*DT
D2 = B82*DT
D3 = B83*DT
D4 = B84*DT
D5 = B85*DT
D6 = B86*DT
D7 = B87*DT
DO 80 NEQ = 1,N
   X(NEQ) = D0*WORK(N1+NEQ)+D1*WORK(N2+NEQ)+D2*WORK(N3+NEQ)+
1          D3*WORK(N4+NEQ)+D4*WORK(N5+NEQ)+D5*WORK(N6+NEQ)+
2          D6*WORK(N7+NEQ)+D7*WORK(N8+NEQ)+WORK(NEQ)
80 CONTINUE
   CALL F(T,X,WORK(N9+1))

```

C
C
C

TENTH EVALUATION

```

T = T0+A9*DT
D0 = B90*DT
D1 = B91*DT
D2 = B92*DT
D3 = B93*DT
D4 = B94*DT
D5 = B95*DT
D6 = B96*DT
D7 = B97*DT

```

```

D8 = B98*DT
DO 85 NEQ = 1,N
  X(NEQ) = D0*WORK(N1+NEQ)+D1*WORK(N2+NEQ)+D2*WORK(N3+NEQ)+
1      D3*WORK(N4+NEQ)+D4*WORK(N5+NEQ)+D5*WORK(N6+NEQ)+
2      D6*WORK(N7+NEQ)+D7*WORK(N8+NEQ)+D8*WORK(N9+NEQ)+
3      WORK(NEQ)
85 CONTINUE
  CALL F(T,X,WORK(N10+1))

```

```

C
C  ELEVENTH EVALUATION
C

```

```

T = T0+A10*DT
D0 = B100*DT
D1 = B101*DT
D2 = B102*DT
D3 = B103*DT
D4 = B104*DT
D5 = B105*DT
D6 = B106*DT
D7 = B107*DT
D8 = B108*DT
D9 = B109*DT
DO 90 NEQ = 1,N
  X(NEQ) = D0*WORK(N1+NEQ)+D1*WORK(N2+NEQ)+D2*WORK(N3+NEQ)+
1      D3*WORK(N4+NEQ)+D4*WORK(N5+NEQ)+D5*WORK(N6+NEQ)+
2      D6*WORK(N7+NEQ)+D7*WORK(N8+NEQ)+D8*WORK(N9+NEQ)+
3      D9*WORK(N10+NEQ)+WORK(NEQ)
90 CONTINUE
  CALL F(T,X,WORK(N11+1))

```

```

C
C  TWELFTH EVALUATION
C

```

```

T = T0+A11*DT
D0 = B110*DT
D1 = B111*DT
D2 = B112*DT
D3 = B113*DT
D4 = B114*DT
D5 = B115*DT
D6 = B116*DT
D7 = B117*DT
D8 = B118*DT
D9 = B119*DT
D10 = B1110*DT
DO 95 NEQ = 1,N
  X(NEQ) = D0*WORK(N1+NEQ)+D1*WORK(N2+NEQ)+D2*WORK(N3+NEQ)+
1      D3*WORK(N4+NEQ)+D4*WORK(N5+NEQ)+D5*WORK(N6+NEQ)+
2      D6*WORK(N7+NEQ)+D7*WORK(N8+NEQ)+D8*WORK(N9+NEQ)+
3      D9*WORK(N10+NEQ)+D10*WORK(N11+NEQ)+WORK(NEQ)
95 CONTINUE
  CALL F(T,X,WORK(N12+1))

```

```

C

```

C THIRTEENTH EVALUATION
C

```

T = T0+A12*DT
D0 = B120*DT
D1 = B121*DT
D2 = B122*DT
D3 = B123*DT
D4 = B124*DT
D5 = B125*DT
D6 = B126*DT
D7 = B127*DT
D8 = B128*DT
D9 = B129*DT
D10 = B1210*DT
D11 = B1211*DT
DO 100 NEQ = 1,N
  X (NEQ) = D0*WORK (N1+NEQ) +D1*WORK (N2+NEQ) +D2*WORK (N3+NEQ) +
1      D3*WORK (N4+NEQ) +D4*WORK (N5+NEQ) +D5*WORK (N6+NEQ) +
2      D6*WORK (N7+NEQ) +D7*WORK (N8+NEQ) +D8*WORK (N9+NEQ) +
3      D9*WORK (N10+NEQ) +D10*WORK (N11+NEQ) +D11*WORK (N12+NEQ) +
4      WORK (NEQ)
100 CONTINUE
  CALL F(T,X,WORK(N13+1))

```

C
C COMPUTE STATE AT T + DT
C

```

D0 = CH0*DT
D1 = CH1*DT
D2 = CH2*DT
D3 = CH3*DT
D4 = CH4*DT
D5 = CH5*DT
D6 = CH6*DT
D7 = CH7*DT
D8 = CH8*DT
D9 = CH9*DT
D10 = CH10*DT
D11 = CH11*DT
D12 = CH12*DT
DO 105 NEQ = 1,N
  X (NEQ) = D0*WORK (N1+NEQ) +D1*WORK (N2+NEQ) +D2*WORK (N3+NEQ) +
1      D3*WORK (N4+NEQ) +D4*WORK (N5+NEQ) +D5*WORK (N6+NEQ) +
2      D6*WORK (N7+NEQ) +D7*WORK (N8+NEQ) +D8*WORK (N9+NEQ) +
3      D9*WORK (N10+NEQ) +D10*WORK (N11+NEQ) +D11*WORK (N12+NEQ) +
4      D12*WORK (N13+NEQ) +WORK (NEQ)
105 CONTINUE

```

C
C IF FIXED STEP SIZE IS DESIRED GO TO 140
C

IF (DTFIX) GO TO 140

C


```

C      COMPUTE MAX LOCAL TRUNCATION ERROR
C
      RTE = 0.D0
      D0 = E0*DT
      D1 = E1*DT
      D2 = E2*DT
      D3 = E3*DT
      D4 = E4*DT
      D5 = E5*DT
      D6 = E6*DT
      D7 = E7*DT
      D8 = E8*DT
      D9 = E9*DT
      D10 = E10*DT
      D11 = E11*DT
      D12 = E12*DT
      DO 110 NEQ = 1,N
          RER = DMAX1 (RELERR (NEQ) , 1.D-16+REMIN)
          SCALE = RER/2.D0
          TE = DABS (D0*WORK (N1+NEQ) +D1*WORK (N2+NEQ) +D2*WORK (N3+NEQ) +
1          D3*WORK (N4+NEQ) +D4*WORK (N5+NEQ) +D5*WORK (N6+NEQ) +
2          D6*WORK (N7+NEQ) +D7*WORK (N8+NEQ) +D8*WORK (N9+NEQ) +
3          D9*WORK (N10+NEQ) +D10*WORK (N11+NEQ) +D11*WORK (N12+NEQ) +
4          D12*WORK (N13+NEQ) )
          XMAG = (DABS (X (NEQ) ) +DABS (WORK (NEQ) ) ) *SCALE+ABSERR (NEQ) +
1          1.0D-15
          RTE = DMAX1 (RTE, TE/XMAG)
110 CONTINUE
      IF (RTE.LT.1.D0) GO TO 140

C
C      REJECT THIS STEP
C
      DTFAIL = .TRUE.
      NREJ = NREJ+1
      NREJT = NREJT+1
      IF (NREJ.LT.MAXREJ) GO TO 130
      DO 120 NEQ = 1,N
          X (NEQ) = WORK (NEQ)
120 CONTINUE
      T = T0
      IFLAG = 7
      RETURN
130 PCT = DTDEC
      IF (RTE.LT.BUP) THEN
          ZZZUTV = .125D0
          PCT = B/RTE**ZZZUTV
      ENDIF
      DT = PCT*DT
      DTOLD = DT
      GO TO 40

C
C      THIS STEP IS ACCEPTABLE - EIGHTH ORDER EVALUATION.
C

```

```

140 T = T0+DT
    NSTP = NSTP+1
    IF (DABS(TOUT-T).GT.1.0D-15) GO TO 150
    DT = DTOLD
    IFLAG = 2
    RETURN
150 IF (DTFIX) GO TO 20
    PCT = DTINC
    IF (RTE.GT.BLO) THEN
        ZZZUTV = .125D0
        PCT = B/RTE**ZZZUTV
    ENDIF
    IF (DTFAIL) THEN
        ZZZUTV = 1.D0
        PCT = DMIN1(PCT,ZZZUTV)
    ENDIF
    DT = DT*PCT
    DTOLD = DT
    GO TO 20

```

```

C
C CHECK FOR TOO SMALL A STEP SIZE
C

```

```

160 IF (DABS(DELT).GT.DABS(DT)) GO TO 180

```

```

C
C STRAIGHT LINE EXTRAPOLATION (EULER'S METHOD OF ORDER 1)
C

```

```

    CALL F(T,X,WORK(N1+1))
    DO 170 NEQ = 1,N
        X(NEQ) = DT*WORK(N1+NEQ)+X(NEQ)
170 CONTINUE
    T = T+DT
    DT = DTOLD
    IFLAG = 2
    RETURN

```

```

C
C ATTEMPTED TO USE TOO SMALL A STEP SIZE
C

```

```

180 IFLAG = 8
    RETURN
    END

```