A numerical procedure based on the fourth order Runge-Kutta method for two point boundary value ordinary differential equation (ODE) systems with applications to a model in flight mechanics

Fabio Silva Botelho
Department of Mathematics
Federal University of Santa Catarina, UFSC
Florianópolis, SC - Brazil
e-mail: fabio.botelho@ufsc.br

Abstract

This short communication develops a numerical procedure suitable for a large class of ordinary differential equation systems found in models in physics and engineering. More specifically, we have utilized the standard fourth order Runge-Kutta method combined with a MAT-LAB function for obtaining a method suitable for a class of two point boundary value ODE non-linear systems. In the last section, we apply the method to a model in flight mechanics.

Keywords: Two point boundary value ODE system, fourth order Runge-Kutta method, model in flight mechanics.

MSC: 65L10, 65L12

1 Indroduction

In this article we develop an application of the fourth order Runge-Kutta method adapted for a two point boundary value ODE (ordinary differential equation) system to a model in flight mechanics.

The model in flight mechanics here addressed may be found in [1].

Details on the Sobolev spaces involved may be found in [2].

Related results to those concerning the generalized method of lines, may be found in the here referenced books of 2011 and 2020, [3, 4] respectively.

For a reference on standard finite differences schemes, please see [5].

2 The main numerical method

Consider the first order system of ordinary differential equations given by

$$\frac{du_j}{dt} = f_j(\{u_l\}), \text{ on } [0, t_f] \; \forall j \in \{1, 2, 3\},$$

with the boundary conditions

$$u_1(0) = 0, \;\; u_2(0) = 0.15, \; u_1(t_f) = u_f.$$

Here $\mathbf{u} = \{u_l\} \in W^{1,2}([0, t_f], \mathbb{R}^3)$ and f_j are functions at least of C^1 class, $\forall j \in \{1, 2, 3\}$.

Concerning a finite differences scheme, we denote the number of nodes on $[0, t_f]$ by m_8 and set $d = t_f/m_8$.

Based on the fourth order Runge-Kutta method, we propose the following algorithm.

In finite differences, (indeed we have made it through an appropriate MAT-LAB function), we define

$$u_1(0) = 0, \; u_2(0) = 0.15, \; u_3(0) = V_0;$$

$$\begin{aligned}
(k_1)_j &= f_j(u_1(0), u_2(0), u_3(0)) = f_j(0, 0.15, V_0), \\
(k_2)_j &= f_j(u_1(0) + (k_1)_1 d/2, u_2(0) + (k_1)_2 d/2, u_3(0) + (k_1)_3 d/2), \\
(k_3)_j &= f_j(u_1(0) + (k_2)_1 d/2, u_2(0) + (k_2)_2 d/2, u_3(0) + (k_2)_3 d/2), \\
(k_4)_j &= f_j(u_1(0) + (k_3)_1 d, u_2(0) + (k_3)_2 d, u_3(0) + (k_3)_3 d),
\end{aligned}$$

$\forall j \in \{1, 2, 3\}$, so that

$$(u_j)_1 = u_j(0) + \frac{d}{6}((k_1)_j + 2(k_2)_j + 2(k_3)_j + (k_4)_j),$$

$\forall j \in \{1, 2, 3\}$.

Inductively, for $n \in \mathbb{N}$, $n \geq 1$, we set

$$\begin{aligned}
(k_1)_j &= f_j((u_1)_n, (u_2)_n, (u_n)_3) \\
(k_2)_j &= f_j((u_1)_n + (k_1)_1 d/2, (u_2)_n + (k_1)_2 d/2, (u_3)_n + (k_1)_3 d/2), \\
(k_3)_j &= f_j((u_1)_n + (k_2)_1 d/2, (u_2)_n + (k_2)_2 d/2, (u_3)_n + (k_2)_3 d/2), \\
(k_4)_j &= f_j((u_1)_n + (k_3)_1 d, (u_2)_n + (k_3)_2 d, (u_3)_n + (k_3)_3 d),
\end{aligned}$$

$\forall j \in \{1, 2, 3\}$, so that

$$(u_j)_{n+1} = (u_j)_n + \frac{d}{6}((k_1)_j + 2(k_2)_j + 2(k_3)_j + (k_4)_j),$$

$\forall j \in \{1, 2, 3\}$, $\forall n \in \{1, \cdots, m_8 - 1\}$.

Finally, having obtained

$$(u_1)_{m8}(V_0),$$

we obtain V_0 by minimizing the function

$$J(V_0) = ((u_1)_{m8}(V_0) - u_f)^2.$$

The problem is then approximately solved.

3 Applications to a flight mechanics model

We present numerical results for the following system of equations, which models the in plane climbing motion of an airplane (please, see more details in [1]).

$$\begin{cases} \dot{h} = V \sin\gamma, \\ \dot{\gamma} = \frac{1}{m_f V}(F\sin[a + a_F] + L) - \frac{g}{V}\cos\gamma, \\ \dot{V} = \frac{1}{m_f}(F\cos[a + a_F]) - D) - g\sin\gamma \\ \dot{x} = V\cos\gamma, \end{cases} \tag{1}$$

with the boundary conditions,

$$\begin{cases} h(0) = h_0, \\ \gamma(0) = \gamma_0 \\ x(0) = x_0 \\ h(t_f) = h_f, \end{cases} \tag{2}$$

where $t_f = 739s$, h is the airplane altitude, V is its speed, γ is the angle between its velocity and the horizontal axis, and finally x denotes the horizontal coordinate position.

For numerical purposes, we assume (data close to those of an Air bus 320)
$m_f = 120,000Kg$, $S_f = 260m^2$, $a = 0.138$ rad, $g = 9.8m/s^2$,

$$\rho(h) = 1.225(1 - 0.0065h/288.15)^{4.225}Kg/m^3,$$

$$a_F = 0.0175,$$

$$(C_L)_a = 5$$

$$(C_D)_0 = 0.0175,$$

$$K_1 = 0.0,$$

$$K_2 = 0.06$$

$$(C_L)_0 = 0,$$

$$C_D = (C_D)_0 + K_1 C_L + K_2 C_L^2,$$

$$C_L = (C_L)_0 + (C_L)_a a,$$

$$L = \frac{1}{2}\rho(h)V^2 C_L S_f,$$

$$D = \frac{1}{2}\rho(h)V^2 C_D S_f,$$

$$F = 240000$$

and where units refer to the International System.

To simplify the analysis, we redefine the variables as below indicated:

$$\begin{cases} h = u_1, \\ \gamma = u_2 \\ V = u_3 \\ x = u_4. \end{cases} \tag{3}$$

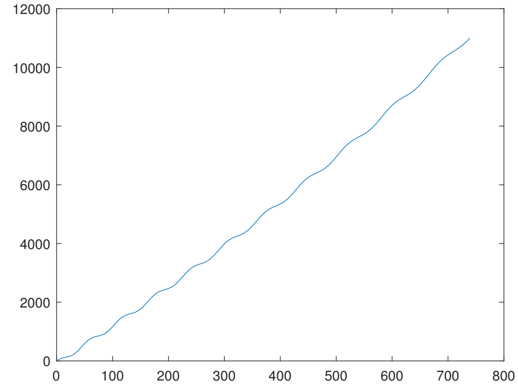

Figure 1: The solution h (in m) for $t_f = 739s$.

Thus, denoting $\mathbf{u} = (u_1, u_2, u_3, u_4) \in U = W^{1,2}([0, t_f]; \mathbb{R}^4)$, the system above indicated may be expressed by

$$\begin{cases} \dot{u}_1 = f_1(\mathbf{u}) \\ \dot{u}_2 = f_2(\mathbf{u}) \\ \dot{u}_3 = f_3(\mathbf{u}) \\ \dot{u}_4 = f_4(\mathbf{u}), \end{cases} \tag{4}$$

where,

$$\begin{cases} f_1(\mathbf{u}) = u_3 \sin(u_2), \\ f_2(\mathbf{u}) = \frac{1}{m_f u_3}(F \sin[a + a_F] + L(\mathbf{u})) - \frac{g}{u_3}\cos(u_2), \\ f_3(\mathbf{u}) = \frac{1}{m_f}(F \cos[a + a_F] - D(\mathbf{u})) - g \sin(u_2) \\ f_4(\mathbf{u}) = u_3 \cos(u_2). \end{cases} \tag{5}$$

We solve this last system for the following boundary conditions:

$$\begin{cases} h(0) = 0 \ m, \\ \gamma(0) = 0.15, \\ x(0) = 0 \ m, \\ h(t_f) = 11000 \ m. \end{cases} \tag{6}$$

Observe that having calculated a solution (h, γ, V) we may easily calculate x through the fourth equation, that is,

$$x(t) = x(0) + \int_0^t V(t) \cos(\gamma(t)) \, dt$$

Thus, for a solution, we may consider just the first three equations in the three variables (h, γ, V). We have numerically obtained the solutions for h, γ, V. Please see figures 1, 2 and 3, respectively. We have set $m_8 = 10000$ nodes.

Here we present the software in MAT-LAB entitled "FlightMech20Aug2026C" through which such numerical results have been obtained.

****

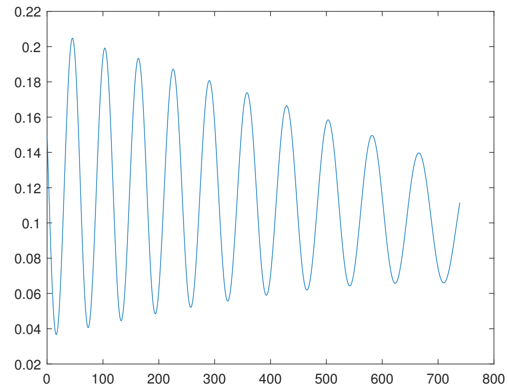

Figure 2: The solution γ (in rad) for $t_f = 739s$.

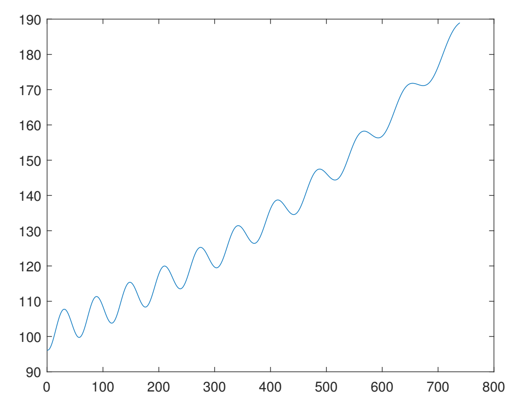

Figure 3: The solution V (in m/s) for $t_f = 739s$.

1. clear all
global m8 d tf CL CD Sf mf a aF F h v g W5 G
m8=10000;
tf=739;
d=tf/m8;
mf=120000;
Sf=260;
G=9.81;
a=0.138;
aF=0.0175;
F=240000;
CLa=5;
CDo=0.0175;
K1=0.0;
K2=0.06;
CLo=0;
CL=CLo+CLa*a;
$CD = CDo + K1 * CL + K2 * CL^2$;
xo(1,1)=120;
b12=1;
k1=1;
while $(b12 > 10^{-4})$ && $(k1 < 10)$
X=lsqnonlin('fun20Aug2026C',xo);
b12=abs(X-xo);
xo=X;
h(m8/2,1);
end;
for i=1:m8
x8(i,1)=i*d;
end;
plot(x8,h)

****
With the main function entitled "fun20Aug2026C"
****

1. function W5=fun20Aug2026C(x)
global m8 d tf CL CD Sf mf a aF F h v g W5 G
Ho(1,1)=0;
Vo(1,1)=x(1,1);
Go(1,1)=0.15;

k=1;

$r1(k) = abs(1.225 * (1 - 0.0065 * Ho(1, 1)/288.15)^{(}4.225));$

$L1 = 1/2 * r1(k) * Vo(1, 1)^2 * CL * Sf;$

$D1 = 1/2 * r1(k) * Vo(1, 1)^2 * CD * Sf;$

h1=Vo(1,1)*sin(Go(1,1));

g1=1/mf*(F*sin(a+aF)+L1)/Vo(1,1)-G/Vo(1,1)*cos(Go(1,1));

v1=1/mf*(F*cos(a+aF)-D1)-G*sin(Go(1,1));

$r2(k) = abs(1.225 * (1 - 0.0065 * (Ho(1, 1) + d/2 * h1)/288.15)^{(}4.225));$

$L2 = 1/2 * r2(k) * (Vo(1, 1) + d/2 * v1)^2 * CL * Sf;$

$D2 = 1/2 * r2(k) * (Vo(1, 1) + d/2 * v1)^2 * CD * Sf;$

h2=(Vo(1,1)+d/2*v1)*sin(Go(1,1)+d/2*g1);

g2=1/mf*(F*sin(a+aF)+L2)/(Vo(1,1)+d/2*v1)-G/(Vo(1,1)+d/2*v1)*cos(Go(1,1)+d/2*g1);

v2=1/mf*(F*cos(a+aF)-D2)-G*sin(Go(1,1)+d/2*g1);

$r3(k) = abs(1.225 * (1 - 0.0065 * (Ho(1, 1) + d/2 * h2)/288.15)^{(}4.225));$

$L3 = 1/2 * r3(k) * (Vo(1, 1) + d/2 * v2)^2 * CL * Sf;$

$D3 = 1/2 * r3(k) * (Vo(1, 1) + d/2 * v2)^2 * CD * Sf;$

h3=(Vo(1,1)+d/2*v2)*sin(Go(1,1)+d/2*g2);

g3=1/mf*(F*sin(a+aF)+L3)/(Vo(1,1)+d/2*v2)-G/(Vo(1,1)+d/2*v2)*cos(Go(1,1)+d/2*g2);

v3=1/mf*(F*cos(a+aF)-D3)-G*sin(Go(1,1)+d/2*g2);

$r4(k) = abs(1.225 * (1 - 0.0065 * (Ho(1, 1) + d * h3)/288.15)^{(}4.225));$

$L4 = 1/2 * r4(k) * (Vo(1, 1) + d * v3)^2 * CL * Sf;$

$D4 = 1/2 * r4(k) * (Vo(1, 1) + d * v3)^2 * CD * Sf;$

h4=(Vo(1,1)+d*v3)*sin(Go(1,1)+d*g3);

g4=1/mf*(F*sin(a+aF)+L4)/(Vo(1,1)+d*v3)-G/(Vo(1,1)+d*v3)*cos(Go(1,1)+d*g3);

v4=1/mf*(F*cos(a+aF)-D4)-G*sin(Go(1,1)+d*g3);

h(k,1)=(Ho(1,1)+d/6*(h1+2*h2+2*h3+h4));

g(k,1)=(Go(1,1)+d/6*(g1+2*g2+2*g3+g4));

v(k,1)=(Vo(1,1)+d/6*(v1+2*v2+2*v3+v4));

for k=2:m8

$r1(k) = abs(1.225 * (1 - 0.0065 * h(k - 1, 1)/288.15)^{(}4.225));$

$L1 = 1/2 * r1(k) * v(k - 1, 1)^2 * CL * Sf;$

$D1 = 1/2 * r1(k) * v(k - 1, 1)^2 * CD * Sf;$

h1=v(k-1,1)*sin(g(k-1,1));

g1=1/mf*(F*sin(a+aF)+L1)/v(k-1,1)-G/v(k-1,1)*cos(g(k-1,1));

v1=1/mf*(F*cos(a+aF)-D1)-G*sin(g(k-1,1));

$r2(k) = abs(1.225 * (1 - 0.0065 * (h(k - 1, 1) + d/2 * h1)/288.15)^{(}4.225));$

$L2 = 1/2 * r2(k) * (v(k - 1, 1) + d/2 * v1)^2 * CL * Sf;$

$D2 = 1/2 * r2(k) * (v(k - 1, 1) + d/2 * v1)^2 * CD * Sf;$

h2=(v(k-1,1)+d/2*v1)*sin(g(k-1,1)+d/2*g1); g 2=1/mf*(F*sin(a+aF)+L2)/(v(k-1,1)+d/2*v1)-G/(v(k-1,1)+d/2*v1)*cos(g(k-1,1)+d/2*g1);

v2=1/mf*(F*cos(a+aF)-D2)-G*sin(g(k-1,1)+d/2*g1);

$r3(k) = abs(1.225 * (1 - 0.0065 * (h(k-1,1) + d/2 * h2)/288.15)^{(}4.225));$

$L3 = 1/2 * r3(k) * (v(k-1,1) + d/2 * v2)^2 * CL * Sf;$

$D3 = 1/2 * r3(k) * (v(k-1,1) + d/2 * v2)^2 * CD * Sf;$

h3=(v(k-1,1)+d/2*v2)*sin(g(k-1,1)+d/2*g2);

g3=1/mf*(F*sin(a+aF)+L3)/(v(k-1,1)+d/2*v2)-G/(v(k-1,1)+d/2*v2)*cos(g(k-1,1)+d/2*g2);

v3=1/mf*(F*cos(a+aF)-D3)-G*sin(Go(1,1)+d/2*g2);

$r4(k) = abs(1.225 * (1 - 0.0065 * (h(k-1,1) + d * h3)/288.15)^{(}4.225));$

$L4 = 1/2 * r4(k) * (v(k-1,1) + d * v3)^2 * CL * Sf;$

$D4 = 1/2 * r4(k) * (v(k-1,1) + d * v3)^2 * CD * Sf;$

h4=(v(k-1,1)+d*v3)*sin(g(k-1,1)+d*g3);

g4=1/mf*(F*sin(a+aF)+L4)/(v(k-1,1)+d*v3)-G/(v(k-1,1)+d*v3)*cos(g(k-1,1)+d*g3);

v4=1/mf*(F*cos(a+aF)-D4)-G*sin(g(k-1,1)+d*g3);

h(k,1)=(h(k-1,1)+d/6*(h1+2*h2+2*h3+h4));

g(k,1)=(g(k-1,1)+d/6*(g1+2*g2+2*g3+g4));

v(k,1)=(v(k-1,1)+d/6*(v1+2*v2+2*v3+v4));

end;

W5=h(m8,1)-11000;

****

4 Acknowledgements

The author is grateful to Professor Pedro Paglione of Technological Institute of Aeronautics, ITA, SP-Brazil, for his valuable suggestions and comments which help me a lot to improve some important parts of this text, in particular on the part concerning the model in flight mechanics addressed.

5 Conclusion

In this article we have presented a numerical procedure for solving a class of two point boundary value ODE systems with applications to a flight mechanics model.

The numerical method is based on the fourth order Runge-Kutta model adapted for a two point boundary value ODE system.

In a near future research we intend to address some other linear and non-linear cases.

1. **Conflict of interest declaration: The author declares no conflict of interest concerning this article.**

References

[1] N.X. Vinh, Flight Mechanics of High Performance Aircraft, (Cambridge University Press, New York, 1993)

[2] R.A. Adams and J.F. Fournier, Sobolev Spaces, 2nd edn. (Elsevier, New York, 2003).

[3] F. Botelho, *Topics on Functional Analysis, Calculus of Variations and Duality*, Academic Publications, Sofia, (2011).

[4] F.S. Botelho, Functional Analysis, Calculus of Variations and Numerical Methods for Models in Physics and Engineering, CRC Taylor and Francis, Florida, 2020.

[5] J.C. Strikwerda, Finite Difference Schemes and Partial Differential Equations, SIAM, second edition (Philadelphia, 2004).