MIRK numerical solution of a BVP which rises in the prediction of meteorological parameters.

This work is co-funded by the European Union (European Social Fund)
and Greek national resources under the framework
of the "Archimedes III: Funding of Research Groups in TEI of Athens"
project of the "Education & Lifelong Learning" Operational Programme.

Framework of the talk:

1. The Physical Problem and Information Geometry
2. The Mathematical Problem and its Numerical Solution
3. Numerical solution using Mono Implicit Runge Kutta (MIRK) methods
4. Numerical Tests and Observations

Τεχνολογικό Εκπαιδευτικό Ίδρυμα Αθήνας

Environmental Parameter Forecasting

Need for high quality environmental predictions-simulations due to important **applications**:

Climate change, Renewable energy production,Transportation,Marine pollution,Ship safety

Two are the main approaches today:

1. Use of in site or remote sensing **observations** (e.g. satellite).
2. Use of numerical **predictions models** governing the atmospheric and wave evolution solved numerically.

Weather and wave forecasting models are successful in simulating general environmental conditions on global or intermediate scale but not on local conditions due to

1. the strong dependence on the initial and lateral conditions,
2. the inability to capture sub-scale phenomena,
3. the parametrization of certain atmospheric or wave procedures.

To overcome this drawback someone can

1. **increase the model resolution**,
2. **improve the initial conditions** based on assimilation systems,
3. **filter-optimize the outputs** of the model using statistical models (MOS methods, Neural networks, Kalman filters).

In all previous options a **"cost function"** measuring the **bias ("the distance")** of the model should be minimized.

When the distance/cost-function is measured by means of classical Euclidean Geometry tools is it correctly estimated?

The role of Information Geometry (IG)

- **IG** is a relatively new branch of Mathematics which applies methods and techniques of non-Euclidean geometry to stochastic processes.
- Given two probability distributions or two data sets we can define a notion of **distance** between them.
- In Euclidean/flat geometry functions are based on least square methods.
- **IG** shows that this assumption is false, in general, and provides a theoretical recipe to avoid such simplifications.
- **IG** naturally introduces geometrical entities (Riemannian metrics, distances, curvature and affine connections) for families of probability distributions (manifolds).

The **minimum distance** between two elements f_1 *and* f_2 of a statistical manifold S is defined by the corresponding ***geodesic*** ω which is the minimum length curve that connects them. Such a curve

$$\omega = (\omega_i) \quad : \quad \mathbb{R} \to S \tag{1}$$

satisfies the following system of **2**nd **order differential equations**:

$$\omega_i^{''}(t) + \sum_{j,k=1}^{n} \Gamma_{jk}^{i}(t)\, \omega_j^{'}(t)\, \omega_k^{'}(t) = 0, \quad i = 1,\ 2, \ldots,\ n. \tag{2}$$

under the conditions $\omega(0) = f_1, \;\; \omega(1) = f_2$.

The two parameter Weibull distributions have been proved a good choice for fitting wind and wave data.
These distributions form a 2-dimensional statistical manifold with $\xi=[\alpha,\beta]$, $\Xi = \{[\alpha,\beta];\ \alpha \text{ and } \beta>0\}$ (where α is the shape and β the scale parameter) and

$$p(x) = \frac{\alpha}{\beta}\left(\frac{x}{\beta}\right)^{\alpha-1} \mathrm{e}^{-\left(\frac{x}{\beta}\right)^{\alpha}}, \quad \alpha,\ \beta > 0. \tag{3}$$

Let us have ξ_0=[α_0,β_0], ξ_1=[α_1,β_1] two members of the Weibull statistical manifold, then the previous system becomes:

$$\begin{aligned}
\omega_1^{''}(t) + \frac{6\left(\gamma\alpha_0 - \alpha_0 - \frac{\pi^2}{6}\right)}{\pi^2\beta_0}\left(\omega_1^{'}(t)\right)^2 + \frac{12\left(\gamma^2 - 2\gamma + \frac{\pi^2}{6} + 1\right)}{\pi^2\alpha_0}\omega_1^{'}(t)\,\omega_2^{'}(t) & - \\
\frac{6(1-\gamma)\beta_0\left(\gamma^2 - 2\gamma + \frac{\pi^2}{6} + 1\right)}{\pi^2 a^3}\left(\omega_2^{'}(t)\right)^2 & = 0 \\
\omega_2^{''}(t) - \frac{\alpha_0^3}{\pi^2\beta_0^2}\left(\omega_1^{'}(t)\right)^2 + \frac{12\alpha_0(1-\gamma)}{\pi^2\beta_0}\omega_1^{'}(t)\,\omega_2^{'}(t) & - \\
\frac{6\left(\gamma^2 - 2\gamma + \frac{\pi^2}{6} + 1\right)}{\pi^2\alpha_0}\left(\omega_2^{'}(t)\right)^2 & = 0
\end{aligned}$$

under the conditions $\omega(0) = \begin{bmatrix}\alpha_0 \\ \beta_0\end{bmatrix}, \ \omega(1) = \begin{bmatrix}\alpha_1 \\ \beta_1\end{bmatrix}$

where $\omega(t) = \begin{bmatrix}\omega_1(t) \\ \omega_2(t)\end{bmatrix}$ and is $\gamma =$ the Euler gamma.

So, we need to study the numerical solution of the following system of differential equations

$$\begin{aligned} \omega_1'' + a_{11}(\omega_1')^2 + a_{12}\omega_1'\omega_2' + a_{22}(\omega_2')^2 &= 0 \\ \omega_2'' + b_{11}(\omega_1')^2 + b_{12}\omega_1'\omega_2' + b_{22}(\omega_2')^2 &= 0 \end{aligned} \qquad (4)$$

under the conditions

$$\omega_1(0) = \omega_1^0, \quad \omega_2(0) = \omega_2^0, \quad \omega_1(1) = \omega_1^{N+1}, \quad \omega_2(1) = \omega_2^{N+1}.$$

This is a **second order Boundary Value Problem** of a form

$$\widetilde{\omega}'' = F(\widetilde{\omega}, \widetilde{\omega}') \text{ where } \widetilde{\omega} = \begin{bmatrix} \omega_1 \\ \omega_2 \end{bmatrix} \text{ defined on the interval } [0,1].$$

It is common to transform this second order system in the form of a first order system of the form:

$$
\begin{aligned}
y_1^{'} &= y_3 \\
y_2^{'} &= y_4 \\
y_3^{'} &= a_{11}y_3^2 - a_{12}y_3y_4 - a_{22}y_4^2 \\
y_4^{'} &= b_{11}y_3^2 - b_{12}y_3y_4 - b_{22}y_4^2
\end{aligned} \tag{5}
$$

under the conditions

$$y_1(0) = \omega_1^0, \quad y_2(0) = \omega_2^0, \quad y_1(1) = \omega_1^{N+1}, \quad y_2(1) = \omega_2^{N+1}.$$

where $y_1 = \omega_1$, $y_2 = \omega_2$, $y_3 = \omega_1^{'}$ and $y_4 = \omega_2^{'}$.

So, this problem can be considered as a problem of the more general class

$$y'(t) = f(t, y(t)), \quad g(y(a), y(b)) = 0 \tag{6}$$

where $t \in [a, b]$, $y : \mathbb{R} \to \mathbb{R}^n$, $f : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n$ and $f : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$.

In our case $[a, b] = [0, 1]$, $n = 4$ and f is a quadratic function.

Our problem has separable boundary conditions e.g.

$$g(y(a), y(b)) = (g_0(y(a)), g_1(y(b)))^T = (0, 0)^T$$

where $g_0(y(a))y(a) - y_a$ and $g_1(y(b)) = y(b) - y_b$.

Numerical Solution of BVPs

can be divided into two classes:

- initial value methods e.g. **multiple shooting methods**.
  Mathematica NDsolve uses such methods.
- global methods e.g. **finite difference, collocation and Runge-Kutta schemes**.
  We have studied finite difference methods.
  Collocation methods for our problem can be included in the class of Runge-Kutta schemes.

Runge Kutta method approach

Weiss, Cash, Shampine, Enright, Muir have worked on various classes of Implicit RK methods for the numerical solution of two point BVPs.

Mono-Iplicit RK schemes (MIRK) are the most popular.

Popular fortran package MIRKDC uses A-stable symmetric MIRK schemes and their continuous extensions (CMIRK) which provide C^1 continuous approximate solutions.

Muir, Owren, Burrage (from a classical Runge Kutta point of view) and Cash have worked on order condition theory and the derivation of MIRK and CMIRK methods and classes of such methods have been proposed.

Home Page

Title Page

Page 15 of 37

Go Back

Full Screen

Close

Quit

Runge Kutta method approach

can be described in terms of a **two-level iteration scheme**:

Initialisation: We determine an initial mesh, $\{t_i\}_i^N = 0$, of [a, b] and an initial discrete solution approximation, $Y^{(0)} = [y_0^{(0)}, y_1^{(0)}, \ldots, y_N^{(0)}]$, where $Y_i^{(0)} \approx y(t_i)$.

Home Page

Title Page

Page 16 of 37

Go Back

Full Screen

Close

Quit

Upper-level iteration:

Setup and solve a discrete system,

$$\Phi(Y) = [g_0(y_0), \phi_1, \phi_2, \ldots, \phi_{N-1}, g_1(y_b)]^T = [0, 0, \ldots, 0, 0]^T.$$

where in the residual function $\Phi(Y)$, each vector ϕ_i is of size n and is defined by a Runge-Kutta scheme.

Solve this discrete system using Newton's method.

For each step of the Newton iteration we have to solve the system

$$\begin{pmatrix} \frac{\vartheta g_0}{\vartheta y_0}^{(m)} & 0 & 0 & \dots & 0 & 0 \\ \frac{\vartheta \phi_0}{\vartheta y_0}^{(m)} & \frac{\vartheta \phi_0}{\vartheta y_1}^{(m)} & 0 & \dots & 0 & 0 \\ 0 & \ddots & \ddots & \dots & 0 & 0 \\ 0 & 0 & 0 & \dots & \frac{\vartheta \phi_{N-1}}{\vartheta y_{N-1}}^{(m)} & \frac{\vartheta \phi_{N-1}}{\vartheta y_N}^{(m)} \\ 0 & 0 & 0 & \dots & 0 & \frac{\vartheta g_0}{\vartheta y_N}^{(m)} \end{pmatrix} \begin{pmatrix} \Delta y_0^{(m)} \\ \vdots \\ \Delta y_N^{(m)} \end{pmatrix} = - \begin{pmatrix} g_0(y_0^{(m)}) \\ \phi_1^{(m)} \\ \vdots \\ \phi_{N-1}^{(m)} \\ g_1(y_N^{(m)}) \end{pmatrix}$$

and update the solution vector using $y_i^{(m+1)} = y_i^{(m)} + \Delta y_i^{(m)}$ for $i = 0, 1, \dots, N$.

The matrix above is the Jacobian matrix of $\Phi(Y)$.

Τεχνολογικό Εκπαιδευτικό Ίδρυμα Αθήνας

When a MIRK scheme is used as the underlying discretization the i_{th} component of the residual function takes the form

$$\phi_i = y_{i+1} - y_i - h_i \sum_{j=1}^{s} b_j K_j$$

where the internal stages

$$K_j = f\left(t_i + c_j h_i, (1 - v_j) y_i + v_j y_{i+1} + h_i \sum_{r=1}^{j-1} x_{jr} K_r\right)$$

An advantage of these formulas over the collocation or general implicit RK formulas is that the calculations on each subinterval, which use MIRK formulas in the setup of the Newton system, are explicit and therefore can be implemented more efficiently.

So, the elements of the Jacobian matrix are easily computed

$$\frac{\vartheta \phi_i}{\vartheta y_i} = -I - h_i \sum_{j=1}^{s} b_j \frac{\vartheta K_j}{\vartheta y_i}, \frac{\vartheta \phi_i}{\vartheta y_{i+1}} = I - h_i \sum_{j=1}^{s} b_j \frac{\vartheta K_j}{\vartheta y_{i+1}}$$

where

$$\frac{\vartheta K_j}{\vartheta y_i} = \mathbf{J}_{j,i} \cdot \left((1 - v_j) I + h_i \sum_{r=1}^{j-1} x_{jr} \frac{\vartheta K_r}{\vartheta y_i} \right), \frac{\vartheta K_j}{\vartheta y_{i+1}} = \mathbf{J}_{j,i+1} \cdot \left(v_j + h_i \sum_{r=1}^{j-1} x_{jr} \frac{\vartheta K_r}{\vartheta y_{i+1}} \right).$$

and

$$\mathbf{J}_{j,i} = \left. \frac{\vartheta f}{\vartheta y_i} \right|_{(t_i + c_j h_i, (1-v_j) y_i + v_j y_{i+1} + h_i \sum_{r=1}^{j-1} x_{jr} K_r)}$$

If the Newton iteration fails to converge we consider a new mesh by halving each subinterval of the current mesh, and with the same current solution approximation repeat the Newton iteration.

If the Newton iteration converges we proceed to the lower level iteration.

Lower-level iteration:

The converged Newton iteration yields a discrete solution approximation for the given mesh.

Then we use an associated **CMIRK scheme** to construct a C^1 continuous solution approximation $u(t)$ over the entire problem interval with a relative small extra cost and the same order of accuracy as the underlying discrete solution.

The defect,

$$\delta(t) = u(t) - f(t, u(t))$$

is estimated on a sample of $[a, b]$ and **terminate the algorithm** if its norm is less than a given user-defined tolerance.

It has been suggested that monitoring the defect may be appropriate in situations where difficulties arise in estimating the global error since it arises in the analysis of the mathematical conditioning of the underlying problem where appropriate condition numbers are introduced to quantify the sensitivity of the global error to perturbations of the ODEs.

Τεχνολογικό Εκπαιδευτικό Ίδρυμα Αθήνας

Algorithm Termination

If the termination condition is not met, the relative sizes of the maximum defect estimates associated with each subinterval are examined in the mesh selection algorithm to determine a more appropriate mesh.

The algorithm is terminated unsuccessfully, if the predicted number of mesh points for the new mesh is too large.

When a new mesh is determined, the continuous solution approximation is used to compute an initial discrete solution approximation for the next discrete problem and associated Newton iteration.

Constructing MIRK and CMIRK

The standard form of a MIRK method advances the solution from t_i to $t_{i+1} = t_i + h_i$ using the formula

$$y_{i+1} = y_i + h_i \sum_{j=1}^{s} b_j K_j$$

where

$$K_j = f\left(t_i + c_j h_i, (1 - v_j) y_i + y_j y_{i+1} + h_i \sum_{r=1}^{j-1} x_{jr} K_r\right)$$

and $c_j = v_j + \sum_{r=1}^{j-1} x_{jr}$.

They are usually represented by a **modified Butcher tableau**

$$\begin{array}{c|c|ccccc} c_1 & v_1 & 0 & 0 & \dots & 0 & 0 \\ c_2 & v_2 & x_{2,1} & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{s-1} & v_{s-1} & x_{s-1,1} & x_{s-1,2} & \dots & 0 & 0 \\ c_s & v_s & x_{s,1} & x_{s,2} & \dots & 0 & 0 \\ \hline & & b_1 & b_2 & \dots & b_{s-1} & b_s \end{array}$$

or in a matrix form

$$\begin{array}{c|c|c} c & v & X \\ \hline & & b^T \end{array}$$

X strictly lower triangular matrix and $c = Xe + v$

e is a vector of $1's$ of length s.

A MIRK method is equivalent to the general IRK method

$$y_{i+1} = y_i + h_i \sum_{j=1}^{s} b_j K_j$$

where

$$K_j = f\left(t_i + c_j h_i, y_i + h_i \sum_{r=1}^{j-1} a_{jr} K_r\right)$$

and $c_j = \sum_{r=1}^{j-1} a_{jr}$ with Butcher representation tableau

$$\begin{array}{c|c} c & A \\ \hline & b^T \end{array}$$

with $A = X + vb^T$.

So, a MIRK method has a **full implicit RK Butcher tableau** :

$$\begin{array}{c|ccccc} c_1 & x_{1,1}+v_1b_1 & x_{1,2}+v_1b_2 & \dots & x_{1,s-1}+v_1b_{s-1} & x_{1,s}+v_1b_s \\ c_2 & x_{2,1}+v_2b_1 & x_{2,2}+v_2b_2 & \dots & x_{2,s-1}+v_2b_{s-1} & x_{2,s}+v_2b_s \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{s-1} & x_{s-1,1}+v_{s-1}b_1 & x_{s-1,2}+v_{s-1}b_2 & \dots & x_{s-1,s-1}+v_{s-1}b_{s-1} & x_{s-1,s}+v_{s-1}b_s \\ c_s & x_{s,1}+v_sb_1 & x_{s,2}+v_sb_2 & \dots & x_{s,s-1}+v_sb_{s-1} & x_{s,s}+v_sb_s \\ \hline & b_1 & b_2 & \dots & b_{s-1} & b_s \end{array}$$

The **stability function** of an MIRK method can be expressed in the form

$$R(z) = \frac{P(z, e-y)}{P(z-v)} \text{ where } P(z,w) = 1 + zb^T(I - zX)^{-1}w$$

$$w \in \mathbb{R}^n.$$

For IRK methods we consider the following **order conditions**

$$B(p): \quad b^T c^{k-1} = \frac{1}{k}, \quad k = 1, 2, \ldots, p$$

and the **stage order conditions**

$$C(p): \quad A^T c^{k-1} = \frac{c^k}{k}, \quad k = 1, 2, \ldots, p.$$

For MIRK methods we consider the **same order conditions** and the **equivalent stage order conditions**

$$C(p): \quad v + kxc^{k-1} = c^k, \quad k = 1, 2, \ldots, p.$$

An IRK method (and consequently) a MIRK method has order at least $p+1$ if $B(p+1)$ and $C(p)$ are satisfied

because then for its local truncation error holds

$$|y(t_{i-1}+h_i)-y_i| = O(h^{p+1})$$

Similar conditions hold for CMIRK methods.

It can be **important for a IRK scheme to have as high a stage order** as possible on order to avoid an **order reduction phenomenon** when solve a system of stiff differential equations.

The **maximum stage** order of a pth order MIRK scheme is min(p, 3).

Considering the stage order conditions up to 3 **has been proved to be restrictive to use the order condition theory for quadratic problems (see Iserles) and construct MIRK and CMIRK methods with better characteristics specially suited for our problem.**

24 Test Problems

We choose data from Levantive are (eastern Mediterranean Sea).

For every month of a year we have modeled **wind speed and wave height data** either includes in the simulation the impact of **sea currents** either not.

Second source of data is the available corresponding **satellite data**.

The **data are fitted by a 2-parameter Weibull distribution** to get their Weibull parameters.

Data for the 24 Test Problems based on Weibull distribution

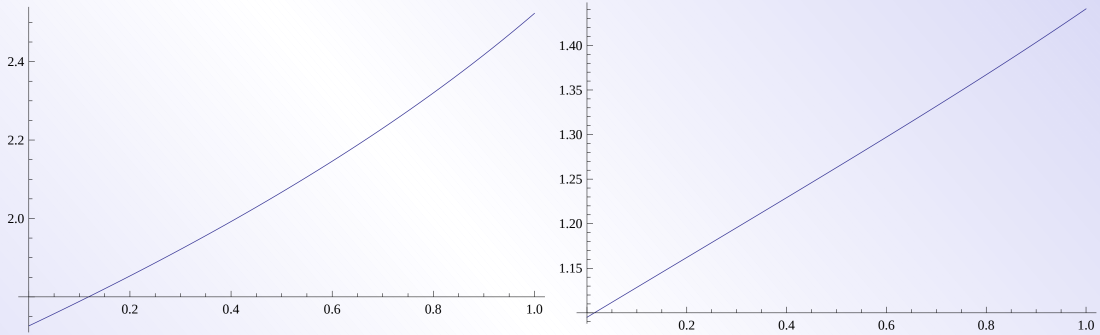
	model data no current		model data with current		satelite data	
Weibull Parameters	shape α_0	scale β_0	shape α_0	scale β_0	shape α_1	scale β_1
Jan	1.600	1.010	1.726	1.095	2.523	1.441
Feb	1.500	1.400	1.571	1.464	2.450	1.762
Mar	1.462	1.132	1.578	1.225	2.560	1.509
Apr	1.564	0.695	1.719	0.754	2.140	1.012
May	1.533	0.608	1.608	0.661	1.576	0.780
Jun	2.333	0.633	2.542	0.680	3.759	0.759
Jul	2.557	0.837	2.688	0.876	3.515	0.960
Aug	3.099	0.716	3.341	0.759	4.938	0.889
Sep	2.418	0.754	2.580	0.800	3.491	0.968
Oct	1.629	0.551	1.850	0.609	2.204	0.665
Nov	1.446	0.892	1.499	0.919	1.911	1.224
Dec	1.435	1.216	1.512	1.283	2.208	1.442

When we consider the**minimum length curve** which connects the each modeled and its corresponding satellite data we **conclude in** 24 **BVP problems**.

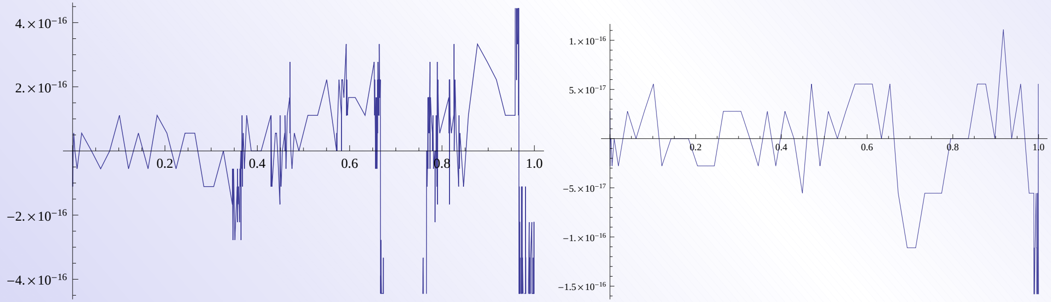
Reference Solutions using Mathematica

- Use NDSolve of Mathematica to solve the 24 test problems.
- Shooting method with proper accuracy options (Working Precision, Accuracy Goal, Accuracy Goal) to get an considerably accurate solution.
- Produce a ”continuous” interpolating form of the solution.
- The defect for an abscissae on $[0, 1]$ of width 10^{-5} has been recorded.
- So, produce high accurate reference solutions for the comparison to the other numerical methods which attain a significantly lower precision.

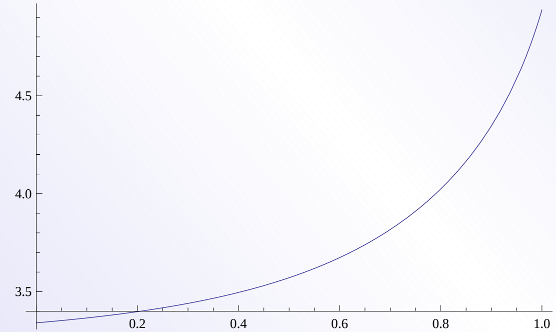
the reference solution of problem Jun with current

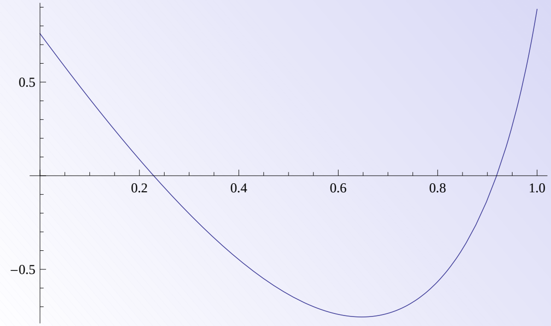


the error 1.11×10^{-15}

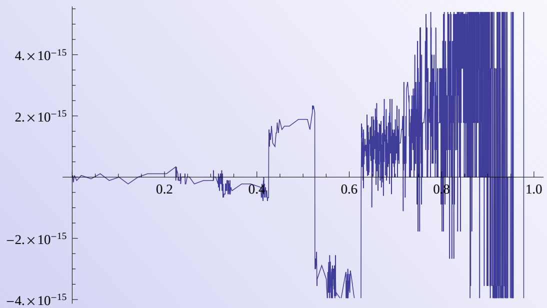


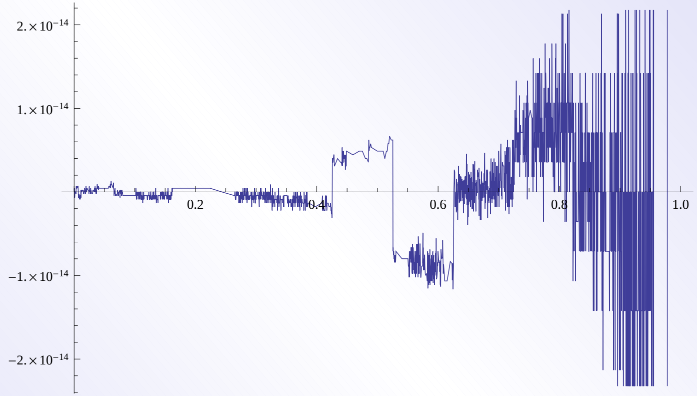
the reference solution of Aug with current (stiffness)





the error 6.25×10^{-13}





Numerical tests

- For the 24 problems we produce a reference solution.
- For an initial guess we use a perturbation with random numbers of the initial conditions on $t = 0$.
- We solve numerically the 24 test problems, using the 5 stage, 6th order, stage order 3 MIRK formula and its 5 stage, 6th order, stage order 3 continuous extension of Muir and Shampine, for tolerances $10^{-6}, 10^{-7}, \ldots, 10^{-11}$. We use two error measures at an abscissae of 101 grid points.
  - The first one is $\|\widehat{F}(\widehat{\omega}_{sol})\|_\infty$ the maximum absolute value that the numerical solution fails to satisfy the nonlinear problem e.g. the defect.
  - The second one is the $\|\widehat{\omega}_{so} - \widehat{\omega}_{ref}\|_\infty$ maximum absolute value of the difference of the numerical solution and the reference solution.

Τεχνολογικό Εκπαιδευτικό Ίδρυμα Αθήνας

Defect for the 24 problems for various tolerances

	$\Vert\widehat{F}(\widehat{\omega}_{sol})\Vert_\infty$		
TOL	max	min	average
10^{-6}	0.162021×10^{-07}	0.362470×10^{-08}	0.362470×10^{-08}
10^{-7}	0.837087×10^{-09}	0.173195×10^{-13}	0.141507×10^{-09}
10^{-8}	0.101302×10^{-09}	0421885×10^{-14}	0.865360×10^{-11}
10^{-9}	0.295586×10^{-11}	0.244249×10^{-14}	0.486983×10^{-12}
10^{-10}	0.204636×10^{-11}	0.144329×10^{-14}	0.168809×10^{-12}
10^{-11}	0.113687×10^{-11}	0.430211×10^{-15}	0.906469×10^{-13}

Home Page

Title Page

Page 37 of 37

Go Back

Full Screen

Close

Quit

Reference error for the 24 problems for various tolerances

TOL	$\|\widehat{\omega}_{so} - \widehat{\omega}_{ref}\|_\infty$ max	min	average
10^{-6}	0.112400×10^{-07}	0.342813×10^{-09}	0.373460×10^{-08}
10^{-7}	0.362932×10^{-08}	0.102763×10^{-09}	0.635162×10^{-09}
10^{-8}	0.303723×10^{-09}	0.129772×10^{-10}	0.730099×10^{-10}
10^{-9}	0.156603×10^{-10}	0.976108×10^{-12}	0.419232×10^{-11}
10^{-10}	0.219255×10^{-11}	0.128120×10^{-12}	0.578689×10^{-12}
10^{-11}	0.501599×10^{-12}	0.159872×10^{-13}	0.824618×10^{-13}