



Universidad
del Atlántico

CÓDIGO: FOR-DO-109

VERSIÓN: 0

FECHA: 03/06/2020

**AUTORIZACIÓN DE LOS AUTORES PARA LA CONSULTA, LA
REPRODUCCIÓN PARCIAL O TOTAL, Y PUBLICACIÓN ELECTRÓNICA DEL
TEXTO COMPLETO**

Puerto Colombia, 23 de Junio de 2021

Señores

DEPARTAMENTO DE BIBLIOTECAS

Universidad del Atlántico

Asunto: Autorización Trabajo de Grado

Cordial saludo,

Yo, **MAYRA ALEJANDRA VEGA NIÑO**, identificado(a) con **C.C. No. 1.143.466.357** de **BARRANQUILLA**, autor(a) del trabajo de grado titulado **ESTUDIO COMPARATIVO DE SOLUCIONES NUMÉRICAS DE SISTEMAS DE ECUACIONES DIFERENCIALES ORDINARIAS TIPO STIFF** presentado y aprobado en el año **2021** como requisito para optar al título Profesional de **MATEMÁTICO**; autorizo al Departamento de Bibliotecas de la Universidad del Atlántico para que, con fines académicos, la producción académica, literaria, intelectual de la Universidad del Atlántico sea divulgada a nivel nacional e internacional a través de la visibilidad de su contenido de la siguiente manera:

- Los usuarios del Departamento de Bibliotecas de la Universidad del Atlántico pueden consultar el contenido de este trabajo de grado en la página Web institucional, en el Repositorio Digital y en las redes de información del país y del exterior, con las cuales tenga convenio la Universidad del Atlántico.
- Permitir consulta, reproducción y citación a los usuarios interesados en el contenido de este trabajo, para todos los usos que tengan finalidad académica, ya sea en formato CD-ROM o digital desde Internet, Intranet, etc., y en general para cualquier formato conocido o por conocer.

Esto de conformidad con lo establecido en el artículo 30 de la Ley 23 de 1982 y el artículo 11 de la Decisión Andina 351 de 1993, "Los derechos morales sobre el trabajo son propiedad de los autores", los cuales son irrenunciables, imprescriptibles, inembargables e inalienables.

Atentamente,

Firma *Mayra Vega*.

MAYRA ALEJANDRA VEGA NIÑO

C.C. No. 1.143.466.357 de BARRANQUILLA

DECLARACIÓN DE AUSENCIA DE PLAGIO EN TRABAJO ACADÉMICO PARA GRADO

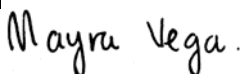
Este documento debe ser diligenciado de manera clara y completa, sin tachaduras o enmendaduras y las firmas consignadas deben corresponder al (los) autor (es) identificado en el mismo.

Puerto Colombia, **23 de Junio de 2021**

Una vez obtenido el visto bueno del director del trabajo y los evaluadores, presento al **Departamento de Bibliotecas** el resultado académico de mi formación profesional o posgradual. Asimismo, declaro y entiendo lo siguiente:

- El trabajo académico es original y se realizó sin violar o usurpar derechos de autor de terceros, en consecuencia, la obra es de mi exclusiva autoría y detento la titularidad sobre la misma.
- Asumo total responsabilidad por el contenido del trabajo académico.
- Eximo a la Universidad del Atlántico, quien actúa como un tercero de buena fe, contra cualquier daño o perjuicio originado en la reclamación de los derechos de este documento, por parte de terceros.
- Las fuentes citadas han sido debidamente referenciadas en el mismo.
- El (los) autor (es) declara (n) que conoce (n) lo consignado en el trabajo académico debido a que contribuyeron en su elaboración y aprobaron esta versión adjunta.

Título del trabajo académico:	ESTUDIO COMPARATIVO DE SOLUCIONES NUMÉRICAS DE SISTEMAS DE ECUACIONES DIFERENCIALES ORDINARIAS TIPO STIFF
Programa académico:	MATEMÁTICAS

Firma de Autor 1:							
Nombres y Apellidos:	MAYRA ALEJANDRA VEGA NIÑO						
Documento de Identificación:	CC	X	CE	PA	Número:	1.143.466.357	
Nacionalidad:	COLOMBIANA			Lugar de residencia:	BARRANQUILLA		
Dirección de residencia:	CL 35B # 17 - 106						
Teléfono:	3019181			Celular:	3023147018		



FORMULARIO DESCRIPTIVO DEL TRABAJO DE GRADO

TÍTULO COMPLETO DEL TRABAJO DE GRADO	ESTUDIO COMPARATIVO DE SOLUCIONES NUMÉRICAS DE SISTEMAS DE ECUACIONES DIFERENCIALES ORDINARIAS TIPO STIFF
AUTOR(A) (ES)	MAYRA ALEJANDRA VEGA NIÑO
DIRECTOR (A)	JEINNY MARIA PERALTA POLO
CO-DIRECTOR (A)	MIGUEL ANTONIO CARO CANDEZANO
JURADOS	JORGE ISAAC ROBINSON EVILLA JORGE ELIECER OSPINO PORTILLO
TRABAJO DE GRADO PARA OPTAR AL TÍTULO DE	MATEMÁTICO
PROGRAMA	MATEMÁTICAS
PREGRADO / POSTGRADO	PREGRADO
FACULTAD	CIENCIAS BÁSICAS
SEDE INSTITUCIONAL	SEDE NORTE – PUERTO COLOMBIA
AÑO DE PRESENTACIÓN DEL TRABAJO DE GRADO	2021
NÚMERO DE PÁGINAS	44
TIPO DE ILUSTRACIONES	TABLAS Y GRÁFICOS
MATERIAL ANEXO (VÍDEO, AUDIO, MULTIMEDIA O PRODUCCIÓN ELECTRÓNICA)	NO APLICA
PREMIO O RECONOCIMIENTO	NO APLICA



**ESTUDIO COMPARATIVO DE SOLUCIONES NUMÉRICAS DE
SISTEMAS DE ECUACIONES DIFERENCIALES ORDINARIAS
TIPO *STIFF***

MAYRA ALEJANDRA VEGA NIÑO

TRABAJO DE GRADO PARA OPTAR AL TÍTULO DE MATEMÁTICO

**PROGRAMA DE MATEMÁTICAS
FACULTAD DE CIENCIAS BÁSICAS
UNIVERSIDAD DEL ATLÁNTICO
PUERTO COLOMBIA**

2021



**ESTUDIO COMPARATIVO DE SOLUCIONES NUMÉRICAS DE
SISTEMAS DE ECUACIONES DIFERENCIALES ORDINARIAS
TIPO *STIFF***

MAYRA ALEJANDRA VEGA NIÑO

TRABAJO DE GRADO PARA OPTAR AL TÍTULO DE MATEMÁTICO

JEINNY MARIA PERALTA POLO

PhD. CIENCIA DE LA COMPUTACIÓN Y MATEMÁTICA COMPUTACIONAL

MIGUEL ANTONIO CARO CANDEZANO

PhD. CIENCIA DE LA COMPUTACIÓN Y MATEMÁTICA COMPUTACIONAL

**PROGRAMA DE MATEMÁTICAS
FACULTAD DE CIENCIAS BÁSICAS
UNIVERSIDAD DEL ATLÁNTICO
PUERTO COLOMBIA**

2021

NOTA DE ACEPTACIÓN

DIRECTOR(A)(S)

JURADO(A)(S)

DEDICATORIA

*A mis padres,
Nevis y Alexander.*

AGRADECIMIENTOS

Son varias las personas a las que deseo agradecer por su acompañamiento en todo el desarrollo de este trabajo de grado y también en mi carrera como matemática.

Inicialmente, deseo agradecer a mis dos asesores, la profesora Jeinny Peralta y el profesor Miguel Caro. Muchas gracias por haberme confiado la realización de este trabajo, por su paciencia en todo el proceso, por su ejemplo de estudio y dedicación, y muchos otros aspectos que fueron de beneficio para mí. Estaré siempre agradecida con ustedes por haber sido mi guía.

En segundo lugar, quiero agradecer a mi familia y en especial a mis padres. Desde el inicio de la carrera han sido una fuente de motivación para mí, me han escuchado y apoyado en todo este proceso. No tengo palabras para describir lo afortunada que soy por tener unos padres como ustedes, este logro también es suyo. Le agradezco a Dios por su vida y le pido que les permita seguir presente en mis próximos logros.

De igual forma, me gustaría agradecerle a mi novio, Daniel, por sus palabras de apoyo y motivación, por confiar tanto en mí y en mis capacidades. Muchas gracias por tu comprensión y ayuda en todo momento. Asimismo, deseo agradecer a mis amigos y futuros colegas por su ayuda desinteresada y su buena voluntad.

Finalmente, agradezco a la universidad y a todos los profesores del programa de matemáticas por sus enseñanzas en el ámbito académico y en la vida profesional, muchas gracias por ayudarme a seguir construyendo mi proyecto de vida.

ESTUDIO COMPARATIVO DE SOLUCIONES NUMÉRICAS DE SISTEMAS DE ECUACIONES DIFERENCIALES ORDINARIAS TIPO *STIFF*

Resumen

Los sistemas de ecuaciones diferenciales han contribuido al modelamiento de problemas físicos, químicos, epidemiológicos, entre otros, gracias a la relación que exponen entre varias funciones y sus derivadas. Encontrar una solución analítica de este tipo de ecuaciones ha sido un reto para los matemáticos, debido a la dificultad que presenta algunas veces. Sin embargo, en muchas ocasiones sólo es necesario conocer el comportamiento de la solución y no su expresión explícita; es aquí donde los métodos numéricos juegan un papel importante al permitir encontrar la solución numérica de este tipo de problemas. En este trabajo, resolvemos numéricamente sistemas de ecuaciones diferenciales ordinarias tipo *Stiff*, para las cuales, la mayoría de métodos numéricos tradicionales, que se dividen en multipasos y multietapas presentan dificultades. Este tipo de ecuaciones se caracteriza principalmente por tener componentes con tasas de cambio muy variadas entre sí, lo que dificulta la obtención de su solución numérica. Los General Linear Methods (GLM) surgen con la intención de solucionar de manera adecuada estos sistemas de ecuaciones. En este trabajo, se hace uso de un tipo de GLM llamado Método de integración multietapa diagonalmente implícito (DIMSIM) para solucionar numéricamente algunos sistemas de ecuaciones diferenciales tipo *Stiff* que tiene origen en reacciones químicas (reacciones de OREGO, Robertson y HIRES) y procesos físicos (problema de Van der Pol). Posteriormente, con los resultados obtenidos se realiza una comparación numérica con la solución hallada por los *solvers ode45* y *ode15s* del software MatLab.

Palabras clave: General Linear Methods, sistemas de ecuaciones diferenciales ordinarias, problemas de valor inicial de tipo *Stiff*.

Abstract

Ordinary differential equation systems have contributed to the modeling of physical, chemical and epidemiological problems, among others, thanks to the relation they expose between the functions and their derivatives. Finding an analytical solution to this type of equations has been a challenge for mathematicians, due to the difficulty it sometimes presents. However, in many occasions it is only necessary to know the behavior of the solution and not its explicit expression; it is here where numerical methods play an important role by allowing to find the numerical solution of this type of problems. In this work, we solve numerically stiff ODE systems, for which most traditional numerical methods, which are divided into multistep and multistage, present difficulties. This type of equations is mainly characterized by having components with very different rates of change among them, which makes it difficult to obtain their numerical solution. The General Linear Methods (GLM) arise with the intention of solving these systems of equations in an adequate way. In this work, a type of GLM called Diagonally Implicit Multistage Integration Method (DIMSIM) is used to solve numerically some stiff ODE systems that originate in chemical reactions (OREGO, Robertson and HIRES reactions) and physical processes (Van der Pol problem). Subsequently, with the results obtained, a numerical comparison is made with the solution found by the *ode45* and *ode15s*

solvers of the MatLab software.

Keywords: General Linear Methods, ordinary differential equation systems, stiff ordinary differential equation systems.

Índice

1. Introducción	4
1.1. Ecuaciones Diferenciales tipo <i>Stiff</i>	4
1.2. Consistencia, Estabilidad y Convergencia.	5
1.2.1. Consistencia	5
1.2.2. Estabilidad	6
1.2.3. Convergencia	6
2. General Linear Methods (GLM)	7
2.1. Origen y representación de los General Linear Methods	7
2.2. Consistencia, estabilidad y convergencia	8
3. Método de integración multietapa diagonalmente implícito (DIMSIM)	11
3.1. Representación de DIMSIM	11
3.2. Matriz de estabilidad	13
3.3. Condiciones de orden	13
3.4. Estabilidad Inherente de Runge-Kutta	16
3.5. Escogencia de los valores para λ	18
3.6. Ejemplos de DIMSIM	20
4. Algoritmo y Código Computacional	22
4.1. Algoritmo y Código Computacional	22
5. Resultados numéricos	28
5.1. Oscilador de Van der Pol	28
5.2. Problema de OREGO	30
5.3. Problema de Robertson	33
5.4. Problema de HIRES	37
6. Conclusión	40
Bibliografía	41

Índice de figuras

1.	Diagrama de flujo del código que implementa un DIMSIM de orden $p = q = 4$ y $s = r = 5$	23
2.	Solución Numérica Oscilador de Van Der Pol usando DIMSIM de orden $p = q = 4$ y $s = r = 5$	29
3.	Solución del problema del Oscilador de Van der Pol utilizando <code>ode45</code>	30
4.	Solución del problema del Oscilador de Van der Pol utilizando <code>ode15s</code>	30
5.	Solución Numérica Oscilador del problema de OREGO usando DIMSIM de orden $p = q = 4$ y $s = r = 5$	31
6.	Solución del problema de OREGO utilizando <code>ode45</code>	32
7.	Solución del problema de OREGO utilizando <code>ode15s</code>	33
8.	Solución Numérica problema de Robertson usando DIMSIM de orden $p = q = 4$ y $s = r = 5$	34
9.	Solución del problema de Robertson utilizando <code>ode45</code>	35
10.	Ampliación a componente y_2 de la solución del problema de Robertson hallada con <code>ode45</code>	36
11.	Solución del problema de Robertson utilizando <code>ode15s</code>	36
12.	Solución Numérica problema de HIRES usando DIMSIM de orden $p = q = 4$ y $s = r = 5$	38
13.	Solución del problema de HIRES utilizando <code>ode45</code>	39
14.	Solución del problema de HIRES utilizando <code>ode15s</code>	39

Índice de tablas

1.	Tipos de DIMSIM.	12
2.	Posibles valores de λ dependiendo del orden p del GLM.	20
3.	Solución numérica y solución de referencia del Oscilador de Van der Pol en $t = 2$	29
4.	Resumen resultados de varios métodos al solucionar el problema del Oscilador de Van der Pol.	29
5.	Solución numérica y solución de referencia del problema de OREGO en $t = 360$	31
6.	Resumen resultados de varios métodos al solucionar el problema de OREGO.	32
7.	Solución numérica y solución de referencia del problema de Robertson en $t = 40$	34
8.	Resumen resultados de varios métodos al solucionar el problema de Robertson.	35
9.	Solución numérica y solución de referencia del problema de HIRES en $t = 321.8122$	37
10.	Resumen implementación de DIMSIM de orden de orden $p = q = 4$ y $s = r = 5$ al problema de HIRES.	38

1. Introducción

1.1. Ecuaciones Diferenciales tipo *Stiff*

Las ecuaciones diferenciales tipo *Stiff* fueron descritas por primera vez en 1952 por Curtiss y Hirschfelder (Curtiss y Hirschfelder, 1952). Al momento de resolver sistemas de ecuaciones que tenían origen en reacciones químicas y teoría de circuitos eléctricos, se dieron cuenta que el método explícito que estaban utilizando presentaba problemas para resolver el sistema. Esto lo notaron estudiando el comportamiento de cada componente, los cuales exhibieron una tasa de cambio bastante rápida en periodos cortos de tiempo, en comparación con el resto de elementos del problema, que tenían tasas de cambio mucho más lentas. En general, lo anterior caracteriza a un sistema de ecuaciones diferenciales tipo *Stiff*. Para comprender un poco mejor la situación, consideremos el siguiente sistema de ecuaciones tomado de Lambert (1991),

$$\begin{cases} y_1' = -2y_1 + y_2, \\ y_2' = 998y_1 - 999y_2. \end{cases} \quad (1.1)$$

El sistema tiene como solución general

$$\mathbf{y}(x) = c_1 \begin{bmatrix} 1 \\ 1 \end{bmatrix} e^{-x} + c_2 \begin{bmatrix} -1 \\ 998 \end{bmatrix} e^{-1000x}. \quad (1.2)$$

Notamos que, para valores relativamente grandes de x , el término e^{-1000x} se vuelve insignificante en comparación con e^{-x} , lo cual lo hace un sistema *Stiff*. De manera numérica esto implica que para poder observar el cambio del componente e^{-1000x} , se necesitaría un paso h bastante pequeño, lo que en sistemas con un número elevado de ecuaciones ralentizaría el proceso de encontrar la solución.

En general, consideremos el siguiente problema de valor inicial

$$y' = Ay + \varphi(t), \quad t \geq t_0, \quad (1.3)$$

$$y(t_0) = y_0 \in \mathbb{R}^m, \quad (1.4)$$

donde $A \in \mathbb{R}^{m \times m}$ es una matriz constante y $\varphi(t)$ es una función que depende de t (Jackiewicz, 2009). Asumamos que A tiene m autovalores $\lambda_1, \dots, \lambda_m$ todos distintos, con autovectores v_1, \dots, v_m . La solución general de (1.3) viene dada por

$$y(t) = \sum_{i=1}^m c_i v_i e^{\lambda_i(t-t_0)} + \psi(t), \quad t \geq t_0, \quad (1.5)$$

donde c_i son constantes que dependen de la condición inicial y $\psi(t)$ es la solución particular. Si $|\operatorname{Re}(\lambda_i)| \gg 0$ y λ_i es negativo para algún i , el término $c_i v_i e^{\lambda_i(t-t_0)}$ de (1.5) tenderá a cero

mucho más rápido que los demás componentes de la solución, lo cual clasifica al sistema como *Stiff*.

Bajo las hipótesis anteriores podemos definir el *radio de Stiff* como

$$R = \frac{\max |Re(\lambda_j)|}{\min |Re(\lambda_j)|}, \quad j = 1, \dots, m.$$

Si $R \gg 1$, se dice que el problema es *Stiff*. En el caso de que el sistema de ecuaciones sea no lineal, se consideran los autovalores de la matriz del Jacobiano J del problema (Huang, 2005).

Observación 1.1. 1. A pesar de que el *radio de Stiff* permite identificar si un problema es *Stiff* o no, este puede llegar a presentar dificultades en el caso de que algún autovalor sea igual a cero (Huang, 2005).

2. Usualmente, para sistemas lineales sólo se consideran los autovalores de la matriz A para comprobar si es *Stiff*. En el caso de los sistemas no lineales, es frecuente que se realice un análisis del comportamiento de la estabilidad de los componentes (Garfinkel *et al.*, 1977).

3. La propiedad de ser *Stiff* es propia del sistema y no de la solución del mismo (Lambert, 1991).

Debido a que no se cuenta con un única definición o caracterización para sistemas de este tipo, usualmente se dice que un sistema es *Stiff* si cumple la siguiente definición.

Definición 1.2. (Lambert, 1991) Si un método numérico es forzado a usar, en un intervalo de integración, un paso h excesivamente pequeño relativo al intervalo de tiempo de la solución, para obtener una solución suave de la solución real, se dice que el problema es *Stiff* en ese intervalo.

1.2. Consistencia, Estabilidad y Convergencia.

En esta sección se presentarán las definiciones de consistencia y estabilidad que servirán como base para el estudio teórico de los General Linear Methods, además, se enunciará el teorema fundamental del análisis numérico.

1.2.1. Consistencia

Definición 1.3. (Butcher, 2016) Sea el problema de valor inicial $y'(x) = 0$ con solución $y(x) = 1$. Se dice que un método numérico es preconsistente si luego de hallar la solución en n iteraciones consecutivas, es capaz de encontrar la solución exacta en la iteración $n + 1$.

Definición 1.4. (Venkateshan y Swaminathan, 2014) Un método numérico se dice que es consistente si las aproximaciones numéricas tienden al valor exacto de la solución cuando el paso h tiende a cero.

1.2.2. Estabilidad

Definición 1.5. (Venkateshan y Swaminathan, 2014) Un método numérico se dice que es estable si el error no crece en cada iteración.

Definición 1.6. (Dahlquist G., 2003) La región de estabilidad de un método numérico para un problema de valor inicial es el conjunto de valores (complejos) de qh para los cuales todas las soluciones del problema $y' = qy$ permanecen acotadas cuando $n \rightarrow \infty$.

Para cada método numérico existe una función (compleja) $f(z)$ que, bajo ciertas condiciones, define la región de estabilidad del método (Butcher, 2016).

Ejemplo 1.7. Para un método explícito Runge-Kutta de orden p la función de estabilidad está dada por el siguiente Teorema.

Teorema 1.8. (Huang, 2005) La función de estabilidad de un método explícito Runge-Kutta de orden p está dada por

$$R(z) = 1 + z + \frac{z^2}{2!} + \cdots + \frac{z^p}{p!} + O(z^{p+1}). \quad (1.6)$$

Definición 1.9. (Huang, 2005) La región de estabilidad de un método explícito Runge-Kutta de orden p está dada por

$$S = \{z \in \mathbb{C} : |R(z)| \leq 1\}. \quad (1.7)$$

La siguiente definición se considera esencial en el estudio de métodos numéricos que resuelven ecuaciones diferenciales tipo *Stiff* (Butcher, 2016).

Definición 1.10. (Butcher, 2016) Un método numérico se dice que es A -estable si su región de estabilidad contiene la parte negativa del plano complejo ($Re(z) \leq 0$).

Definición 1.11. (Huang, 2005) Un método se dice que es L -estable si es A -estable y si su función de estabilidad $f(z)$ satisface que

$$\lim_{|z| \rightarrow \infty} |f(z)| = 0. \quad (1.8)$$

1.2.3. Convergencia

Definición 1.12. (Butcher, 2016) Un método numérico es convergente si es capaz de aproximar la solución de una ecuación diferencial aún tomando pasos suficientemente pequeños y sin importar la precisión requerida.

Teorema 1.13 (Teorema fundamental del análisis numérico). (*Strang y Aarikka, 1986*)
Un método numérico es convergente si, y sólo si, es estable y consistente.

2. General Linear Methods (GLM)

2.1. Origen y representación de los General Linear Methods

Los General Linear Methods (GLM) nacen como respuesta a las dificultades computacionales y numéricas presentadas por los distintos tipos de métodos (multipasos y multietapas), siendo la idea principal de estos, poder generar aproximaciones suficientemente precisas con bajo costo computacional (Butcher, 1966). Este tipo de métodos fueron introducidos por Butcher en 1966, los cuales integraban distintas características de los métodos ya mencionados. Este tipo de método es caracterizado por una matriz de tamaño $(s+r) \times (s+r)$ que contiene cuatro matrices, A, B, U y V , de la siguiente forma:

$$\left[\begin{array}{c|c} A_{s \times s} & U_{s \times r} \\ \hline B_{r \times s} & V_{r \times r} \end{array} \right].$$

Para un sistema N -dimensional de ecuaciones diferenciales ordinarias, la matriz anterior es equivalente a

$$\begin{aligned} Y_i^{[n]} &= \sum_{j=1}^s a_{ij} h f(Y_j^{[n]}) + \sum_{j=1}^r u_{ij} y_j^{[n-1]}, \quad i = 1, 2, \dots, s. \\ y_i^{[n]} &= \sum_{j=1}^s b_{ij} h f(Y_j^{[n]}) + \sum_{j=1}^r v_{ij} y_j^{[n-1]}, \quad i = 1, 2, \dots, r. \end{aligned} \tag{2.1}$$

donde

- $Y_i^{[n]}$: representa las etapas internas como en un método Runge-Kutta.
- $f(Y_i^{[n]})$: representa la derivada de cada etapa.
- $y_i^{[n-1]}$: es la aproximación entrante en el paso n .
- $y_i^{[n]}$: es la aproximación de salida en el paso n .

Usualmente las ecuaciones en (2.1) se suelen escribir en forma matricial de la siguiente manera

$$\begin{bmatrix} Y_1^{[n]} \\ \vdots \\ Y_s^{[n]} \\ y_1^{[n]} \\ \vdots \\ y_r^{[n]} \end{bmatrix} = \left[\begin{array}{c|c} A \otimes I & U \otimes I \\ \hline B \otimes I & V \otimes I \end{array} \right] \begin{bmatrix} hF_1 \\ \vdots \\ hF_s \\ y_1^{[n-1]} \\ \vdots \\ y_r^{[n-1]} \end{bmatrix}. \tag{2.2}$$

Aquí, I representa la matriz identidad de tamaño $N \times N$ y el producto tensorial de dos matrices $A \in \mathbb{R}^{m_1 \times n_1}$ y $B \in \mathbb{R}^{m_2 \times n_2}$ definido por

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1,n_1}B \\ a_{21}B & a_{22}B & \cdots & a_{2,n_1}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{m_1,1}B & a_{m_1,2}B & \cdots & a_{m_1,n_1}B \end{bmatrix} \in \mathbb{R}^{m_1 m_2 \times n_1 n_2}.$$

Las ecuaciones en (2.2) se pueden escribir de manera reducida como

$$\begin{aligned} Y^{[n]} &= (A \otimes I)hF + (U \otimes I)y^{[n-1]}, \\ y^{[n]} &= (B \otimes I)hF + (V \otimes I)y^{[n-1]}, \end{aligned} \tag{2.3}$$

donde $Y^{[n]} = [Y_1^{[n]}, Y_2^{[n]}, \dots, Y_s^{[n]}]^T$, $F = [F_1, F_2, \dots, F_s]^T$, $F_i = f(Y_i^{[n]})$ para $i = 1, \dots, s$, $y^{[n-1]} = [y_1^{[n-1]}, y_2^{[n-1]}, \dots, y_r^{[n-1]}]^T$ y $y^{[n]} = [y_1^{[n]}, y_2^{[n]}, \dots, y_r^{[n]}]^T$.

Los enteros p, q, r, s son tales que

- s : representa el número de etapas del método.
- r : representa la cantidad de valores utilizados en cada paso n del método.
- p : representa el orden del método, lo cual indica que si $y(\bar{x}_n)$ es la solución exacta, se tiene que

$$y(\bar{x}_n) = y(x_n) + O(h^{p+1}).$$

- q : representa el orden de las etapas, lo cual indica que el valor Y_i aproxima la solución en $x_{n-1} + c_i h$ con orden q , esto implica que $Y_i = y(x_{n-1} + c_i h) + O(h^{q+1})$. Aquí, $c = [c_1, \dots, c_s]^T$ es el vector de abscisas.

En la implementación utilizaremos un tipo de GLM para el cual se cumple que $p = q$ y $p = s + 1$. En la siguiente sección estudiaremos la consistencia, estabilidad y convergencia de los GLM en general.

2.2. Consistencia, estabilidad y convergencia

Asumamos que existen vectores q_0 y q_1 con

$$q_0 = [q_{1,0}, q_{2,0}, \dots, q_{r,0}]^T, \quad q_1 = [q_{1,1}, q_{2,1}, \dots, q_{r,1}]^T,$$

tales que los componentes del vector de entrada $y^{[n-1]}$ satisfacen

$$y_i^{[n-1]} = q_{i,0}y(x_{n-1}) + q_{i,1}hy'(x_{n-1}) + O(h^2), \quad i = 1, \dots, r. \tag{2.4}$$

Pedimos que los componentes del vector de etapas $Y^{[n]}$ y el vector de salida $y^{[n]}$ satisfagan

$$\begin{aligned} Y_i^{[n]} &= y(x_{n-1} + c_i h) + O(h^2), \quad i = 1, \dots, s \\ &= y(x_{n-1}) + hc_i y'(x_{n-1}) + O(h^2), \quad y(x_{n-1} + c_i h) = \sum_i \frac{(c_i h)^i}{i!} y^{(i)}(x_{n-1}) \end{aligned} \quad (2.5)$$

$$y_i^{[n]} = q_{i,0} y(x_n) + q_{i,1} h y'(x_n) + O(h^2), \quad i = 1, \dots, r. \quad (2.6)$$

Reemplazando (2.4), (2.5) y (2.6) en las ecuaciones del método (2.1), se obtiene lo siguiente

$$\begin{aligned} y(x_{n-1}) + hc_i y'(x_{n-1}) &= \\ h \sum_{j=1}^s a_{ij} y'(x_{n-1}) + \sum_{j=1}^r u_{ij} (q_{i,0} y(x_{n-1}) + h q_{j,1} y'(x_{n-1})) + O(h^2), \quad i = 1, \dots, s, \end{aligned} \quad (2.7)$$

$$\begin{aligned} q_{i,0} y(x_n) + q_{i,1} h y'(x_n) &= \\ h \sum_{j=1}^s b_{ij} y'(x_n) + \sum_{j=1}^r v_{ij} (q_{i,0} y(x_{n-1}) + h q_{j,1} y'(x_{n-1})) + O(h^2), \quad i = 1, \dots, r. \end{aligned} \quad (2.8)$$

Ahora, la aproximación de $y(x_n)$ usando derivadas está dada por $y(x_n) \approx y(x_{n-1}) + (x_n - x_{n-1})y'(x_{n-1})$. Como $h = x_n - x_{n-1}$, entonces $y(x_n) \approx y(x_{n-1}) + h y'(x_{n-1})$. Reemplazando esta aproximación en (2.8) tenemos que

$$\begin{aligned} q_{i,0} y(x_{n-1}) + h q_{i,0} y'(x_{n-1}) + q_{i,1} h y'(x_n) &= \\ h \sum_{j=1}^s b_{ij} y'(x_n) + \sum_{j=1}^r v_{ij} q_{i,0} y(x_{n-1}) + \sum_{j=1}^r h q_{j,1} y'(x_{n-1}) + O(h^2), \quad i = 1, \dots, r. \end{aligned} \quad (2.9)$$

Para tener un error de orden $O(h^2)$ en las expresiones (2.7) y (2.9), los términos que acompañan a $y(x_{n-1})$, $h y'(x_{n-1})$ y $h y'(x_n)$ en ambas partes de las igualdades deben ser iguales, de esta manera se tiene lo siguiente

$$\begin{aligned} \sum_{j=1}^r u_{ij} q_{j,0} &= 1, \quad i = 1, 2, \dots, s, \\ \sum_{j=1}^r v_{ij} q_{j,0} &= q_{i,0}, \quad i = 1, 2, \dots, r, \\ \sum_{j=1}^s a_{ij} + \sum_{j=1}^r u_{ij} q_{j,1} &= c_i, \quad i = 1, \dots, s, \\ \sum_{j=1}^s b_{ij} + \sum_{j=1}^r v_{ij} q_{j,1} &= q_{i,0} + q_{i,1}, \quad i = 1, \dots, r. \end{aligned} \quad (2.10)$$

Las relaciones en (2.10) inducen las siguientes definiciones

Definición 2.1. (Jackiewicz, 2009) GLM es preconsistente si existe un vector q_0 tal que

$$\begin{aligned} Uq_0 &= e, \quad e = [1, \dots, 1]^T \in \mathbb{R}^s, \\ Vq_0 &= q_0. \end{aligned} \tag{2.11}$$

A q_0 se le llama vector de preconsistencia.

Definición 2.2. (Jackiewicz, 2009) Un GLM es consistente si es preconsistente, con vector de preconsistencia q_0 , y si existe un vector q_1 tal que

$$Be + Vq_1 = q_0 + q_1,$$

donde $e = [1, \dots, 1]^T \in \mathbb{R}^r$. A q_1 se le llama vector de consistencia.

Definición 2.3. (Jackiewicz, 2009) Un GLM es etapa-consistente si

$$Ae + Uq_1 = c.$$

Dado un GLM del cual sólo se tienen las cuatro matrices que lo definen, A, B, U y V , entonces por (2.11) la matriz V debe tener un autovalor igual a 1 con autovector igual a q_0 (Jackiewicz, 2009).

A continuación, se presenta la definición de matriz estable, la cual será usada en la definición de estabilidad de un GLM.

Definición 2.4. (Jackiewicz, 2009) La matriz V es estable si existe una constante $M > 0$ tal que para todo $n \in \mathbb{N}$ se cumple que $\|V^n\| \leq M$.

Definición 2.5. (Jackiewicz, 2009) Un General Linear Method es estable si V es una matriz estable.

Finalmente, se presenta la definición de convergencia de un GLM.

Definición 2.6. (Huang, 2005) Un General Linear Method es convergente si para cualquier valor inicial, el problema

$$y'(x) = f(y(x)), \quad y(x_0) = y_0,$$

sujeto a la condición de Lipschitz $\|f(y) - f(z)\| \leq L\|y - z\|$, existe un vector no nulo u , tal que si $y_i^{[0]} = u_i y(x_0) + O(h)$, con $i = 1, \dots, r$, entonces $y_i^{[n]} = u_i y(x_0 + nh) + O(h)$, para todo n donde nh está acotado.

La definición anterior garantiza que podemos obtener soluciones numéricas precisas en cualquier punto tomando h suficientemente pequeño, para condiciones iniciales suficientemente cercanas (Huang, 2005).

El siguiente teorema enuncia la relación entre consistencia, estabilidad y convergencia de un GLM.

Teorema 2.7. *(Jackiewicz, 2009) Un General Linear Method es convergente si, y sólo si, es estable y consistente.*

3. Método de integración multietapa diagonalmente implícito (DIMSIM)

3.1. Representación de DIMSIM

Como hemos visto, un GLM está caracterizado por cuatro matrices, A, B, U , y V . En este trabajo, como en Huang (2005), A es una matriz triangular inferior; los elementos de su diagonal son todos iguales a una constante positiva λ , de modo que se puedan encontrar los valores de cada etapa por separado y de forma secuencial y así reducir el costo de implementación. Al método que satisface las hipótesis anteriores se le conoce como Método de integración multietapa diagonalmente implícito (DIMSIM por sus siglas en inglés).

Un DIMSIM está definido como

$$\begin{aligned} Y_i^{[n]} &= \sum_{j=1}^s a_{ij} h f(Y_j^{[n]}) + \sum_{j=1}^r u_{ij} y_j^{[n-1]}, \quad i = 1, 2, \dots, s, \\ y_i^{[n]} &= \sum_{j=1}^s b_{ij} h f(Y_j^{[n]}) + \sum_{j=1}^r v_{ij} y_j^{[n-1]}, \quad i = 1, 2, \dots, r, \end{aligned}$$

donde

$$Y_i^{[n]} = y(x_{n-1} + c_i h) + O(h^{q+1}),$$

y además

$$\begin{aligned} y_i^{[n-1]} &= \sum_{k=0}^p \alpha_{ik} y^{(k)}(x_{n-1}) h^k + O(h^{p+1}), \\ y_i^{[n]} &= \sum_{k=0}^p \alpha_{ik} y^{(k)}(x_n) h^k + O(h^{p+1}). \end{aligned} \tag{3.1}$$

Dependiendo de la estructura de la matriz A , los DIMSIM se pueden dividir en los siguientes 4 tipos

Tabla 1: Tipos de DIMSIM.

Tipo	Estructura	Problema
Tipo 1	$A = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ a_{21} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_{s1} & a_{s2} & \cdots & 0 \end{bmatrix}$	No- <i>Stiff</i>
Tipo 2	$A = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}$	No- <i>Stiff</i>
Tipo 3	$A = \begin{bmatrix} \lambda & 0 & \cdots & 0 \\ a_{21} & \lambda & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_{s1} & a_{s2} & \cdots & \lambda \end{bmatrix}$	<i>Stiff</i>
Tipo 4	$A = \begin{bmatrix} \lambda & 0 & \cdots & 0 \\ 0 & \lambda & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda \end{bmatrix}$	<i>Stiff</i>

Observación 3.1. 1. En lo que sigue, se considerará un DIMSIM tal que $p = q$ y $s = r = p + 1$. Considerando (3.1), si $r = p + 1$, entonces el vector $y_i^{[n]}$ estará en forma de un vector de Nordsieck (Huang, 2005). Un vector de Nordsieck tiene la siguiente forma

$$\tilde{y}^{[n]} = \begin{bmatrix} y(x_n) \\ hy'(x_n) \\ \vdots \\ h^p y^{(p)}(x_n) \end{bmatrix}, \quad (3.2)$$

lo que facilita el cambio del paso h en cada iteración (Huang, 2005).

- Debido a que los métodos implícitos de Runge-Kutta tienen una amplia región de estabilidad, se desea que este tipo de GLM tenga la misma función de estabilidad que el método implícito de Runge-Kutta del mismo orden, a esta propiedad se le conoce como estabilidad inherente de Runge-Kutta (Huang, 2005).

3.2. Matriz de estabilidad

Un GLM, y en particular un DIMSIM, se puede representar de la siguiente forma

$$\begin{aligned} Y^{[n]} &= AhF + Uy^{[n-1]}, \\ y^{[n]} &= BhF + Vy^{[n-1]}. \end{aligned} \tag{3.3}$$

Para estudiar la estabilidad de estos métodos, se aplica (3.3) al siguiente problema lineal $y' = Ly$, $L \in \mathbb{C}$. Reemplazando $y' = Ly$ en la primera ecuación de (3.3) se tiene que

$$\begin{aligned} Y^{[n]} &= AhLY^{[n]} + Uy^{[n-1]} \\ \implies Uy^{[n-1]} &= Y^{[n]} - AhLY^{[n]} \\ \implies Uy^{[n-1]} &= (I - AhL)Y^{[n]} \\ \implies Y^{[n]} &= (I - AhL)^{-1}Uy^{[n-1]}. \end{aligned} \tag{3.4}$$

Haciendo $z = Lh$ se obtiene que $Y^{[n]} = (I - zA)^{-1}Uy^{[n-1]}$ ($(I - zA)$ es no singular debido a la forma de la matriz A). Reemplazando esto en la segunda ecuación de (3.3) se tiene que

$$\begin{aligned} y^{[n]} &= BhL(I - zA)^{-1}Uy^{[n-1]} + Vy^{[n-1]}, \\ y^{[n]} &= Bz(I - zA)^{-1}Uy^{[n-1]} + Vy^{[n-1]}, \quad z = Lh \\ y^{[n]} &= (Bz(I - zA)^{-1}U + V)y^{[n-1]}. \end{aligned} \tag{3.5}$$

Así, se obtiene que $y^{[n]} = My^{[n-1]}$, donde

$$M(z) = V + zB(I - zA)^{-1}U,$$

y esta es llamada la matriz de estabilidad del método (Huang, 2005).

3.3. Condiciones de orden

En esta sección se presentan las condiciones suficientes y necesarias para que un GLM tenga el orden de etapas igual al orden del método, es decir, $p = q$.

Teorema 3.2. (Huang, 2005; Wright, 2002) *Un GLM con matrices de coeficiente A, B, U y V en forma de Nordsieck, tiene orden de etapas q igual al orden del método p si, y sólo si,*

$$\begin{aligned} \exp(cz) &= zA \exp(cz) + UZ + O(z^{p+1}), \\ \exp(z)Z &= zB \exp(cz) + VZ + O(z^{p+1}), \end{aligned}$$

donde

$$\exp(cz) = \begin{bmatrix} \exp(c_1 z) \\ \vdots \\ \exp(c_s z) \end{bmatrix}.$$

Demostración. Para satisfacer que las etapas tengan orden igual a p , los valores de estas, que representan las aproximaciones de la solución en los puntos $x_{n-1} + hc_i$, $i = 1, \dots, s$, deben tener la siguiente forma

$$Y_i^{[n]} = y(x_{n-1} + hc_i) + O(h^{p+1}).$$

Usando la expansión en serie de Taylor de $y(x_{n-1} + hc_i)$ alrededor del punto x_{n-1} se tiene que

$$Y_i^{[n]} = y(x_{n-1}) + c_i h y'(x_{n-1}) + \dots + \frac{(c_i h)^p}{p!} y^{(p)}(x_{n-1}) + O(h^{p+1}). \quad (3.6)$$

Como las aproximaciones de entrada $y^{[n-1]}$ están en forma de Nordsieck, representamos el lado derecho de la igualdad de (3.6) en forma vectorial y en términos de $y^{[n-1]}$, de donde (3.6) queda como

$$Y^{[n]} = C y^{[n-1]} + O(h^{p+1}), \quad (3.7)$$

donde C es la matriz de Vandermonde de tamaño $(p+1) \times (p+1)$ dada por

$$C = \begin{bmatrix} 1 & c_1 & \frac{c_1^2}{2!} & \cdots & \frac{c_1^p}{p!} \\ 1 & c_2 & \frac{c_2^2}{2!} & \cdots & \frac{c_2^p}{p!} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & c_s & \frac{c_s^2}{2!} & \cdots & \frac{c_s^p}{p!} \end{bmatrix}.$$

Ahora, para la derivada de cada etapa se tiene que

$$\begin{aligned} hf(Y_i^{[n]}) &= h y'(x_{n-1}) + O(h^{p+2}) \\ &= \sum_{k=1}^{p+1} \frac{c_i^{k-1}}{(k-1)!} y^{(k)}(x_{n-1}) h^k + O(h^{p+2}) \\ &= \sum_{k=1}^p \frac{c_i^{k-1}}{(k-1)!} y^{(k)}(x_{n-1}) h^k + O(h^{p+1}). \end{aligned} \quad (3.8)$$

Análogamente, la ecuación anterior se puede representar en forma vectorial como

$$hF = CK y^{[n-1]} + O(h^{p+1}), \quad (3.9)$$

donde K es la siguiente matriz de cambio de tamaño $(p+1) \times (p+1)$

$$K = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \end{bmatrix}.$$

Por otra parte, para un método de orden p las aproximaciones de salida en el paso n tienen la forma

$$y_i^{[n]} = h^i y^{(i)}(x_{n-1} + h) + O(h^{p+1}), \quad i = 1, \dots, p, \quad (3.10)$$

y las aproximaciones de entrada tienen la forma

$$y_i^{[n-1]} = h^{i-1} y^{(i-1)}(x_{n-1}) + O(h^{p+1}), \quad i = 1, \dots, p. \quad (3.11)$$

Haciendo la expansión de Taylor de $y^{(i)}(x_{n-1} + h)$ alrededor del punto x_{n-1} en la aproximación de salida y representandola en términos de $y_i^{[n-1]}$, se tiene que

$$y^{[n]} = Ey^{[n-1]} + O(h^{p+1}), \quad (3.12)$$

donde E es la siguiente matriz de Toeplitz de tamaño $(p+1) \times (p+1)$ dada por

$$E = \begin{bmatrix} 1 & \frac{1}{1!} & \frac{1}{2!} & \cdots & \frac{1}{p!} \\ 0 & 1 & \frac{1}{1!} & \cdots & \frac{1}{(p-1)!} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \frac{1}{1!} \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} = \exp(K).$$

Sustituyendo (3.7), (3.9), (3.12) en (3.3) se obtienen las siguientes condiciones de orden

$$\begin{aligned} Cy^{[n-1]} &= ACKy^{[n-1]} + Uy^{[n-1]} + O(h^{p+1}), \\ Ey^{[n-1]} &= BCKy^{[n-1]} + Vy^{[n-1]} + O(h^{p+1}). \end{aligned} \quad (3.13)$$

Se define ahora un vector Z dado por

$$Z = \begin{bmatrix} 1 \\ z \\ \vdots \\ z^p \end{bmatrix},$$

donde z^k representa $h^k y^{(k)}(x_{n-1})$. Luego, el vector de Nordsieck $y^{[n-1]}$ puede ser representado por el vector Z . Así, haciendo este cambio en las condiciones de orden (3.13) tenemos que

$$\begin{aligned} CZ &= ACKZ + UZ + O(z^{p+1}), \\ EZ &= BCKZ + VZ + O(z^{p+1}). \end{aligned} \quad (3.14)$$

Las matrices K, C y E tienen las siguientes propiedades (Wright, 2002),

$$\begin{aligned} KZ &= zZ + O(z^{p+1}), \\ CZ &= \exp(cz) + O(z^{p+1}), \\ EZ &= \exp(z)Z + O(z^{p+1}), \end{aligned}$$

donde z es un parámetro complejo. Sustituyendo las propiedades anteriores en (3.14) tenemos que

$$\begin{aligned}\exp(cz) &= zACZ + UZ + O(z^{p+1}) \\ &= zA \exp(cz) + UZ + O(z^{p+1}), \\ \exp(z)Z &= zBCZ + VZ + O(z^{p+1}) \\ &= zB \exp(cz) + VZ + O(z^{p+1}).\end{aligned}$$

□

3.4. Estabilidad Inherente de Runge-Kutta

Se mencionó anteriormente que se deseaba trabajar con DIMSIM que tuvieran la propiedad de estabilidad de Runge-Kutta. En esta sección se brindarán las definiciones y teoremas que garantizan esta propiedad. Previamente se vio que la matriz de estabilidad de un GLM es de la forma $M(z) = V + zB(I - zA)^{-1}U$, se define a partir de esta la región de estabilidad como

$$T = \{z \in \mathbb{C}, \exists K : \|M(z)\|^n \leq K, \forall n \geq 1\}.$$

Ahora, sea w un autovalor de $M(z)$, entonces $|w| \leq \|M(z)\|$. Como $\|M(z)\|^n \leq K, \forall n \geq 1$, se tiene que $\|M(z)\| \leq 1$. Así, para cualquier autovalor w de $M(z)$ se tiene que $|w| \leq 1$. El polinomio característico, el cual es conocido como la función de estabilidad de $M(z)$, se puede escribir como

$$\begin{aligned}P(w) &= \det(wI - M(z)) \\ &= w^r + P_1(z)w^{r-1} + \dots + P_r(z),\end{aligned}\tag{3.15}$$

donde el grado de $P_i(z)$ es al menos s para todo $i = 1, \dots, r$. Por lo anterior, se tiene la siguiente definición.

Definición 3.3. Un GLM se dice que posee estabilidad de Runge-Kutta si la función de estabilidad tiene la forma

$$P(w) = \det(wI - M(z)) = w^{r-1}(w - R(z)),$$

donde $R(z)$ es la función de estabilidad del método Runge-Kutta.

Definición 3.4. (Huang, 2005) Un GLM que satisface $Ve_1 = e_1$ tiene la propiedad de estabilidad inherente de Runge-Kutta si

$$\det(wI - V) = (w - 1)w^{r-1},\tag{3.16a}$$

$$BA \equiv JB, \quad (3.16b)$$

$$BU \equiv JV - VJ, \quad (3.16c)$$

donde \equiv denota la igualdad de dos matrices, excepto la primera fila, además J es igual a la siguiente matriz de cambio,

$$J = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 & 0 \end{bmatrix}, \quad (3.17)$$

y e_1 es el primer vector de la base canónica de tamaño r , es decir, $e_1 = (1, 0, \dots, 0)_{1 \times r}$.

Observación 3.5. La propiedad $Ve_1 = e_1$ junto con la condición de que la matriz V sólo tenga un autovalor distinto de cero, garantizan que esta matriz es estable (Huang, 2005).

Teorema 3.6. (Huang, 2005; Wright, 2002) *Si un GLM tiene la propiedad de estabilidad inherente de Runge-Kutta, entonces el polinomio característico de la matriz de estabilidad $M(z)$ tiene la forma*

$$P(w) = w^{r-1}(w - R(z)).$$

Demostración. En lugar de considerar la matriz de estabilidad $M(z)$, se considera una matriz similar a esta, la cual viene dada por

$$M(z) \sim (I - zJ)M(z)(I - zJ)^{-1},$$

de esta manera se tiene que

$$\begin{aligned} M(z) &\sim (I - zJ)M(z)(I - zJ)^{-1} \\ &= (I - zJ)[V + zB(I - zA)^{-1}U](I - zJ)^{-1} \\ &= (I - zJ)V(I - zJ)^{-1} + z(I - zJ)B(I - zA)^{-1}U(I - zJ)^{-1} \\ &= (I - zJ)V(I - zJ)^{-1} + z(B - zJB)(I - zA)^{-1}U(I - zJ)^{-1} \\ &\equiv (I - zJ)V(I - zJ)^{-1} + zB(I - zA)(I - zA)^{-1}U(I - zJ)^{-1}, && \text{por (3.16b)} \\ &\equiv (I - zJ)V(I - zJ)^{-1} + zBU(I - zJ)^{-1}, && (I - zA)(I - zA)^{-1} = I \\ &\equiv ((I - zJ)V + z(JV - VJ))(I - zJ)^{-1}, && \text{por (3.16c)} \\ &= (V - zJV + zJV - zVJ)(I - zJ)^{-1} \\ &= V(I - zJ)(I - zJ)^{-1} = V. \end{aligned}$$

Esto implica que $(I - zJ)M(z)(I - zJ)^{-1}$ es idéntica a V excepto por la primera fila. Por observación 3.5 e hipótesis, V tiene solo un autovalor distinto de cero, así, $M(z)$ tiene un único autovalor distinto de cero, el cual es igual a $R(z)$. \square

Teorema 3.7. (*Wright, 2002*) Dado un GLM en forma de Nordsieck con $q = p$, la matriz X más general que satisface

$$\begin{aligned} BA &\equiv XB, \\ BU &\equiv XV - VX, \end{aligned}$$

es la matriz de doble compañía dada por

$$X = \begin{bmatrix} -\alpha_1 & -\alpha_2 & \cdots & -\alpha_{p-1} & -\alpha_p & -\alpha_{p+1} - \beta_{p+1} \\ 1 & 0 & \cdots & 0 & 0 & -\beta_p \\ 0 & 1 & \cdots & 0 & 0 & -\beta_{p-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & -\beta_3 \\ 0 & 0 & \cdots & 1 & 0 & -\beta_2 \\ 0 & 0 & \cdots & 0 & 1 & -\beta_1 \end{bmatrix}. \quad (3.18)$$

La matriz J dada en (3.17) es un caso particular de la matriz X dada en (3.18) (Wright, 2002).

3.5. Escogencia de los valores para λ

Como se mencionó anteriormente, dado un GLM de orden p , si este tiene estabilidad inherente de Runge-Kutta, su función de estabilidad será la misma a la del método implícito Runge-Kutta de mismo orden. De esta manera, la función de estabilidad tiene la siguiente forma

$$R(z) = \frac{(-1)^{p+1} \sum_{i=0}^p L_{p+1}^{(p+1-i)}\left(\frac{1}{\lambda}\right)(\lambda z)^i}{(1 - \lambda z)^{p+1}} = \frac{N(z)}{D(z)},$$

donde $L_n(x)$ representa el polinomio de Laguerre de grado n dado por

$$L_n(x) = \sum_{m=0}^n (-1)^m \binom{n}{m} \frac{x^m}{m!}.$$

Por lo anterior, se tiene que $N(z)$ es un polinomio de grado p , el cual por definición es escogido de modo que se cumpla que

$$R(z) = \exp(z) + C_p z^{p+1} + O(z^{p+2}),$$

donde C_p es la constante de error dada por

$$C_p = (-\lambda)^{p+1} L_{p+1}\left(\frac{1}{\lambda}\right).$$

Ejemplo 3.8. Sea $p = 4$, entonces

$$\begin{aligned} L_5(x) &= \sum_{m=0}^5 (-1)^m \binom{5}{m} \frac{x^m}{m!} = 1 - 5x + 5x^2 - \frac{5}{3}x^3 + \frac{5}{24}x^4 - \frac{x^5}{120}, \\ L_5^{(1)}(x) &= -5 + 10x - 5x^2 + \frac{5}{6}x^3 - \frac{x^4}{24}, \\ L_5^{(2)}(x) &= 10 - 10x + \frac{5}{2}x^2 - \frac{x^3}{6}, \\ L_5^{(3)}(x) &= -10 + 5x - \frac{x^2}{2}, \\ L_5^{(4)}(x) &= 5 - x, \quad L_5^{(5)} = -1. \end{aligned}$$

Reemplazando en la expresión para $N(z)$ se tiene que

$$\begin{aligned} N(z) &= (-1)^5 \sum_{i=0}^4 L_5^{(5-i)} \left(\frac{1}{\lambda} \right) \lambda^i z^i \\ &= (-1) \left[L_5^{(5)} \left(\frac{1}{\lambda} \right) + L_5^{(4)} \left(\frac{1}{\lambda} \right) \lambda z + L_5^{(3)} \left(\frac{1}{\lambda} \right) \lambda^2 z^2 + L_5^{(2)} \left(\frac{1}{\lambda} \right) \lambda^3 z^3 + L_5^{(1)} \left(\frac{1}{\lambda} \right) \lambda^4 z^4 \right] \\ &= (-1) \left[-1 + \left(5 - \frac{1}{\lambda} \right) \lambda z + \left(-10 + \frac{5}{\lambda} - \frac{1}{2\lambda^2} \right) \lambda^2 z^2 + \left(10 - \frac{10}{\lambda} + \frac{5}{2\lambda^2} - \frac{1}{6\lambda^3} \right) \lambda^3 z^3 + \right. \\ &\quad \left. + \left(-5 + \frac{10}{\lambda} - \frac{5}{\lambda^2} + \frac{5}{6\lambda^3} - \frac{1}{24\lambda^4} \right) \lambda^4 z^4 \right] \\ &= 1 + (1 - 5\lambda)z + \left(\frac{1}{2} - 5\lambda + 10\lambda^2 \right) z^2 + \left(\frac{1}{6} - \frac{5\lambda}{2} + 10\lambda^2 - 10\lambda^3 \right) z^3 + \\ &\quad + \left(\frac{1}{24} - \frac{5\lambda}{6} + 5\lambda^2 - 10\lambda^3 + 5\lambda^4 \right) z^4. \end{aligned}$$

Por otra parte, la constante de error para $p = 4$ viene dada por

$$\begin{aligned} C_4 &= (-\lambda)^5 L_5 \left(\frac{1}{\lambda} \right) \\ &= -\lambda^5 \left[1 - \frac{5}{\lambda} + \frac{5}{\lambda^2} - \frac{5}{3\lambda^3} + \frac{5}{24\lambda^4} - \frac{1}{120\lambda^5} \right] \\ &= -\lambda^5 + 5\lambda^4 - 5\lambda^3 + \frac{5}{3}\lambda^2 - \frac{5}{24}\lambda + \frac{1}{120}. \end{aligned}$$

Este tipo de GLM se desea emplear para la solución de sistemas diferenciales tipo *Stiff*, por lo cual se desea que sean métodos A -estables. Para satisfacer esta necesidad, la función de estabilidad debe satisfacer $|R(z)| \leq 1$ para todo $z \in \mathbb{C}$ con $Re(z) < 0$ (Huang, 2005). Haciendo $z = iy$, tenemos que lo anterior es equivalente a que el siguiente polinomio

$$E(y) = D(iy)D(-iy) - N(iy)N(-iy) \quad (3.19)$$

sea mayor o igual a cero (Butcher, 2016).

Dado p , para encontrar los posibles valores de λ se debe hallar el polinomio $N(z)$, luego reemplazarlo en (3.19) y resolver la desigualdad $E(y) \geq 0$. Para los valores de $p = 1, \dots, 6$, tenemos que el valor de λ en cada caso debe estar en los siguientes intervalos

Tabla 2: Posibles valores de λ dependiendo del orden p del GLM.

p	Intervalo para λ
1	[0.29289321, 1.70710678]
2	[0.18042530, 2.18560009]
3	[0.22364780, 0.57281606]
4	[0.24799464, 0.67604239]
5	[0.18391465, 0.33414236]
6	[0.20408345, 0.37886489]

3.6. Ejemplos de DIMSIM

A continuación, se presentan algunos ejemplos de este tipo de métodos los cuales fueron obtenidos de Butcher (2001). Todos los ejemplos están en su forma de Nordsieck, lo cual reduce el costo computacional en la ejecución de cada método.

Ejemplo 3.9. 1. Método con $p = q = 2$, $s = 3$, $c = [0, \frac{1}{2}, 1]^T$, $\lambda = \frac{4}{9}$,

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array} \right] = \left[\begin{array}{ccc|ccc} \frac{4}{9} & 0 & 0 & 1 & -\frac{4}{9} & 0 \\ \frac{29}{72} & \frac{4}{9} & 0 & 1 & -\frac{25}{72} & -\frac{7}{36} \\ \frac{1295}{522} & -\frac{248}{261} & \frac{4}{9} & 1 & -\frac{509}{522} & \frac{277}{261} \\ \hline \frac{9827}{4698} & -\frac{644}{783} & \frac{44}{81} & 1 & -\frac{3817}{4698} & \frac{1729}{2349} \\ \frac{292}{261} & -\frac{98}{261} & \frac{13}{18} & 0 & -\frac{27}{58} & \frac{27}{29} \\ -\frac{3}{58} & \frac{1}{29} & \frac{1}{4} & 0 & -\frac{27}{116} & \frac{27}{58} \end{array} \right] \quad (3.20)$$

2. Método con $p = q = 3$, $s = 4$, $c = [1, \frac{2}{3}, \frac{1}{3}, 1]^T$, $\lambda = \frac{1}{4}$,

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array} \right] = \left[\begin{array}{cccc|cccc} \frac{1}{4} & 0 & 0 & 0 & 1 & \frac{3}{4} & \frac{1}{2} & \frac{1}{4} \\ -\frac{7}{216} & \frac{1}{4} & 0 & 0 & 1 & \frac{97}{216} & \frac{19}{108} & \frac{13}{216} \\ \frac{53}{432} & -\frac{3}{8} & \frac{1}{4} & 0 & 1 & \frac{145}{432} & \frac{43}{216} & \frac{37}{432} \\ -\frac{39}{128} & \frac{63}{64} & -\frac{9}{32} & \frac{1}{4} & 1 & \frac{45}{128} & -\frac{1}{64} & -\frac{7}{128} \\ \hline 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \frac{3}{4} & -\frac{9}{2} & \frac{9}{4} & 2 & 0 & -\frac{1}{2} & 0 & 0 \\ \frac{1}{3} & -\frac{15}{2} & 6 & \frac{8}{3} & 0 & -\frac{3}{2} & 0 & 0 \end{array} \right] \quad (3.21)$$

3. Método con $p = q = 4$, $s = 5$, $c = [1, \frac{3}{4}, \frac{1}{2}, \frac{1}{4}, 1]^T$, $\lambda = \frac{1}{4}$,

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array} \right] = \left[\begin{array}{ccccc|cccc} \frac{1}{4} & 0 & 0 & 0 & 0 & 1 & \frac{3}{4} & \frac{1}{2} & \frac{1}{4} & 0 \\ -\frac{27}{1024} & \frac{1}{4} & 0 & 0 & 0 & 1 & \frac{539}{1024} & \frac{123}{512} & \frac{81}{1024} & 0 \\ \frac{1}{32} & -\frac{1}{9} & \frac{1}{4} & 0 & 0 & 1 & \frac{95}{288} & \frac{5}{48} & \frac{1}{32} & 0 \\ \frac{129}{1024} & -\frac{5}{12} & \frac{3}{8} & \frac{1}{4} & 0 & 1 & -\frac{259}{3072} & -\frac{33}{512} & \frac{13}{1024} & 0 \\ -\frac{29}{96} & \frac{7}{9} & -\frac{1}{4} & \frac{1}{3} & \frac{1}{4} & 1 & \frac{55}{288} & \frac{1}{48} & -\frac{1}{32} & 0 \\ \hline -\frac{29}{96} & \frac{7}{9} & -\frac{1}{4} & \frac{1}{3} & \frac{1}{4} & 1 & \frac{55}{288} & \frac{1}{48} & -\frac{1}{32} & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \frac{13}{6} & -8 & 6 & -\frac{8}{3} & 2 & 0 & \frac{1}{2} & 0 & 0 & 0 \\ \frac{43}{9} & -\frac{64}{3} & -\frac{64}{3} & -\frac{64}{9} & \frac{8}{3} & 0 & -\frac{1}{3} & -\frac{2}{3} & 0 & 0 \\ \frac{21}{4} & -\frac{800}{27} & 40 & -\frac{32}{3} & \frac{8}{3} & 0 & -\frac{823}{108} & -\frac{109}{18} & -\frac{7}{4} & 0 \end{array} \right] \quad (3.22)$$

Los coeficientes anteriores serán utilizados en la implementación del método.

4. Algoritmo y Código Computacional

En esta sección se explicará el funcionamiento del código creado por Shirley Huang en Huang (2005), el cual implementa un DIMSIM de orden $p = q = 4$ y $s = r = 5$. El código fue escrito en el lenguaje de programación FORTRAN y para su ejecución se instalaron el compilador gfortran, junto con los paquetes de Lapack y BLAS, en una máquina virtual de Linux. El computador donde se realizaron los experimentos posee las siguientes características:

- Sistema operativo: Windows 10.
- Procesador: Intel Core i3-1005G1.
- Velocidad: 3.40 GHz.
- Memoria Ram: 4GB DDR4.
- SSD: 512GB.

La máquina virtual posee las siguientes características:

- Sistema operativo: Linux Ubuntu.
- Procesador: Intel Core i3-1005G1.
- Velocidad: 3.40 GHz.
- Memoria Ram: 2GB DDR4.
- SSD: 2GB.

4.1. Algoritmo y Código Computacional

El diagrama de flujo de la Figura 1 muestra el ciclo realizado por la función principal que realiza las iteraciones de un DIMSIM de orden $p = q = 4$ y $s = r = 5$.

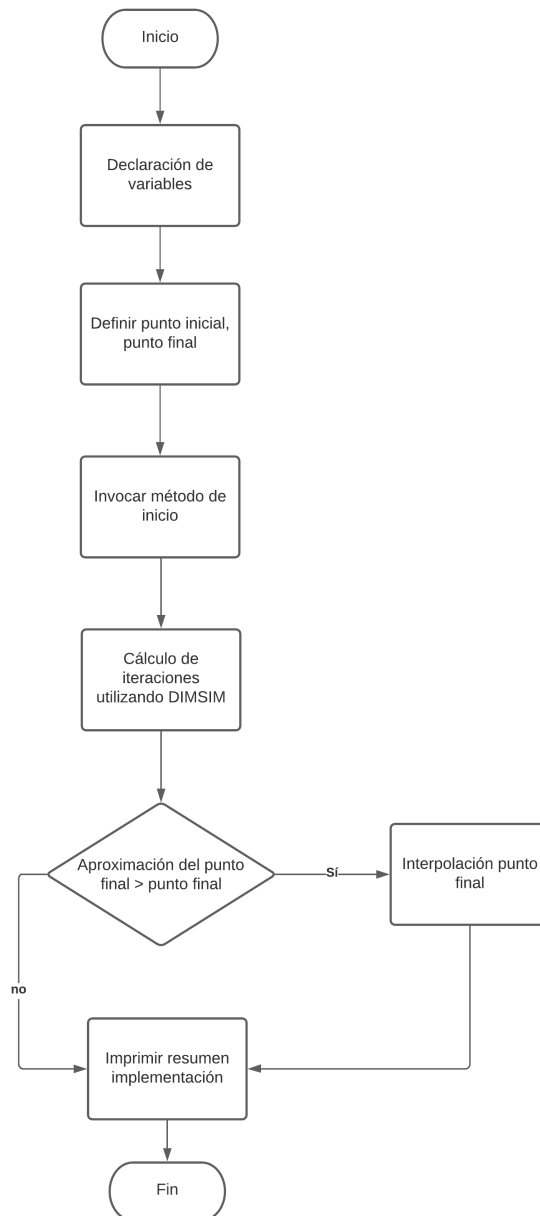


Figura 1: Diagrama de flujo del código que implementa un DIMSIM de orden $p = q = 4$ y $s = r = 5$.

Observación 4.1. La información presentada a continuación fue obtenida de Alexander (1991); Huang (2005); Mokhtary y Hosseini (2008).

1. **Declaración de variables:** En el lenguaje de programación FORTRAN se deben inicializar todas las variables que se utilizarán en el código, esto implica definir el tipo, el tamaño (si es necesario) y el nombre de cada variable a utilizar.
2. **Definición del punto inicial y el punto final:** El código necesita que se le proporcionen los valores del punto inicial y del punto final para poder encontrar la solución numérica del sistema de ecuaciones. En este proceso, el código llama y almacena los datos anteriores, los cuales son almacenados en subrutinas para su uso posterior en el cálculo de las iteraciones.

3. **Método de inicio:** Como se ha mencionado anteriormente, los valores en cada paso están en la forma de Nordsieck, esto implica que para un método de orden p , la primera aproximación de entrada viene dada por

$$\begin{bmatrix} y(x_0) \\ h_0 y'(x_0) \\ \vdots \\ h_0^p y^{(p)}(x_0) \end{bmatrix}, \quad (4.1)$$

donde h_0 es el paso inicial. Debido a que en los problemas de condición inicial sólo se tiene disponible el valor de $y(x_0) = y_0$, se hace necesario el uso de un método de la forma

$$\begin{bmatrix} \widehat{A} & \widehat{U} \\ \widehat{B} & \widehat{V} \end{bmatrix} \quad (4.2)$$

que proporcione la primera aproximación de la forma (4.1) (Huang, 2005). Este tipo de método de inicio puede aproximar la solución en los puntos x_0 o $x_0 + h_0$ (Huang, 2005). En la tesis de Huang (2005) se utiliza un tipo de método implícito de Runge-Kutta con $p + 1$ salidas, el cual fue propuesto en Gear (1980); Huang (2005). Este método utiliza el mismo λ que el método principal y, debido a que sólo se cuenta con el valor de y_0 , se considera $\widehat{U} = [1, 1, \dots, 1]$ y $\widehat{V} = [1, 0, \dots, 0]$.

Ejemplo 4.2. Para el método con $p = q = 4$ y $s = 5$ que se utilizará en la implementación, se tiene que el método de inicio viene dado por

$$\left[\begin{array}{cccccc|cc} \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{1}{4} - \frac{\sqrt{2}}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 1 \\ -\frac{31}{36} + \frac{17\sqrt{2}}{36} & \frac{4}{9} - \frac{2\sqrt{2}}{9} & \frac{1}{4} & 0 & 0 & 0 & 0 & 1 \\ 0 & \frac{3}{8} + \frac{9\sqrt{2}}{32} & -\frac{3}{8} - \frac{9\sqrt{2}}{32} & \frac{1}{4} & 0 & 0 & 0 & 1 \\ 0 & -\frac{9}{8} - \frac{3\sqrt{2}}{4} & \frac{129}{56} + \frac{45\sqrt{2}}{28} & -\frac{13}{14} - \frac{6\sqrt{2}}{7} & \frac{1}{4} & 0 & 0 & 1 \\ 0 & 0 & -\frac{261}{1288} - \frac{351\sqrt{2}}{2576} & \frac{25}{28} + \frac{9\sqrt{2}}{56} & -\frac{35}{184} - \frac{9\sqrt{2}}{368} & \frac{1}{4} & 0 & 1 \\ 0 & 0 & 0 & \frac{5}{12} & \frac{5}{12} & -\frac{1}{12} & \frac{1}{4} & 1 \\ \hline 0 & 0 & 0 & \frac{2}{3} & -\frac{1}{3} & \frac{2}{3} & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -\frac{4}{3} & 6 & -12 & \frac{22}{3} & 0 \\ 0 & 0 & 0 & -16 & 64 & -80 & 32 & 0 \\ 0 & 0 & 0 & -64 & 192 & -192 & 64 & 0 \end{array} \right]$$

con $\widehat{C} = \left[\frac{1}{4}, \frac{1}{2} - \frac{\sqrt{2}}{4}, \frac{\sqrt{2}}{4} - \frac{1}{6}, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1 \right]^T$.

4. **Cálculo de iteraciones utilizando DIMSIM:** Para el cálculo de las iteraciones se utiliza una modificación del método de Newton que se explica a continuación: Consideremos un sistema de ecuaciones diferenciales ordinarias de N ecuaciones. Las etapas de un GLM de etapa s vienen dadas por

$$Y_i^{[n]} = \sum_{j=1}^s a_{ij} h f(Y_j^{[n]}) + \sum_{j=1}^r u_{ij} y_j^{[n-1]}, \quad i = 1, 2, \dots, s. \quad (4.3)$$

La matriz A del DIMSIM a utilizar es tal que $a_{ii} = \lambda$ y $a_{ij} = 0$ para $i < j$, por lo anterior en (4.3) se tiene que

$$Y_i^{[n]} - \lambda h f(Y_i^{[n]}) = \sum_{j=1, j \neq i}^s a_{ij} h f(Y_j^{[n]}) + \sum_{j=1}^r u_{ij} y_j^{[n-1]}, \quad i = 1, 2, \dots, s. \quad (4.4)$$

Denotando como φ la parte derecha de la igualdad en (4.4) e introduciendo una variable η se tiene que

$$\eta - \lambda h f(\eta) = \varphi, \quad (4.5)$$

de donde

$$\Phi(\varphi, \eta) = \eta - \lambda h f(\eta) - \varphi = 0. \quad (4.6)$$

De esta manera, encontrar el valor de cada etapa es equivalente a resolver la ecuación $\Phi(\varphi, \eta) = 0$. El método de Newton modificado introduce la siguiente matriz

$$M = I - \lambda h J, \quad (4.7)$$

donde J representa el Jacobiano del sistema de ecuaciones diferenciales ordinarias. A partir de una aproximación inicial de $\eta^{(0)}$, el método encuentra la solución mediante la siguiente sucesión iterativa

$$\eta^{(m+1)} = \eta^{(m)} + M^{-1} \Phi(\varphi, \eta^{(m)}), \quad m = 0, 1, 2, \dots \quad (4.8)$$

Observación 4.3. Para el cálculo de la matriz inversa de M se utiliza la descomposición LU .

Para la aproximación inicial de $\eta_0 = Y_i^{[0]}$, se utiliza la interpolación de Hermite. Para las dos primeras etapas tenemos que $Y_1^{[0]} = y_1^{[n-1]}$ y $Y_2^{[0]} = y_1^{[n-1]} + c_2 y_2^{[n-1]} + \frac{c_2^2}{2!} y_3^{[n-1]} + \dots$. Las etapas con $i \geq 3$ son calculadas de la siguiente manera

$$Y_i^{[0]} = a Y_{i-2} + b h f(Y_{i-2}) + c Y_{i-1} + d h f(Y_{i-2}). \quad (4.9)$$

Los valores de a, b, c y d se calculan expandiendo en serie de Taylor los valores de $Y_{i-2}, f(Y_{i-2}), Y_{i-1}$ y $f(Y_{i-1})$ y resolviendo el sistema de ecuaciones resultante.

Observación 4.4. En el proceso del cálculo de cada iteración es donde se realiza el cambio del paso h que depende del comportamiento del problema a trabajar. Para poder realizar el cambio se define el error de truncamiento local para métodos de orden 4 con $\lambda = \frac{1}{4}$ como

$$E_n = -\frac{13}{60}(hf(Y_1) - 4hf(Y_2) + 6hf(Y_3) - 4hf(Y_4) + hf(Y_5)). \quad (4.10)$$

Ahora, denotando h_n como el paso en la iteración n y h_{n+1} como el siguiente paso esperado, se tiene que

$$h_{n+1} = \min\left(2, \max\left(\widehat{\theta}_n, \frac{1}{2}\right)\right), \quad (4.11)$$

donde

$$\theta_n = 0.9\left(\frac{Tol}{\|E_n\|}\right)^{\frac{1}{p+1}}. \quad (4.12)$$

Finalmente, como los vectores de entrada y de salida están en forma de Nordsieck, se hace necesario el cambio de $y^{[n+1]} \approx D(\widehat{\theta}_n)y^{[n]}$ donde

$$D(\widehat{\theta}_n) = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & \widehat{\theta}_n & 0 & \cdots & 0 \\ 0 & 0 & \widehat{\theta}_n^2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & \cdots & \widehat{\theta}_n^p \end{bmatrix}. \quad (4.13)$$

5. Interpolación: Como el método utiliza un paso h que cambia en cada iteración, se hace prácticamente imposible calcular de manera directa la solución en el punto final del intervalo de integración x , debido a que el valor de este usualmente es tal que $x_n < x < x_{n+1}$. Como la solución ya fue calculada en los puntos x_n y x_{n+1} , se realiza una interpolación de la solución $y(x)$ utilizando los valores de $y(x_n), y(x_{n+1}), y'(x_n), y'(x_{n+1})$ de modo que

$$y(x) = ay(x_n) + by(x_{n+1}) + chy'(x_n) + dhy'(x_{n+1}).$$

Expandiendo en serie de Taylor los términos de $y(x_n), y(x_{n+1}), y'(x_n), y'(x_{n+1})$ y haciendo $t = (x - x_n)/(x_{n+1} - x_n)$, se obtiene que

$$a = 2t^3 - 3t^2 + 1,$$

$$b = 3t^2 - 2t^3,$$

$$c = t(1 - t)^2,$$

$$d = t^2(t - 1).$$

Con los valores anteriores, se podrá realizar la interpolación de la solución en el punto final del intervalo de integración.

6. **Resumen implementación:** En este proceso se imprimen los siguientes valores que sirven para caracterizar el algoritmo

- a) Error en el punto final.
- b) *scd* (*significant correct digits*) representa el número mínimo de cifras significativas correctas de la solución numérica en el punto final x , es decir,

$$scd = -\log_{10}(\|\text{error en } x\|_{\infty}).$$

- c) Número de iteraciones totales.
- d) Número de iteraciones rechazadas.
- e) Número de descomposiciones LU realizadas para resolver el sistema implícito resultante en cada iteración.
- f) Número de evaluaciones del Jacobiano.
- g) Número de evaluaciones del sistema de ecuaciones.

5. Resultados numéricos

En esta sección se mostrarán los resultados obtenidos al resolver distintos sistemas de ecuaciones diferenciales tipo *Stiff* utilizando un DIMSIM de orden $p = q = 4$ y $s = r = 5$, además de presentar el resumen de la implementación de estos problemas utilizando los solvers `ode45` y `ode15s` del software Matlab.

Observación 5.1. 1. En todas las soluciones de la implementación del DIMSIM de orden $p = q = 4$ y $s = r = 5$, se utilizaron los valores de $tol = h = 10^{-3}$.

2. Si denotamos como \hat{y} a la aproximación de la solución y en el punto final del intervalo, entonces el error estará dado por

$$\|E\|_{rel} = \max_{\hat{y} \neq 0} \frac{|\hat{y} - y|}{\hat{y}}.$$

3. Se utilizó la versión R2020a del software Matlab.

5.1. Oscilador de Van der Pol

Este problema corresponde a la siguiente ecuación diferencial de segundo orden

$$\mu(1 - z^2)z' - z = z'', \quad \mu > 0, \quad (5.1)$$

donde $z(0) = z_0$, $z'(0) = z'_0$, la cual describe las oscilaciones en un circuito eléctrico de válvulas termoiónicas (Jackiewicz, 2009; Mazzia e Iavernaro, 2003). La ecuación de segundo orden puede ser escrita como un sistema de ecuaciones de primer orden de la siguiente manera

$$\begin{cases} y_1' = y_2, \\ y_2' = ((1 - y_1^2)y_2 - y_1)/k, \end{cases} \quad (5.2)$$

con $k = 10^{-6}$, $y_0 = (2, 0)$, $t \in [0, 2]$.

Solución numérica

En la Tabla 3 se presenta la solución numérica en el punto $t = 2$ hallada con el DIMSIM antes mencionado, junto con la solución de referencia del problema obtenida de Mazzia e Iavernaro (2003).

Tabla 3: Solución numérica y solución de referencia del Oscilador de Van der Pol en $t = 2$

y_i	Solución numérica en $t = 2$	Solución de referencia en $t = 2$.
y_1	1.7061677321704829	1.706167732170483
y_2	-0.89280970102479751	-0.8928097010247975

Se puede observar que la solución numérica se aproxima bastante a la solución de referencia conocida.

A continuación, en la Figura 2 se presenta la gráfica de la solución numérica obtenida en el intervalo de integración.

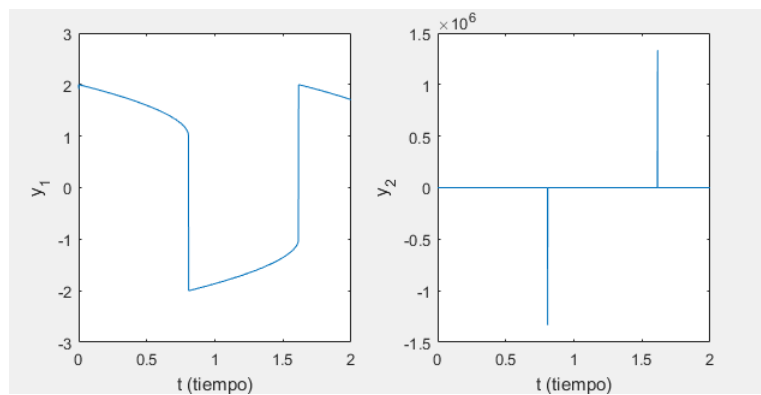


Figura 2: Solución Numérica Oscilador de Van Der Pol usando DIMSIM de orden $p = q = 4$ y $s = r = 5$.

Comparación

En la Tabla 4 se presenta el resumen de la implementación del problema del Oscilador de Van der Pol utilizando el DIMSIM antes mencionado y los solvers `ode45` y `ode15s` del software Matlab. Esta tabla incluye el error (mencionado en 5.1), el número de iteraciones totales y el tiempo de ejecución de cada método para hallar la solución.

Tabla 4: Resumen resultados de varios métodos al solucionar el problema del Oscilador de Van der Pol.

Solver	Error	Iteraciones totales	Tiempo de ejecución
DIMSIM	0.000000225109884	202104	1.61563003 s
<code>ode45</code>	0.002247249818733	4631969	53.9140700 s
<code>ode15s</code>	0.001427237268948	431	0.40160100 s

En este problema en particular notamos que el DIMSIM realizó más iteraciones que si se utilizara el paso fijo $h = 10^{-3}$, esto debido al comportamiento del sistema. Se puede

observar que, a pesar de haber realizado un número mayor de iteraciones, tuvo el menor error en el punto final del intervalo. En cuanto a tiempo de ejecución, el solver más lento fue el `ode45` que tardó casi un minuto en encontrar una solución al problema. A continuación se presentan las gráficas de las soluciones numéricas halladas con `ode45` y `ode15s`.

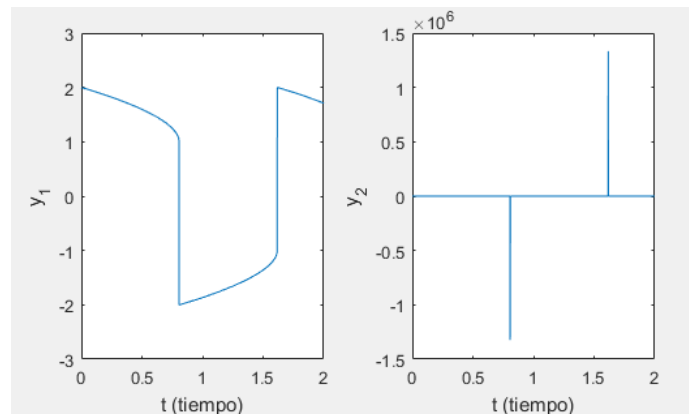


Figura 3: Solución del problema del Oscilador de Van der Pol utilizando `ode45`.

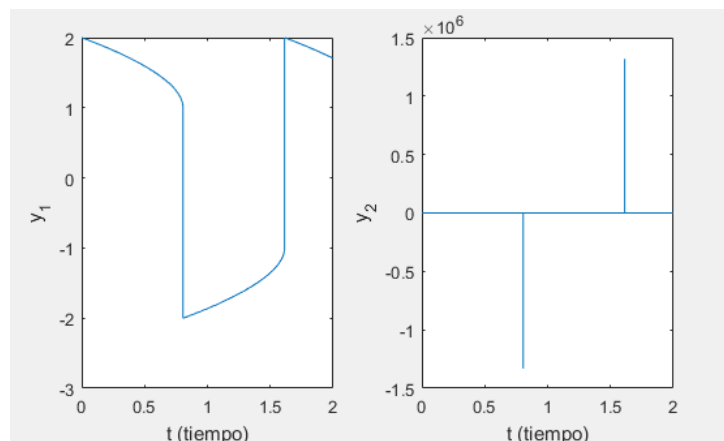


Figura 4: Solución del problema del Oscilador de Van der Pol utilizando `ode15s`.

Observando las gráficas de las Figuras 3 y 4 se puede notar que el comportamiento de las soluciones halladas por `ode45` y `ode15s` son bastante parecidas a la encontrada por el DIMSIM (Figura 2).

A pesar de que los tres solver resolvieron el problema del Oscilador de Van der Pol, por los resultados presentados en la Tabla 4 y la Figura 4, se recomienda utilizar el solver `ode15s` para solucionar este problema.

5.2. Problema de OREGO

Este sistema de ecuaciones diferenciales ordinarias describe la reacción de Belusov-Zhabotinskii la cual modela las oscilaciones que se presentan en el ácido malónico en un medio ácido por medio de bromatos (Mazzia e Iavernaro, 2003). A continuación, se presenta el sistema

de ecuaciones que describe este problema

$$\begin{cases} y_1' = 77.27(y_2 + y_1(1 - 8.375 \times 10^{-6}y_1 - y_2)), \\ y_2' = \frac{1}{77.27}(y_3 - (1 + y_1)y_2), \\ y_3' = 0.161(y_1 - y_2), \end{cases} \quad (5.3)$$

con $y_0 = (1, 2, 3)^T$, $t \in [0, 360]$ (Jackiewicz, 2009).

Solución numérica

En la Tabla 5 se presenta la solución numérica en el punto $t = 360$ hallada con el DIMSIM antes mencionado, junto con la solución de referencia del problema obtenida de Mazzia e Iavernaro (2003).

Tabla 5: Solución numérica y solución de referencia del problema de OREGO en $t = 360$

y_i	Solución numérica en $t = 360$	Solución de referencia en $t = 360$
y_1	1.0008148688420000	1.00081487031852
y_2	1228.1807448950001	1228.17852154988
y_3	132.05683398389999	132.055494284651

Seguidamente, en la Figura 5, se presenta la gráfica de la solución numérica obtenida en el intervalo de integración.

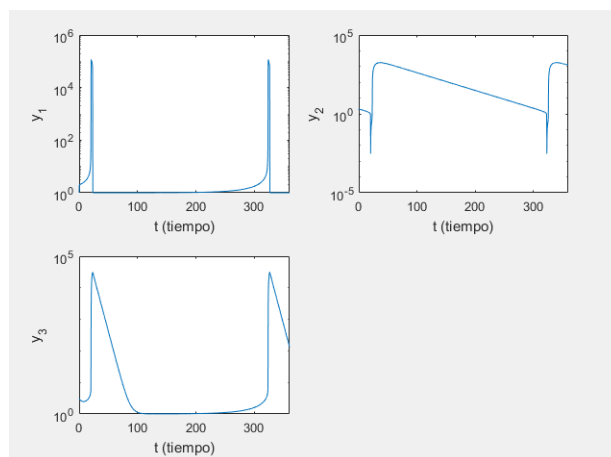


Figura 5: Solución Numérica Oscilador del problema de OREGO usando DIMSIM de orden $p = q = 4$ y $s = r = 5$.

Comparación

En la Tabla 6 se presenta el resumen de la implementación del código a este problema utilizando el DIMSIM antes mencionado y los solvers `ode45` y `ode15s` del software Matlab. De igual forma, ésta tabla incluye el error (mencionado en 5.1), el número de iteraciones totales y el tiempo de ejecución del código para hallar la solución.

Tabla 6: Resumen resultados de varios métodos al solucionar el problema de OREGO.

Solver	Error	Iteraciones totales	Tiempo de ejecución
DIMSIM	$1.193361856000000 \times 10^{-7}$	143026	1.89863896 s
<code>ode45</code>	$7.879608107408603 \times 10^{-5}$	13580769	292.014364 s
<code>ode15s</code>	$1.974911335578400 \times 10^{-2}$	539	0.35330500 s

Si se hubiera utilizado un método con paso fijo h para resolver el problema de OREGO, al hacer $h = 10^{-3}$, se hubieran obtenido un total de 360000 iteraciones. Por la Tabla 6 sabemos que el DIMSIM realizó un total de 143026, es decir, 216974 iteraciones menos de las esperadas. De igual forma que en el problema anterior, el DIMSIM tuvo el menor error en el punto final del intervalo, superando notablemente el error del `ode15s`. El solver que más tardó en encontrar la solución fue `ode45`, aproximadamente 291.661059 s, lo cual es bastante alto teniendo en cuenta que los otros dos solvers tardaron menos de 2 s cada uno. A continuación se presentan las gráficas de las soluciones numéricas halladas con `ode45` y `ode15s`.

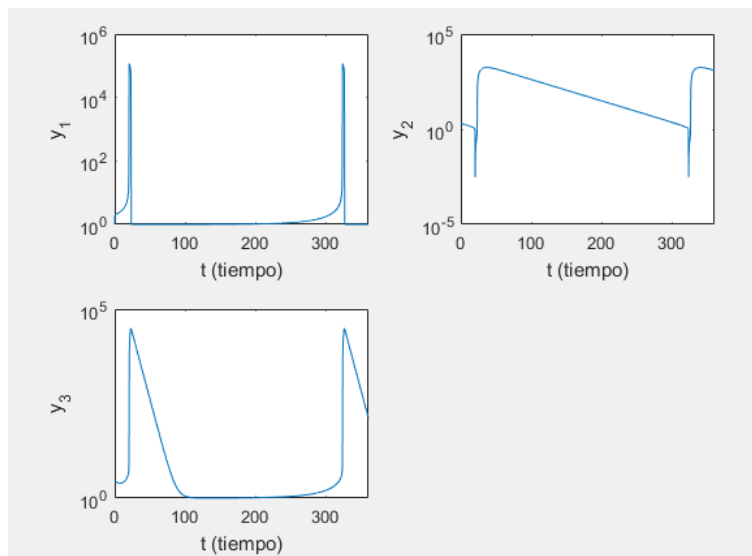


Figura 6: Solución del problema de OREGO utilizando `ode45`.

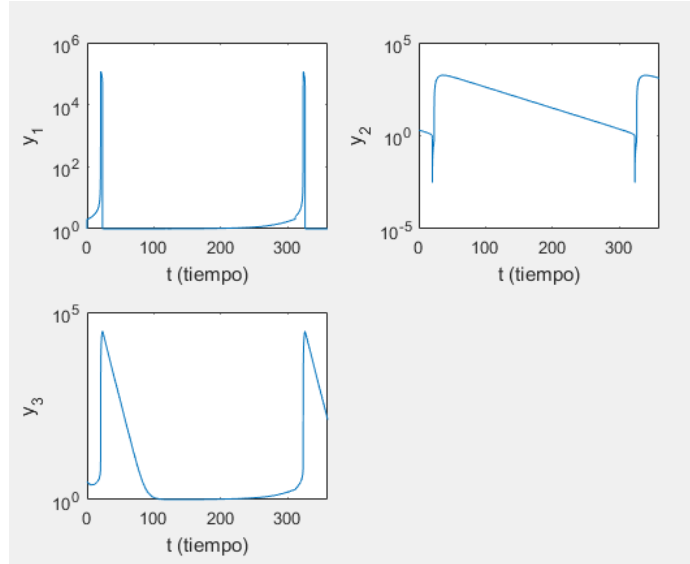


Figura 7: Solución del problema de OREGO utilizando `ode15s`.

Observando las gráficas de las Figuras 6 y 7 puede notar que el comportamiento de las soluciones halladas por `ode45` y `ode15s` son bastante parecidas a la encontrada por el DIMSIM (Figura 5).

Aunque los tres solver resolvieron el problema de OREGO, por los resultados presentados en la Tabla 6 y la Figura 7, se recomienda utilizar el solver `ode15s` para solucionar este problema.

5.3. Problema de Robertson

Este sistema describe la concentración de 3 reactivos en una reacción autocatalítica, fue dada por Robertson en 1966 y está descrita por el siguiente sistema de ecuaciones diferenciales ordinarias (Amat *et al.*, 2019)

$$\begin{cases} y_1' = -0.04y_1 + 10^4y_2y_3, \\ y_2' = 0.04y_1 - 10^4y_2y_3 - 3 \times 10^7y_2^2, \\ y_3' = 3 \times 10^7y_2^2, \end{cases} \quad (5.4)$$

con $y_0 = (1, 0, 0)^T$, $t \in [0, 40]$.

Solución numérica

En la Tabla 7 se presenta la solución numérica en el punto $t = 360$ hallada con el DIMSIM antes mencionado, junto con la solución de referencia del problema obtenida de (Amat *et al.*, 2019).

Tabla 7: Solución numérica y solución de referencia del problema de Robertson en $t = 40$

y_i	Solución numérica en $t = 40$	Solución de referencia en $t = 40$
y_1	0.72524595260620117	0.72524595
y_2	0.00000932999955693	0.00000933
y_3	0.28331312537193298	0.28331313

A continuación, en la Figura 8 se presenta la gráfica de la solución numérica obtenida en el intervalo de integración.

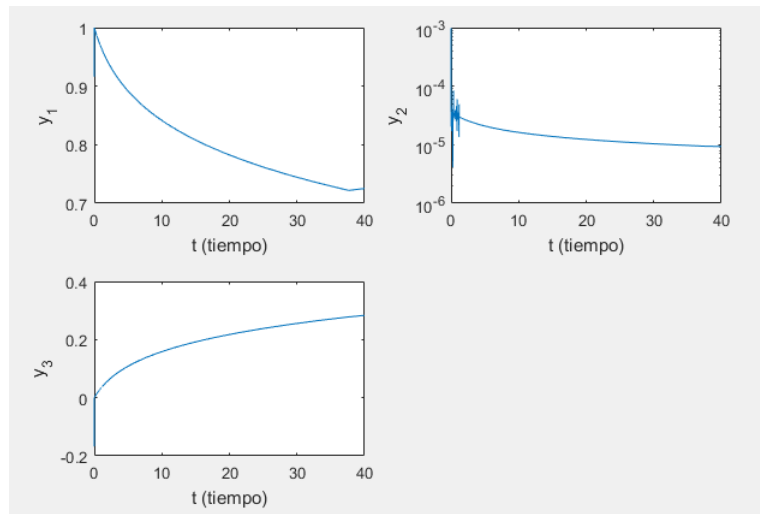


Figura 8: Solución Numérica problema de Robertson usando DIMSIM de orden $p = q = 4$ y $s = r = 5$.

Observando el comportamiento de y_2 en la Figura 8, se puede notar que en las primeras iteraciones realizó algunas oscilaciones, esto ocurre por las variaciones causadas por ser un sistema *Stiff*. A pesar de esta dificultad, el método se estabiliza en las iteraciones posteriores y se aproxima considerablemente a la solución de referencia.

Comparación

En la Tabla 7 se presenta el resumen de la implementación del código a este problema utilizando el DIMSIM antes mencionado y los solvers `ode45` y `ode15s` del software Matlab. De igual forma, ésta tabla incluye el error (mencionado en 5.1), el número de iteraciones totales y el tiempo de ejecución del código para hallar la solución.

Tabla 8: Resumen resultados de varios métodos al solucionar el problema de Robertson.

Solver	Error	Iteraciones totales	Tiempo de ejecución
DIMSIM	0.0157037347900000	557	0.006022999 s
ode45	0.0944734453694130	141413	2.132995000 s
ode15s	10.601023295086355	156	0.171542000 s

Si hubieramos utilizados un método con paso fijo h para resolver el problema de Robertson, al hacer $h = 10^{-3}$, se hubieran obtenido un total de 40000 iteraciones. Por la Tabla 8 sabemos que el DIMSIM realizó un total de 557, es decir, 39443 iteraciones menos de las esperadas. En este problema, podemos observar que el tiempo de ejecución de todos los solvers fue bastante bajo, siendo el menor de todos el DIMSIM. Antes de mencionar los resultados obtenidos del error en el punto $t = 40$, observemos las gráficas de las soluciones encontradas por `ode45` y `ode15s`.

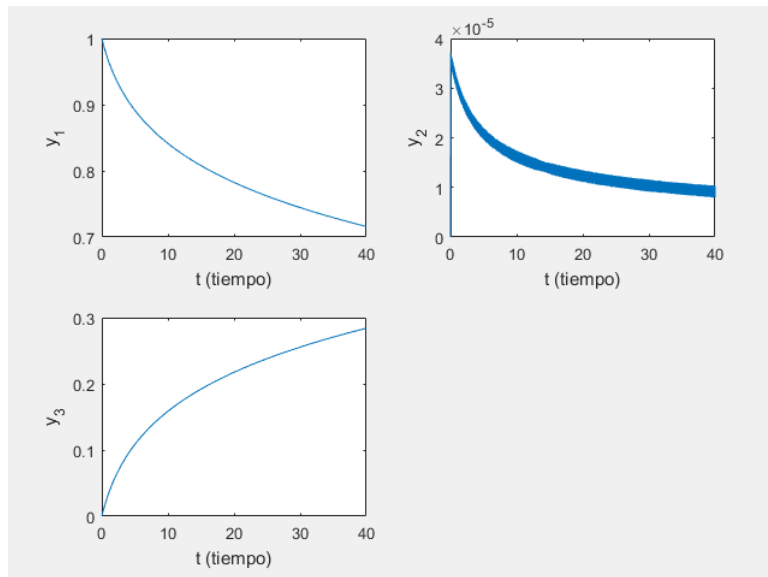


Figura 9: Solución del problema de Robertson utilizando `ode45`.

Observando la solución hallada por `ode45` notamos que la gráfica de la solución del componente y_2 es una línea más gruesa, esto es causado porque el método no es estable y realizó oscilaciones en todo el intervalo de integración (Ver Figura 10). Para las otras dos componentes, y_1 y y_3 , su comportamiento es parecido a la solución hallada por el DIMSIM que se puede observar en la Figura 8.

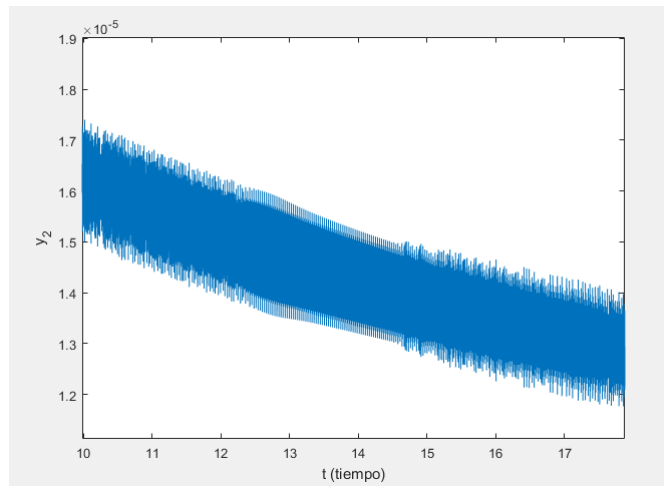


Figura 10: Ampliación a componente y_2 de la solución del problema de Robertson hallada con `ode45`.

A continuación se presentan las gráficas de la solución hallada por `ode15s`.

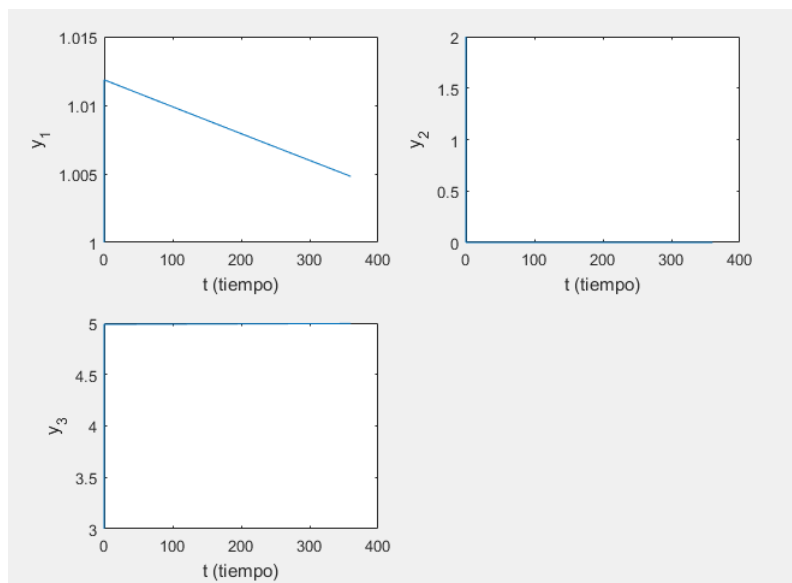


Figura 11: Solución del problema de Robertson utilizando `ode15s`.

Observando los componentes y_2 y y_3 de la solución hallada por `ode15s` notamos que sus gráficas parecen líneas constantes cercanas a los valores de 0 y 5 respectivamente, esto es causado debido a que el *solver* no describe de manera eficiente las variaciones de este sistema *Stiff*. De igual forma, el comportamiento del componente y_1 es bastante diferente al descrito en la solución hallada por el DIMSIM en la Figura 8, lo cual, teniendo en cuenta la magnitud de su error en el punto $t = 40$, que podemos ver en la Tabla 8, nos indica que el *solver* `ode15s` no resolvió el problema de Robertson de manera eficiente. Lo anterior indica que sólo el DIMSIM pudo encontrar la solución de este problema.

5.4. Problema de HIRES

Este sistema de ecuaciones diferenciales ordinarias está conformado por 8 ecuaciones diferenciales que modelan la morfogénesis, proceso por el cual los órganos de las plantas adquieren formas distintivas que les permitirán desarrollar funciones específicas (Jackiewicz, 2009; Mazzia e Iavernaro, 2003). A continuación se presenta el conjunto de ecuaciones

$$\left\{ \begin{array}{l} y'_1 = -1.71y_1 + 0.43y_2 + 8.32y_3 + 0.0007, \\ y'_2 = 1.71y_1 - 8.75y_2, \\ y'_3 = -10.03y_3 + 0.43y_4 + 0.035y_5, \\ y'_4 = 8.32y_2 + 1.71y_3 - 1.12y_4, \\ y'_5 = -1.745y_5 + 0.43y_6 + 0.43y_7, \\ y'_6 = -280y_6y_8 + 0.69y_4 + 1.71y_5 - 0.43y_6 + 0.69y_7, \\ y'_7 = 280y_6y_8 - 1.81y_7, \\ y'_8 = -280y_6y_8 + 1.81y_7, \end{array} \right.$$

con $y_0 = (1, 0, 0, 0, 0, 0, 0, 0.0057)^T$, $t \in [0, 321.8122]$.

Solución numérica

En la Tabla 9 se presenta la solución numérica en el punto $t = 321.8122$ hallada con el DIMSIM antes mencionado, junto con la solución de referencia del problema obtenida de Mazzia e Iavernaro (2003).

Tabla 9: Solución numérica y solución de referencia del problema de HIRES en $t = 321.8122$

y_i	Solución numérica en $t = 321.8122$	Solución de referencia en $t = 321.8122$
y_1	$0.73713125733256685 \times 10^{-3}$	$0.7371312573325668 \times 10^{-3}$
y_2	$0.14424857263161851 \times 10^{-3}$	$0.1442485726316185 \times 10^{-3}$
y_3	$0.58887297409675752 \times 10^{-4}$	$0.5888729740967575 \times 10^{-4}$
y_4	$0.11756513432831491 \times 10^{-2}$	$0.1175651343283149 \times 10^{-2}$
y_5	$0.23863561988313308 \times 10^{-2}$	$0.2386356198831331 \times 10^{-2}$
y_6	$0.62389682527427964 \times 10^{-2}$	$0.6238968252742796 \times 10^{-2}$
y_7	$0.28499983951857689 \times 10^{-2}$	$0.2849998395185769 \times 10^{-2}$
y_8	$0.28500016048142308 \times 10^{-2}$	$0.2850001604814231 \times 10^{-2}$

A continuación, en la Figura 12 se muestra la gráfica de la solución numérica obtenida en el intervalo de integración.

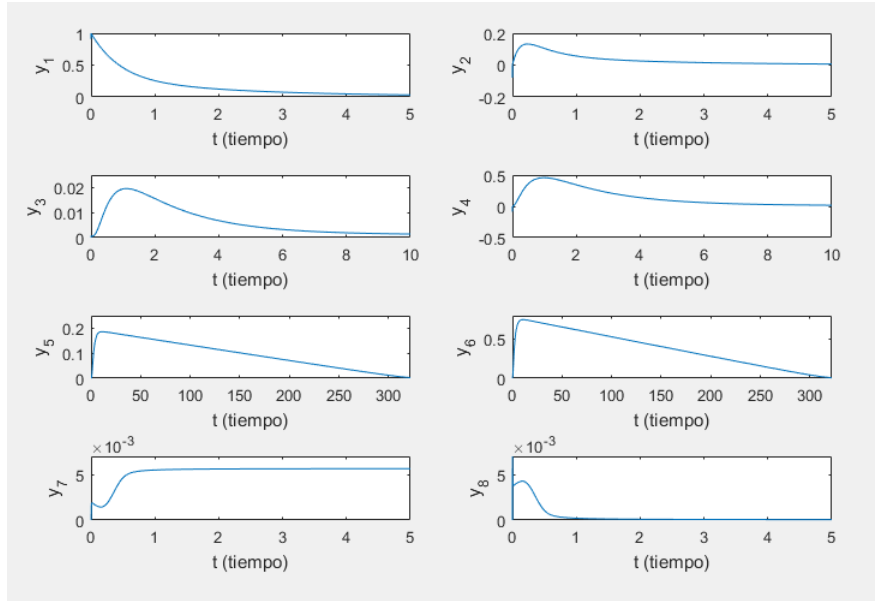


Figura 12: Solución Numérica problema de HIRES usando DIMSIM de orden $p = q = 4$ y $s = r = 5$.

Observación 5.2. No todos los componentes y_i de la solución fueron graficadas en el intervalo $[0, 321.8122]$, debido a que luego de $t = 5$ o $t = 10$ decaen y se mantienen muy cercanas a cero lo cual dificulta percibir gráficamente su comportamiento.

Comparación

En la Tabla 10 se muestra el resumen de la implementación del código a este problema utilizando el DIMSIM antes mencionado y los solvers `ode45` y `ode15s` del software Matlab. De igual forma, esta tabla incluye el error (mencionado en 5.1), el número de iteraciones totales y el tiempo de ejecución del código para hallar la solución.

Tabla 10: Resumen implementación de DIMSIM de orden de orden $p = q = 4$ y $s = r = 5$ al problema de HIRES.

Solver	Error	Iteraciones totales	Tiempo de ejecución
DIMSIM	0.008179834018000	90	0.002056 s
<code>ode45</code>	0.001318115504546	41521	0.557389 s
<code>ode15s</code>	0.006751333881008	119	0.406032 s

Se puede notar que el DIMSIM es el solver que realizó menos iteraciones en menor tiempo. El solver `ode45` fue el que presentó el menor error en el punto final del intervalo

$t = 321.8122$, siendo este el que realizó el mayor número de iteraciones totales. En general, los tres solvers, tuvieron un buen desempeño en el cálculo de la solución de este problema. A continuación se presentan las gráficas de las soluciones numéricas halladas con `ode45` y `ode15s`.

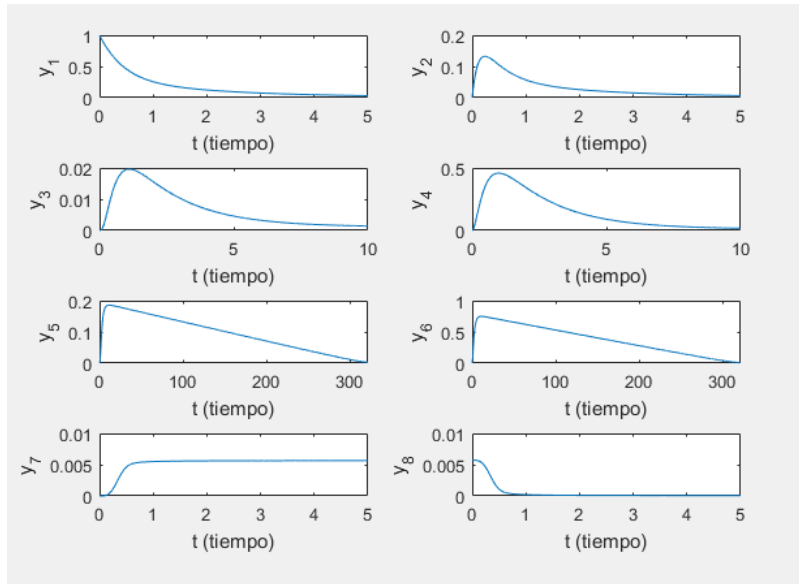


Figura 13: Solución del problema de HIRES utilizando `ode45`.

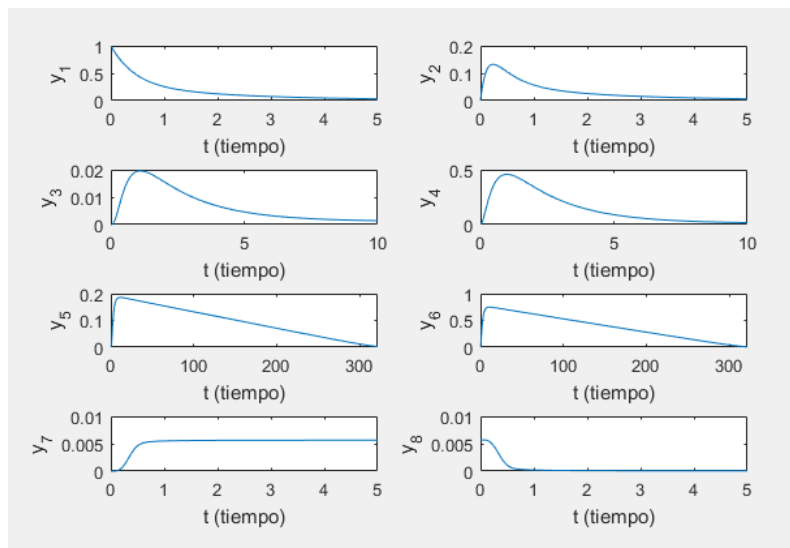


Figura 14: Solución del problema de HIRES utilizando `ode15s`.

Observando las gráficas de las Figuras 13 y 14 se puede notar que el comportamiento de las soluciones halladas por `ode45` y `ode15s` son bastante parecidas a la encontrada por el DIMSIM (Figura 12).

A pesar de que los tres solvers resolvieron el problema de HIRES, por los resultados presentados en la Tabla 10 y la Figura 12, se recomienda utilizar el DIMSIM para solucionar este problema.

6. Conclusión

El propósito principal de este trabajo era encontrar la solución de algunos sistemas de ecuaciones diferenciales tipo *Stiff* utilizando un tipo de GLM y comparar estos resultados con las soluciones halladas en la implementación de dos solvers del software Matlab. Luego de presentar las gráficas de las soluciones encontradas con el DIMSIM y las tablas comparativas que mostraban el resumen de la ejecución de cada *solver*, podemos plantear las siguientes conclusiones y observaciones

1. A pesar de que los GLM realizan el cambio del paso h en cada iteración con el propósito de reducir el número de iteraciones totales, se pueden presentar problemas, como el Oscilador de Van der Pol, donde el método realiza muchas más iteraciones de las esperadas si se trabajara con un paso h fijo.
2. El solver `ode15s` tuvo mejores resultados que el DIMSIM en los aspectos de iteraciones totales y tiempo de ejecución en los problemas del Oscilador de Van der Pol y de OREGO, esto se puede explicar debido a que este utiliza métodos de orden variable (orden 1 a orden 5) lo cual favorece su rendimiento en algunos problemas.
3. Teniendo en cuenta que el solver `ode45` no fue diseñado para resolver sistemas de ecuaciones diferenciales ordinarias tipo *Stiff*, pudimos observar que, aunque puede ser más demorado (puede tardar hasta 291.661059 s más que el método DIMSIM y el solver `ode15s`), es igualmente capaz de encontrar la solución de algunos sistemas de este tipo.
4. Una diferencia notable entre el DIMSIM utilizado y los solvers `ode45` y `ode15s`, es que estos últimos no necesitan que el usuario proporcione el Jacobiano del sistema, ellos lo aproximan internamente, lo cual puede ser una ventaja en sistemas con un gran número de ecuaciones debido a que calcular su Jacobiano analíticamente puede ser un proceso costoso computacionalmente.

En general, pudimos observar que, aunque el DIMSIM redujo en la mayoría de casos el número de iteraciones totales, en algunos problemas puede llegar a realizar más de las iteraciones esperadas en un tiempo de ejecución más alto que el solver `ode15s`. A pesar de lo anteriormente mencionado, el DIMSIM fue el único de los tres solvers que pudo hallar la solución numérica de los cuatro problemas estudiados, solo teniendo algunas dificultades en las primeras iteraciones del problema de Robertson. Se hace necesario seguir experimentando con otro tipo de problemas más demandantes que tengan un mayor número de ecuaciones o que su solución esté definida en un intervalo más grande para evaluar el desempeño de este tipo de métodos.

Bibliografía

- Alexander, R. (1991). The modified newton method in the solution of stiff ordinary differential equations. *Mathematics of Computation*, 57(196):673–701.
- Amat, S., Legaz, M. J., y Ruiz-Álvarez, J. (2019). On a variational method for stiff differential equations arising from chemistry kinetics. *Mathematics*, 7(5):459.
- Butcher, J. (1966). On the convergence of numerical solutions to ordinary differential equations. *Mathematics of Computation*, 20(93):1–10.
- Butcher, J. (2001). General linear methods for stiff differential equations. *BIT Numerical Mathematics*.
- Butcher, J. C. (2016). *Numerical Methods For Ordinary Differential Equations*. Wiley.
- Curtiss, C. F. y Hirschfelder, J. O. (1952). Integration of stiff equations.
- Dahlquist G., B. A. (2003). *Numerical Methods*.
- Garfinkel, D., Marbach, C., y Shapiro, N. (1977). Stiff differential equations. *Annual Review of Biophysics*.
- Gear, C. W. (1980). Runge-kutta starters for multistep methods. *ACM Trans. Math. Softw.*, 6(3):263–279.
- Huang, S. Y. (2005). *Implementation of General Linear Methods for Stiff Ordinary Differential Equations*. Tesis doctoral, University of Auckland.
- Jackiewicz, Z. (2009). *General linear methods for ordinary differential equations*. Wiley.
- Lambert, J. D. (1991). *Numerical Methods for Ordinary Differential Systems*. John Wiley and Sons.
- Mazzia, F. e Iavernaro, F. (2003). *Test Set for Initial Value Problem Solvers*. Università degli studi di Bari.
- Mokhtary, P. y Hosseini, S. M. (2008). Some implementation aspects of the general linear methods withinherent runge-kutta stability. *Iranian Journal of Mathematical Sciences and Informatics*, 3:63–76.
- Strang, G. y Aarikka, K. (1986). *Introduction to Applied Mathematics*. Wellesley-Cambridge Press.

Venkateshan, S. y Swaminathan, P. (2014). *Computational Methods in Engineering*. Academic Press.

Wright, W. (2002). *General linear methods with inherent Runge-Kutta stability*. Thesis doctoral, University of Auckland.