

00365



UNIVERSIDAD NACIONAL  
AUTONOMA DE MEXICO

*Juj*

FACULTAD DE CIENCIAS

División de Estudios de Posgrado

METODOS DE INTEGRACION MULTITASA  
PARA SIMULACION DE PROCESOS EN  
TIEMPO REAL

T E S I S  
Que para obtener el Grado Académico de  
MAESTRO EN CIENCIAS (MATEMATICAS)

p r e s e n t a

GUSTAVO RODRIGUEZ GOMEZ

258081

Director de Tesis: Dr. Jesús López Estrada

TESIS CON  
FALLA DE ORIGEN

1998



Universidad Nacional  
Autónoma de México



## **UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso**

### **DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

# INDICE

INTRODUCCION .....	1
CAPITULO I <i>SIMULADORES PARA ENTRENAMIENTO</i> .....	I-1
§ I.1 OBJETIVO DE LA SIMULACION .....	I-1
§ I.2 DESCRIPCION DE UN SIMULADOR .....	I-1
§ I.3 FUNCIONAMIENTO DE UN SIMULADOR .....	I-7
CAPITULO II <i>SIMULACION EN TIEMPO REAL</i> .....	II-1
§ II.1 SIMULACION .....	II-1
§ II.1.1 <i>Tiempo Real</i> .....	II-3
§ II.1.2 <i>Operación en Tiempo Real</i> .....	II-4
§ II.1.3 <i>Integración Numérica en Tiempo Real</i> .....	II-5
§ II.1.4 <i>Control del Error en Tiempo Real</i> .....	II-7
§ II.2 PROBLEMAS NUMERICOS EN TIEMPO REAL .....	II-8
§ II.2.1 <i>Problemas Físicos que Originan Dificultades Numéricas</i> .....	II-8
§ II.2.2 <i>Problemas en la Actualización de la Información que Originan Dificultades Numéricas</i> .....	II-9
§ II.2.3 <i>Ecuaciones con Discontinuidades</i> .....	II-9
§ II.2.4 <i>Retrasos en el Flujo de la Información</i> .....	II-12
§ II.3 ESTRATEGIA DE SOLUCION EN TIEMPO REAL .....	II-13
§ II.3.1 <i>Solución Modular Secuencial</i> .....	II-13
§ II.4 EJEMPLOS .....	II-15
§ II.4.1 <i>Primer Modelo: Tres Reactores Tipo Tanque con Control PI</i> .....	II-15
§ II.4.2 <i>Control de Presión en una Tubería</i> .....	II-17
§ II.4.3 <i>Simulaciones de Transitorios con Método Integrador de Orden y Paso Variable</i> .....	II-19
CAPITULO III <i>METODOS MULTITASA</i> .....	III-1
§ III.1 INTEGRACION DE SISTEMAS DINAMICOS .....	III-1
§ III.2 METODOS DE INTEGRACION MULTITASA .....	III-3
§ III.3 DESCRIPCION DE LOS METODOS MULTITASA .....	III-3
§ III.4 APLICABILIDAD DE LOS METODOS MULTITASA .....	III-5
§ III.4.1 <i>Sistemas con Dos Escalas Temporales: Caso Lineal Homogéneo</i> .....	III-6
§ III.4.2 <i>Sistemas con Dos Escalas Temporales: Caso Lineal no Homogéneo</i> .....	III-18
§ III.4.3 <i>Sistemas con Dos escalas Temporales: Caso no Lineal</i> .....	III-20
§ III.5 ANALISIS DE LOS ALGORITMOS .....	III-20
§ III.5.1 <i>Análisis del Número de Operaciones de los Algoritmos</i> .....	III-25

CAPITULO IV. ANALISIS DE ERROR LOCAL EN METODOS DE INTEGRACION MULTITASA .....	IV-1
§ IV.1 ERROR DE TRUNCAMIENTO LOCAL PARA MIM's .....	IV-1
§ IV.1.1 Error de Truncamiento Local Algoritmos III.5.1, III.5.2, III.5.3 .....	IV-12
§ IV.1.2 Error de Truncamiento Local Algoritmos III.5.4, III.5.5, III.5.6 .....	IV-22
§ IV.1.3 Análisis de Error de Truncamiento Local para un Sistema Lineal .....	IV-29
§ IV.2 DISCUSION DE LOS ERRORES .....	IV-35
CAPITULO V. ESTABILIDAD ABSOLUTA EN METODOS DE INTEGRACION MULTITASA .....	V-1
§ V.1 ESTABILIDAD ABSOLUTA EN METODOS MULTIPASOS LINEALES ...	V-2
§ V.2 ESTABILIDAD ABSOLUTA EN METODOS MULTITASA .....	V-8
§ V.3 DISCUSION DE LOS POLINOMIOS DE ESTABILIDAD .....	V-12
CAPITULO VI. EXPERIMENTOS NUMERICOS .....	VI-1
§ VI.1 CONSIDERACIONES DE RAPIDEZ .....	VI-1
§ VI.2 EXPERIMENTOS NUMERICOS .....	VI-3
§ VI.2.1 Primer Experimento .....	VI-3
§ VI.2.2 Segundo Experimento: Aplicación de los MIM's al Modelo Simplificado de una Planta Termoeléctrica .....	VI-9
§ VI.3 TABLAS .....	VI-26
CONCLUSIONES .....	C-1
BIBLIOGRAFIA .....	B-1

A la memoria de mis dos madres y de mi padre

Piedad e Isabel Rodríguez Rivero

Gustavo Rodríguez Rivero

quienes no vivieron lo suficiente para ver esto, ya que me tarde demasiado

Con especial cariño a mi  
esposa por su paciencia y mis hijos

Silvia M. Rodríguez Véjar  
Criseida Rodríguez Rodríguez  
Gustavo Rodríguez Rodríguez

## **Agradecimientos**

La idea de escribir la experiencia tecnológica de haber participado como matemático en el desarrollo construcción y puesta en servicio del primer simulador de alcance total de una planta termoeléctrica hecho en Latino América fue concebida en pláticas con el finado Dr. Diego Bricio H. a quién agradezco sus estímulos y comentarios que dieron el inicio a éste trabajo.

Agradezco al Instituto de Investigaciones Eléctricas el haberme brindado la oportunidad de encontrar un área en donde aplicar las matemáticas a problemas tecnológicos de mi país.

Agradezco a mi tutor de la maestría el Prof. José A. Canavati Ayub su apoyo y orientación brindada tanto a nivel licenciatura como en la maestría.

Finalmente quiero agradecer al Dr. Jesús López Estrada su colaboración y apoyo brindado para culminar el presente trabajo.

# Introducción

Los simuladores de procesos han sido reconocidos desde hace muchos años como una herramienta de enseñanza de gran utilidad en el entrenamiento de operadores de equipos complejos. Estos fueron introducidos por la industria aeronáutica, después de la primera guerra mundial, en la actualidad se producen simuladores de equipos tan diversos como locomotoras, sistemas de transporte colectivo, vehículos militares, plantas generadoras de energía eléctrica.

En el Departamento de Simulación del Instituto de Investigaciones Eléctricas (IIE) se trabaja en la construcción de simuladores y emuladores para entrenamiento de operadores principalmente de centrales generadoras de energía eléctrica. Estos están orientados para la capacitación de personal en el manejo de la planta, y para pruebas de maquinaria o equipo eléctrico.

A la fecha se han construido los siguientes simuladores:

1. Simulador de alcance total de la Unidad 1 (300 MW) de la Central Termoeléctrica de Tula (1984).
2. Simulador de alcance total del Sistema de Transporte Colectivo (1990).
3. Simulador de alcance total de la Unidad 1 (660 MW) de la Central Nucleoeléctrica de Laguna Verde (1991).
4. Simulador parcial de alcance total para el rodado de turbina de Centrales Termoeléctricas (1992).

5. Emulador de máquinas síncronas para pruebas de Reguladores Automáticos de Tensión (1995).

Los modelos matemáticos que representan a los procesos y equipos que conforman a una planta de potencia están constituidos por sistemas de ecuaciones algebraico-diferenciales cuyo número es muy grande. Desde el punto de vista del análisis numérico, en la construcción del primer simulador, el de la Central Termoeléctrica de Tula (Termos I), el principal problema fue encontrar las estrategias y algoritmos numéricos que permitieran aproximar la solución del sistema de ecuaciones algebraico-diferencial en tiempo real. Estos algoritmos en los años 80's no eran del dominio público ya que las empresas que construían simuladores para comercializarlos, no informaban que algoritmos usaban. Se puede afirmar que todos los simuladores construidos en el departamento de Simulación usan las estrategias y algoritmos que se implantaron en Termos I. Estos algoritmos se implantaron bajo ciertas consideraciones heurísticas sin haber realizado un análisis teórico para comprenderlos en forma más profunda.

**El objetivo de este trabajo es:**

Presentar en forma rigurosa los algoritmos de integración numérica utilizados para alcanzar las simulaciones en tiempo real, lo que incluye definir los conceptos de simulación de procesos en tiempo real, error y estabilidad asociadas a los algoritmos de integración llamados *Métodos de Integración Multitasa (MIM)*, así como caracterizar sus zonas de estabilidad absoluta, y analizar su desempeño en la simulación de plantas generadoras de energía eléctrica en tiempo real.



En el capítulo I *Simuladores para Entrenamiento* se presenta en forma breve los objetivos de la simulación y la descripción de un simulador.

En el capítulo II *Simulación en Tiempo Real* se definen los conceptos de simulación y operación en tiempo real, y se indican las diferencias con el de emulación, se plantean los problemas numéricos originados por los elementos modelados y se presenta la técnica numérica utilizada para alcanzar las simulaciones en tiempo real.

En el capítulo III *Métodos Multitasa* se presentan los Métodos de Integración Multitasa, se definen las condiciones de aplicabilidad y se hace un análisis de su costo computacional respecto a las técnicas convencionales.

En el capítulo IV *Análisis de Error Local en Métodos de Integración Multitasa* se desarrolla la teoría y se presentan los análisis de error de truncamiento local para los algoritmos definidos en el capítulo III, y se define el orden de estos.

En el capítulo V *Estabilidad Absoluta en Métodos de Integración Multitasa* se desarrolla la teoría de estabilidad absoluta para los Métodos de Integración Multitasa y se caracteriza sus zonas de estabilidad absoluta.

Finalmente en el capítulo VI *Experimentos Numéricos* se realizan dos experimentos numéricos para medir el desempeño de los algoritmos. El primero de ellos es un sistema de ecuaciones diferenciales ordinarias que permite evaluar los algoritmos bajo diferentes grados de acoplamiento. El segundo es un modelo simplificado de una planta generadora de energía eléctrica, éste permite evaluar el desempeño de los algoritmos de integración multitasa en ambientes de simulación para tiempo real.

# **Capítulo I**

## **Simuladores para Entrenamiento**

En éste primer capítulo se presenta el objetivo principal de los simuladores para entrenamiento y se da una breve descripción de su funcionamiento y sus principales características.

### **§ I.1 OBJETIVO DE LA SIMULACIÓN**

El objetivo de todos los simuladores de entrenamiento es reproducir fielmente el ambiente de trabajo de un operador, con el fin de familiarizarlo con el funcionamiento del equipo y adiestrarlo en su operación a base de ejercicios repetidos. El entrenamiento se basa en estímulos visuales, auditivos y motrices, por medio de los cuales el estudiante adquiere una habilidad específica. Por lo tanto, el éxito de un simulador depende, en gran medida, de la precisión con la que reproduce el aspecto y funcionamiento de los equipos simulados [4].

### **§ I.2 DESCRIPCIÓN DE UN SIMULADOR**

El local de un simulador en general se encuentra separado en tres secciones. En una se encuentran los tableros de control de la planta, donde se desarrollan las sesiones

de entrenamiento. La segunda corresponde a un cuarto comunicado con la sección de tableros por una ventana de vidrios oscuros donde se localiza una consola desde la cual el instructor controla la simulación. Por último, el cuarto de computadoras donde se encuentra el equipo de cómputo asociado.

Una parte fundamental del simulador son los modelos matemáticos que aproximan el estado de la planta dentro de cierta tolerancia de error, de acuerdo a un conjunto de condiciones iniciales y a las acciones tomadas en los tableros. Las salidas de estos modelos son tomadas por un sistema de entrefases que actualiza los instrumentos del tablero con una frecuencia superior a la que pueda distinguir el ojo humano, dando así la sensación de suavidad en los movimientos de los instrumentos. Además de simular el funcionamiento normal de la planta, se modelan una serie de fallas en los equipos más importantes, entre las que se cuentan roturas de tubos en la caldera, disparo de turbina y del generador y apagón, para lo cual la computadora controla al alumbrado del cuarto de tableros.

Los tableros, externamente, son una replica de los de una planta real, tanto en forma y tamaño, como en el funcionamiento de interruptores y botones. Internamente, sin embargo, todos los instrumentos analógicos han sido sustituidos por amperímetros. Es decir, si en la planta se tienen medidores de presión y temperatura, en el simulador se cambian por medidores de corriente, cuyo funcionamiento puede ser controlado fácilmente por la computadora. De esta manera, todos los medidores, alarmas, registradores y demás indicadores responden precisamente en la misma forma que se observa en la central generadora de energía eléctrica.

El instructor tiene a su disposición una pantalla y un teclado para diálogos, con los que se comunica con la computadora y controla la simulación. Varias pantallas adicionales le permiten desplegar la información necesaria para conocer el estado de la simulación, ya sea por medio de gráficas dinámicas o de listas de variables. La información más importante está resaltada con claves de colores para permitir su reconocimiento inmediato.

Una característica muy importante del simulador es la reproducción de los ruidos de la planta. Se ha reconocido que los estímulos auditivos producen reflejos condicionados que le permiten al operador reaccionar más rápidamente, sobre todo en situaciones de emergencia. En el simulador denominado Termos I se puede escuchar el ruido producido por el vapor al abrirse una válvula, por los motores de las bombas incluyendo cambios de velocidad durante el arranque o paro y la vibración del turbogenerador. Estas operaciones producen un ruido de fondo típico en las plantas reales.

En el cuarto de tableros se cuenta también con un sistema de adquisición de datos y registro de eventos (SADRE), que en las centrales modernas se emplea para recopilar y presentar al operador la información de los parámetros más importantes de la planta. Esta función normalmente la realiza una computadora a la que se alimenta con mediciones realizadas directamente en la planta. La computadora despliega esta información en pantallas a color en forma de gráficas, diagrama de barras, tablas y diagramas dinámicos que representan con claves de color a los sistemas de interés. En el simulador, esta computadora es sustituida por un

programa, tal como se hace con el resto de los equipos de la central. Este programa toma información de los modelos matemáticos y los despliega en formatos muy flexibles, de acuerdo con lo solicitado por el operador, por medio de un teclado especial.

El instructor dispone de un conjunto de funciones para realizar las sesiones de entrenamiento. A continuación se describen estos grupos de funciones:

### ***FUNCIONES DE CONTROL DE LA SIMULACIÓN***

1. **Empieza.** Alista el simulador para iniciar la simulación.
2. **Termina.** Da por terminado el período de entrenamiento.
3. **Simula.** Inicia la simulación. El instructor puede definir previamente la condición inicial empleando la función de estado, (ver funciones auxiliares).
4. **Auto.** Esta función ejecuta una simulación en forma completamente automática, de acuerdo con los comandos almacenados en un archivo.
5. **Repite.** Hace una repetición instantánea de la simulación, presentando estados del simulador almacenados en forma periódica.
6. **Paro.** Detiene la simulación momentáneamente. Esta se puede reiniciar con las instrucciones Simula, Auto o Repite.

### ***FUNCIONES DE ESCALA DE TIEMPO***

1. **Normal.** Se ejecutan los modelos y funciones del instructor en tiempo real.

2. **Rápido.** Se ejecutan algunos modelos y funciones del instructor diez veces más rápido que tiempo real.
3. **Lento.** Se ejecutan los modelos y funciones del instructor diez veces más lento que tiempo real.

El objetivo de la función **Rápido** es permitir un aprovechamiento más eficiente del simulador, agilizando el paso por los transitorios lentos de la planta. Esta función es particularmente útil durante el calentamiento de la caldera, cuando el operador debe esperar varias horas hasta que el sistema se estabilice. La función **Lento** se emplea para mostrar al operador alguna maniobra complicada que requiera un análisis más detallado del que se puede obtener con la escala de tiempo **Normal**.

### ***FUNCIONES DE DINÁMICA DE MÓDULOS***

1. **Fallas.** El objetivo de esta función es simular comportamientos anormales de la unidad. Una falla como la rotura de tubos puede ser degradable desde 0 hasta 100 por ciento.
2. **Alteraciones.** Esta función se emplea para alterar el valor de las variables de salida al tablero, lo cual permite simular instrumentos descompuestos o un foco fundido.
3. **Parámetros internos.** Permite al instructor realizar operaciones que normalmente llevan a cabo desde los tableros de control, tales como apertura de válvulas manuales, restablecimiento de interruptores, etc.

4. **Parámetros externos.** El objetivo de esta función es simular el efecto de agentes o fenómenos externos a la unidad, como son las variaciones en la temperatura y presión atmosférica, frecuencia en la red eléctrica.

### ***FUNCIONES AUXILIARES***

1. **Fotografiado.** Esta función permite obtener una copia del estado de la simulación en un momento dado. Una fotografía en este contexto se refiere a un archivo de datos donde se almacena la información de todas las variables que definen el estado del sistema, que puede ser etiquetado y almacenado en disco para emplearse posteriormente.
2. **Fotografiado periódico.** Obtiene una secuencia de fotografías a intervalos que pueden seleccionarse entre 15 segundos y 5 minutos.
3. **Estado.** Permite definir las condiciones iniciales del simulador, transfiriendo los valores almacenados en una fotografía a la memoria principal de la computadora.
4. **Información.** El objetivo de esta función es generar despliegues con datos sobre las fotografías existentes, el intervalo del fotografiado periódico, las fallas introducidas por el instructor, etc.
5. **Evaluación.** Por medio de esta función se obtiene un reporte en la impresora con las acciones efectuadas por el instructor y el operador, y los valores de las variable monitoreadas. Todas las acciones se reportan clasificadas por la hora a la que tuvieron lugar.

6. **Monitoreo.** Presenta al instructor, en las pantallas de su consola, el valor actual de las variables elegidas. Estos valores se actualizan cada dos segundos.
7. **Graficación.** Genera gráficas dinámicas de las variables elegidas, en pantallas a color. Es posible también obtener una copia en papel de cualquier imagen presentada en las pantallas de la consola del instructor.

En una sesión de entrenamiento típica, el instructor empieza por definir las condiciones iniciales y alista el simulador para iniciar el entrenamiento. Una vez iniciada la simulación, el instructor puede detenerla en cualquier momento para examinar con todo detalle el estado de la planta y analizar las acciones del operador. La simulación se puede continuar en el estado que se detuvo o en algún punto anterior, permitiendo la repetición instantánea de un cierto transitorio y el análisis de operaciones alternativas. Todas las acciones del operador y del instructor son registradas continuamente por la computadora para su evaluación posterior.

### **§ I.3 FUNCIONAMIENTO DEL SIMULADOR**

Para que un simulador cumpla plenamente con su objetivo de capacitar en forma práctica y eficiente a los operadores, debe cumplir con los siguientes requerimientos:

1. Simular en forma realista el funcionamiento de la unidad a partir de cualquier condición inicial.
2. Comenzar, detener y continuar la simulación en cualquier momento durante la sesión de entrenamiento.
3. Simular la ocurrencia de fallas del sistema.



4. Restablecer estados de operación en los que se encontró el simulador anteriormente y reiniciar su operación a partir de uno de dichos estados.
5. Efectuar en forma automática sesiones de simulación preprogramadas.
6. Facilitar el análisis del estado de la planta por medio de registros de información gráficos.
7. Registrar y facilitar el análisis de las acciones de los operadores.
8. Contar con los medios de diagnóstico y mantenimiento del simulador.

# Capítulo II

## Simulación en Tiempo Real

En éste capítulo se presentan los conceptos de simulación y operación en tiempo real señalando las diferencias con el de emulación. Se da la definición de tiempo real y el de integración numérica en tiempo real, se plantean los problemas originados por los elementos modelados, se presentan las técnicas numéricas para alcanzar las simulaciones en tiempo real y finalmente a través de dos modelos simples se exhibe el alto costo computacional que dentro de ambientes de simulación en tiempo real, tienen los algoritmos de integración que emplean técnicas de paso variable.

### § II.1 SIMULACIÓN

En el desarrollo de simuladores de tiempo real, se requiere verificar la interacción del sistema simulado con su medio ambiente. El medio ambiente [9] consiste del equipo, y de la persona (operador) que interviene en la operación del proceso, ver figura 1. Es necesario, para nuestros propósitos, distinguir los objetivos de emulación y simulación en tiempo real. El objetivo de la simulación es:

- **desarrollar un programa de cómputo que se comporte como el sistema físico a ser modelado,**

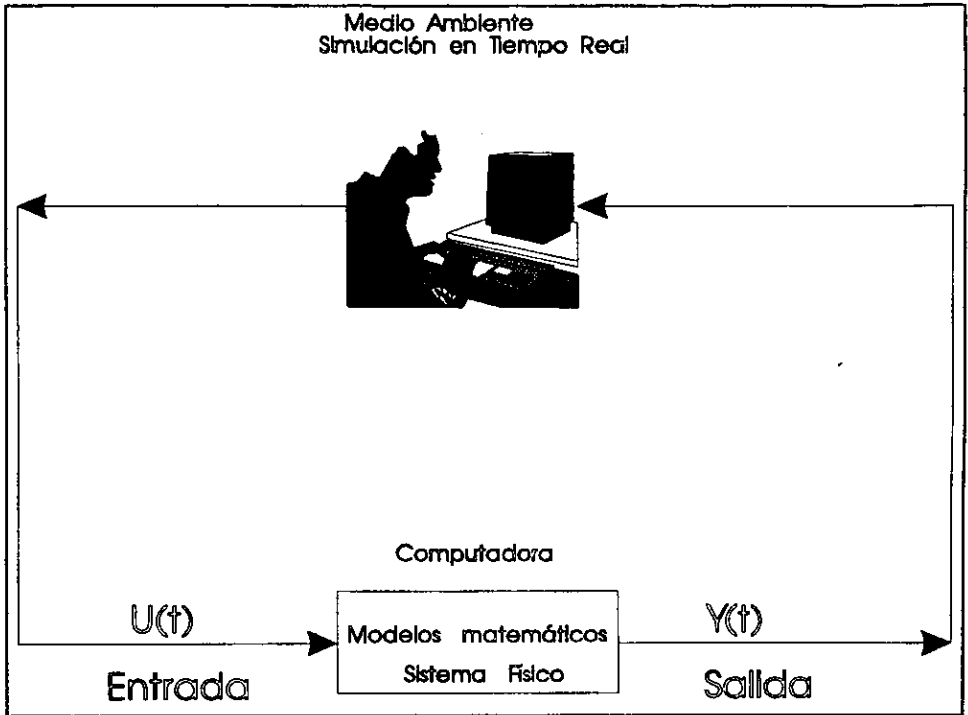


Figura 1

el "como" está en función del punto de vista de la persona o máquina en el medio ambiente; por otra parte, el objetivo de la emulación se puede plantear como sigue

- **analizar y probar un diseño propuesto.**

Los objetivos anteriores tienen una repercusión importante en la generación de los modelos que configuran el sistema a ser simulado (emulado). En el modo simulación es necesario modelar únicamente lo "suficiente" para alcanzar el comportamiento externo requerido. No se requiere que el modelo refleje fielmente el diseño del objeto simulado. La simulación es entonces un proceso que se desarrolla de arriba hacia

abajo, los detalles del sistema físico se añaden hasta que el modelo sea indistinguible del mundo real dentro de una aproximación deseada. Por el contrario, la emulación se desarrolla de abajo hacia arriba, el modelo es construido de componentes individuales y se requiere que se modele con un alto grado de aproximación.

En las siguientes secciones se hará un uso indistinto de estos dos conceptos, a menos que se aclare lo contrario.

### **§ II.1.1 Tiempo Real**

En simulación de procesos el concepto de tiempo real se refiere a la restricción de conservar el tiempo de computación empleado en realizar las simulaciones dentro de cierto límite que no debe excederse, es decir, si en una planta un proceso se efectúa en cierto período de tiempo, al realizar la simulación de éste el tiempo empleado por la computadora no debe excederlo. Consecuentemente, se debe cumplir la siguiente condición [9]

- **el tiempo de computación no debe de exceder el tiempo del período simulado**

El tiempo real delimita el ámbito de trabajo, lo cual no se debe perderse de vista. La capacidad de manejar problemas más complejos depende de disponer de computadoras más rápidas y/o modelos más rápidos conjuntamente con métodos numéricos eficientes en ambientes de simulación en tiempo real, con la restricción del costo-beneficio.

### § II.1.2 Operación en Tiempo Real

Para analizar las simulaciones en tiempo-real, es indispensable tomar en cuenta el lazo que conforma la persona (operador) y la máquina, ver figura (1). En general se espera que la entrada al sistema físico de una respuesta retardada a la salida de éste. Por ejemplo, una persona al operar el proceso puede estar siguiendo una determinada señal, la entrada al sistema físico puede tener la siguiente forma

$$u(t) = \mu(d(t) - y(t - \Delta_p)) + \nu d(t) \quad (1)$$

donde

$d(t)$	señal en seguimiento
$\Delta_p$	retardo en la respuesta de la persona
$\mu, \nu$	coeficientes de la habilidad de la persona
$y$	salida a tablero
$u$	entrada al sistema físico

Cuando el sistema físico es simulado digitalmente, las entradas al sistema físico son muestreadas en tiempos discretos. Por tanto los valores de  $u(t)$  se conocen únicamente en una sucesión de puntos  $t_n$ . Lo anterior tiene como consecuencia natural que en la simulación de procesos cualquier acción externa hecha por una persona o máquina no se detectará al tiempo  $t_n$  sino en el mejor de los casos en  $t_{n+1} = t_n + h$ . Después que la entrada  $u(t)$  ha sido detectada, se requiere de una cantidad de tiempo cómputo, que denotamos por  $\Delta_c$ , para aproximar numéricamente la salida  $y(t)$ , luego el retardo originado por la persona o máquina, se ve incrementado a

$$\Delta_p + \Delta_c + \theta h, \quad 0 < \theta \leq 1,$$

donde  $h$  es el paso de integración y  $\theta$  un parámetro.

Desde el punto de vista de la estabilidad del sistema físico, estos retardos, pueden ocasionar que el sistema se vuelva inestable.

### § II.1.3 Integración Numérica en Tiempo Real

La operación en tiempo real impone restricciones a los métodos de integración. En general éstos no pueden ser usados en el "sentido usual" [9]. La principal restricción es que el tiempo simulado se conserve dentro de los márgenes de tiempo real, es decir

- **el tiempo de computación no debe exceder la longitud del tiempo simulado para un paso de integración,**

en la práctica significa que se dispone de un recurso fijo del tiempo de computación para realizar los cálculos numéricos, este hecho no da cabida a usar códigos de propósito general para resolver sistemas de ecuaciones diferenciales (algebraico-diferenciales) con valores iniciales, que tengan técnicas de paso y orden variable con control de error, en los cuales el usuario no tiene (o si lo tiene es de forma restringida) control sobre la selección del paso y orden del método, se puede afirmar que no se puede tener en línea ningún control de error en la solución numérica del sistema de ecuaciones diferenciales ordinarias con valores iniciales cuando se está operando en tiempo real.

Es común encontrarse con el razonamiento que se menciona a continuación para justificar el uso de códigos de paso y orden variable con control de error. "Considérese

un código con las características anteriores, tal que en una parte problemática de la simulación seleccione un paso de integración pequeño que viole la condición de tiempo real, por ejemplo en un transitorio, posteriormente seleccione un paso de integración lo "suficientemente grande" que no sólo recupera el tiempo perdido sino aún más, el tiempo de cómputo en el período de salida seleccionado es menor que la longitud del tiempo simulado. Sin embargo, de acuerdo a lo señalado en §II.1.2 en los simuladores de entrenamiento se realizan alteraciones externas al sistema físico que pueden cambiar totalmente las condiciones de la simulación y consecuentemente la solución obtenida con el paso de integración grande ya no corresponde a la solución que se debe alcanzar. Es necesario recordar que no estamos ante un problema autocontenido, en donde, se ejecuta la simulación y nos dedicamos a observar los resultados para posteriormente analizarlos, la simulación en tiempo real se analiza dentro de su medio ambiente, en el cual existe una interacción entre el operador y el proceso representado por la computadora.

Por otra parte en tiempo real, las salidas son requeridas en períodos previamente determinados, por ejemplo cada 0.125 seg. Para lograr la sensación de suavidad en la actualización de los instrumentos analógicos, un código de propósito general de paso y orden variable con control de error, para satisfacer las necesidades predefinidas del usuario para las variables de salida, usa interpolación ya que es muy poco probable que con los pasos de integración que selecciona se satisfagan los requerimientos predeterminados. Sin embargo, la interpolación no se puede usar libremente [9], hay que tener especial cuidado en su aplicación y evaluar cuando se puede usar

sin afectar a la operación en tiempo real. La salida  $y(t)$  se necesita antes que el siguiente paso de integración se estime.

La evaluación de las derivadas del modelo, es en general, la parte dominante del costo computacional, siempre y cuando se esté usando un algoritmo cuyo costo computacional sea despreciable. Bajo esta suposición se tiene que:

- *tamaño del paso  $h = k \cdot \text{número de evaluaciones del sistema}$ ,*  
donde  $k$  es el tiempo de ejecución del modelo.

Es necesario asegurar que el equipo de cómputo tenga la suficiente capacidad en rapidez de cálculos numéricos y memoria para lograr la simulaciones en tiempo real. Indudablemente la expresión sobre el tamaño del paso de integración  $h$  impone una restricción fuerte a los algoritmos numéricos. Hay que tener atención especial con aquellos algoritmos que llaman al modelo más de una vez por paso de integración ya que se incrementan los retardos en la información y el costo computacional, quizás a niveles no tolerables. Si los retardos y el costo computacional pueden ser tolerados, el método puede ser considerado.

#### **§ II.1.4 Control del Error en Tiempo Real**

En §II.1.3 se mencionó la restricción de que el recurso del tiempo de computo por paso de integración es fijo, éste hecho impone una gran limitación al uso de técnicas de paso variable en los métodos de integración numéricos, luego es prácticamente imposible introducir técnicas para el control de error en los algoritmos de integración.



## **§ II.2 PROBLEMAS NUMÉRICOS EN TIEMPO REAL**

Los equipos modelados para realizar simulaciones de los procesos plantean problemas de diversos grados de dificultad a los métodos numéricos que se emplean, por otra parte, la ordenación incorrecta del conjunto de ecuaciones algebraico-diferencial introducen dificultades adicionales. A continuación se presenta un conjunto de problemas que dan origen a dificultades numéricas.

### **§ II.2.1 Problemas Físicos que Originan Dificultades Numéricas**

En la simulación de procesos en tiempo real, existen ciertas condiciones de operación y límites físicos de los equipos que dan origen a restricciones en las variables de estado, que imponen limitaciones a los métodos numéricos que se emplean para aproximar la solución. Entre estas restricciones se pueden citar las siguientes:

1. El valor de algunas variables de estados están limitadas a un rango predefinido de valores máximos y mínimos, por ejemplo
  - apertura y cierre de válvulas,
  - velocidad de apertura y cierre de válvulas de control,
  - volúmenes de tanques.
2. Rangos de valores prohibidos, por ejemplo
  - válvulas de no-retorno que impiden flujos negativos.
3. Algunas condiciones de emergencias y operación errónea del proceso pueden originar situaciones en las cuales los flujos tienden a cero.

4. La simulación de fallas de los equipos introducen cambios abruptos en el comportamiento del proceso.
5. Cambios de fase del proceso, por ejemplo, de líquido a vapor.

Estos problemas físicos se traducen en las siguientes dificultades numéricas:

- Discontinuidades en las derivadas,
- Presencia de matrices singulares en el sistema algebraico-diferencial que representa al proceso simulado.

### **§ II.2.2 Problemas en la Actualización de la Información que Originan Dificultades Numéricas**

Un ordenamiento correcto de las ecuaciones del sistema algebraico diferencial significa, que la evaluación de variables que aparecen en ellas no se efectúan en términos de otras que todavía no hayan sido calculadas [21], [23]. El ordenamiento incorrecto del sistema algebraico diferencial que representa al proceso que se desea simular, da origen a un problema que plantea grandes dificultades a los métodos numéricos empleados, éste origina:

- retrasos en el flujo de la información que afectan a la evaluación de las derivadas introduciendo dificultades a los métodos numéricos empleados.

### **§ II.2.3 Ecuaciones con Discontinuidades**

Las discontinuidades en las derivadas originan dificultades a los algoritmos de integración para aproximar las soluciones del sistema de ecuaciones diferenciales

ordinarias, éstas tienen que manipularse en forma apropiada [12]. El problema tiene origen cuando se violan hipótesis de trabajo que garantizan la existencia y unicidad de la solución, en general se asume que la solución puede ser aproximada por alguna función de clase  $\mathcal{C}^k$ , por ejemplo un polinomio. Sin embargo, cuando existe una discontinuidad en una derivada de orden menor que el grado del polinomio, se viola la hipótesis de trabajo. Si se está empleando un método integrador de un paso, el problema se puede resolver haciendo que la partición que se da al intervalo de integración, coincida con el punto donde se presenta la discontinuidad, posibilitando el uso de aproximaciones separadas de cada lado de ésta (se está asumiendo que el brinco en la discontinuidad es finito). Sin embargo el programa tiene que organizarse adecuadamente, por ejemplo, supóngase que se está usando una regla trapezoidal para integrar el siguiente problema

$$y' = \begin{cases} f_0(y,t), & \text{si } t < t_c \\ f_1(y,t), & \text{si } t \geq t_c \end{cases} \quad (3)$$

el paso terminal en  $t_c$  debe ser integrado con  $f_0(y,t)$  aún cuando  $y'$  esté evaluada en  $t_c$  en ese paso. Después que el paso ha sido completado se debe cambiar a la función  $f_1(y,t)$ . Lo anterior implica que debe existir coordinación entre el programa que evalúa las derivadas y el método integrador; el primero debe comunicar al segundo de la existencia de la discontinuidad y el segundo debe informar al primero cuando la discontinuidad ha sido sobrepasada. Si se usa un método integrador multipaso, el orden de éste no debe exceder el orden de la derivada en que se presenta la

discontinuidad, en cualesquiera de los pasos que use información de ambos lados de la discontinuidad, lo que significa que se debe conocer el orden de la derivada en donde se presenta la discontinuidad. Alternativamente, se puede reiniciar el método multipaso para evitar el problema, no obstante, todo reinicio es costoso en tiempo de cómputo, situación que no es conveniente en la operación en tiempo real.

Se puede concluir, que para resolver el problema es necesario conocer la discontinuidad y posiblemente el orden de la derivada en donde se presenta. Si la discontinuidad ocurre en la variable independiente no representa complicación en salvarla, ya que se puede determinar anticipadamente el tiempo en que se dará. Si la discontinuidad se presenta en valores conocidos de las variables dependientes o en funciones de ellas, es necesario localizar la discontinuidad rastreando los valores de estas funciones, para prevenir al método integrador. En todos los casos anteriores la detección de las discontinuidades es posible debido a que el usuario ha indicado la presencia de éstas, sin embargo, en el peor de los escenarios **no se tiene información de la presencia de éstas**. En la práctica esto suele suceder cuando se tienen códigos complejos de los modelos, que dificultan el análisis de la evaluación de las derivadas y no se puede obtener resultados adecuados, también se tienen casos en que las derivadas son funciones de tablas y los datos de éstas tienen rangos de invalidez.

Se puede resumir diciendo que existen tres requerimientos para manipular adecuadamente las discontinuidades

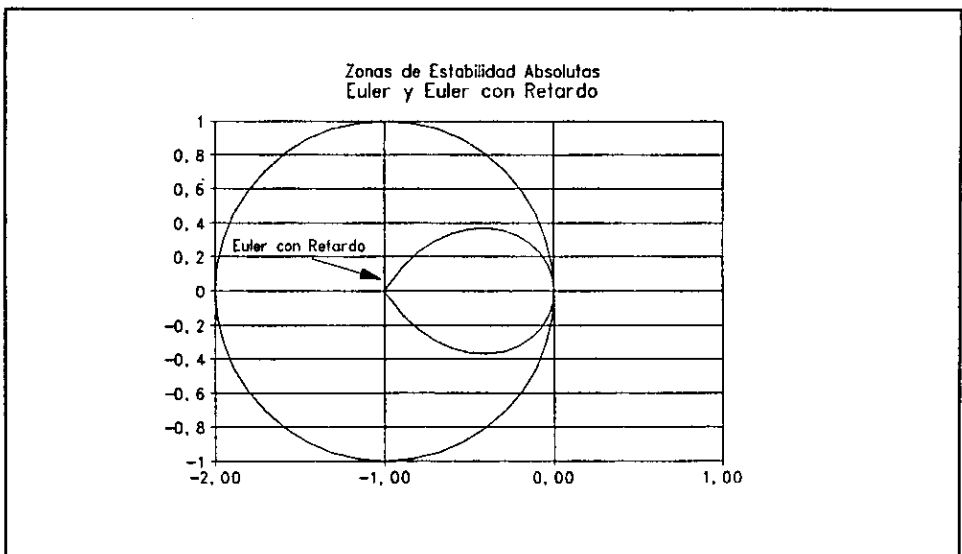
- Detección
- Localización
- Reinicio

## § II.2.4 Retrasos en el Flujo de la Información

Los retrasos en el flujo de la información reducen las zonas de estabilidad absoluta de los métodos numéricos empleados [9]. Si, por ejemplo, el método de Euler explícito es usado pero la información para actualizar la derivada es retrasada por un tiempo adicional  $h$ , el método realmente usado es

$$y_{n+1} = y_n + h y'_{n-1} \quad (4)$$

La gráfica 1, presenta la zona de estabilidad de éste método comparado con el método de Euler.



Gráfica 1

Estos decrementos en las zonas de estabilidad absoluta de los métodos de integración numérica, obligan a reducir el paso de integración  $h$  para lograr la convergencia, como consecuencia el esfuerzo computacional se incrementa notablemente.

## § II.3 ESTRATEGIA DE SOLUCIÓN EN TIEMPO REAL

En esta sección se presenta la estrategia:

- **modular secuencial,**

que se ha utilizado en el departamento de Simulación del Instituto de Investigaciones Eléctricas en el desarrollo de sus diferentes simuladores.

### § II.3.1 Solución Modular Secuencial

Los modelos matemáticos de los sistemas físicos modelados, en general están representados por un sistema de ecuaciones algebraicas no lineales acoplado con un conjunto de ecuaciones diferenciales ordinarias. La representación matemática es la siguiente

$$\begin{aligned} \dot{y} &= f(y,z) \\ 0 &= g(y,z) \end{aligned} \tag{5}$$

donde

$y$  es un vector de  $n$  variables de estado,

$z$  es un vector de  $m$  variables algebraicas

La aproximación secuencial consiste en:

1. Aproximar una solución de  $z$  al tiempo  $t_n$  manteniendo  $y$  constante. En general, los métodos utilizados son el algoritmo de Newton o variantes de éste.
2. Aproximar una solución de  $y$  al tiempo  $t_n$  manteniendo  $z$  constante. La aproximación de  $y$  se hace por medio de un algoritmo apropiado para la solución de ecuaciones diferenciales ordinarias con valores iniciales. El ciclo anterior es repetido por paso de integración  $h$ , hasta alcanzar el tiempo final de la simulación.

En los simuladores desarrollados en el departamento de Simulación del Instituto de Investigaciones Eléctricas, se introduce al esquema anterior una modularización que consiste en partir el sistema de ecuaciones algebraico diferenciales en un conjunto de módulos en función del grado de acoplamiento entre ellos. En general esta partición del sistema físico es hecha heurísticamente explotando las diferentes dinámicas que se encuentran presentes en una planta generadora de energía eléctrica. La partición del sistema físico a través de su dinámica es utilizada para asociar a cada módulo (o conjuntos de módulos) el método y paso de integración más adecuado de acuerdo a las características de su dinámica (constantes de tiempo) para reducir el costo computacional, este método de integración se le conoce con el nombre de Método de Integración Multitasa [8], [10], [11], [14] (Integration Multirate Method). La aproximación secuencial, es entonces aplicada a cada módulo.

## § II.4 EJEMPLOS

En esta sección se ejemplifica, a través de dos modelos simples, el alto costo computacional de los métodos de integración numérica con paso y orden variables, y con control de error local, en ambientes de simulación en tiempo real, señalados en §II.1.3, así como, las dificultades numéricas presentadas en §II.2. Con estos dos modelos se realizaron cuatro pruebas numéricas, los resultados obtenidos presentan el esfuerzo computacional asociado al método integrador con las características mencionadas, y los efectos originados por las discontinuidades e información retrasada debida al ordenamiento no adecuado de las ecuaciones que representan a los procesos simulados. En las siguientes secciones estos resultados son presentados.

### § II.4.1 Primer Modelo: Tres Reactores Tipo Tanque con Control PI

Se consideran tres reactores tipo tanque [22] con un control proporcional integral (PI) retroalimentado como se muestra en la figura 2.

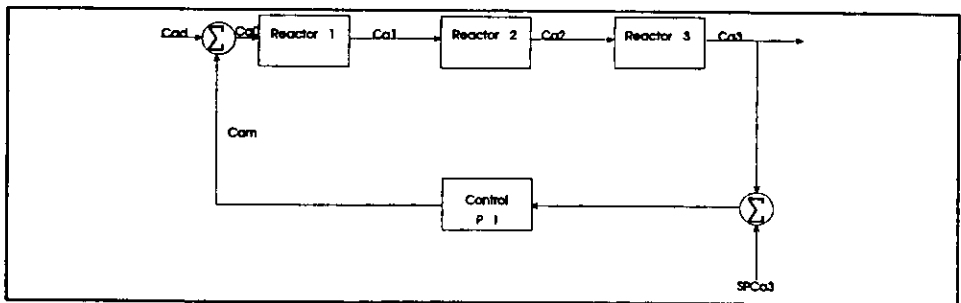


Figura 2



En estos reactores el producto B es producido y el reactante A es consumido por una reacción de primer orden que se produce en el líquido. Se asume que las temperaturas y volúmenes de los tres tanques pueden ser diferentes, pero la temperatura y el volumen del líquido en cada tanque se considera constante. La densidad se considera constante a lo largo del sistema, el cual es una mezcla binaria de A y B. El control mantiene la corriente  $Ca_3$  en el valor deseado  $SPCa_3$ .

La representación matemática del modelo es la siguiente:

$$\begin{pmatrix} D_t Ca_1 \\ D_t Ca_2 \\ D_t Ca_3 \end{pmatrix} = \begin{pmatrix} -(\frac{1}{\tau} + k) & 0 & 0 \\ \frac{1}{\tau} & -(\frac{1}{\tau} + k) & 0 \\ 0 & \frac{1}{\tau} & -(\frac{1}{\tau} + k) \end{pmatrix} \begin{pmatrix} Ca_1 \\ Ca_2 \\ Ca_3 \end{pmatrix} + \begin{pmatrix} \frac{1}{\tau} \\ 0 \\ 0 \end{pmatrix} Ca_0 \quad (6)$$

La concentración  $Ca_0$  en la entrada está formada por  $Ca_d$  que es una concentración de disturbio y por  $Ca_m$  que es la concentración manipulada por el controlador

$$\begin{aligned} Ca_0 &= Ca_d + Ca_m \\ Ca_m &= 0.8 + k_c \left( e + \frac{1}{\tau_i} \int e(t) dt \right) \end{aligned} \quad (7)$$

donde

- $D_t$  denota la derivada de la función respecto a  $t$ ,
- $Ca_i$  concentración del producto  $a_i$ ,  $i = 1, 2, 3$ ,
- $e$   $SPCa_3 - Ca_3$ , error,

- $SP_{aj}$  punto de ajuste,
- $k_c$  ganancia del controlador,
- $\tau_i$  constante integral del controlador.

### § II.4.2 Segundo Modelo: Control de Presión en una Tubería

La figura 3 representa el control de presión en una tubería efectuado con un transmisor de presión (PT) un controlador de presión proporcional integral (PI), una válvula de control y la tubería. El control PI mantiene la presión en la tubería en el valor deseado  $SP$ .

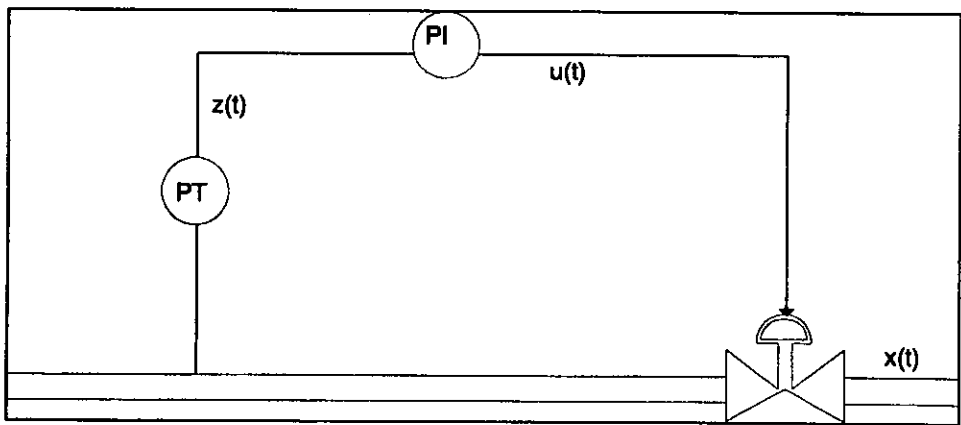


Figura 3

La representación matemática del modelo es la siguiente:

$$u(t) = k_c \left( e + \frac{1}{\tau_i} \int e(t) dt \right)$$

$$\ddot{x} + a\dot{x} + bx = bu(t)$$

$$a = \frac{1}{c_2} 2s$$

$$b = \frac{1}{C_2^2}$$

$$\dot{y} = \frac{1}{PR}(x-y)$$

$$\dot{z} = \frac{1}{T_1}(y-z)$$

(8)

donde

- $u(t)$  salida del controlador,
- $x(t)$  flujo de la válvula,
- $y(t)$  presión en la tubería,
- $z(t)$  salida del transmisor de presión,
- $e$   $SP - z$ , error,
- $SP$  punto de ajuste,
- $k_c$  ganancia del controlador,
- $\tau_i$  constante integral del controlador,
- $C_2$  constante de tiempo de la válvula,
- $s$  factor de amortiguamiento de la válvula,
- $PR$  constante de tiempo del proceso,

- $T_i$  constante de tiempo del transmisor de presión.

El valor del flujo de la válvula está limitado en el siguiente rango de valores

$$\hat{x} = \begin{cases} 1 & \text{si } x > 1 \\ x & \text{si } 0 \leq x \leq 1 \\ 0 & \text{si } x < 0 \end{cases} \quad (9)$$

### § II.4.3 Simulaciones de Transitorios con Método Integrador de Orden y Paso Variable.

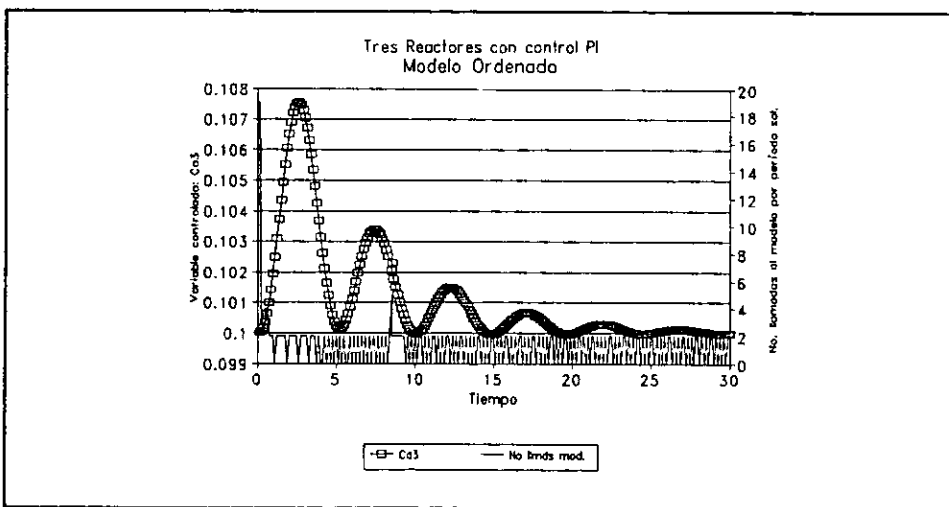
A continuación se presentan los resultados de las cuatro pruebas numéricas realizadas a los sistemas (6) y (8), el método integrador utilizado es el código ODE de Shampine [28], éste integra un sistema de ecuaciones diferenciales ordinarias a valores iniciales planteadas en forma explícita. Usa un algoritmo de aproximación a la solución basado en las fórmulas de Adams programadas en la forma PECE (Predice-Evalúa-Corrige-Evalúa). Ajusta el orden del método y paso de integración en función del control del error local.

Los parámetros dados a ODE en las cuatro pruebas numéricas son

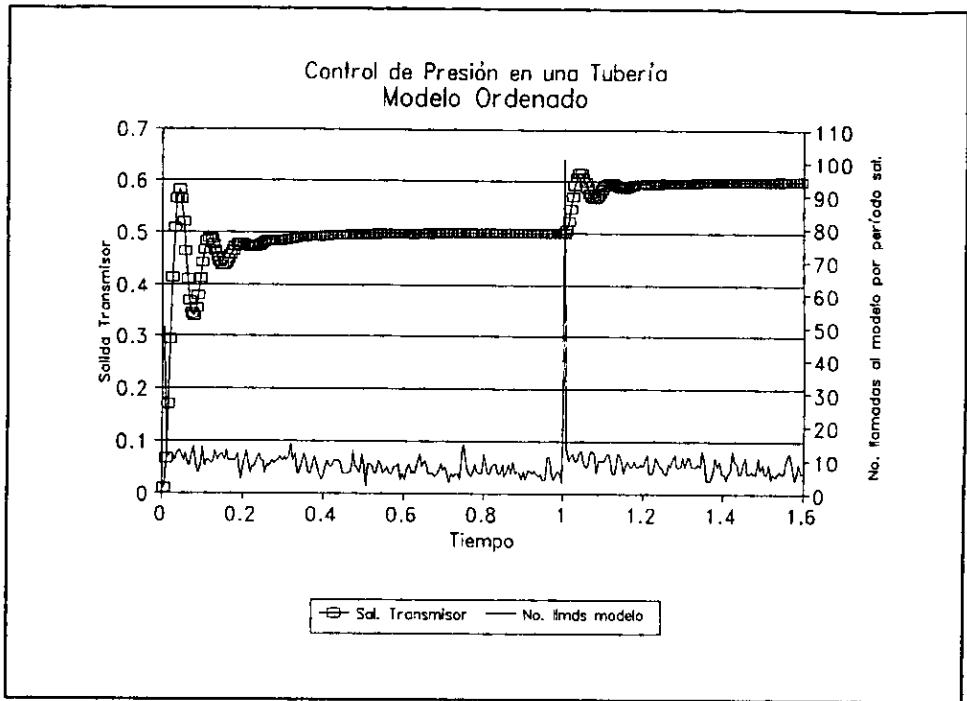
Parámetros Método ODE de Shampine					
	Tiempo Inicio	Tiempo Final	Error Abs	Error Rltvo	Puntos Sal.
Tres Reactores	0	30	$10^6$	$10^{11}$	0.125
Control Presión	0	1.6	$10^6$	$10^{11}$	0.053

El primer experimento consiste en integrar el primer y segundo modelo, sistemas (6) y (8), con el ordenamiento correcto de sus ecuaciones. Para el primer modelo, **Tres Reactores con Control Proporcional Integral (PI)**, se realizó una simulación de 30 minutos tomándose muestras cada 0.125 minutos de los datos. La prueba inicia en un transitorio y por medio de la acción del control PI se obliga a la corriente  $Ca_3$  ir al valor deseado  $SPCa_3$ . En la gráfica 2 se presentan los resultados obtenidos, en el eje primario se gráfica la variable  $Ca_3$  la cual oscila entre 0.1 y 0.108, esta oscilación se amortigua y tiende al valor de 0.1 que es el valor deseado  $SPCa_3$ , en el eje secundario se gráfica el costo computacional por cada 0.125 minutos, éste se mide por el número de llamadas hechas por el código ODE al programa que contiene a las derivadas. El rango de valores del costo computacional por cada 0.125 minutos varía entre 0 y 20, la parte con más costo se da al inicio de la simulación con 19 llamadas, posteriormente cuando se supera el transitorio y se alcanza el nuevo estado estacionario el número de llamadas varía entre 0 y 2. Para el segundo modelo, **Control de Presión en una Tubería**, se realizó una simulación de 1.6 seg., tomándose muestras cada 0.053 seg. de los datos. La prueba inicia en un transitorio y por medio de la acción del control PI se obliga a la variable  $z(t)$  salida del transmisor de presión, ir al valor deseado  $SP$ , además se introduce una discontinuidad en las derivadas cambiando el valor deseado  $SP$ , de su valor original 0.5, al de 0.6 en el tiempo 1.0 seg. La gráfica 3 presentan los resultados obtenidos, en el eje primario se gráfica la variable  $z(t)$  la cual oscila entre 0 y 0.6, esta oscilación es amortiguada por el control PI que obliga a  $z(t)$  ir al valor de 0.5 que es el valor deseado  $SP$ , al llegar a un

segundo simulado se hace un cambio instantáneo en valor de  $SP$  de 0.5 a 0.6, lo que origina un nuevo transitorio que es amortiguado nuevamente por el control PI que obliga a  $x(t)$  ir al nuevo valor deseado 0.6 de  $SP$ . Análogamente a la gráfica 2 del primer modelo, en el eje secundario de la gráfica 3, se gráfica el costo computacional por cada 0.053 seg., éste varía entre 0 y 110 llamadas al programa que contiene las derivadas, la parte con el costo computacional mayor es cuando se hace el cambio instantáneo a  $SP$ , en éste punto se realizan 100 llamadas, la discontinuidad obliga al código ODE a realizar un esfuerzo considerable para sobrepasarla; con excepción del inicio en donde se hacen 50 llamadas, en las partes en donde se ha superado el transitorio se efectúan alrededor de 15 llamadas.



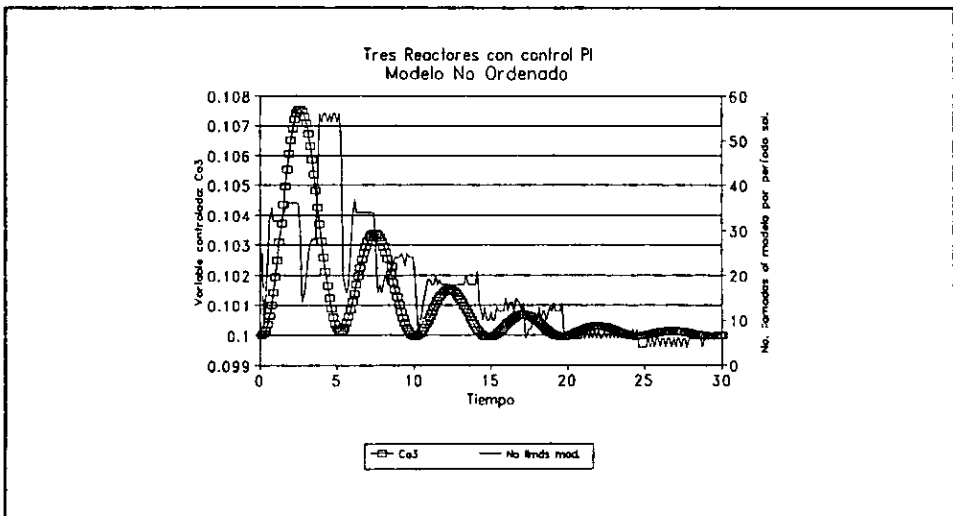
Gráfica 2



Gráfica 3

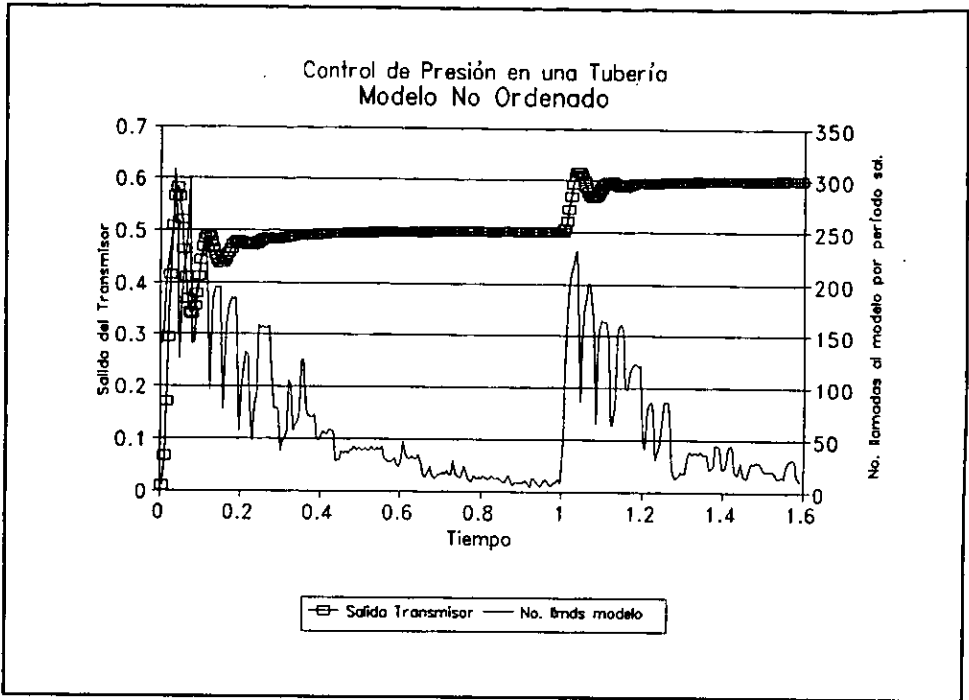
El segundo experimento es análogo al primero, excepto que se integran los modelos con un orden incorrecto en las ecuaciones, esto se logra cambiando de posición a una de las ecuaciones en la secuencia de ejecución en la programación, tal que la evaluación de sus variables se haga en términos de otras que todavía no han sido calculadas. Las gráficas 4 y 5 presentan los resultados del segundo experimento para ambos modelos respectivamente, se gráficas las mismas variables. El costo computacional es mayor para los experimentos con los modelos no ordenados. En el caso del modelo de los tres reactores el número máximo de llamadas es de 60 presen-

tándose al tiempo 5, éste corresponde a la parte crítica del transitorio, en comparación de las 20 llamadas respecto al primer experimento. Para el modelo de control de presión en una tubería se tienen un máximo de 310 llamadas en el tiempo 0.1 que corresponde a la parte crítica del transitorio, y 225 llamadas cuando se introduce el cambio instantáneo en el valor de  $SP$ . En general el costo computacional es superior en el segundo experimento respecto al primero. Lo anterior confirma lo expuesto en §II.2.4, debido al retardo de la información originada por el ordenamiento incorrecto de las ecuaciones se decrementa la zona de estabilidad absoluta asociada al método y en consecuencia el código ODE es obligado a disminuir el paso de integración para converger.



Gráfica 4





**Gráfica 5**

# Capítulo III

## Métodos Multitasa

En éste capítulo se definen los conceptos de los métodos de integración multitasa para métodos multipasos lineales, a los cuales se les identifica con la siguiente abreviación **MIM's**, se asume que el lector esta familiarizado con los conceptos de los métodos de integración multipaso convencionales [19]. Se definen los sistemas de ecuaciones diferenciales ordinarias de dos escalas temporales, a los que se les puede aplicar éste tipo de **MIM's**. Se presentan seis algoritmos multitasa utilizados en los ambientes de simulación en tiempo real y se hace un análisis de su costo computacional respecto al número de operaciones realizadas.

### § III.1 INTEGRACION DE SISTEMAS DINAMICOS

La simulación de sistemas dinámicos en **tiempo real** nos lleva a considerar problemas con valores iniciales para sistemas de ecuaciones diferenciales ordinarias cuya solución numérica plantea dificultades ya señaladas en §II.1.1, §II.1.2, §II.1.3, §II.2.

Entre éstas, se encuentran:

1. Restricciones en la longitud del paso de integración.

2. El tiempo de computación de un paso de integración no debe de exceder al tiempo representado por el paso.
3. Las alteraciones externas al proceso producidas por la relación hombre-sistema que conforman el medio ambiente de simulación del proceso (consultar II.1).
4. La dimensión del sistema de ecuaciones diferenciales y de las ecuaciones algebraicas que representan al modelo simulado.
5. Retardos en la información para actualizar las ecuaciones.
6. La amplia variedad de dinámicas que exhibe el vector solución del sistema de ecuaciones diferenciales ordinarias.

Estos problemas imponen fuertes restricciones a los métodos numéricos usados y al máximo paso de integración posible. Por otra parte, algunas componentes de la solución pueden tener una alta frecuencia de oscilación y/o transitorios rápidos, mientras que las componentes restantes son relativamente lentas. A este tipo de sistemas se les conoce como **"SISTEMAS CON DOS ESCALAS TEMPORALES"** [1], [18]. Con técnicas de integración numérica convencionales, las componentes que varían rápidamente impondrían el uso de un paso de integración pequeño, con el resultado de que las componentes lentas son integradas con un alto costo computacional. Esta situación se remedia partiendo el sistema de ecuaciones diferenciales ordinarias en dos subsistemas, la parte rápida y la lenta; la integración se realiza entonces usando un paso de integración grande para el subsistema lento y un paso pequeño para el rápido, ésta técnica de integración se llama **MÉTODO DE INTEGRACIÓN MULTITASA (MIM)** [8], [10], [11], [14].

### § III.2 METODOS DE INTEGRACION MULTITASA (MIM)

En un sentido más general, un método de integración multitasa es una técnica que permite integrar cada ecuación de un sistema de ecuaciones diferenciales ordinarias con el paso y método de integración más adecuado a su comportamiento. Esto será posible si no hay una dependencia fuerte entre las componentes rápidas y lentas. Al considerar los problemas mencionados se eligen, métodos de integración de orden bajo, que en general son explícitos, y en algunas ocasiones se eligen métodos implícitos que se implementan en esquemas  $P(EC)^m$  o  $P(EC)^mE$  [19], y se programan a paso fijo sin control del error. Los requerimientos de aproximación en simulación en tiempo real son mínimos; además los modelos incluyen múltiples simplificaciones y aproximaciones (algunos presentan discontinuidades en las derivadas debido a las operaciones de control), así pues, se justifica el uso de métodos de bajo orden.

### § III.3 DESCRIPCION DE LOS METODOS MULTITASA

Considérese el siguiente problema  $n$  dimensional con valor inicial

$$w' = P(w,t); \quad w_0 = w(0); \quad (1)$$

supóngase que (1) se puede partir en dos subsistemas

$$w' = \begin{pmatrix} y' \\ z' \end{pmatrix} = \begin{pmatrix} f(y,z,t) \\ g(y,z,t) \end{pmatrix}; \quad w_0 = \begin{pmatrix} y_0 \\ z_0 \end{pmatrix} \quad (2)$$

donde  $\dim(y) + \dim(z) = n$ . El subsistema  $f(y,z,t)$  representa la parte rápida y  $g(y,z,t)$  la lenta.

Sea  $M$  un método multipaso lineal conocido aplicado a (1) con longitud de paso constante  $h$

$$w_{n+p} = M (w_n, \dots, w_{n+p-1}, f_n, \dots, f_{n+p-1}, h; P), \quad (3)$$

donde  $w_{n+p}$   $w_n$ , almacenan la información necesaria para llevar a cabo el método multipaso lineal,  $p$  es el número de pasos del método.

El método multitasa puede ser descrito con  $M$  de la siguiente manera

$$\begin{aligned} y_{nk+i+1} &= M (y_{nk+i}, f_{nk+k+i}, h; \hat{f}); \quad i=0, \dots, k-1 \\ z_{nk+k} &= M (z_{nk}, f_{nk}, kh; g) \end{aligned} \quad (4)$$

donde

$n$  representa el ciclo (ver figura 1) en el que se efectúa la integración  
 $k$  factor multitasa, es el número de marcos (ver figura 1, en ésta se encuentran representados nueve marcos) en que se divide el ciclo (entero positivo).

$i$  representa el marco en el que se calcula  $y_{nk+i+1}$ ,  $i=0, \dots, k-1$

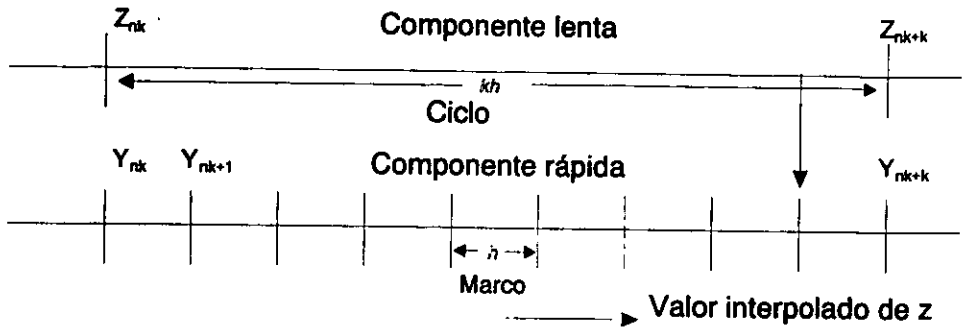
$h$  paso de integración del subsistema rápido

$kh$  paso de integración del subsistema lento

$\hat{f} = f(y, z, t)$

$\hat{z}$ : aproximación a  $z$

Una aproximación  $\hat{z}$  de los valores de  $z$  es necesaria para evaluar el subsistema rápido  $f(y, z, t)$  en los tiempos,  $t_{nk+i+1}$ , ya que en estos puntos se desconocen los valores de  $z$ . Para  $n$  entero positivo, la ecuación (4) describe una regla de integración para un ciclo de longitud  $kh$ . El MIM es considerado como un método de paso  $kh$ .



Sucesión cronológica del método multitasa  
figura 1

Si se escoge un paso  $h$  para el subsistema rápido, entonces el número de evaluaciones de las derivadas del tiempo inicial,  $t_{nk}$  al tiempo final,  $t_{nk+k}$  es:

a) método de integración usual

$$k \text{ (evaluaciones de } f) + k \text{ (evaluaciones de } g)$$

b) método multitasa

$$k \text{ (evaluaciones de } f) + \text{una evaluación de } g + \text{operaciones de interpolación}$$

Observe que en éste caso el método multitasa descrito por (4) esta usando el mismo método de integración para la componente rápida y lenta, sin embargo, bajo un esquema multitasa se pueden usar diferentes métodos de integración para cada subsistema.

### § III.4 APLICABILIDAD DE LOS METODOS MULTITASA

Un método multitasa como el descrito en la sección anterior, no se puede aplicar a cualquier tipo de sistemas de ecuaciones diferenciales ordinarias, éste debe satisfacer

ciertas condiciones de las que hasta ahora se ha hablado vagamente. En ésta sección se definen los sistemas de ecuaciones diferenciales ordinarias con dos escalas temporales, siendo a éste tipo de sistemas a los que se aplican los métodos multitasa.

### § III.4.1 Sistemas con Dos Escalas Temporales: Caso Lineal Homogéneo

Sea

$$y' = Ay; \quad y_0 = y(0), \quad (5)$$

un sistema lineal de ecuaciones diferenciales ordinarias homogéneo,  $A$  es una matriz de orden  $n \geq 2$ , con valores característicos  $\lambda_i$  distintos y parte real negativa.

Un resultado conocido (págs. 47-48)[17] es que la solución de (5) está dada por

$$y(t) = \sum_{i=1}^n k_i e^{\lambda_i t} v_i \quad (6)$$

donde  $v_i$  es el vector característico asociado a  $\lambda_i$ , en tanto que las  $k_i$  son constantes que dependen de la condición inicial y de los vectores característicos,  $i = 1, 2, \dots, n$ .

Supóngase que los valores característicos  $\lambda_i$  de la matriz  $A$  pueden ser partidos en dos clases ajenas  $\mathfrak{K}$  y  $\mathfrak{L}$  tal que cumplan las siguientes condiciones (para no complicar la notación asumiremos que los valores y vectores característicos así como las constantes, se han ordenado y enumerado para que los primeros  $m$  valores característicos pertenezcan a la clase  $\mathfrak{K}$  y los restantes a la clase  $\mathfrak{L}$ )

$$a) \quad |\lambda_i| > |\lambda_j| \quad i = 1, 2, \dots, m, \quad j = m+1, \dots, n, \quad \lambda_i \in \mathfrak{R} \text{ y } \lambda_j \in \mathfrak{L}$$

$$b) \quad \left| \frac{\lambda}{\lambda^*} \right| = 1 \text{ para toda } \lambda, \lambda^* \in \mathfrak{R}$$

$$c) \quad \left| \frac{\lambda}{\lambda^*} \right| = 1 \text{ para toda } \lambda, \lambda^* \in \mathfrak{L}$$

Los valores característicos que pertenecen a  $\mathfrak{R}$  se llaman los rápidos y los de  $\mathfrak{L}$  los lentos. Entonces, la ecuación (6) se puede expresar como

$$y(t) = \sum_{i=1}^m k_i e^{\lambda_i t} v_i + \sum_{j=m+1}^n k_j e^{\lambda_j t} v_j \quad (7)$$

Los términos  $k_i e^{\lambda_i t}$  para  $i = 1, 2, \dots, m$  se llaman los **modos rápidos**, en tanto que para  $j = m+1, \dots, n$  se llaman los **modos lentos** de (1).

**Definición III.4.1.1.** El sistema (5) tiene dos escalas temporales [1], [18] en el intervalo  $[a, b]$  si

1. Los valores característicos  $\lambda_i$  de la matriz  $A$  pueden ser partidos en dos clases ajenas  $\mathfrak{R}$  y  $\mathfrak{L}$  que satisfagan las condiciones (a), (b) y (c).
2. Exista al menos una componente, digamos

$$y_i(t) = \sum_{i=1}^m k_i v_i^r e^{\lambda_i t} + \sum_{j=m+1}^n k_j v_j^l e^{\lambda_j t} \quad (8)$$

de la solución de (5) que satisface



$$\left| \sum_{i=1}^m k_i v_i^r \right| > 0 \quad (9)$$

donde  $v_i^r$  es la  $r$ -ésima componente del vector característico  $v_i$ , análogamente para

$v_j^r$ , y

$$\frac{\left| \sum_{j=m+1}^n k_j v_j^r \right|}{\left| \sum_{i=1}^m k_i v_i^r \right|} < \varepsilon \quad (10)$$

con  $\varepsilon > 0$  lo suficientemente pequeña; en otras palabras que la dependencia de  $y$  con los modos lentos de la solución no sea significativa. A  $y$ , se le llama la **componente rápida**.

3. Exista al menos una componente, digamos

$$y(t) = \sum_{i=1}^m k_i v_i^t e^{\lambda_i t} + \sum_{j=m+1}^n k_j v_j^t e^{\lambda_j t} \quad (11)$$

de la solución de (5) que satisface

$$\left| \sum_{j=m+1}^n k_j v_j^t \right| > 0 \quad (12)$$

y

$$\frac{\left| \sum_{i=1}^m k_i v_i^t \right|}{\left| \sum_{j=m+1}^n k_j v_j^t \right|} < \varepsilon \quad (13)$$

con  $\varepsilon > 0$  lo suficientemente pequeña; en otras palabras, que la dependencia de  $y$  con los modos rápidos de la solución no sea significativa. A  $y_1$  se le llama la **componente lenta**.

A continuación se presenta el análisis de un sistema bidimensional con dos escalas temporales, cuya matriz es triangular inferior. Se demuestra que las escalas temporales dependen también de las condiciones iniciales. Finalmente se da un ejemplo específico en donde se aplican los resultados obtenidos del análisis.

Consideremos el siguiente sistema de ecuaciones diferenciales ordinarias (EDO's) con valores iniciales

$$\begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \end{pmatrix} = \begin{pmatrix} a_{11} & 0 \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}; \quad y_0 = \begin{pmatrix} y_{01} \\ y_{02} \end{pmatrix}, \quad (14)$$

ya que la matriz es triangular inferior, los valores característicos son los elementos de la diagonal

$$\begin{aligned} \lambda_1 &= a_{11}, \\ \lambda_2 &= a_{22}, \end{aligned} \quad (15)$$

supondremos que  $\lambda_1 \neq \lambda_2$ .

Denotemos por  $V_i$ ,  $i = 1, 2$  a los vectores característicos de la matriz, estos se encuentran resolviendo el sistema

$$(A - \lambda_i I)V_i = 0; \quad i=1,2 \quad (16)$$

donde  $I$  es la matriz identidad y  $A$  la matriz triangular inferior de (14). Los vectores característicos asociados a  $\lambda_1$ ,  $\lambda_2$ , son respectivamente

$$V_1 = \begin{pmatrix} 1 \\ -\frac{a_{21}}{\lambda_2 - \lambda_1} \end{pmatrix}, \quad V_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad (17)$$

La solución del sistema (14) de ecuaciones diferenciales ordinarias está dado por [17]

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = k_1 e^{\lambda_1 t} V_1 + k_2 e^{\lambda_2 t} V_2, \quad (18)$$

desarrollando (18), obtenemos

$$\begin{aligned} y_1(t) &= k_1 e^{\lambda_1 t} \\ y_2(t) &= -\frac{a_{21}}{(\lambda_2 - \lambda_1)} k_1 e^{\lambda_1 t} + k_2 e^{\lambda_2 t} \end{aligned} \quad (19)$$

Las constantes  $k_1$  y  $k_2$  son funciones de las condiciones iniciales  $y_0$ , y se encuentran resolviendo el sistema (19) para  $t = t_0$ , por simplicidad supondremos que  $t_0 = 0$ , sustituyendo en (19) el sistema a resolver es el siguiente

$$\begin{pmatrix} 1 & 0 \\ -\frac{a_{21}}{\lambda_2 - \lambda_1} & 1 \end{pmatrix} \begin{pmatrix} k_1 \\ k_2 \end{pmatrix} = \begin{pmatrix} y_{01} \\ y_{02} \end{pmatrix} \quad (20)$$

luego

$$\begin{aligned} k_1 &= y_{01}, \\ k_2 &= y_{02} + \frac{a_{21}}{\lambda_2 - \lambda_1} k_1, \end{aligned} \quad (21)$$

entonces sí

$$y_{02} = -\frac{a_{21}}{\lambda_2 - \lambda_1} y_{01}, \quad (22)$$

de (21) obtenemos que  $k_2 = 0$ .

Supongamos que  $|\lambda_1| > |\lambda_2|$ , y que sus partes reales son negativas, entonces de acuerdo a la definición III.4.1.1, claramente la componente rápida es  $y_1$  ya que no tiene contribución de la parte lenta y consecuentemente la expresión (10) es igual a cero,  $y_2$  es la componente lenta sí se satisface la condición (13), esto es

$$\left| \frac{a_{21} k_1}{(\lambda_2 - \lambda_1) k_2} \right| < \epsilon, \quad \text{con } k_2 \neq 0, \quad (23)$$

Hay que notar que en el caso de una matriz  $2 \times 2$ , los valores característicos deben ser reales negativos diferentes, ya que si uno es complejo necesariamente su conjugado es el segundo valor característico asociado al sistema (14), y  $|\lambda_1| = |\lambda_2|$  serían iguales, por lo que los valores característicos no estarían separados.

Es importante observar que la separabilidad del sistema depende también de sus condiciones iniciales, ya que si escogemos  $y_{02}$  como en (22), entonces  $y_2(t)$  se reduce a

$$y_2(t) = -\frac{a_{21}}{\lambda_2 - \lambda_1} k_1 e^{\lambda_1 t}, \quad (24)$$

ya que  $k_2 = 0$  por (21). La solución del sistema (14) está entonces en función de  $\lambda_1$  únicamente, y en consecuencia a pesar que la matriz  $A$  de (14) sea triangular inferior y tenga valores característicos separados no es un sistema a dos escalas temporales, debido a la elección de la condición inicial.

Consideremos ahora, el siguiente ejemplo específico:

Sea

$$\begin{pmatrix} y_1' \\ y_2' \end{pmatrix} = \begin{pmatrix} -25 & 0 \\ 3 & -5 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}; \quad y_0 = \begin{pmatrix} y_{01} \\ y_{02} \end{pmatrix}. \quad (25)$$

De (15) y (17) obtenemos los valores y vectores característicos asociados al sistema (25)

$$\begin{aligned} \lambda_1 = -25, \quad V_1 &= \begin{pmatrix} 1 \\ -\frac{3}{20} \end{pmatrix}, \\ \lambda_2 = -5, \quad V_2 &= \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \end{aligned} \quad (26)$$

La solución general de (25) es

$$\begin{aligned} y_1 &= k_1 e^{-25t} \\ y_2 &= -\frac{3}{20} k_1 e^{-25t} + k_2 e^{-5t} \end{aligned} \quad (27)$$

Claramente  $y_1$  es la componente rápida del sistema, y  $y_2$  es la componente lenta si se cumple que la contribución del modo rápido no es significativa comparada con el modo lento, esto es

$$\left| -\frac{3}{20} \frac{k_1}{k_2} \right| < \epsilon \quad (28)$$

para  $\epsilon$  pequeño.

A continuación se presentan las gráficas de (25) para la siguiente pareja de condiciones iniciales, la segunda pareja de condiciones iniciales se eligió de acuerdo a (22) para mostrar la dependencia en las condiciones iniciales de las escalas temporales

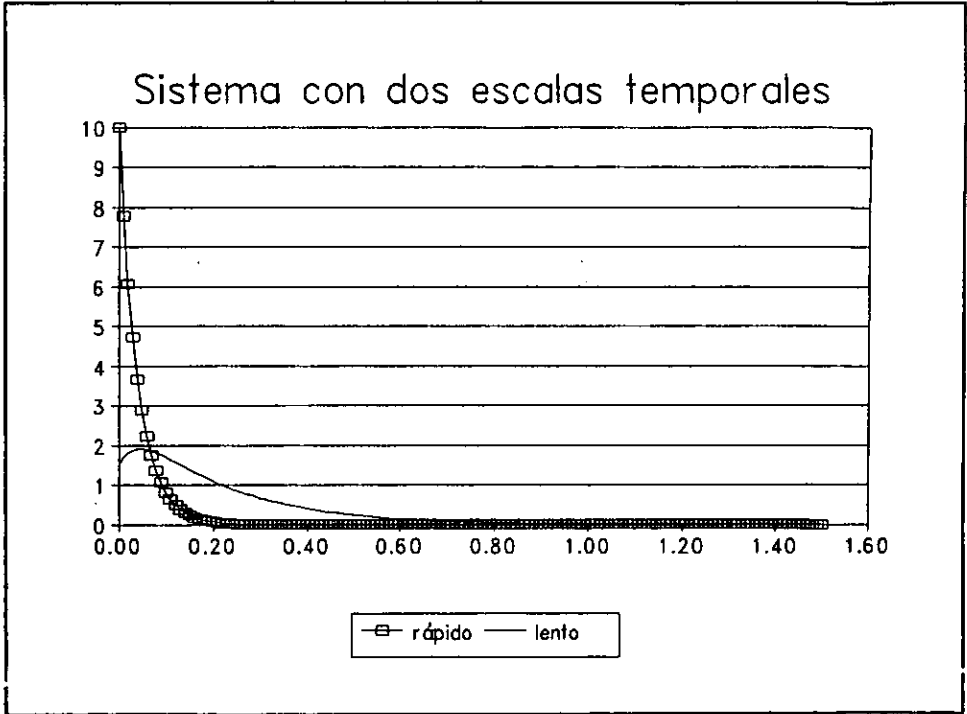
$$y_0 = \begin{pmatrix} 10 \\ 1.5 \end{pmatrix}, \quad y_0 = \begin{pmatrix} 10.11 \\ -1.52 \end{pmatrix} \quad (29)$$

En el primer caso, sustituyendo en (21) la primera pareja de condiciones iniciales dadas por (29) obtenemos

$$k_1 = 10, \quad k_2 = 3, \quad (30)$$

sustituyendo (30) en (28) obtenemos la contribución de los coeficientes del modo rápido al lento, éste es de 0.5, ya que es menor que la unidad se considera que es

una dependencia débil, lo que demuestra que para la componente  $y_2$  la contribución del modo rápido en su formación no es significativa. En la gráfica 1, se aprecia cualitativamente el comportamiento de la componente rápida y lenta.

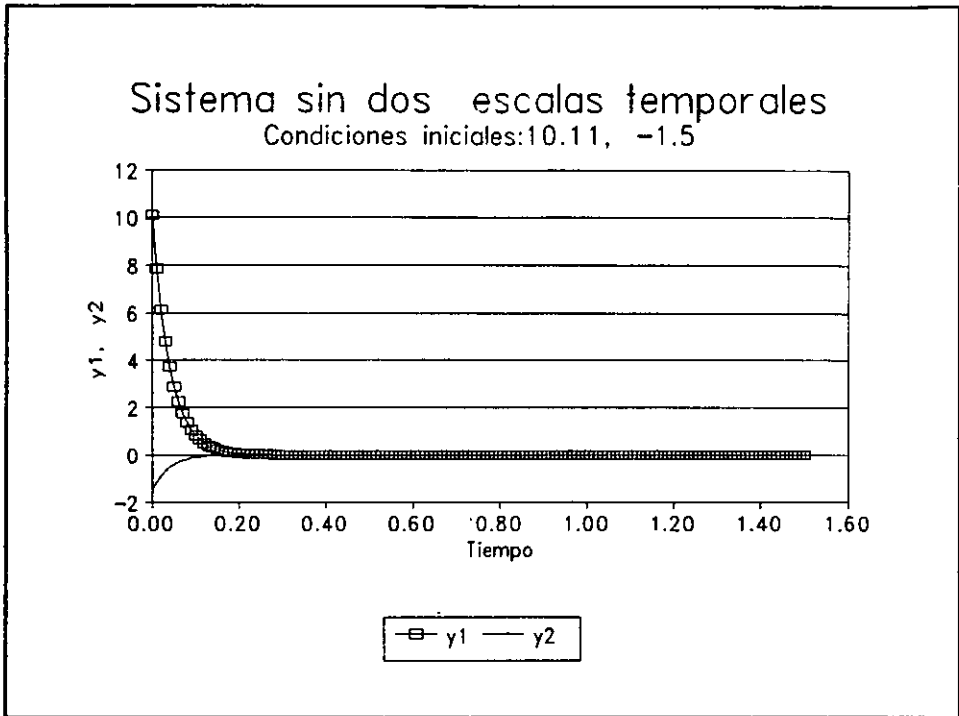


**Gráfica 1**

En forma análoga procedemos para la segunda pareja de condiciones iniciales, en éste caso se escogieron para que

$$k_1 = 10.11, \quad k_2 = 0, \tag{31}$$

de acuerdo a (27) desaparece el modo lento. La gráfica 2 presenta el comportamiento de  $y_1, y_2$ , se aprecia claramente que no existe componente lenta.





Considérese ahora el siguiente sistema

$$\begin{pmatrix} y_1' \\ y_2' \end{pmatrix} = \begin{pmatrix} -45.3912 & 23.3844 \\ -35.221 & 15.3912 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}, \quad y_0 = \begin{pmatrix} 10 \\ 1.5 \end{pmatrix}. \quad (32)$$

Los valores característicos de la matriz jacobiana de (32) y (25) son los mismos, esto es, las matrices son semejantes. La solución de (32) está dada por (pág. 51)

[17]

$$\begin{aligned} y_1 &= 18.4409e^{-25t} - 8.4418e^{-5t} \\ y_2 &= 16.0805e^{-25t} - 14.5812e^{-5t} \end{aligned} \quad (33)$$

De la definición III.4.1.1, y de (10) y (13) obtenemos la dependencia de la componente rápida  $y_1$  respecto a la lenta  $y_2$  y viceversa, estas se presentan a continuación respectivamente

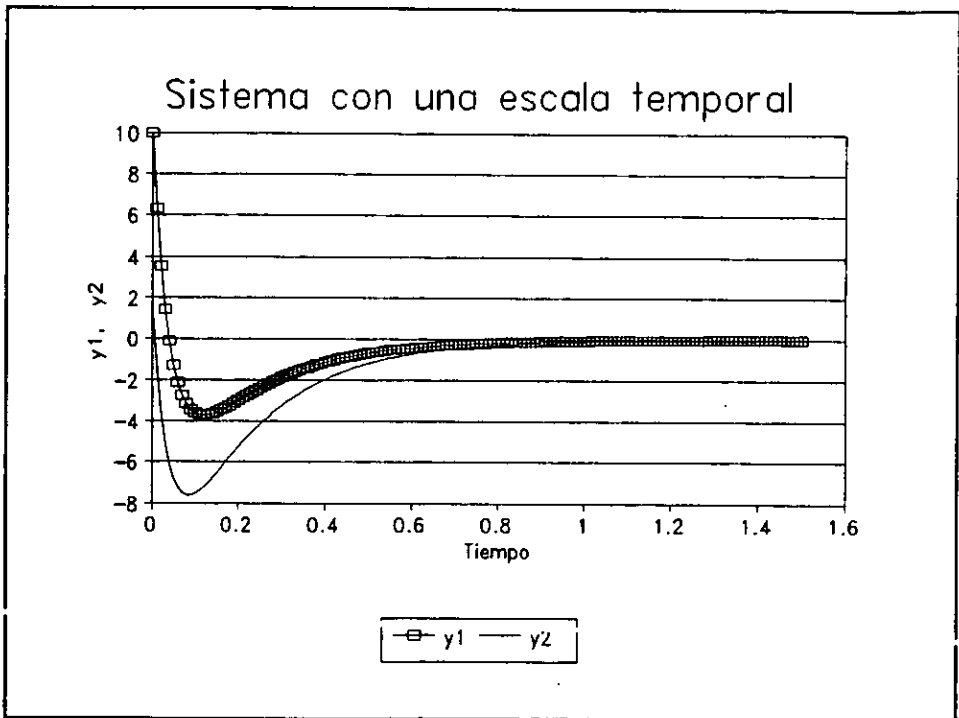
$$\left| \frac{-8.4418}{18.4409} \right| < 1 \quad (34)$$

$$\left| \frac{16.0805}{-14.5812} \right| > 1 \quad (35)$$

En consecuencia, de (34) se obtiene que el modo lento influye poco en la formación de la componente rápida. Sin embargo, de (35) se obtiene que el modo rápido influye en forma apreciable en la constitución de la componente lenta. Podemos concluir que el sistema (32) tiene una escala temporal, por lo que no se le puede aplicar un MIM

para aproximar su solución a pesar de que los valores característicos de la matriz jacobiana del sistema (32) pueden ser partidos en dos clases ajenas.

La gráfica 3 presenta el comportamiento de  $y_1$ ,  $y_2$ , se aprecia claramente que hay una sola escala temporal.



Gráfica 3

### § III.4.2 Sistemas con Dos Escalas Temporales: Caso Lineal no Homogéneo

Considérese ahora el sistema de ecuaciones (5) pero ahora incluyendo la parte no homogénea

$$y'(t) = Ay(t) + f(t); \quad y(0) = y_0 \quad (36)$$

donde  $A(t)$  es una matriz de  $n \times n$ .

La solución general de (36) es (págs.99-101)[17]

$$y(t) = e^{At} \left[ \int_0^t e^{-As} f(s) ds + K \right], \quad K \in \mathbb{R}^n \quad (37)$$

**Definición III.4.2.1.** El sistema (36) de ecuaciones diferenciales ordinarias no homogéneo es de dos escalas temporales si

- la ecuación homogénea satisface la definición III.4.1.1 y
- las componentes de la función  $f$  asociada a la parte no homogénea satisfacen

$$\left| f_i^{(0)} \right| < \left| f_r^{(0)} \right|, \quad \text{para } i = 0, 1, \dots, n \quad (38)$$

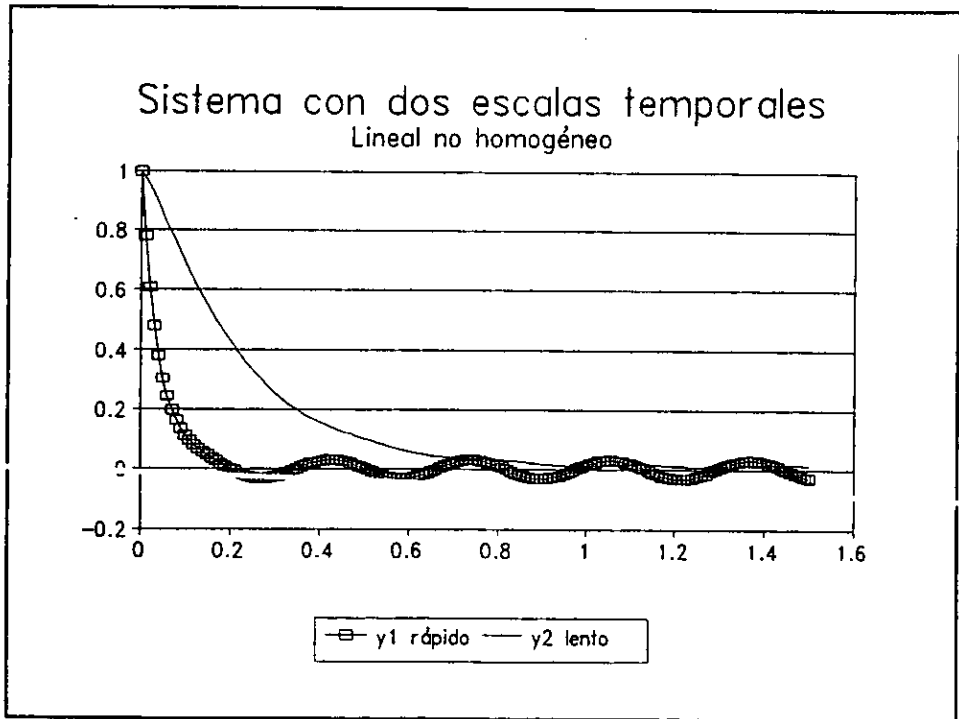
esto es, la función de excitación del subsistema rápido es más grande en magnitud y varía más rápidamente que la del subsistema lento.

El siguiente sistema de ecuaciones diferenciales ordinarias es el ejemplo dado en (25) con parte no homogénea satisfaciendo b) de la definición III.4.2.1, luego representa

un sistema lineal no homogéneo con dos escalas temporales,  $y_1$  es la componente rápida  $y_2$  es la componente lenta

$$\begin{pmatrix} y_1' \\ y_2' \end{pmatrix} = \begin{pmatrix} -25 & 0 \\ 3 & -5 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} + \begin{pmatrix} \text{sen}(20t) \\ \text{sen}(0.05t) \end{pmatrix}; \quad y_0 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}. \quad (39)$$

En la gráfica 4 se presenta el comportamiento de la componente rápida y lenta.



Gráfica 4

### § III.4. 3 Sistemas con Dos Escalas Temporales: Caso no Lineal

Sea

$$\dot{y} = f(y,t), \quad y_0 = y(0), \quad t \in [a,b] \quad (40)$$

$f: \mathbb{R}^n \times I \rightarrow \mathbb{R}^n$ ,  $I$  intervalo abierto  $(a,b)$  de  $\mathbb{R}$ , y supóngase que  $f$  puede ser aproximada por una serie de Taylor alrededor del punto  $(y(t_0), t_0)$ , es decir

$$\dot{y} = f(y(t_0), t_0) + f_y(y(t_0), t_0)(y(t) - y(t_0)) + f_t(y(t_0), t_0)(t - t_0) + R_n \quad (41)$$

donde se están considerando los dos primeros términos de la serie,  $R_n$  es el residuo en la fórmula de Taylor y  $f_y$  es la matriz jacobiana del sistema (40), esto es

$$f_y = \left( \frac{\partial f_i}{\partial y_j} \right)_{i,j=1,\dots,n} \quad (42)$$

**Definición III.4.3.1.** El sistema (40) tiene dos escalas temporales en  $I$  si para toda  $(y(t), t)$  en  $\mathbb{R}^n \times I$ , su representación lineal dado por (41) cumple con la definición III.4.2.1.

### § III.5 ANALISIS DE LOS ALGORITMOS

En esta sección se presentan los algoritmos multitasas que se analizaron para su implementación en los simuladores desarrollados en el Instituto de Investigaciones Eléctricas. Primero presentamos un resultado sobre interpolación lineal el cual se utiliza en los algoritmos que interpolan del sistema lento al rápido y posteriormente

las convenciones que se seguirán respecto a la notación utilizada en los algoritmos para no sobrecargarla.

El lema III.5.1 que se presenta a continuación, demuestra la equivalencia entre interpolación y extrapolación cuando se ésta usando el método de Euler para aproximar la solución de la ecuación  $\dot{z}=g(z,t)$ ,  $z(0)=z_0$ . Este resultado simplifica la implementación de los MIM's que utilizan el método de Euler e interpolan del sistema lento al rápido.

**Lema III.5.1.** Sea  $g(z,t)$  una función definida en una región

$$\mathfrak{R} = \{(z,t): -\infty < z < +\infty, a \leq t \leq b\}$$

continua en  $t$  y Lipschitz en  $z$ , y  $\phi$  una partición del intervalo  $[a,b]$

$$\phi = \left\{ t_j : t_j = a + jh, j=0,1,\dots,k, k = \frac{(b-a)}{h} \right\}. \quad (43)$$

Definimos  $z' = g(z,t)$ ,  $z(0) = z_0$ , y sea  $z_k$  una aproximación a  $z(t_k)$  dada por medio del algoritmo de Euler

$$z_k = z_0 + \sum_{i=0}^{k-1} h g(z_i, t_i) \quad (44)$$

Entonces si  $\hat{z}_j$ ,  $1 \leq j \leq k-1$  es la aproximación a  $z(t_j)$  por medio de una interpolación lineal a través de  $z_0$  y  $z_k$ , ésta se puede expresar como

$$\hat{z}_j = z_0 + jhg(z_0, t_0), \quad (45)$$

**Demostración.** De la fórmula de interpolación lineal (págs. 38-39)[29] se tiene

$$\begin{aligned}
\hat{z}_j &= \frac{(t_k - t_j)z_0 + (t_j - t_0)z_k}{(t_k - t_0)} \\
&= \frac{(k-j)z_0 + jz_k}{k} \\
&= z_0 + jhg(z_0, t_0)
\end{aligned} \tag{46}$$

en consecuencia el lema se cumple.

De acuerdo a la notación introducida en la sección §III.3, los algoritmos se analizan en el ciclo  $nk$ ,  $[s]$  indica la  $s$ -ésima iteración,  $f^{(s)} = f(y^{(s)}, t)$ . Por simplicidad denotamos por **P** la aplicación del predictor, por **C** la del corrector y por **E** la evaluación de la función  $f$  en términos de los argumentos conocidos. Con esta convención, la notación **P(EC)<sup>2</sup>E** significa un cálculo en el orden **PECECE**.

A continuación se presentan seis algoritmos, los tres primeros utilizan el método de Euler explícito para el sistema rápido y lento, los restantes utilizan el Euler implícito implementado en la forma **P(EC)<sup>2</sup>E** para el subsistema rápido y Euler explícito para el lento. Los algoritmos III.5.1 y III.5.4 hacen uso del lema III.5.1

**Algoritmo III.5.1.** Euler explícito para ambos subsistemas, con **interpolación lineal** del subsistema lento al rápido

$$\left. \begin{aligned}
y_{nk+i+1} &= y_{nk+i} + hf(y_{nk+i}, \hat{z}_{nk+i}, t_{nk+i}) \\
\hat{z}_{nk+i+1} &= z_{nk} + (i+1)hg(y_{nk}, z_{nk}, t_{nk})
\end{aligned} \right\} i=0,1,\dots,k-1 \tag{47}$$

$$z_{nk+k} = \hat{z}_{nk+k}$$

**Algoritmo III.5.2.** Euler explícito para ambos subsistemas, con **información adelantada**. El valor de  $\hat{z}_{nk,t}$  se toma siempre como  $z_{nk,k}$ , la información del subsistema lento al rápido no se actualiza, es decir, se toma durante cada marco del ciclo siempre el valor adelantado del subsistema lento al final del ciclo.

$$\begin{aligned} z_{nk+k} &= z_{nk} + khg(y_{nk}, z_{nk}, f_{nk}) \\ y_{nk+i+1} &= y_{nk+i} + hf(y_{nk+i}, z_{nk+k}, f_{nk+i}) \end{aligned} \quad \} i=0,1,\dots,k-1 \quad (48)$$

**Algoritmo III.5.3.** Euler explícito para ambos subsistemas, con **información atrasada**. El valor de  $\hat{z}_{nk+k}$  se toma siempre como  $z_{nk}$ , es decir, la información del subsistema lento al rápido no se actualiza y se toma siempre el valor atrasado del subsistema lento al principio del ciclo.

$$\begin{aligned} z_{nk+k} &= z_{nk} + khg(y_{nk}, z_{nk}, f_{nk}) \\ y_{nk+i+1} &= y_{nk+i} + hf(y_{nk+i}, z_{nk}, f_{nk+i}) \end{aligned} \quad \} i=0,1,\dots,k-1 \quad (49)$$

Los MIM's definidos por los algoritmos III.5.4 interpolación lineal, III.5.5 información adelantada y III.5.6 información atrasada, se presentan a continuación



$$\left. \begin{aligned}
 P: y_{nk+i+1}^{[0]} &= y_{nk+i}^{[m]} + hf(y_{nk+i}^{[m]}, \hat{z}_{nk+i}, t_{nk+i}), \\
 I: \hat{z}_{nk+i+1} &= p(t_{nk+i+1}), \\
 E: f_{nk+i+1}^{[s]} &= f(y_{nk+i+1}^{[s]}, \hat{z}_{nk+i+1}, t_{nk+i+1}), \\
 C: y_{nk+i+1}^{[s+1]} &= y_{nk+i}^{[m]} + hf_{nk+i+1}^{[s]}, \\
 E: f_{nk+i+1}^{[m]} &= f(y_{nk+i+1}^{[m]}, \hat{z}_{nk+i+1}, t_{nk+i+1}),
 \end{aligned} \right\} i=0,1,\dots,k-1 \quad (50)$$

con  $s=0,1,2,\dots,m-1$ ,  $i=0,1,2,\dots,k-1$ , los paréntesis [ ] denotan iteraciones y

$$p(t_{nk+i+1}) = \begin{cases} z_{nk} + (i+1)hg(y_{nk}^{[m]}, \hat{z}_{nk}, t_{nk}), & \text{interpolación} \\ z_{nk+k}, & \text{información adelantada} \\ z_{nk}, & \text{información atrasada} \end{cases} \quad (51)$$

definiéndose a  $z_{nk+k} = \hat{z}_{nk+k}$  al terminar el ciclo.

En los algoritmos III.5.5 y III.5.6  $z_{nk+k}$  se estima por medio del método del Euler

$$z_{nk+k} = z_{nk} + khg(y_{nk}^{[m]}, \hat{z}_{nk}, t_{nk})$$

Con la notación anterior se engloban los tres algoritmos III.5.4 **interpolación lineal del lento al rápido**, III.5.5 **información atrasada** y III.5.6 **información adelantada**, dentro de una estrategia **PI(EC)<sup>m</sup>E**.

### § III.5.1 Análisis del Número de Operaciones de los Algoritmos

Considérese el sistema de ecuaciones diferenciales ordinarias lineal homogéneo

$$\begin{aligned}y' &= A_{11}y + A_{12}z \\z' &= A_{21}y + A_{22}z\end{aligned}\tag{52}$$

donde

- $A_{11}$  es una matriz  $n_1 \times n_1$ ,
- $A_{12}$  es una matriz de  $n_1 \times n_2$ ,
- $A_{21}$  es una matriz de  $n_2 \times n_1$ ,
- $A_{22}$  es una matriz de  $n_2 \times n_2$ ,
- $y$  es una matriz de  $n_1 \times I$
- $z$  es una matriz de  $n_2 \times I$

se asume que el sistema es de dos escalas temporales donde  $y'$  representa la parte rápida y  $z'$  la lenta.

A continuación presentamos un análisis comparativo del número de multiplicaciones involucradas en los algoritmos presentados, éstos se comparan respecto del algoritmo de Euler explícito.

Los algoritmos presentados en la sección §III.5 así como los convencionales se aplican al sistema de ecuaciones (52), integrados desde  $t_0$  a  $t_k$  con paso de integración fijo  $h$ . El Euler implícito se implementa como P(EC)<sup>2</sup>E. Tomando en cuenta las siguientes relaciones presentadas en la tabla I, que se presenta a continuación:

Expresión	Número de multiplicaciones
$A_{11}y + A_{12}z$	$n_1^2 + n_1 n_2$
$h(A_{11}y + A_{12}z)$	$n_1^2 + n_1 n_2 + n_1$
$A_{21}y + A_{22}z$	$n_2^2 + n_1 n_2$
$h(A_{21}y + A_{22}z)$	$n_2^2 + n_1 n_2 + n_2$

Tabla I

se deducen el número de multiplicaciones de los algoritmos Euler explícito, implícito y de los algoritmos de integración multitasa III.5.1 hasta III.5.6, como se ve en la siguiente tabla:

Euler Convencional Explícito	$k((n_1+n_2)^2+n_1+n_2)$
Euler Convencional Implícito	$k(3((n_1+n_2)^2+n_1+n_2))$
Algoritmo 1	$k(n_1^2 + n_1 n_2 n_1 + n_2 + 1) + n_1 n_2 + n_2^2$
Algoritmo 2	$k(n_1^2 + n_1) + n_2^2 + 2n_1 n_2 + 1$
Algoritmo 3	$k(n_1^2 + n_1) + n_2^2 + 2n_1 n_2 + 1$
Algoritmo 4	$k(3n_1^2 + 2n_1 n_2 3n_1 + n_2 + 1) + n_1 n_2 + n_2^2$
Algoritmo 5	$3k(3n_1^2 + n_1) + n_2^2 + 2n_1 n_2 + 1$
Algoritmo 6	$3k(3n_1^2 + n_1) + n_2^2 + 2n_1 n_2 + 1$

En la tabla III, se presenta el número de multiplicaciones para un ejemplo hipotético, los valores dados a  $n_1$ ,  $n_2$ ,  $k$  son respectivamente 100, 200 y 8. Todos los porcentajes se comparan respecto al método de Euler explícito.

Ejemplo Hipotético		
Algoritmo Numérico	Número de Multiplicaciones	Porcentaje
Euler explícito convencional	722.440	100%
Algoritmo 1	302.408	41.86%
Algoritmo 2	161.601	22.37%
Algoritmo 3	161.601	22.37%
Euler implícito P(EC)2E convencional	2.162.400	299.32%
Algoritmo 4	624.008	86.86%
Algoritmo 5	325.001	44.99%
Algoritmo 6	325.001	44.99%

Tabla III

De las tablas presentadas se concluye que respecto al costo computacional los algoritmos multitasa representan un ahorro considerable respecto a los algoritmos convencionales. Sin embargo, es necesario que el mayor número de ecuaciones se concentre en el subsistema lento para que el ahorro en número de operaciones sea significativo.

# Capítulo IV

## Análisis de Error Local en Métodos de Integración Multitasa

En este capítulo se presentan las definiciones de error de truncamiento local, y se estiman los errores locales para los seis algoritmos presentados en el capítulo III. La notación empleada es la definida en III y se supone que  $\dim(y) + \dim(z) = 2$  para evitar recargar demasiado la notación empleada.

### § IV.1 ERROR DE TRUNCAMIENTO LOCAL PARA MIM's

Considérese el sistema de ecuaciones diferenciales ordinarias definidas por §III.3.(2), esto es

$$w'(t) = \begin{pmatrix} y' \\ z' \end{pmatrix} = \begin{pmatrix} f(y, z, t) \\ g(y, z, t) \end{pmatrix}; \quad w_0 = \begin{pmatrix} y_0 \\ z_0 \end{pmatrix} \quad (1)$$

Suponga que  $w'(t)$ ,  $t \in [a, b]$ , es un sistema con dos escalas temporales en el sentido de la definición III.4.3.1. La componente rápida está representada por  $y$ , y la lenta por  $z$ . Para aproximar la solución numérica del sistema (1) en el intervalo  $[a, b]$ , se integra a  $y'$  con un paso de integración constante  $h$ , mientras que para integrar  $z'$

se usa un paso de integración constante  $kh$ , donde  $k$  es un entero, en consecuencia el conjunto de puntos en donde se integra la componente lenta  $z$  es un subconjunto de los puntos en donde se integra la componente rápida  $y$ , cuando se da esta situación se dice que los pasos de integración están sintonizados.

Sea

$$\hat{f}(y, z, t) = f(y, \hat{z}, t) \quad (2)$$

donde  $\hat{z}(t)$  es cualquier función que aproxima a  $z(t)$ .

Considere la aplicación del siguiente MIM al sistema representado por (1), el cual está definido por los siguientes métodos multipasos lineales de  $p$  y  $m$ -pasos respectivamente

$$\begin{aligned} \sum_{j=0}^p \alpha_j y_{\sigma+j-p} &= h \sum_{j=0}^p \beta_j \hat{f}_{\sigma+j-p}, \quad p < k, \quad \sigma = nk+i+1, \quad i=0,1,\dots,k-1, \\ \sum_{l=0}^m \alpha_l^* z_{v+l} &= kh \sum_{l=0}^m \beta_l^* g_{v+l}, \quad v=(n+1+l-m)k \end{aligned} \quad (3)$$

Cada vez que se necesite evaluar  $f(y,z,t)$  se requiere de los valores interpolados de la componente lenta  $z$ .

En los métodos multitasa aquí presentados, no se está considerando interpolar de la componente rápida  $y$  hacia la lenta  $z$ , ya que se está suponiendo sintonización entre los pasos de integración y no es necesario, por lo que el MIM consiste entonces de los siguientes pasos:

1. para el tiempo  $t = t_{nk}$  se integra la componente lenta  $z$  una vez con paso  $kh$ , y
2. se integra la componente rápida  $y$ ,  $k$  veces con un paso de integración  $h$ .

Con el paso 1 y 2 se completa un ciclo de amplitud  $kh$  con  $k$  marcos de longitud  $h$ , avanzando la integración al tiempo  $t_{nk+k} = t_{nk} + kh$ ,  $\{t_{nk} = a + nkh : n=0, 1, 2, \dots, N, Nkh=b-a\}$ . El ciclo es repetido hasta alcanzar el final de la simulación. Es necesario entonces, interpolar del subsistema lento al rápido por cada marco.

A continuación se definen los siguientes operadores de diferencias asociados al método multitasa (3), que se utilizan en la definición de error de truncamiento local y global para los MIM,

$$\begin{aligned} \mathcal{Q}_1[y(t);h] &= \sum_{j=0}^p [\alpha_j y(t+(k+j-p)h) - h\beta_j y'(t+(k+j-p)h)], \\ \mathcal{Q}_2[z(t);kh] &= \sum_{l=0}^m [\alpha_l^* z(t+\nu h) - kh\beta_l^* z'(t+\nu h)], \quad \nu=(n+1+l-m)k \end{aligned} \quad (4)$$

donde  $y(t)$ ,  $z(t)$  son funciones arbitrarias, continuamente diferenciables en el intervalo de integración  $[a,b]$ .

**Definición IV.1.1.** El error de truncamiento local introducido en un ciclo de longitud  $kh$ , correspondiente al  $k$ -ésimo marco, para el MIM definido por (3) es

$$T_{nk+k} = \begin{pmatrix} T_1 \\ T_2 \end{pmatrix} = \begin{pmatrix} \mathcal{Q}_1[y(t_{nk});h] \\ \mathcal{Q}_2[z(t_{nk});kh] \end{pmatrix} \quad (5)$$

donde  $y(t)$ ,  $z(t)$  son las soluciones teóricas del sistema (1).

Considere la aplicación de (3) en  $[a,b]$  al sistema (1), con las siguientes suposiciones locales:

$$\begin{aligned} y_{\sigma+j} &= y(t_{\sigma+j}), \quad \sigma = nk+i+1, \\ z_{v+l} &= z(t_{v+l}), \quad v = (n+1+l-m)k \end{aligned} \quad (6)$$

con  $i=0,1,2,\dots,k-1$ ,  $j=0,1,2,\dots,p-1$ ,  $l=0,1,2,\dots,m-1$ . Además, se está suponiendo que no se ha cometido ningún error de truncamiento previo, y se sustituye  $\hat{f}$  por

$$\bar{f} = f(y, \bar{z}, t), \quad (7)$$

donde la aproximación  $\bar{z}(t)$  a  $z(t)$  usa los valores exactos en lugar de los aproximados.

A continuación se procede a encontrar el error de truncamiento local para el MIM definido por (3). Con las suposiciones anteriores (6), (7), se analiza primero la componente rápida  $y$ , entonces de (3) para  $i=k-1$  y de (4), se obtiene

$$\sum_{j=0}^p \alpha_j y(t_{nk+k-p+j}) = h \sum_{j=0}^p \beta_j f(y(t_{nk+k-p+j}), z(t_{nk+k-p+j}), t_{nk+k-p+j}) + \mathcal{O}_y[y(t_{nk}); h] \quad (8)$$

donde  $y(t)$  es la solución teórica. El valor de  $y_{nk+k}$  dado por (3) satisface para  $i = k-1$

$$\sum_{j=0}^p \alpha_j y_{nk+k-p+j} = h \sum_{j=0}^p \beta_j f(y_{nk+k-p+j}, \bar{z}_{nk+k-p+j}, t_{nk+k-p+j}) \quad (9)$$

Restando (9) de (8), y de acuerdo a las suposiciones locales dadas por (6) y (7), se obtiene



$$\begin{aligned}
y(t_{nk+k}) - y_{nk+k} &= h[\beta_p(f(y(t_{nk+k}), z(t_{nk+k}), f_{nk+k}) - f(y_{nk+k}, z_{nk+k}, f_{nk+k})) + \\
&\sum_{j=0}^{p-1} [\beta_j(f(y(t_{nk+k-p+j}), z(t_{nk+k-p+j}), f_{nk+k-p+j}) - f(y_{nk+k-p+j}, z_{nk+k-p+j}, f_{nk+k-p+j}))]] + \\
&\mathcal{L}_y[y(t_{nk}); h]
\end{aligned} \tag{10}$$

donde se ésta asumiendo que  $\bar{z}_{nk+k} = z_{nk+k}$ .

Por el teorema del valor medio y de (6), (7), se obtiene

$$\begin{aligned}
f(y(t_{nk+k-p+j}), z(t_{nk+k-p+j}), f_{nk+k-p+j}) - f(y_{nk+k-p+j}, \bar{z}_{nk+k-p+j}, f_{nk+k-p+j}) = \\
[z(t_{nk+k-p+j}) - \bar{z}_{nk+k-p+j}] f_z, \quad j=0, 1, \dots, p-1,
\end{aligned}$$

donde la  $f_z$  se evalúa en el vector

$$\xi_j = (y_{nk+k-p+j}, \eta_{nk+k-p+j}^1, f_{nk+k-p+j})^T, \tag{12}$$

y  $\eta_{nk+k-p+j}^1$  es un punto interior del intervalo cuyos extremos son  $\bar{z}_{nk+k-p+j}$  y  $z(t_{nk+k-p+j})$ .

En forma análoga, del teorema de valor medio se obtiene

$$f(y(t_{nk+k}), z(t_{nk+k}), f_{nk+k}) - f(y_{nk+k}, z_{nk+k}, f_{nk+k}) = [y(t_{nk+k}) - y_{nk+k}] f_y + [z(t_{nk+k}) - z_{nk+k}] f_z, \tag{13}$$

en donde las  $f_y$ ,  $f_z$  se evalúan en el vector

$$\xi = (\eta_{nk+k}^2, \eta_{nk+k}^3, f_{nk+k})^T, \tag{14}$$

y  $\eta_{nk+k}^2$ ,  $\eta_{nk+k}^3$  son puntos interiores de los intervalos cuyos extremos son

$y_{nk+k}$ ,  $y(t_{nk+k})$ ,  $z_{nk+k}$ ,  $z(t_{nk+k})$  respectivamente.

Luego

$$(1-h\beta_p f_y(\xi)) [y(t_{nk+k}) - y_{nk+k}] - h\beta_p [z(t_{nk+k}) - z_{nk+k}] f_z(\xi) - h \sum_{j=0}^{p-1} \beta_j [z(t_{nk+k-p+j}) - \tilde{z}_{nk+k-p+j}] f_z(\xi_j) = \mathcal{L}_y [y(t_{nk}); h], \quad (15)$$

que es la expresión del error de truncamiento local para la componente rápida  $y$ . Procediendo en forma análoga para la componente lenta  $z$ , se obtiene el error de truncamiento local de ésta

$$(1-kh\beta_m^* g_z(\rho)) [z(t_{nk+k}) - z_{nk+k}] - kh\beta_m^* [y(t_{nk+k}) - y_{nk+k}] g_y(\rho) = \mathcal{L}_z [z(t_{nk}); kh], \quad (16)$$

$$\rho = (\zeta_{nk+k}^1, \zeta_{nk+k}^2, t_{nk+k})^T,$$

$\zeta_{nk+k}^1, \zeta_{nk+k}^2$  son puntos interiores de los intervalos cuyos extremos son  $y_{nk+k}, y(t_{nk+k}), z_{nk+k}, z(t_{nk+k})$  respectivamente.

Se concluye entonces de (15) y (16), que para un MIM explícito ( $\beta_p=0$  y/o  $\beta_m^*=0$ ), esto es

$$(y(t_{nk+k}) - y_{nk+k}) - h \sum_{j=0}^{p-1} \beta_j [z(t_{nk+k-p+j}) - \tilde{z}_{nk+k-p+j}] f_z(\xi_j) = \mathcal{L}_y [y(t_{nk}); h],$$

$$z(t_{nk+k}) - z_{nk+k} = \mathcal{L}_z [z(t_{nk}); kh],$$

el error de truncamiento local para la componente rápida  $y$  es la diferencia de la solución teórica y la solución dada por el método multipaso para la componente rápida  $y$ , más la contribución del error introducido por el polinomio de interpolación

de la componente lenta  $z$  a la rápida  $y$  por cada marco, multiplicado por el grado de acoplamiento del subsistema rápido respecto al lento, bajo las suposiciones locales dadas en (6) y (7), y para la componente lenta  $z$  es la diferencia de la solución teórica y la solución dada por el método multipaso de la componente lenta, bajo las suposiciones locales dadas en (6) y (7).

En el caso de los **MIM** implícitos el error de truncamiento local para la componente rápida  $y$ , y lenta  $z$ , se conforma del error de truncamiento local asociado al **MIM** explícito, más la contribución del error originado por la parte no explícita, el cual es, la diferencia de la solución teórica y la solución dada por el método multipaso para la componente rápida  $y$ , y lenta  $z$  por sus respectivas derivadas parciales de  $f$  respecto a  $y$ , y  $z$  en el caso de la componente rápida ( $g$  respecto a  $y$ , y  $z$  en el caso de la componente lenta).

Finalmente, se concluye que el error de truncamiento local de un **MIM** definido en (3), está formado por el error de truncamiento del método multipaso lineal, el error del polinomio de interpolación y los coeficientes de acoplamiento definidos por las derivadas parciales del subsistema rápido y lento.

Considérese el siguiente sistema de dos ecuaciones lineal homogéneo:

$$\begin{pmatrix} y' \\ z' \end{pmatrix} = \begin{pmatrix} a_{11} & \mu \\ \epsilon & a_{22} \end{pmatrix} \begin{pmatrix} y \\ z \end{pmatrix} = \begin{pmatrix} f(y,z,t) \\ g(y,z,t) \end{pmatrix}, \quad (17)$$

se asume que la componente rápida es  $y$ , y la lenta es  $z$ .

Entonces

$$\begin{aligned} f_y &= a_{11}, & f_z &= \mu, \\ g_y &= \epsilon, & g_z &= a_{22}. \end{aligned} \tag{18}$$

Sustituyendo (18) en las ecuaciones (15) y (16) de los errores de truncamiento local de la componente rápida  $y$ , y lenta  $z$  del sistema (17), se deduce que:

1. Sí no existe acoplamiento entre la componente rápida  $y$ , y lenta  $z$ , esto es cuando  $\epsilon=0$  y  $\mu=0$ , al sustituir estos valores en (15) y (16) obtenemos que los errores de truncamiento local para ambas componentes se reducen a los de los métodos multipasos lineales convencionales [19], es decir

$$(1-h\beta_p f_y)[y(t_{nk+k})-y_{nk+k}] = \mathcal{Q}_y[y(t_{nk});h], \tag{19}$$

y

$$(1-kh\beta_m^* g_z)[z(t_{nk+k})-z_{nk+k}] = \mathcal{Q}_z[z(t_{nk});kh]. \tag{20}$$

Luego, el comportamiento de los errores locales en ambas componentes son independientes uno del otro y la integración del sistema (17) por medio de un **MIM** es adecuado.

2. Sí existe acoplamiento entre ambas componentes, esto es  $\epsilon \neq 0$  y  $\mu \neq 0$ , entonces de (15) y (16) obtenemos que los errores de truncamiento local dependen de  $\mu$  en el caso de la componente rápida  $y$ , y de  $\epsilon$  para la componente lenta. Luego, si las magnitudes de  $\epsilon$  y  $\mu$  son "grandes", entonces los errores de truncamiento local para las componentes rápida  $y$ , y lenta  $z$  son también "grandes" en magnitud. En éste caso no es adecuado integrar el sistema (17) con un **MIM**.

Aún cuando  $\epsilon$  y  $\mu$  sean en magnitud pequeños hay que evaluar cuidadosamente la aplicación de un **MIM** al sistema (17).

3. Sí no existe acoplamiento de la componente rápida  $y$  con la lenta  $z$ , y el acoplamiento de la componente lenta con la rápida es "débil", esto es cuando  $\mu=0$  y  $\epsilon$  es de magnitud moderada, el error de truncamiento local de la componente rápida  $y$  está dado por (19) y no es necesario interpolar de la componente lenta  $z$  a la rápida  $y$ ; consecuentemente éste no se ve afectado por los errores de la componente lenta  $z$ . En el caso del error de truncamiento local de la componente lenta  $z$ , éste tiene la contribución adicional, respecto al error de truncamiento local de los métodos multipasos convencionales, de

$$kh\beta_m^*[y(t_{nk,k})-y_{nk,k}]\epsilon, \quad (21)$$

sí  $\beta_m^* \neq 0$ . Este término es pequeño en magnitud sí la magnitud de  $\epsilon$  es moderada, y no existe sí  $\beta_m^* = 0$ . Luego, el error en la componente lenta  $z$  está dominado por (20). En éste caso, el comportamiento de los errores de truncamiento local de ambas componentes es similar al caso desacoplado, siendo entonces adecuada la integración del sistema (17) por un **MIM**.

4. Sí existe acoplamiento entre la componente rápida  $y$ , y la lenta  $z$ , y no existe acoplamiento de la componente lenta  $z$  con la rápida  $y$ , esto es cuando  $\mu \neq 0$  y  $\epsilon = 0$ , el error de truncamiento local de la componente lenta  $z$  está dado por (20). El error de truncamiento local para la componente rápida  $y$  tiene la contribu-

ción adicional, respecto al error de truncamiento local de los métodos multipasos convencionales, de

$$h\beta_p[z(t_{nk+k}) - z_{nk+k}]\mu, \quad (22)$$

sí  $\beta_p \neq 0$ , y de

$$h \sum_{j=0}^{p-1} \beta_j [z(t_{nk+k-p+j}) - \bar{z}_{nk+k-p+j}]\mu. \quad (23)$$

Luego, el error en la componente rápida  $y$  tiene la contribución adicional de (22) y (23) sí  $\beta_p \neq 0$  y de (23) únicamente sí  $\beta_p = 0$ . En éste caso podría no ser adecuado la integración del sistema (17) por medio de un MIM, aún cuando la magnitud de  $\mu$  sea pequeña, ya que el error de truncamiento local de la componente rápida  $y$  siempre se ve afectado por los errores cometidos en la componente lenta  $z$ .

Regresemos al caso general, si además, suponemos que las soluciones teóricas  $y(t)$ ,  $z(t)$  del sistema (1) son de clase  $\mathcal{E}^s$ , para  $s$  suficientemente grande, entonces se deduce de (10) y (16) que

$$\begin{aligned} y(t_{nk+k}) - y_{nk+k} &= C_{qr+1} h^{qr+1} y^{(qr+1)}(t_{nk}) + O(h^{qr+2}) + O(h(kh)^{qr+2}) + O(h^{qr+2}), \\ z(t_{nk+k}) - z_{nk+k} &= D_{ql+1} (kh)^{ql+1} z^{(ql+1)}(t_{nk}) + O((kh)^{ql+2}) + O(h^{qr+2}) + O(h^{ql+3}), \end{aligned} \quad (24)$$

con

$qr$  es el orden del método multipaso de la componente rápida  $y$ ,

$ql$  es el orden del método multipaso de la componente lenta  $z$ ,

$qi$  es el orden del polinomio de interpolación empleado,

$C_{qr}, D_{qi}$  son constantes.

Sea  $q = \min(qr, ql, qi+1)$ ,  $I=[a,b]$ , y definimos a

$$M_1 = |C_{qr+1}| \max_{t \in I} |y(t)^{(qr+1)}|, \quad (25)$$

$$M_2 = |D_{qi+1}| \max_{t \in I} |z(t)^{(qi+1)}|,$$

$$M = \max \{M_1+M_3+M_4+M_5, M_2+M_3+M_6+M_7\}, \quad (26)$$

las constantes  $M_3, M_4, M_5, M_6, M_7$  son las asociadas a  $O(h^{qr+2}), O(h(kh)^{qi+2}), O(h^{qi+2}), O((kh)^{qi+2}), O(h^{qi+3})$ .

Entonces

$$\|T_{nk+1}\| \leq Mh^{q+1}, \quad (27)$$

donde la norma utilizada es la del máximo y  $T_{nk+1}$  está definida por (5).

De (27), se define el orden del método multitasa (3) como

$$q = \min(qr, ql, qi+1),$$

es decir el orden de un MIM es el mínimo de los ordenes de los métodos multipasos lineales empleados y del orden del polinomio de interpolación incrementado en uno, esto debido a que el orden del método de interpolación podría tener un error local de una potencia de  $h$  menor que la del método de integración. Lo anterior es posible ya que el polinomio de interpolación  $\hat{z}$  usado está en términos de  $hg(y, \hat{z}, t)$ , por lo que se incrementa  $h$  en una potencia.

§ IV.1.1 Error de Truncamiento Local Algoritmos III.5.1, III.5.2, III.5.3.

A continuación se estiman los errores de truncamiento local para los algoritmos III.5.1 (interpolación del lento al rápido), III.5.2 (información adelantada) y III.5.3 (información atrasada).

**Teorema IV.1.1.1.** Sean  $y(t)$ ,  $z(t)$  las soluciones teóricas del sistema (1), y suponga que son de clase  $\mathcal{C}^s$ ,  $s$  lo suficientemente grande, entonces los errores de truncamiento local de los MIM definidos por los algoritmos III.5.1, III.5.2, y III.5.3 son respectivamente

$$T_y = \begin{cases} k \frac{h^2}{2} y''(t_{nk}) + O(h^3), \\ k \frac{h^2}{2} \bar{y}_2''(t_{nk}) + \frac{(kh)^2}{2} (y''(t_{nk}) - \bar{y}_2''(t_{nk})) - (k^2 f_z(\zeta) + f_z(\xi_0)) h^2 z'(t_{nk}) + O(h^3), \\ k \frac{h^2}{2} \bar{y}_3''(t_{nk}) + \frac{(kh)^2}{2} (y''(t_{nk}) - \bar{y}_3''(t_{nk})) + (k-1) h^2 z'(t_{nk}) f_z(\xi_0) + O(h^3), \end{cases} \quad (28)$$

$$T_z = \frac{(kh)^2}{2} z''(t_{nk}) + O(h^3), \quad (29)$$

donde  $\bar{y}_l$ , para  $l = 1, 2, 3$ , se definen a continuación:

$$\begin{aligned} \bar{y}_1'(t) &= f(y(t), \bar{z}(t), t), \\ \bar{y}_2'(t) &= f(y(t), \bar{z}(t_{nk+k}), t), \\ \bar{y}_3'(t) &= f(y(t), \bar{z}(t_{nk}), t), \end{aligned} \quad (30)$$

**Demostración.** Las expresiones generales de los errores de truncamiento local de las componente rápida  $y$ , y lenta  $z$ , para un MIM del tipo definido por (3), están dados



por (15) y (16). Los algoritmos III.5.1, III.5.2 y III.5.3 son MIM's del tipo definido por (3), siendo sus coeficientes y parámetros los siguientes

Coeficientes Método Componente Rápida y				
$p$	$\alpha_0$	$\alpha_1$	$\beta_0$	$\beta_1$
1	-1	1	1	0

Tabla 1

Coeficientes Método Componente Lenta z				
$m$	$\alpha_0^z$	$\alpha_1^z$	$\beta_0^z$	$\beta_1^z$
1	-1	1	1	0

Tabla 2

Sustituyendo estos coeficientes y parámetros en (15), (16) y junto con las suposiciones locales definidas en (6), (7), se obtiene

$$\begin{aligned} \mathcal{L}[y(t_{nk}); h] &= y(t_{nk+k}) - y_{nk+k} - h[z(t_{nk+k-1}) - \bar{z}_{nk+k-1}]M_2(\xi_0), \\ \mathcal{L}[z(t_{nk}); kh] &= z(t_{nk+k}) - z_{nk+k} \end{aligned} \quad (31)$$

donde  $\bar{z}_{nk+k-1} = p(t_{nk+k-1})$  y

$$p(t) = \begin{cases} \frac{(t_{nk+k}-t)z(t_{nk})+(t-t_{nk})z(t_{nk+k})}{t_{nk+k}-t_{nk}}, & \text{interpolación} \\ z(t_{nk+k}), & \text{información adelantada} \\ z(t_{nk}), & \text{información atrasada} \end{cases} \quad (32)$$

para toda  $t \in [t_{nk}, t_{nk+k}]$ , y  $\xi_0$  esta definida por (12),  $p(t)$  es el polinomio de interpolación empleado,  $\bar{z}$  se evalúa en los valores exactos en lugar de los aproximados.

Las expresiones dadas en (31) son los errores de truncamiento local para la componente rápida  $y$ , y lenta  $z$  de los MIM definidos por algoritmos III.5.1, III.5.2 y III.5.3. El término

$$h[z(t_{nk+k-1}) - \bar{z}_{nk+k-1}]f'_z(\xi_0), \quad (33)$$

presente en (31) es el error adicional introducido por el MIM, éste se encuentra en función del error del algoritmo de interpolación empleado, así como del grado de acoplamiento del subsistema rápido respecto al lento  $z$ .

Desarrollando en serie de Taylor a la solución teórica  $y(t_{nk+k})$  alrededor del punto  $t_{nk}$  y teniendo en cuenta que  $t_{nk+k} = t_{nk} + kh$ , se obtiene

$$y(t_{nk+k}) = y(t_{nk}) + kh y'(t_{nk}) + \frac{(kh)^2}{2} y''(t_{nk}) + O(h^3), \quad (34)$$

Considérese ahora el algoritmo III.5.1, entonces para  $i = 0$ , se obtiene para la componente rápida

$$y_{nk+1} = y_{nk} + hf(y_{nk}, \bar{z}_{nk}, f_{nk}), \quad (35)$$

Para simplificar las expresiones definimos a

$$\bar{y}'_1(t) = f(y(t), \bar{z}(t), t) \quad (36)$$

Sustituyendo (36) en (35) y con las suposiciones dadas en (6), (7), esto es,  $y_{nk+i} = y(t_{nk+i})$ ,  $i = 0, 1, 2, \dots, k-1$ , y  $\bar{z}(t_{nk}) = \bar{z}_{nk}$ , se obtiene

$$\begin{aligned} y_{nk+1} &= y_{nk} + h\bar{y}'_1(t_{nk}), \\ y_{nk+2} &= y_{nk+1} + h\bar{y}'_1(t_{nk+1}) \\ &= y_{nk} + h\bar{y}'_1(t_{nk}) + h\bar{y}'_1(t_{nk+1}). \end{aligned} \quad (37)$$

Desarrollando en serie de Taylor a  $\bar{y}'_1(t_{nk+1})$  alrededor del punto  $t_{nk}$  y ya que  $t_{nk+1} = t_{nk} + h$ , se encuentra

$$\bar{y}'_1(t_{nk+1}) = \bar{y}'_1(t_{nk}) + h\bar{y}''_1(t_{nk}) + \frac{h^2}{2}\bar{y}'''_1(t_{nk}) + O(h^3). \quad (38)$$

Sustituyendo (38) en (37), se deduce

$$\begin{aligned} y_{nk+2} &= y_{nk} + 2h\bar{y}'_1(t_{nk}) + h^2\bar{y}''_1(t_{nk}) + O(h^3) \\ &= y_{nk} + 2h\bar{y}'_1(t_{nk}) + 2(2-1)\frac{h^2}{2}\bar{y}''_1(t_{nk}) + O(h^3). \end{aligned} \quad (39)$$

Procediendo en forma análoga para  $i=1, 2, \dots, k-1$ , se obtiene

$$y_{nk+k} = y_{nk} + kh\bar{y}'_1(t_{nk}) + k(k-1)\frac{h^2}{2}\bar{y}''_1(t_{nk}) + O(h^3). \quad (40)$$

Aplicando la regla de la cadena a  $y'(t) = f(y(t), z(t), t)$ , se encuentra

$$y''(t) = f_y y'(t) + f_z z'(t) + f_t, \quad (41)$$

donde todas las parciales están evaluadas en el punto  $(y(t), z(t), t)$ .

Análogamente, aplicando la regla de la cadena a (36), se obtiene

$$\tilde{y}_1''(t) = f_y y'(t) + f_z \tilde{z}'(t) + f_p \quad (42)$$

pero en este caso todas las parciales están evaluadas en el punto  $(y(t), \tilde{z}(t), t)$ .

**Primer caso algoritmo III.5.1: Interpolación del Lento al Rápido:**

En este caso se está suponiendo que en  $t = t_{nk}$  se cumple

$$z(t_{nk}) = \tilde{z}(t_{nk}) \quad (43)$$

Entonces, se concluye de (30) que para  $t = t_{nk}$

$$y'(t_{nk}) = \tilde{y}'_1(t_{nk}), \quad (44)$$

y de (41) y (42) para  $t = t_{nk}$  se obtiene

$$y''(t_{nk}) = \tilde{y}''_1(t_{nk}) \quad (45)$$

Luego, restando (40) de (34), y teniendo en cuenta a (44) y (45), se obtiene

$$y(t_{nk+h}) - y_{nk+h} = k \frac{h^2}{2} y''(t_{nk}) + O(h^3). \quad (46)$$

Por otra parte, del teorema del error general del polinomio de interpolación (págs. 49-52)[29] y del teorema del valor medio se encuentra

$$\begin{aligned}
 z(t_{nk+k-1}) - \bar{z}_{nk+k-1} &= (t_{nk+k-1} - t_{nk})(t_{nk+k-1} - t_{nk+k}) \frac{z''(\zeta_1)}{2} \\
 &= -(k-1)h^2 \frac{z''(\zeta_1)}{2},
 \end{aligned}
 \tag{47}$$

donde  $\zeta_1$  es un punto intermedio entre  $t_{nk}$  y  $t_{nk+k}$ , multiplicando ambos miembros por  $h$  y  $f_z$ , se obtiene

$$h[z(t_{nk+k-1}) - \bar{z}_{nk+k-1}] f_z(\xi_0) = -(k-1)h^3 \frac{z''(\zeta)}{2} f_z(\xi_0),
 \tag{48}$$

Sustituyendo (46) y (48) en (31) se encuentra la expresión del error de truncamiento local para la componente rápida y del algoritmo III.5.1, esto es

$$T_y = k \frac{h^2}{2} y''(t_{nk}) + O(h^3).
 \tag{49}$$

### Segundo caso algoritmo III.5.2: Información Adelantada

Considérese ahora el algoritmo III.5.2, entonces para  $i=0$ , se obtiene para la componente rápida y

$$y_{nk+1} = y_{nk} + hf(y_{nk}, z_{nk+k}, t_{nk})
 \tag{50}$$

Sea

$$\bar{y}_2'(t) = f(y(t), z(t_{nk+k}), t),
 \tag{51}$$

donde  $z(t_{nk+k}) = z_{nk+k}$  es constante para toda  $t$ .

Procediendo en forma análoga al caso del algoritmo III.5.1 para obtener (40), se encuentra

$$y_{nk+k} = y_{nk} + kh\bar{y}'_2(t_{nk}) + k(k-1)\frac{h^2}{2}\bar{y}''_2(t_{nk}) + O(h^3). \quad (52)$$

Aplicando la regla de la cadena a (51), se deduce

$$\bar{y}'_1(t) = f_y y'(t) + f_p \quad (53)$$

las derivadas parciales se evalúan en el punto  $(y(t), z(t_{nk+k}), t)$  para toda  $t$ .

En consecuencia

$$\begin{aligned} y'(t_{nk}) & \neq \bar{y}'_2(t_{nk}), \\ y''(t_{nk}) & \neq \bar{y}''_2(t_{nk}). \end{aligned} \quad (54)$$

Restando (52) de (40) y teniendo en cuenta a (54), se obtiene

$$y(t_{nk+k}) - y_{nk+k} = kh(y'(t_{nk}) - \bar{y}'_2(t_{nk})) + \frac{(kh)^2}{2}(y''(t_{nk}) - \bar{y}''_2(t_{nk})) + k\frac{h^2}{2}\bar{y}''_2(t_{nk}) + O(h^3). \quad (55)$$

Desarrollando en serie de Taylor a  $z(t_{nk+k,l})$  alrededor del punto  $t_{nk}$ , y ya que

$t_{nk+k,l} = t_{nk} + (k-1)h$  se obtiene

$$z(t_{nk+k-1}) = z(t_{nk}) + (k-1)hz'(t_{nk}) + O(h^2) \quad (56)$$

De la definición del algoritmo III.5.2 se tiene

$$z_{nk+k} = z_{nk} + khz'(t_{nk}) \quad (57)$$

Restando (57) de (56), y ya que  $z_{nk} = z(t_{nk})$ , se encuentra

$$\begin{aligned}
 z(t_{nk+k-1}) - \bar{z}_{nk+k-1} &= z(t_{nk+k-1}) - p(t_{nk+k-1}) \\
 &= z(t_{nk+k-1}) - z_{nk+k} \\
 &= -hz'(t_{nk}) + O(h^2)
 \end{aligned}
 \tag{58}$$

Multiplicando ambos miembros de (58) por  $h$  y  $f_z$  se obtiene

$$h[z(t_{nk+k-1}) - z_{nk+k}]f_z = -h^2 z'(t_{nk})f_z + O(h^3) \tag{59}$$

Por el teorema del valor medio, teniendo en cuenta que  $\bar{z}(t_{nk+k}) = z_{nk+k}$ ,  $z(t_{nk}) = z_{nk}$  y de

(57), se encuentra

$$\begin{aligned}
 y'(t_{nk}) - \bar{y}'_2(tnk) &= f(y(t_{nk}), z(t_{nk}), t_{nk}) - f(y(t_{nk}), \bar{z}(t_{nk+k}), t_{nk}) \\
 &= (z(t_{nk}) - \bar{z}(t_{nk+k}))f'_z(\zeta) \\
 &= -khz'(t_{nk})f'_z(\zeta)
 \end{aligned}$$

donde  $\zeta$  es un punto interior del intervalo cuyos extremos son  $z_{nk+k}$ ,  $z_{nk}$ .

Multiplicando por  $kh$  ambos miembros de la expresión anterior se obtiene

$$kh(y'(t_{nk}) - \bar{y}'_2(tnk)) = -(kh)^2 z'(t_{nk})f'_z(\zeta) \tag{60}$$

Finalmente sustituyendo (55), (59) y (60) en (31) se encuentra la expresión del error de truncamiento local correspondiente a la componente rápida y del algoritmo III.5.2, esto es

$$T_y = \frac{(kh)^2}{2} (y''(t_{nk}) - \bar{y}''_2(t_{nk})) + k \frac{h^2}{2} \bar{y}''_2(t_{nk}) - (k^2 f'_z(\zeta) + f'_z(\xi_0)) h^2 z'(t_{nk}) + O(h^3). \tag{61}$$

### Tercer caso algoritmo III.5.3: Información Atrasada

Considérese el algoritmo III.5.3, entonces para  $i = 0$  para la componente rápida  $y$ , se obtiene

$$y_{nk+1} = y_{nk} + hf(y_{nk}, z_{nk+k}, t_{nk}) \quad (62)$$

Sea

$$\bar{y}'_3(t) = f(y(t), z(t_{nk}), t), \quad (63)$$

donde  $z(t_{nk}) = z_{nk}$  es constante para toda  $t$ .

Procediendo en forma análoga a los dos casos anteriores, se obtiene

$$y_{nt+k} = y_{nk} + kh\bar{y}'_3(t_{nk}) + k(k-1)\frac{h^2}{2}\bar{y}''_3(t_{nk}) + O(h^3). \quad (64)$$

Además, en este caso se tiene

$$\begin{aligned} y'(t_{nk}) &= \bar{y}'_3(t_{nk}), \\ y''(t_{nk}) &= \bar{y}''_3(t_{nk}), \end{aligned} \quad (65)$$

para  $t = t_{nk}$ .

Restando (64) de (34) y junto con (65), se obtiene

$$y(t_{nk+k}) - y_{nk+k} = \frac{(kh)^2}{2}(y''(t_{nk}) - \bar{y}''_3(t_{nk})) + k\frac{h^2}{2}\bar{y}''_3(t_{nk}) + O(h^3). \quad (66)$$

De (56) se deduce



$$\begin{aligned}
 z(t_{nk+k-1}) - \bar{z}_{nk+k-1} &= z(t_{nk+k-1}) - P(t_{nk+k-1}) \\
 &= z(t_{nk+k-1}) - z_{nk} \\
 &= (k-1)hz'(t_{nk}) + O(h^2)
 \end{aligned}
 \tag{67}$$

ya que  $z(t_{nk}) = z(t_{nk})$ .

Multiplicando ambos miembros de (67) por  $h$  y  $f_z$ , se encuentra

$$h[z(t_{nk+k-1}) - z_{nk}]f_z(\xi_0) = (k-1)h^2z'(t_{nk})f_z(\xi_0) + O(h^3) \tag{68}$$

Sustituyendo (66) y (68) en (31) se encuentra la expresión del error de truncamiento local correspondiente a la componente rápida  $y$  del algoritmo III.5.3, esto es

$$T_y = \frac{(kh)^2}{2}(y''(t_{nk}) - \bar{y}_3''(t_{nk})) + k\frac{h^2}{2}\bar{y}_3''(t_{nk}) + (k-1)h^2z'(t_{nk})f_z(\xi_0) + O(h^3). \tag{69}$$

Finalmente, el error de truncamiento local para la componente lenta  $z$  es el mismo para los tres algoritmos. Este se obtiene desarrollando en serie de Taylor a la solución teórica  $z(t_{nk+k})$  alrededor del punto  $t_{nk}$  y se le resta la ecuación de diferencias definida por  $z_{nk+k}$ , esto es

$$z(t_{nk+k}) = z(t_{nk}) + khz'(t_{nk}) + \frac{(kh)^2}{2}z''(t_{nk}) + O(h^3), \tag{70}$$

$$z_{nk+k} = z_{nk} + khz'_{nk} \tag{71}$$

Restando (71) de (70), se obtiene

$$\begin{aligned}
 T_z &= z(t_{nk+k}) - z_{nk+k} \\
 &= \frac{(kh)^2}{2}z''(t_{nk}) + O(h^3),
 \end{aligned}
 \tag{72}$$

con lo cual el teorema queda demostrado.

**§ IV.1.2 Error de Truncamiento Local Algoritmos III.5.4, III.5.5, III.5.6.**

En los algoritmos III.5.4 (interpolación del lento al rápido), III.5.5 (información adelantada), III.5.6 (información atrasada) se proponen **MIM**'s en los cuales la componente rápida  $y$  se integra por medio de un Euler implícito implementado por medio de una estrategia de la forma **PI(EC)<sup>m</sup>E** con  $m=2$  (I de interpolación), la componente lenta  $z$  se integra con un Euler explícito.

A continuación se estiman los errores de truncamiento local asociados a los algoritmos III.5.4, III.5.5 y III.5.6.

**Teorema IV.1.2.1.** Sean  $y(t)$ ,  $z(t)$  las soluciones teóricas del sistema definido en (1), y suponga que son de clase  $\mathcal{C}^s$ ,  $s$  lo suficientemente grande, entonces los errores de truncamiento local de los **MIM**'s definidos por los algoritmos III.5.4, III.5.5, y III.5.6 son respectivamente

$$y(t_{nk+k}) - y_{nk+k}^{[m]} = \begin{cases} -k \frac{h^2}{2} y''(t_{nk}) + O(h^3), \\ -k \frac{h^2}{2} \bar{y}_2''(t_{nk}) + \frac{(kh)^2}{2} (y''(t_{nk}) - \bar{y}_2''(t_{nk})) - (kh)^2 z'(t_{nk}) f_z(\zeta) + O(h^3), \\ -k \frac{h^2}{2} \bar{y}_3''(t_{nk}) + \frac{(kh)^2}{2} (y''(t_{nk}) - \bar{y}_3''(t_{nk})) - kh^2 z'(t_{nk}) f_z(\zeta_0) + O(h^3), \end{cases} \quad (73)$$

$$T_z = \frac{(kh)^2}{2} z''(t_{nk}) + O(h^3), \quad (74)$$

donde  $\bar{y}_l$ , para  $l = 1, 2, 3$ , están definidas por (30).

Demostración. A los operadores de diferencias asociado al predictor y corrector definidos por (4) se denotan por  $\mathcal{P}$  y  $\mathcal{C}$  respectivamente. Los coeficientes asociados al operador de diferencias del corrector llevan el superíndice  $c$ .

Los coeficientes para el predictor de la componente rápida  $y$  de los algoritmos III.5.4, III.5.5 y III.5.6 están dados por la tabla 1, los del corrector se presentan a continuación

Coeficientes Método Corrector Componente Rápida $y$				
$p$	$\alpha_0^c$	$\alpha_1^c$	$\beta_0^c$	$\beta_1^c$
1	-1	1	1	0

Tabla 3

Los coeficiente asociados al método de la componente lenta  $z$ , están dados por la tabla 2.

Los errores de truncamiento local para las predicciones de los algoritmos III.5.4, III.5.5, y III.5.6, están dados por (28). De las suposiciones locales dadas en (6), (7), y repitiendo el análisis hecho para obtener (10), y además tomando en cuenta que los coeficientes y parámetros asociados con el método multipaso definido para el predictor son los definidos por la tabla 1, se obtiene para cada uno de los algoritmos considerados

$$y(t_{nk+k}) - y_{nk+k}^{[0]} = T_y \quad (75)$$

donde  $T_y$  esta definida por (28).

Sustituyendo los coeficientes del corrector, definidos en la tabla 3, en (15) y de las suposiciones locales definidas en (6), (7), se obtiene para el corrector

$$(1 - hf_y(\xi)) [y(t_{nk+k}) - y_{nk+k}] - h [z(t_{nk+k}) - \bar{z}_{nk+k}] f_z(\xi) = \mathcal{G}_y[y(t_{nk}); h], \quad (76)$$

donde  $\xi$  esta definida por (12),  $\bar{z}_{nk+k} = p(t_{nk+k})$  y

$$p(t) = \begin{cases} \frac{(t_{nk+k} - t)z(t_{nk}) + (t - t_{nk})z(t_{nk+k})}{t_{nk+k} - t_{nk}}, & \text{interpolación} \\ z(t_{nk+k}), & \text{información adelantada} \\ z(t_{nk}), & \text{información atrasada} \end{cases} \quad (77)$$

para toda  $t \in [t_{nk}, t_{nk+k}]$ .

A continuación se encuentra el error introducido por el polinomio de interpolación para cada un de los casos considerados.

Para el algoritmo III.5.4, de la definición (77) del polinomio de interpolación se obtiene

$$\begin{aligned} z(t_{nk+k}) - \bar{z}_{nk+k} &= z(t_{nk+k}) - p(t_{nk+k}) \\ &= 0 \end{aligned} \quad (78)$$

En el algoritmo III.5.5 se toma información adelantada, entonces de (77) se obtiene

$$\bar{z}_{nk+k} = p(t_{nk+k}) = z(t_{nk+k}) \quad (79)$$

para toda  $t \in [t_{nk}, t_{nk+k}]$ . Luego

$$z(t_{nk+k}) - \bar{z}_{nk+k} = 0 \quad (80)$$

Finalmente, en el caso del algoritmo III.5.6 se toma información atrasada, entonces de (77) se obtiene

$$\bar{z}_{nk+k} = p(t_{nk+k}) = z(t_{nk}) \quad (81)$$

para toda  $t \in [t_{nk}, t_{nk+k}]$ .

Desarrollando en serie de Taylor a  $z(t_{nk+k})$  alrededor de  $t_{nk}$ , y ya que  $t_{nk+k} = t_{nk} + kh$ , se obtiene

$$z(t_{nk+k}) = z(t_{nk}) + khz'(t_{nk}) + O(h^2) \quad (82)$$

Luego

$$\begin{aligned} z(t_{nk+k}) - \bar{z}_{nk+k} &= z(t_{nk+k}) - p(t_{nk+k}) \\ &= z(t_{nk+k}) - z_{nk} \\ &= khz'(t_{nk}) + O(h^2) \end{aligned} \quad (83)$$

Resumiendo, los errores introducidos por el polinomio de interpolación usado en los algoritmos III.5.4, III.5.5 y III.5.6, están dados por (78), (80) y (83), esto es

$$z(t_{nk+k}) - \bar{z}_{nk+k} = \begin{cases} 0, & \text{interpolación del tipo al rápido} \\ 0, & \text{información adelantada} \\ khz'(t_{nk}) + O(h^2), & \text{información atrasada} \end{cases} \quad (84)$$

En forma completamente análoga al desarrollo hecho en el teorema IV.1.1.1, se expande en serie de Taylor a la solución teórica  $y(t_{nk+k})$  alrededor del punto  $t_{nk}$ , y se

obtiene  $y_{nk+k}$  para el método de Euler implícito para cada uno de los casos considerados, luego

$$y(t_{nk+k}) = y(t_{nk}) + kh y'(t_{nk}) + \frac{(kh)^2}{2} y''(t_{nk}) + O(h^3), \quad (85)$$

y las ecuaciones de diferencias  $y_{nk+k}$  asociadas a los algoritmos III.5.4, III.5.5 y III.5.6 que utilizan el método de Euler implícito, son respectivamente

$$y_{nk+k} = y_{nk} + kh y'(t_{nk}) + k(k+1) \frac{h^2}{2} y''(t_{nk}) + O(h^3), \quad (86)$$

$$y_{nk+k} = y_{nk} + kh \bar{y}_2'(t_{nk}) + k(k+1) \frac{h^2}{2} \bar{y}_2''(t_{nk}) + O(h^3), \quad (87)$$

$$y_{nk+k} = y_{nk} + kh \bar{y}_3'(t_{nk}) + k(k+1) \frac{h^2}{2} \bar{y}_3''(t_{nk}) + O(h^3), \quad (88)$$

donde  $y_2, y_3$  están definidas por (30).

Realizando un análisis parecido al del teorema IV.1.1.1 para obtener  $T_y$ , restamos (86) de (85), y repetimos la operación para las ecuaciones (87) y (88), entonces se obtiene para la ecuación del corrector para cada uno de los algoritmos considerados

$$y(t_{nk+k}) - y_{nk+k} = \begin{cases} -k \frac{h^2}{2} y''(t_{nk}) + O(h^3), \\ -k \frac{h^2}{2} \bar{y}_2''(t_{nk}) + \frac{(kh)^2}{2} (y''(t_{nk}) - \bar{y}_2''(t_{nk})) - (kh)^2 z'(t_{nk}) f_z(\zeta) + O(h^3), \\ -k \frac{h^2}{2} \bar{y}_3''(t_{nk}) + \frac{(kh)^2}{2} (y''(t_{nk}) - \bar{y}_3''(t_{nk})) + O(h^3). \end{cases} \quad (89)$$

Sustituyendo (84) y (89) en (76) para cada uno de los algoritmos considerados, se encuentra

$$\mathcal{G}_y^c[y(t_{nk});h] = \begin{cases} -k\frac{h^2}{2}y''(t_{nk})+O(h^3), \\ -k\frac{h^2}{2}\bar{y}_2''(t_{nk})+\frac{(kh)^2}{2}(y''(t_{nk})-\bar{y}_2''(t_{nk}))-(kh)^2z'(t_{nk})f_z(\zeta)+O(h^3), \\ -k\frac{h^2}{2}\bar{y}_3''(t_{nk})+\frac{(kh)^2}{2}(y''(t_{nk})-\bar{y}_3''(t_{nk}))-kh^2z'(t_{nk})f(\xi)_z+O(h^3). \end{cases} \quad (90)$$

Por otra parte, sustituyendo los coeficientes del corrector dados en la tabla 3 en (8), se obtiene para el corrector

$$y(t_{nk+k}) = y(t_{nk+k-1}) + hf(y(t_{nk+k}), z(t_{nk+k}), f_{nk+k}) + \mathcal{G}_y^c[y(t_{nk});h] \quad (91)$$

De la definición de los algoritmos III.5.4, III.5.5 y III.5.6 se obtiene para  $i=k-1$ ,  $s=0, 1, 2, \dots, m-1$

$$y_{nk+k}^{[s+1]} = y_{nk+k-1}^{[m]} + hf(y_{nk+k}^{[s]}, \bar{z}_{nk+k}^{[s]}, f_{nk+k}^{[s]}), \quad (92)$$

donde  $\bar{z}_{nk+k} = p(t_{nk+k})$  esta dado por (77).

Restando (92) de (91) y junto con las suposiciones locales dadas en (6) y (7), se encuentra para el corrector

$$y(t_{nk+k}) - y_{nk+k}^{[s+1]} = hf(y(t_{nk+k}), z(t_{nk+k}), f_{nk+k}) - hf(y_{nk+k}^{[s]}, \bar{z}_{nk+k}^{[s]}, f_{nk+k}^{[s]}) + \mathcal{G}_y^c[y(t_{nk});h] \quad (93)$$

Del teorema del valor medio, se obtiene

$$y(t_{nk+k}) - y_{nk+k}^{[s+1]} = h[y(t_{nk+k}) - y_{nk+k}^{[s]}]f_y + h[z(t_{nk+k}) - \bar{z}_{nk+k}^{[s]}]f_z + \mathcal{G}_y^c[y(t_{nk});h], \quad (94)$$

donde  $s=0, 1, 2, \dots, m-1$ .

Sustituyendo (75), (84) y (90) en el miembro derecho de (94) con  $s=0$ , para cada uno de los algoritmos considerados, se obtiene

$$y(t_{nk+k}) - y_{nk+k}^{[1]} = \begin{cases} -k \frac{h^2}{2} y''(t_{nk}) + O(h^3), \\ -k \frac{h^2}{2} \tilde{y}_2''(t_{nk}) + \frac{(kh)^2}{2} (y''(t_{nk}) - \tilde{y}_2''(t_{nk})) - (kh)^2 z'(t_{nk}) f'_z(\zeta) + O(h^3), \\ -k \frac{h^2}{2} \tilde{y}_3''(t_{nk}) + \frac{(kh)^2}{2} (y''(t_{nk}) - \tilde{y}_3''(t_{nk})) - kh^2 z'(t_{nk}) f(\xi)_z + O(h^3). \end{cases} \quad (95)$$

La expresión (95) se sustituye en el lado derecho de (94) con  $s=1$ , para cada uno de los algoritmos considerados. Realizando esta sustitución sucesiva en (94) para  $s=2, 3, \dots, m-1$ , se encuentra que el error de truncamiento local esta dado por

$$y(t_{nk+k}) - y_{nk+k}^{[m]} = \begin{cases} -k \frac{h^2}{2} y''(t_{nk}) + O(h^3), \\ -k \frac{h^2}{2} \tilde{y}_2''(t_{nk}) + \frac{(kh)^2}{2} (y''(t_{nk}) - \tilde{y}_2''(t_{nk})) - (kh)^2 z'(t_{nk}) f'_z(\zeta) + O(h^3), \\ -k \frac{h^2}{2} \tilde{y}_3''(t_{nk}) + \frac{(kh)^2}{2} (y''(t_{nk}) - \tilde{y}_3''(t_{nk})) - kh^2 z'(t_{nk}) f(\xi)_z + O(h^3). \end{cases} \quad (96)$$

para  $m = 1, 2, \dots$

Finalmente, el error de truncamiento local para la componente lenta  $z$  es el mismo para los tres algoritmos. Este se obtiene repitiendo el desarrollo hecho para obtener (72) en el teorema IV.1.1.1,

$$T_z = \frac{(kh)^2}{2} z''(t_{nk}) + O(h^3), \quad (97)$$

con lo cual el teorema queda demostrado.



Es claro entonces, que el error de truncamiento que prevalece es el del corrector, para los algoritmos III.5.4, II.5.5 y III.5.6 con estrategia PI(EC)<sup>m</sup>E.

### § IV.1.3 Análisis de Error de Truncamiento Local para un Sistema Lineal

A la luz de los resultados obtenidos en los teoremas IV.1.1.1 y IV.1.2.1, se realiza un análisis de error de truncamiento local en el siguiente sistema lineal homogéneo

$$\begin{pmatrix} y' \\ z' \end{pmatrix} = \begin{pmatrix} a_{11} & \mu \\ \epsilon & a_{22} \end{pmatrix} \begin{pmatrix} y \\ z \end{pmatrix} \quad (98)$$

la componente rápida es  $y$ , y la lenta  $z$ .

Se consideran únicamente dos casos:

1. No existe acoplamiento entre la componente rápida  $y$  con la lenta  $z$ , esto es,  $\epsilon \neq 0$ ,  $\mu = 0$ . La magnitud de  $\epsilon$  se asume *moderada*.
2. No existe acoplamiento entre la componente lenta  $z$  con la rápida  $y$ , esto es,  $\epsilon = 0$ ,  $\mu \neq 0$ . La magnitud de  $\mu$  se asume *moderada*.

#### Primer Caso

La componente rápida  $y$  se expresa como

$$y'(t) = a_{11}y \quad (99)$$

Entonces de (30) se obtiene

$$\begin{aligned} y'(t) &= \bar{y}'_2(t) \\ &= \bar{y}'_3(t) \end{aligned} \quad (100)$$

Luego, como una consecuencia de (99) y (100) se obtiene

$$\begin{aligned} y'(t_{nk}) &= \bar{y}'_l(t_{nk}), \\ y''(t_{nk}) &= \bar{y}''_l(t_{nk}), \end{aligned} \quad (101)$$

en  $t=t_{nk}$ , donde  $l = 2, 3$ .

Además

$$f_z = 0 \quad (102)$$

Aplicando los resultados obtenidos al teorema IV.1.1.1 se encuentra el error de truncamiento local para los algoritmos III.5.1, III.5.2 y III.5.3, éstos son

$$T_y = k \frac{h^2}{2} a_{11} y'(t_{nk}) + O(h^3), \quad (103)$$

$$T_z = \frac{(kh)^2}{2} \epsilon y'(t_{nk}) + \frac{(kh)^2}{2} a_{22} z'(t_{nk}) + O(h^3) \quad (104)$$

Análogamente, para el teorema IV.1.2.1, el error de truncamiento local para los algoritmos III.5.4, III.5.5 y III.5.6, es

$$y(t_{nk+k}) - y_{nk+k}^{[m]} = -k \frac{h^2}{2} a_{11} y'(t_{nk}) + O(h^3), \quad (105)$$

$$T_z = \frac{(kh)^2}{2} \epsilon y'(t_{nk}) + \frac{(kh)^2}{2} a_{22} z'(t_{nk}) + O(h^3) \quad (106)$$

### Segundo Caso

En éste, las expresiones para la componente rápida  $y$ , y  $\bar{y}$  se expresan como

$$y'(t) = a_{11} y + \mu z(t) \quad (107)$$

$$\bar{y}_2'(t) = a_{11} y(t) + \mu z_{nk+k} \quad (108)$$

$$\bar{y}_3'(t) = a_{11} y(t) + \mu z_{nk} \quad (109)$$

Luego

$$\begin{aligned} y''(t) &= a_{11} y'(t) + \mu z'(t), \\ \bar{y}_2''(t) &= \bar{y}_3''(t) \\ &= a_{11} y'(t) \end{aligned} \quad (110)$$

Además

$$f_z = \mu \quad (111)$$

De las ecuaciones anteriores, se obtiene para  $t=t_{nk}$ , que

$$\begin{aligned}
 y''(t_{nk}) - \bar{y}_2''(t_{nk}) &= y''(t_{nk}) - \bar{y}_3''(t_{nk}) \\
 &= \mu z'(t_{nk})
 \end{aligned}
 \tag{112}$$

Luego, sustituyendo (107), (110), (111) y (112) en el teorema IV.1.1.1 se encuentra que el error de truncamiento local para la componente rápida  $y$  de los algoritmos III.5.1, III.5.2 y III.5.3, son respectivamente

$$T_y = \begin{cases} k \frac{h^2}{2} a_{11} y'(t_{nk}) + k \frac{h^2}{2} z'(t_{nk}) \mu + O(h^3), \\ k \frac{h^2}{2} a_{11} y'(t_{nk}) - \left[ \frac{(kh)^2}{2} + h^2 \right] z'(t_{nk}) \mu + O(h^3), \\ k \frac{h^2}{2} a_{11} y'(t_{nk}) + \left[ \frac{(kh)^2}{2} + (k-1)h^2 \right] z'(t_{nk}) \mu + O(h^3), \end{cases}
 \tag{113}$$

En forma análoga se encuentra el error de truncamiento local para la componente lenta  $z$

$$T_z = \frac{(kh)^2}{2} a_{22} z'(t_{nk}) + O(h^3)
 \tag{114}$$

Análogamente, se encuentran los errores de truncamiento local para los algoritmos III.5.4, III.5.5 y III.5.6, éstos son respectivamente

$$T_y = \begin{cases} - \left[ k \frac{h^2}{2} a_{11} y'(t_{nk}) + k \frac{h^2}{2} z'(t_{nk}) \mu \right] + O(h^3), \\ - k \frac{h^2}{2} a_{11} y'(t_{nk}) - \frac{(kh)^2}{2} z'(t_{nk}) \mu + O(h^3), \\ - k \frac{h^2}{2} a_{11} y'(t_{nk}) + \left[ \frac{(kh)^2}{2} - kh^2 \right] z'(t_{nk}) \mu + O(h^3), \end{cases}
 \tag{115}$$

$$T_z = \frac{(kh)^2}{2} a_{22} z'(t_{nk}) + O(h^3) \quad (116)$$

Si no existiera acoplamiento entre las diferentes componentes de un sistema de ecuaciones diferenciales ordinarias, por ejemplo el sistema (99) con  $\epsilon = \mu = 0$ , su integración se podría hacer independiente para cada componente.

Considere que los valores característicos asociados al sistema (99) nos permiten la aplicación de los algoritmos III.5.1 al III.5.6, y suponga que no existe acoplamiento entre sus diferentes componentes, entonces la aplicación a éste tipo de sistemas de cualquiera de los algoritmos considerados nos daría resultados dentro de márgenes de precisión deseada, ya que los errores de truncamiento local para ambas componentes  $y$ ,  $z$  se podrían mantener dentro de niveles aceptables para pasos de integración  $h$  y factores multitasas  $k$  adecuados.

Del análisis error realizado al sistema lineal (98), se puede observar que el caso que tiene el comportamiento más similar al desacoplado, es el primero. En éste se cumplen los siguientes resultados

- Al no existir acoplamiento entre la componente rápida  $y$  con la lenta  $z$ , no es necesario interpolar de la componente lenta  $z$  a la rápida  $y$ .
- Los errores de  $z$  no afectan a  $y$ .
- El error de truncamiento local de la componente rápida  $y$  es el mismo que para el caso desacoplado (ver (103)).

El error de truncamiento local de la componente lenta  $z$ , tiene el término adicional  $\frac{(kh)^2}{2}\epsilon y'(t_{nk})$ . En consecuencia, si se mantiene el error de  $y$  en niveles

aceptables y la magnitud de  $\epsilon$  es modesta, para valores moderados  $k$  y  $h$  se puede mantener a  $z$  en márgenes de precisión requeridos.

Este tipo de sistemas ( $\epsilon \neq 0, \mu = 0$ ) permite el uso de los algoritmos III.5.2, III.5.3, III.5.5 y III.5.6 en los que no se interpola de la componente lenta  $z$  a la rápida  $y$ .

Considere el segundo caso, en éste se tienen los siguientes resultados

- Los errores cometidos en la aproximación de  $z$  afectan a  $y$ .
- El error de truncamiento local de la componente rápida  $y$  está afectado por  $\mu$  que es el factor de acoplamiento con la componente lenta  $z$ .
- El error de truncamiento local de la componente lenta  $z$  es el mismo que para el caso desacoplado.

De (113) y (115) se obtiene que para los algoritmos en los que no se interpola III.5.2, III.5.3, III.5.5 y III.5.6, valores modestos en la magnitud de  $\mu$  pueden originar errores de truncamiento local no aceptables en la componente rápida  $y$ .

En este caso, lo más conveniente es utilizar los algoritmos que interpolen de la componente lenta  $z$  a la rápida  $y$ , esto es, los algoritmos III.5.1 o III.5.4.

Las características que presentan los modelos matemáticos que representan a los procesos de las plantas generadoras de energía eléctrica, son muy similares al primer

caso, es decir, existe nulo acoplamiento o muy débil de las componentes rápidas con las lentas.

Por lo general, en base a consideraciones heurísticas se seleccionan algoritmos del tipo definido por III.5.2 ó III.5.5, que son los que no interpolan y hacen uso de información adelantada en la componente lenta  $z$ . Estos algoritmos, por su lógica son menos complicados en su programación y requieren de menos memoria física.

Valores típicos para el paso de integración  $h$  y factores  $k$  multitasa en simulaciones en tiempo real de plantas generadoras de energía eléctrica para entrenamiento son

$$h_l = \frac{1}{8}, \frac{1}{4}, \frac{1}{2}, 1, \quad k_l = 1, 2, 4, 8 \quad (117)$$

donde  $l=1, 2, 3, 4$ . El paso de integración  $h$ , se puede expresar en función de  $h_1 = 1/8$ , esto es

$$h_l = k_l h_1 \quad (118)$$

El análisis anterior justifica desde el marco de referencia del error de truncamiento local, la selección de los algoritmos III.5.2 ó III.5.5 para sistemas que tengan un comportamiento similar al primer caso analizado.

## § IV.2 DISCUSION DE LOS ERRORES

En lo que se refiere a los errores de truncamiento local se puede concluir que los algoritmos que usan interpolación tienen el menor error en la componente rápida, el

error en la componente lenta es el mismo en todos los casos. Los algoritmos que usan información adelantada y atrasada (III.5.2 y III.5.5) tienen un comportamiento similar en el error de truncamiento local en la componente rápida. En el caso de los algoritmos III.5.4, III.5.5 y III.5.6 prevalece el error del corrector. En todos los casos el error de truncamiento local  $T_{n+k} \rightarrow 0$  cuando  $h \rightarrow 0$ .

Por otra parte, se concluye de la definición de orden para MIM y de los teoremas IV.1.1.1, IV.1.1.2 que el orden de los algoritmos analizados es de uno respecto al error de truncamiento local, y el polinomio de interpolación se podría seleccionar de un orden más bajo que el de los métodos de integración empleados sin afectar el orden del método multitasas.

Finalmente, los algoritmos analizados III.5.1 hasta III.5.6 tienen su mejor comportamiento en los sistemas de ecuaciones diferenciales ordinarias cuya estructura sea lo más cercana posible a el primer caso de la sección IV.1.3, esto es, sistemas con matriz jacobiana casi triangular inferior, en donde las componentes rápidas no dependan de las lentas (o en forma muy débil), por lo que se recomienda la aplicación de éstos algoritmos en general a este tipo de sistemas.

En caso de existir un acoplamiento entre las componentes lentas y rápidas, el cual no sea débil, no se recomienda aplicar ninguno de los algoritmos analizados, ya que el error de truncamiento local podría crecer a márgenes no aceptables.



## Capítulo V

# Estabilidad Absoluta para Métodos de Integración Multitasa

En el presente capítulo la noción de estabilidad absoluta para los **MIM**'s definidos por los algoritmos III.5.1 a III.5.6 es presentada, y no se establece ningún criterio simple para determinar cuando un sistema lineal está en la región de estabilidad absoluta de un **MIM** dado. Sin embargo, cuando la componente lenta del sistema IV.1.(1) es independiente de la rápida (o viceversa), las regiones de estabilidad absoluta de los **MIM**'s son caracterizadas en términos de las regiones de estabilidad absoluta de los métodos definidos por III.5.1 a III.5.6 que son usados para aproximar numéricamente la solución de los subsistemas.

## §V.1 ESTABILIDAD ABSOLUTA EN METODOS MULTIPASOS LINEALES.

A continuación se da una breve exposición del concepto de estabilidad absoluta en métodos multipasos lineales convencionales. El objetivo de repasar estos conceptos es definir la terminología y notación usada en los conceptos de estabilidad absoluta, para posteriormente usarla en los MIM's y señalar las diferencias que se tienen con los métodos multipasos lineales convencionales.

Considere el siguiente sistema de ecuaciones diferenciales ordinarias con valores iniciales

$$y'(t) = f(y,t), \quad t \in [a,b], \quad y(a) = \eta \quad (1)$$

donde

$$\begin{aligned} y &= (y^1, \dots, y^m)^T, \\ f &= (f^1, \dots, f^m)^T, \\ \eta &= (\eta^1, \dots, \eta^m)^T, \end{aligned} \quad (2)$$

y suponga que el siguiente método multipaso lineal de  $k$ -pasos es aplicado a (1) para aproximar el valor de su solución numérica en  $[a, b]$

$$\sum_{j=0}^k \alpha_j y_{n-j} = h \sum_{j=0}^k \beta_j f_{n-j}, \quad (3)$$

donde  $\alpha_j, \beta_j$  son constantes,  $\alpha_k = 1$ ,  $f_n = f(y_n, t_n)$  y  $\alpha_0, \beta_0$  no son cero al mismo tiempo.

La solución teórica  $y(t)$  de (1) satisface

$$\sum_{j=0}^k \alpha_j y(t_{n,j}) = h \sum_{j=0}^k \beta_j f(y(t_{n,j}), t_{n,j}) + \mathcal{L}[y(t_n); h]. \quad (4)$$

Denotemos por  $\{\bar{y}_n\}$  a la solución de (3) cuando un error de redondeo  $R_{n,k}$  es cometido en la n-ésima aplicación del método, entonces

$$\sum_{j=0}^k \alpha_j \bar{y}_{n,j} = h \sum_{j=0}^k \beta_j f(\bar{y}_{n,j}, t_{n,j}) + R_{n,k}. \quad (5)$$

Restando (5) de (4) y definiendo al error global  $\bar{e}_n$  por

$$\bar{e}_n = y(t_n) - \bar{y}_n, \quad (6)$$

se encuentra

$$\sum_{j=0}^k \alpha_j \bar{e}_{n,j} = h \sum_{j=0}^k \beta_j [f(y(t_{n,j}), t_{n,j}) - f(\bar{y}_{n,j}, t_{n,j})] + \Phi_{n,k}, \quad (7)$$

$\Phi_{n,k}$  representa el error total cometido en el n-ésimo paso. Aplicando el teorema del valor medio a cada componente del lado derecho de (7) se obtiene

$$\sum_{j=0}^k \alpha_j \bar{e}_{n,j} = h \sum_{j=0}^k \beta_j \frac{\partial f}{\partial y} \bar{e}_{n,j} + \Phi_{n,k}, \quad (8)$$

donde  $\frac{\partial f}{\partial y}$  es la matriz jacobiana del sistema (1), esto es

$$\frac{\partial f}{\partial y} = \left( \frac{\partial f_i}{\partial y_j} \right)_{i,j=1,\dots,m}, \quad (9)$$

la cual se evalúa en puntos intermedios adecuados. Supongamos ahora las siguientes simplificaciones en nuestro análisis

$$\begin{aligned} \frac{\partial f}{\partial y} &= J, \text{ matriz constante,} \\ \Phi_n &= \Phi, \text{ vector } n \times 1 \text{ constante.} \end{aligned} \quad (10)$$

Con las simplificaciones anteriores la ecuación (8), se reduce a la siguiente ecuación de error linealizada

$$\sum_{j=0}^k (\alpha_j I - h\beta_j J) \bar{e}_{n+j} = \Phi, \quad (11)$$

donde  $I$  es la matriz unitaria  $m \times m$ . Si además suponemos, que los valores característicos  $\lambda_i$  de  $J$  son distintos para  $i=1, 2, \dots, m$ , entonces [20] existe una matriz

$S$   $m \times m$  tal que

$$S^{-1}JS = \Lambda = \text{diag}(\lambda_1, \dots, \lambda_m), \quad (12)$$

Pre-multiplicando ambos lados de (11) por  $S^{-1}$ , y definiendo  $\bar{d}_n$  por

$$\bar{e}_n = S\bar{d}_n, \quad (13)$$

se obtiene

$$\sum_{j=0}^k S^{-1}(\alpha_j I - h\beta_j J) S \bar{d}_{n+j} = S^{-1}\Phi. \quad (14)$$

De la expresión (14) se obtiene

$$\sum_{j=0}^k (\alpha_j I - h\beta_j \Lambda) d_{n,j} = \Psi, \quad (15)$$

donde  $\Psi = S^{-1}\Phi$ . Ya que  $I, \Lambda$  son matrices diagonales, las componentes de (15) se encuentran desacopladas, consecuentemente (15) se puede escribir como

$$\sum_{j=0}^k (\alpha_j - h\beta_j \lambda_i) d_{n,j}^i = \Psi^i, \quad i=1,2,\dots,m, \quad (16)$$

donde  $d_n^i, \Psi^i$  son las componentes de  $d_n$  y  $\Psi$  respectivamente. Cada ecuación de (16) es independiente de las otras.

El análisis entonces, se puede realizar suponiendo que la  $\dim(y) = 1$  sin pérdida de generalidad, evitándose recargar la notación. En función de los comentarios anteriores se interpretará a (16) como la ecuación de error linealizada en una dimensión, esto es

$$\sum_{j=0}^k (\alpha_j - \lambda h \beta_j) \bar{e}_{n,j} = \Phi, \quad (17)$$

La solución general de la ecuación de diferencias (págs. 7-9)[19] definidas por (17) es

$$\bar{e}_n = \sum_{s=1}^k d_s r_s^n - \Phi \left/ \lambda h \sum_{j=0}^k \beta_j \right., \quad (18)$$

donde las  $d_s$  son constantes arbitrarias y  $r_s$  son las raíces, supuestamente distintas, del polinomio característico asociado a (17), es decir

$$\sum_{j=0}^k (\alpha_j - \lambda h \beta_j) r^j = 0, \quad (19)$$

para el caso de raíces múltiples consultar [15], [19].

Sea

$$\rho(r) = \sum_{j=0}^k \alpha_j r^j, \quad \sigma(r) = \sum_{j=0}^k \beta_j r^j. \quad (20)$$

Desarrollando (19) y combinándola con (20), ésta se escribe como

$$\pi(r, \lambda h) = \rho(r) - \lambda h \sigma(r) = 0. \quad (21)$$

El polinomio definido por (21) lo llamaremos el "polinomio de estabilidad" del método multipaso definido por (3).

**Definición V.1.1.** Un método multipaso lineal o predictor corrector es **absolutamente estable** [relativamente estable] en una región  $\mathfrak{R}$  del plano complejo  $\mathbb{C}$  [19], si para toda  $\lambda h \in \mathfrak{R}$ , todas las raíces del polinomio de estabilidad  $\pi(r, \lambda h)$  del método satisfacen

$$\begin{aligned} |r_s| < 1, \quad s=1,2,\dots,k \\ [ |r_s| < |r_1|, \quad s=1,2,\dots,k ]. \end{aligned} \quad (22)$$

Sea  $D$  el disco abierto unitario en  $\mathbb{C}$ . La región de estabilidad absoluta  $\mathfrak{R}$ , es un conjunto abierto del plano complejo tal que

$$\mathfrak{R} = \{\lambda h \in \mathbb{C} \mid \pi(r, \lambda h) = 0 \rightarrow r \in D\}. \quad (23)$$

La frontera de la región  $\mathfrak{R}$  se denota por  $\partial\mathfrak{R}$ ; ya que las raíces son funciones continuas de  $\lambda h$ , entonces  $\lambda h \in \partial\mathfrak{R}$  cuando  $|r|=1$ , esto es

$$\partial\mathfrak{R} = \{z \in \mathbb{C} \mid \pi(r, z) = 0 \rightarrow |r|=1\}. \quad (24)$$

Una consecuencia inmediata de la definición IV.1.1.1, es el siguiente teorema:

**Teorema 1.** En ausencia de errores de redondeo el error  $\bar{\epsilon}_n \rightarrow 0$  cuando  $n \rightarrow \infty$ , si y solamente si  $|r_s| < 1$  para toda  $s=1, 2, \dots, k$ .

**Demostración.** Es una consecuencia inmediata de (18).

## § V.2 ESTABILIDAD ABSOLUTA EN METODOS MULTITASA

Los **MIM**'s suponen que hay al menos dos ecuaciones diferenciales ordinarias, luego no es posible usar la ecuación prueba escalar  $y' = \lambda y$  para realizar el análisis de estabilidad absoluta, como es en el caso de los métodos multipasos convencionales. Es necesario entonces, considerar al menos una ecuación prueba bidimensional para realizar el análisis de estabilidad absoluta en los **MIM**'s, esto es

$$w'(t) = \begin{pmatrix} y' \\ z' \end{pmatrix} = \begin{pmatrix} \lambda_y & \mu \\ \epsilon & \lambda_z \end{pmatrix} \begin{pmatrix} y \\ z \end{pmatrix}, \quad (25)$$

donde  $\lambda, \lambda_z, \epsilon, \mu$  pertenecen al plano complejo  $\mathbf{C}$ . Supóngase que  $\eta_1, \eta_2$  son los valores característicos del sistema (25) y que la componente rápida es  $y$ , y la lenta  $z$ , entonces se satisface

$$\text{Real}(\eta_1) < \text{Real}(\eta_2) < 0. \quad (26)$$

La aplicación de los **MIM**'s definidos por los algoritmos III.5.1 hasta III.5.6 a la ecuación prueba (25), da la siguiente ecuación de diferencias después de algunas manipulaciones algebraicas

$$\begin{pmatrix} y_{nk+k} \\ z_{nk+k} \end{pmatrix} = M \begin{pmatrix} y_{nk} \\ z_{nk} \end{pmatrix}, \quad (27)$$

donde  $M$  es una matriz  $2 \times 2$  que depende de  $\lambda, \lambda_z, \epsilon, \mu, h, k$ .



La matriz  $M$  tiene la siguiente estructura

$$M = \begin{pmatrix} P^k + m_{11}\epsilon^\mu & m_{12}\epsilon^\mu \\ kh\epsilon & 1 + kh\lambda_z \end{pmatrix}, \quad (28)$$

donde  $P$  y las  $m_j$  de la matriz  $M$  están dados en las tablas 1 y 2 para cada uno de los algoritmos:

Alg.	P	$m_{11}$	$m_{12}$
III.5.1	$1 + h\lambda_y$	$h^2 \left[ \frac{P^k - kh\lambda_y - 1}{h^2 \lambda_y^2} \right]$	$h \left[ \frac{P^{k-1}}{h\lambda_y} + h\lambda_z \frac{P^k - kh\lambda_y - 1}{h^2 \lambda_y^2} \right]$
III.5.2	$1 + h\lambda_y$	$kh^2 \left[ \frac{P^{k-1}}{h\lambda_y} \right]$	$h \left[ (1 + kh\lambda_y) \frac{P^{k-1}}{h\lambda_y} \right]$
III.5.3	$1 + h\lambda_y$	0	$\begin{bmatrix} P^{k-1} \\ h\lambda_y \end{bmatrix}$

Tabla 1

	P	m <sub>11</sub>	m <sub>12</sub>
III.5.4	$1+h\lambda_y+h^2\lambda_y^2+h^3\lambda_y^3$	$h^2 \left[ \frac{P^k - kh\lambda_y - 1}{h^2\lambda_y^2} \right]$	$h \left[ \frac{P^k - 1}{h\lambda_y} + h\lambda_y \frac{P^k - kh\lambda_y - 1}{h^2\lambda_y^2} \right]$
III.5.5	$1+h\lambda_y+h^2\lambda_y^2+h^3\lambda_y^3$	$kh^2 \left[ \frac{P^k - 1}{h\lambda_y} \right]$	$h \left[ (1+kh\lambda_y) \frac{P^k - 1}{h\lambda_y} \right]$
III.5.6	$1+h\lambda_y+h^2\lambda_y^2+h^3\lambda_y^3$	0	$h \left[ \frac{P^k - 1}{h\lambda_y} \right]$

Tabla 2

Supóngase que el método Euler explícito es aplicado en forma convencional a la ecuación prueba (25), entonces se obtiene la siguiente ecuación de diferencias

$$\begin{pmatrix} y_{n+1} \\ z_{n+1} \end{pmatrix} = (\bar{I} + h\bar{J}) \begin{pmatrix} y_n \\ z_n \end{pmatrix} \quad (29)$$

donde  $h$  es el paso de integración,  $I$  es la matriz unitaria y  $J$  es la matriz jacobiana de (25).

La comparación de las matrices  $M$  e  $I+hJ$  no dan los siguientes resultados:

1. La matriz  $I+hJ$  está en función de la matriz original  $J$ , y cualquier transformación que diagonaliza a  $J$  diagonaliza a  $I+hJ$ , por lo tanto sus valores característicos son independientes bajo transformaciones de semejanza de  $J$ .
2. La matriz  $M$  no es diagonalizable por transformaciones que diagonalicen a  $J$ , luego los valores característicos de  $M$  son dependientes de  $J$  bajo transformaciones de semejanza.

La única propiedad que  $M$  hereda de  $J$  es la siguiente:

3. Si  $J$  es triangular superior [inferior]  $M$  también lo es.

En forma similar al análisis de estabilidad absoluta de los métodos multipasos lineales convencionales, se realiza el de los MIM, analizando las raíces del polinomio característico de  $M$ , para verificar si estas son en magnitud estrictamente menores que la unidad.

El caso más simple de analizar es cuando la matriz  $J$  es triangular inferior [superior] ya que de acuerdo al inciso (3)  $M$  también lo es, por lo que en éste caso las raíces del polinomio característico de  $M$  son para los algoritmos III.5.1, III.5.2 y III.5.3

$$\begin{aligned} r_1 &= (1+h\lambda_y)^k, \\ r_2 &= (1+kh\lambda_y) \end{aligned} \tag{30}$$

y para los algoritmos III.5.4, III.5.5 y III.5.6

$$\begin{aligned} r_1 &= (1+h\lambda_y+h^2\lambda_y^2+h^3\lambda_y^3)^k, \\ r_2 &= (1+kh\lambda_y) \end{aligned} \tag{31}$$

Entonces las magnitudes de  $r_1$  y  $r_2$  son menores que la unidad sí y solo sí se satisface que

$$\begin{aligned} |1+h\lambda_y| &< 1, \\ |1+h\lambda_y+h^2\lambda_y^2+h^3\lambda_y^3| &< 1, \\ |1+kh\lambda_z| &< 1 \end{aligned} \quad (32)$$

Las relaciones presentada en (32) son los módulos de las raíces de los polinomios de estabilidad asociados al método de Euler y Euler implícito, éste último con la estrategia P(EC)<sup>2</sup>E, es decir de acuerdo a (20) y (21) se tiene

$$\begin{aligned} \pi_1(r, h\lambda_y) &= \rho(r) - h\lambda_y \sigma(r) = r - (1+h\lambda_y), \\ \pi_2(r, h\lambda_y) &= \rho(r) - h\lambda_y \sigma(r) = r - (1+h\lambda_y + h^2\lambda_y^2 + h^3\lambda_y^3), \\ \pi_3(r, kh\lambda_z) &= \rho(r) - kh\lambda_z \sigma(r) = r - (1+kh\lambda_z) \end{aligned} \quad (33)$$

que son los polinomios de estabilidad para el método de Euler y Euler implícito. Se puede concluir entonces que sí la matriz asociada al sistema (25) es triangular, los MIM's definidos por los algoritmos III.5.1 a III.5.6 son absolutamente estables si y solo si los métodos asociados a la componente rápida  $y$ , y la componente lenta  $z$  son absolutamente estables para las ecuaciones no acopladas, usando un paso  $h$  para  $y$ , y  $kh$  para  $z$ .

### § V.3 DISCUSION DE LOS POLINOMIOS DE ESTABILIDAD

Los polinomios de estabilidad encontrados para los algoritmos III.5.1 a III.5.6, no son funciones de los valores característicos de la matriz jacobiana de la ecuación prueba (25), es ésta propiedad la que origina que las regiones de estabilidad

absoluta no sean conservadas por transformaciones de semejanza de la matriz jacobiana y consecuentemente no se puede caracterizar las regiones de estabilidad absoluta de una forma sencilla como en el caso de los métodos numéricos de integración convencionales. Sin embargo, cuando la estructura de la matriz jacobiana es triangular, las regiones de estabilidad absoluta de los **MIM** se pueden caracterizar en términos de las regiones de estabilidad absoluta de los métodos de integración usados para aproximar la solución de los subsistemas en que se separó al conjunto de ecuaciones diferenciales ordinarias.

De las tablas 1 y 2 se puede inferir que sí existe un acoplamiento fuerte entre las componentes rápidas y lentas los **MIM** podrían tener problemas de estabilidad numérica. Además, se obtiene que los polinomios de estabilidad con más términos son los relacionados con los algoritmos que interpolan de los subsistemas lentos a los rápidos y los que tienen menos términos son lo que utilizan información atrasada de los subsistemas lentos.

Podemos entonces esperar que los algoritmos cuya región de estabilidad absoluta sea la más grande pertenezca a los algoritmos III.5.1 y III.5.4 .

# Capítulo VI

## Experimentos Numéricos

A continuación se presentan los experimentos numéricos realizados para medir el desempeño de los algoritmos presentados en el capítulo III. El primer experimento nos permite evaluar el comportamiento de los algoritmos III.5.1, III.5.2, III.5.3 y III.5.5 bajo diferentes acoplamientos del sistema de ecuaciones diferenciales ordinarias. El segundo experimento consiste en integrar por medio de un MIM el modelo simplificado de una planta termoeléctrica bajo condiciones muy parecidas a la de los simuladores desarrollados en el Instituto de Investigaciones Eléctricas. Por medio de éste segundo experimento se mide el desempeño de los algoritmos de integración multitasa para ambientes de simulación en tiempo real en cuanto a rapidez de ejecución y aproximación de los resultados. Todas las pruebas se realizaron en una computadora personal 486 de 66 Mhz con 8 Mb de RAM, los tiempos de la Unidad de Procesamiento Central (UPC) se dan en segundos.

### § VI.1 CONSIDERACIONES DE RAPIDEZ

La lógica de programación y requerimientos de memoria de los MIM son más complejas que la de los métodos de integración convencionales, lo que origina un costo adicional que se debe conservar dentro de límites razonables. El costo

computacional de un método de integración numérica a paso fijo es debido a los siguientes factores:

- Evaluación de las ecuaciones algebraicas.
- Evaluaciones de las derivadas.
- Aplicación de las fórmulas de integración.
- Interpolación si un **MIM** es usado.
- Solución de ecuaciones implícitas si se usan.

Los algoritmos de integración multitasas que requieren menos memoria y que tienen una lógica de programación menos complicada son aquellos que no interpolan y los subsistemas más rápidos evalúan sus derivadas con la información adelantada de los subsistemas lentos. Además, este tipo de algoritmos evitan la complejidad en la práctica de interpolar las ecuaciones algebraicas (que no forman parte del sistema algebraico-diferencial) presentes en la representación matemática de los modelos de procesos, por otra parte, tienen la flexibilidad de permitir cambiar libremente los métodos y pasos de integración asociados a los modelos con suma facilidad. Es por esta consideración práctica, que en el segundo experimento únicamente se prueba este tipo de algoritmo multitasas para evaluar su desempeño en su aplicación a modelos de plantas generadoras de energía eléctrica.

En la programación de los **MIM** se siguen muy de cerca los lineamientos dados por C. W. Gear en [10], especial énfasis se pone en integrar primero los modelos que tienen los pasos de integración más grandes (los lentos), para el mejor desempeño de los algoritmos. Los pasos de integración más grandes asociados a los subsiste-

mas lentos son escogidos como múltiplos de los pasos menores.

## § VI.2 EXPERIMENTOS NUMERICOS

### § VI.2.1 Primer Experimento

La ecuación prueba por medio de la cual se realizan los experimentos numéricos está definida por (consultar [31])

$$\dot{y} = A(y - \phi(t)) + \phi'(t), \quad y(0) = \phi(0), \quad t \in [0, 4], \quad (1)$$

donde

$$A = \begin{pmatrix} -50 & 49 & a & a & a & a \\ 49 & -50 & a & a & a & a \\ b & b & -5 & 4 & a & a \\ b & b & 4 & -5 & a & a \\ b & b & b & b & -1 & 0 \\ b & b & b & b & 0 & -1 \end{pmatrix} \quad (2)$$

$$y' = (y_1'(t), y_2'(t), \dots, y_6'(t))^T,$$

$$y = (y_1(t), y_2(t), \dots, y_6(t))^T. \quad (3)$$

$$\phi(t) = (\text{sen}(20t), \text{cos}(20t), \text{sen}(t), \text{cos}(t), \text{sen}(0.05t), \text{cos}(0.05t))^T.$$

La solución del sistema es  $y(t) = \phi(t)$ , y en consecuencia de acuerdo a la definición III.4.2.1 se tienen dos componentes que oscilan rápidamente, dos que oscilan a una razón moderada y dos que oscilan lentamente. Los números  $a$ ,  $b$  son reales y estos se varían para ver el comportamiento de los algoritmos III.5.1 a III.5.3 y III.5.5



bajo diferentes acoplamientos del sistema definido en (1).

Los valores característicos asociados a las submatrices 2 por 2 que se localizan en la diagonal

$$\begin{pmatrix} -50 & 49 \\ 49 & -50 \end{pmatrix}, \begin{pmatrix} -5 & 4 \\ 4 & -5 \end{pmatrix} \text{ y } \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}, \quad (4)$$

son -99 y -1, -9 y -1, -1 y -1, respectivamente. De acuerdo a la definición de  $\phi(t)$  y a los valores característicos de la matriz jacobiana del sistema (1), es posible separar a éste en concordancia a sus modos de oscilación, es decir,  $(y_1, y_2)$  constituyen las componentes **rápidas**,  $(y_3, y_4)$  las componentes **moderadas** y  $(y_5, y_6)$  son las componentes **lentas**.

En base a la terminología definida en III.2, para cada prueba se definió un ciclo de longitud de 0.1, pasos de integración iguales a  $h_1=0.1$ ,  $h_2=0.01$  y  $h_3=0.001$  para las componentes lentas, moderadas y rápidas respectivamente, lo que da los siguientes factores multitasa  $k_2=10$ ,  $k_3=100$  para los subsistemas moderados y rápidos, esto es, al final de cada ciclo de longitud 0.1 el subsistema lento se integra una vez por 10 veces del sistema moderado y 100 del rápido. En la tabla 1 se presentan los máximos errores absolutos (en porcentaje) por componente  $y_i$ ,  $i=1, \dots, 6$ , para cada prueba realizada, en la primera columna se indican los valores definidos para  $a$  y  $b$  y el número de la prueba, la segunda columna presenta el algoritmo usado de acuerdo a la siguiente convención:

- a) **1** significa que se usa el algoritmo III.5.1, esto es, se aplica el método de Euler explícito a los tres subsistemas y se interpola de los subsistemas lentos al rápido.
- b) **2** significa que se usa el algoritmo III.5.2, esto es, se aplica el método de Euler explícito a los tres subsistemas sin interpolación de los subsistemas lentos a los rápidos, estos actualizan sus derivadas con la información adelantada del sistema moderado y lento en el caso del subsistema rápido y del sistema lento en el caso del subsistema moderado.
- c) **3** significa que se usa el algoritmo III.5.3, esto es, se aplica el método de Euler explícito a los tres subsistemas sin interpolación de los subsistemas lentos a los rápidos, estos actualizan sus derivadas con la información atrasada del sistema moderado y lento en el caso del subsistema rápido y del sistema lento en el caso del subsistema moderado.
- d) **5** significa que se usa el algoritmo III.5.5 y es análogo a 2, excepto que el método integrador usado para cada uno de los subsistemas es el Euler implícito programado como  $P(EC)^2E$ .
- e) **2,5** significa que el subsistema lento se integra por medio del algoritmo III.5.2, los subsistemas moderado y rápido se integran por medio del algoritmo III.5.5. No se interpola y las derivadas se actualizan con información adelantada como se explico en el inciso (b).
- f) **1(2,3)** significa que la prueba se realizó para cada uno de los algoritmos III.5.1 a III.5.3 y no se obtuvieron diferencias significativas en los errores.

Los pasos de integración se seleccionaron de acuerdo a lo expuesto en el capítulo V, es decir, se asumió que la matriz jacobiana de (1) es triangular y se verificó que se cumpla:

a) subsistema rápido

$$\begin{aligned} |1+h_3\lambda_i| < 1, \\ |1+h_3\lambda_i+h_3^2\lambda_i^2+h_3^3\lambda_i^3| < 1, \quad i=1,2, \quad \lambda_1=-99, \quad \lambda_2=-1, \end{aligned} \quad (5)$$

b) subsistema moderado

$$\begin{aligned} |1+10h_3\lambda_i| < 1, \\ |1+10h_3\lambda_i+(10h_3)^2\lambda_i^2+(10h_3)^3\lambda_i^3| < 1, \quad i=1,2, \quad \lambda_1=-9, \quad \lambda_2=-1, \end{aligned} \quad (6)$$

c) subsistema lento

$$|1+100h_3\lambda_i| < 1, \quad i=1,2, \quad \lambda_1=-1, \quad \lambda_2=-1, \quad (7)$$

lo que garantiza la convergencia de los algoritmos cuando el acoplamiento del sistema (1) es débil.

Ya que el rango de valores de la solución del sistema (1) se encuentra en el intervalo  $[-1, 1]$ , se obtuvieron errores absolutos para cada componente del sistema (1) en el intervalo de integración  $[0, 4]$  con muestras tomadas cada 0.1.

No. Exp.	Alg. Emp.	Max. error abs $y_1$	Max. error abs $y_2$	Max. error abs $y_3$	Max. error abs $y_4$	Máx. error abs $y_5$	Max. error abs $y_6$
1. $a=0, b=1$	(2,3)	1.22%	0.98%	0.65%	0.64%	1.28%	1.29%
2. $a=1, b=1$	(2,3)	Inestable	Inestable	Inestable	Inestable	Inestable	Inestable
3. $a=0.1, b=1$	1	1.92%	1.57%	1.97%	1.95%	4.51%	4.51%
4. $a=0.1, b=1$	2	1.98%	1.63%	2.11%	2.10%	4.79%	4.79%
5. $a=0.1, b=1$	3	2.36%	2.01%	2.68%	2.66%	5.80%	5.80%
6. $a=0.01, b=1$	1	1.22%	0.98%	0.68%	0.65%	1.35%	1.36%
7. $a=0.01, b=1$	2	1.22%	0.98%	0.69%	0.66%	1.36%	1.37%
8. $a=0.01, b=1$	3	1.22%	0.98%	0.66%	0.64%	1.33%	1.34%
9. $a=0, b=1$	5	1.22%	0.49%	1.95%	1.91%	22.46%	22.45%
10. $a=0.1, b=1$	5	Inestable	Inestable	Inestable	Inestable	Inestable	Inestable
11. $a=0.1, b=1$	5	Inestable	Inestable	Inestable	Inestable	Inestable	Inestable
12. $a=0, b=1$	2.5	1.22%	0.99%	1.95%	1.91%	0.84%	0.82%
13. $a=0.1, b=1$	2.5	1.21%	0.98%	1.96%	1.92%	1.24%	1.26%
14. $a=0.1, b=1$	2.5	1.22%	0.98%	1.95%	1.91%	0.85%	0.84%

Tabla 1

De los resultados se concluye que los algoritmos presentan inestabilidad al aumentarse el grado de acoplamiento ( $a, b = 1$ ), lo que está de acuerdo con lo presentado en IV.1 y IV.1.3.

Los algoritmos que presentan la mayor aproximación a la solución analítica de la ecuación prueba (1), corresponden a los algoritmos III.5.1 que interpola de los subsistemas lentos a los rápidos y a la combinación de los algoritmos III.5.2 y III.5.5 correspondiente al experimento 14.

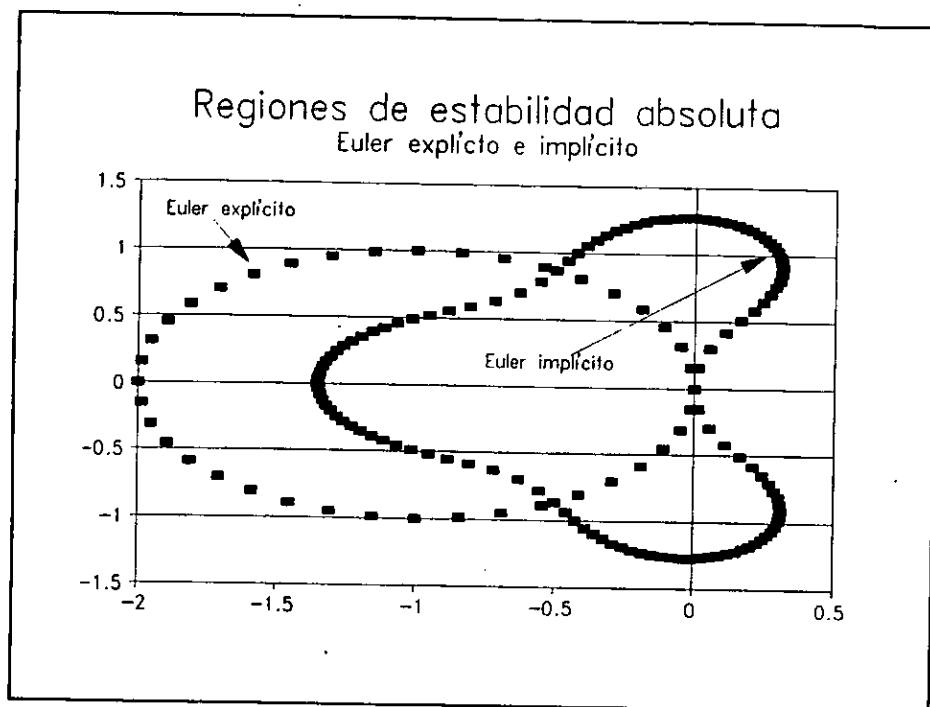
En cuanto a los algoritmos III.5.2 y III.5.3 presentan diferencias mínimas entre ellos respecto a la solución analítica, aunque tiende a ser más aproximado el algoritmo III.5.2 que el III.5.3 cuando es mayor el acoplamiento del sistema (experimentos 4 y 5).

En el experimento 9 el subsistema lento ( $y_5, y_6$ ) presenta errores grandes cercanos al 23%, cuya tendencia es ir creciendo, lo que ya no es aceptable. Los experimentos 9, 10 y 11 son inestables, el algoritmo aplicado a los tres subsistemas es el III.5.5 que usa el método de Euler implícito (información adelantada), por lo que las magnitudes de las raíces del polinomio de estabilidad (en forma desacoplada) están muy próximas a la unidad, esto es

$$|1 + \lambda h + (\lambda h)^2 + (\lambda h)^3| = 1,$$

para los pasos de integración  $h_1, h_2, h_3$  y los valores característicos de los tres subsistemas. Consecuentemente al aumentar el acoplamiento del sistema ( $a=0.01$ , ó  $a = 0.1$ ) se vuelve inestable el algoritmo.

En los experimentos 12, 13 y 14 la situación anterior se evita, integrando al subsistema lento por medio del algoritmo III.5.2 que usa el método Euler explícito. La región de estabilidad absoluta del Euler explícito es más amplia en su intersección con el eje real que la del Euler implícito con estrategia P(EC)<sup>2</sup>C, como se puede apreciar en la gráfica 1 y en consecuencia puede capturar a  $|1 + 100h_3\lambda| < 1$  sin que quede muy próximo a la unidad. Los subsistemas moderado y rápido conservan su método integrador, esto es, se integran por medio del algoritmo III.5.5.



Gráfica 1

### § VI.2.2 Segundo Experimento: Aplicación de los MIM's al Modelo Simplificado de una Planta Termoeléctrica.

Una unidad generadora de energía eléctrica en su forma más simple está compuesta de una caldera que convierte el agua en vapor y una turbina que al recibir el flujo de vapor gira obligando a su vez a girar al generador eléctrico. El modelo matemático que se utiliza en el presente trabajo fue desarrollado por Usoro [30] y posteriormente adaptado en el Instituto de Investigaciones Eléctricas, éste

representa una unidad de 400 Mega Watts (MW) de potencia.

Las ecuaciones correspondientes a la dinámica del proceso están basadas en principios básicos de conservación: balances de momentos, balances de energías y balances de masa [30], además de las relaciones termodinámicas y curvas características del equipo auxiliar [5], [16]. El modelo matemático de la unidad termoeléctrica (UT) está representado por un sistema de 27 ecuaciones diferenciales ordinarias con valores iniciales de primer orden no lineales y 154 ecuaciones algebraicas lineales [6].

Con la finalidad de que el lector se forme una idea general de la estructura de las ecuaciones que forman el modelo matemático de la UT, se presentan las ecuaciones que constituyen al sobrecalentador, atemperador y control de salida de vapor del sobrecalentador:

Entradas:  $W_{sp}, W_{dts}, A_{cv}, h_{dts}, h_{jpo}$

Salidas:  $T_{sso}, P_{sso}, W_{sso}, h_{sso}$

Ecuaciones diferenciales:  $\rho_{sso}, h_{sso}$

Relaciones algebraicas:

$$\begin{aligned}
 P_{sso} &= -291.36 - 964.04 \rho_{sso} + 0.2178 h_{sso} + 1.1815 \rho_{sso} h_{sso}, \\
 S_{sso} &= 1.3136 - 1.799 \times 10^{-3} \rho_{sso} + 6.3573 h_{sso} - 8.3591 \times 10^{-2} \ln(\rho_{sso} h_{sso}), \\
 T_{sso} &= -1285.4 + 129.1 \rho_{sso} + 1.811 h_{sso} - 0.06631 \rho_{sso} h_{sso}, \\
 W_{sso} &= K_{cv} A_{cv} P_{sso}
 \end{aligned} \tag{9}$$

Ecuaciones diferenciales ordinarias:

$$\begin{aligned} \rho'_{sso} &= \frac{(W_{drs} + W_{sy} - W_{sso})}{K_{vsh}} \\ h'_{sso} &= \frac{(h_{drs} W_{drs} + h_{fpo} W_{sy} + Q_{sh} - h_{sso} W_{sso})}{K_{mshe}} \end{aligned} \quad (10)$$

las variables que inician con  $K$  son constantes.

De las relaciones (9) y (10) se observa, que en general la estructura del sistema de ecuaciones algebraico diferencial que representa al modelo de la UT es la siguiente:

$$\begin{aligned} y' &= f(y,z), \\ 0 &= g(y,z) \end{aligned} \quad (11)$$

donde

$$y = (y_1, \dots, y_m)^T, \quad z = (z_1, \dots, z_n)^T \quad (12)$$

son las variables de estado y las ecuaciones algebraicas respectivamente.

Además se cumple que la matriz jacobiana

$$\frac{\partial g(y,z)}{\partial z} \quad (13)$$

es invertible, luego por el teorema de la función implícita la función  $g$  definida en (11) tiene una solución única  $z = \Phi(y)$ , que al sustituirla en (11) se obtiene

$$y' = f(y, \Phi(y)) \quad (14)$$

En nuestro caso particular, el modelo matemático de la UT que es usado en los experimentos las ecuaciones algebraicas  $z$  son lineales y tienen la siguiente estructura



$$Az = By + C,$$

donde  $A$ ,  $B$  y  $C$  son matrices y  $A$  es invertible, lo que facilita el manejo de las ecuaciones algebraicas.

Al lector interesado en el detalle del modelado la UT se le invita a consultar la referencia [30].

La versión del modelo de la UT adaptada en el IIE se programó en forma modular, ésta versión consiste de 23 módulos, cada uno representando una parte del proceso. La modularización del modelo se hizo en base a consideraciones físicas, el método integrador seleccionado (para ésta versión) es un Runge-Kutta de orden cuarto con paso de integración de 0.1 seg. Esta es la versión que se utiliza para realizar comparaciones con las simulaciones realizadas con los métodos de integración multitasa.

Para poder adaptar al modelo matemático de la UT un MIM es necesario desacoplarlo en sus diferentes dinámicas en caso de ser esto posible, por lo que es necesario hacer un análisis de desacoplamiento y de estabilidad del modelo.

El análisis de desacoplamiento nos informa del número de modos (escalas temporales) que se encuentran presentes en el modelo y si es posible separario. En base a ésta información se decide el número de modos en que se divide el modelo y si éste reúne los requisitos para aplicarle un MIM: matriz jacobiana muy cercana a una triangular inferior y que la parte denominada rápida tenga un reducido número de ecuaciones.

El análisis de estabilidad del modelo nos permite seleccionar los algoritmos de

integración más adecuados para tiempo real a los subsistemas en que se divida el modelo.

Para realizar los análisis se seleccionó un transitorio, el cual consiste en ejecutar una simulación con las siguientes características:

- a) Iniciar la simulación al 100% de la demanda de carga (400 MW) durante 10 segundos, usando las condiciones de inicio al 100% de carga.
- b) Reducir la demanda de carga a una razón del 15% por minuto (90 MW/min) hasta alcanzar el 77.5 % de la demanda de carga (310 MW).
- c) Continuar con el 77.5% de carga hasta alcanzar los 400 segundos de simulación.

El análisis de desacoplamiento del modelo de la UT se elabora por medio de los valores y vectores característicos de la matriz jacobiana asociada al modelo linealizado de la UT. En consecuencia, es necesario aproximar numéricamente a la matriz jacobiana del sistema de ecuaciones diferenciales ordinarias que representa a la UT en diferentes tiempos del transitorio simulado.

La selección de los tiempos en donde se aproxima la matriz jacobiana se definieron en base al análisis de la gráfica de la potencia de salida del generador, una de las principales variables a seguir [30]. Se aproximaron 16 matrices jacobianas en los tiempos: 0, 5, 15, 30, 45, 60, 75, 90, 95, 110, 120, 130, 140, 150, 200 y 380 seg.

A continuación se da una breve explicación de la metodología seguida para realizar del desacoplamiento del modelo matemático de la UT, invitamos al lector a consultar las referencias [25] y [26] en donde se exponen con más detalle estos

conceptos.

El desacoplamiento se hace a través del análisis de la **matriz  $P$  de participación** de orden  $n$  [25], [26], la cual está definida por

$$P = (p_{ki}) = (u_{ki}v_{ki}), \quad (16)$$

donde  $v_{ki}$ ,  $[u_{ki}]$  es el  $k$ -ésimo elemento del  $i$ -ésimo vector característico derecho [izquierdo]  $v_i$ ,  $[u_i]$  de  $J$ , la matriz jacobiana del sistema de ecuaciones diferenciales ordinarias que representan al modelo.

Los vectores característicos derechos [izquierdos] están definidos por

$$\begin{aligned} Jv_i &= \lambda_i v_i, \quad v_i \neq 0, \\ u_i^T J &= \lambda_i u_i^T, \quad u_i \neq 0, \end{aligned} \quad (17)$$

donde  $\lambda_i$  es el  $i$ -ésimo valor característico, estos se normalizan para que satisfagan la siguiente relación

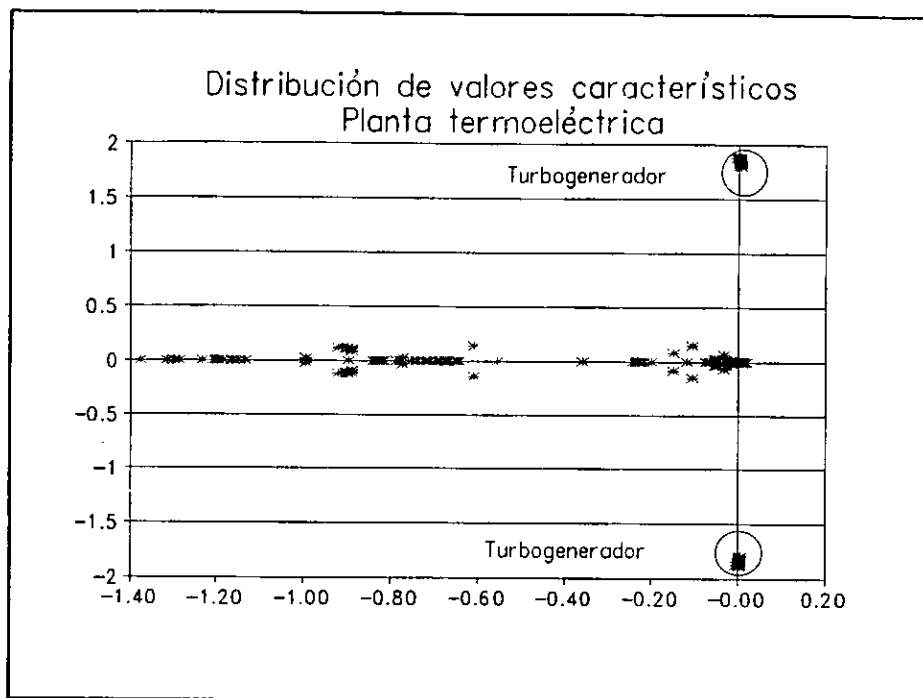
$$u_i^T v_j = \begin{cases} 1, & \text{sí } i=j \\ 0, & \text{sí } i \neq j \end{cases} \quad (18)$$

Consecuentemente, la suma de los valores de todas los elementos de un renglón o columna de  $P$  es siempre igual a 1. A los elementos de  $P$  se les llama los **factores de participación** del sistema y no tienen dimensiones, es decir, son independientes de la elección de unidades de ingeniería en que se representa al modelo de la UT, lo que evita problemas de escalamiento.

La magnitud de las  $p_{ki}$  dan la medida de la participación relativa de la  $k$ -ésima variable de estado en el  $i$ -ésimo estado y viceversa.

A cada una de las matrices jacobianas aproximadas se les calculó su matriz  $P$  de participación, sus vectores y valores característicos.

En la gráfica 2 se presentan la evolución de los valores característicos de todas las matrices jacobianas calculadas en el transitorio simulado:



**Gráfica 2**

La figura 1 presenta un diagrama simbólico de la matriz de participación  $P$ , ésta se interpreta de la siguiente forma, los números en el primer renglón horizontal se refieren a la numeración dada a los valores característicos, la primera columna contiene los nombres asociados a las variables de estado, las siguientes columnas presentan los símbolos . + \* # que se refieren a los valores de las componente de

la matriz  $P$  de participación. Los símbolos tienen el siguiente significado:

Un . (punto) significa que el  $p_{ki}$  elemento de  $P$  su valor está entre  $0 \leq p_{ki} < 0.01$  .

Un + (mas) significa que el  $p_{ki}$  elemento de  $P$  su valor está entre  $0.01 \leq p_{ki} < 0.1$  .

Un \* (asterisco) significa que el  $p_{ki}$  elemento de  $P$  su valor está entre  $0.1 \leq p_{ki} < 1$  .

Un # (gato) significa que el  $p_{ki}$  elemento de  $P$  su valor es  $p_{ki} = 1$  .

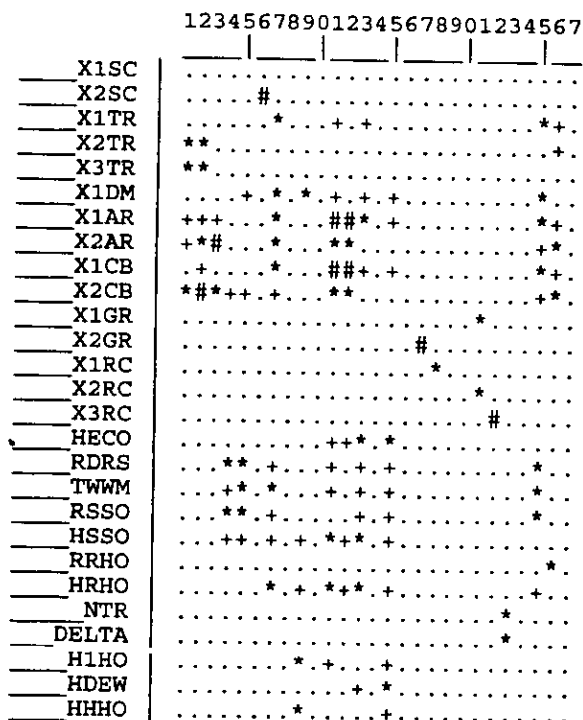


Figura 1

Observe que del diagrama simbólico de la matriz  $P$  de participación se obtiene la asociación de los valores característicos con las variables de estado.

Del análisis de los resultados, consultar gráfica 2 y figura 1, se concluye que

1. Los valores característicos se localizan en zonas claramente definidas del plano complejo a lo largo de toda la simulación de 0 a 400 segundos.
2. Existen dos valores característicos con parte real casi cero y parte imaginaria cercana a  $\pm 2i$ , estos están asociados a las variables de estado NTR y DELTA, correspondientes a la velocidad de la turbina y el ángulo de potencia del generador.
3. El modelo de la UT tiene definidas dos escalas temporales, siendo el subsistema rápido el turbogenerador, variables de estado NTR y DELTA, y el subsistema lento lo identificamos con las restantes variables de estado.
4. El acoplamiento entre el subsistema rápido y lento es muy débil o nulo, ya que sus factores de participación se encuentran entre 0 y 0.01.

El modelo simplificado de la UT se dividió entonces en dos escalas temporales, y sus módulos se dividieron de la siguiente forma:

21 módulos correspondieron a la parte lenta y,

2 módulos a la parte rápida

El número de ecuaciones diferenciales ordinarias correspondientes al modo rápido son 5 y la parte lenta tiene las 22 restantes.

De la conclusión 4, se puede aplicar la teoría presentada en el capítulo V, esto es,

suponemos que el sistema de ecuaciones diferenciales esta desacoplado y escogemos los algoritmos de integración tal que las raíces de su polinomio de estabilidad se encuentre en la región de estabilidad absoluta del método de integración.

Debido a la magnitud de los valores característicos asociados al subsistema lento, y dadas las características de la región de estabilidad absolutas del Euler explícito V.(33), el subsistema lento se integra por medio de éste método con paso de integración  $h_1=0.25$ . Análogamente, el método más adecuado de los algoritmos analizados por la presencia de los valores característicos con parte real casi cero y parte imaginaria  $\pm 2i$ , es por las características de su región de estabilidad absoluta V.(33), para el caso del subsistema rápido el Euler implícito con paso de integración  $0 < h_2 \leq 0.125$ , programado con la estrategia P(EC)<sup>2</sup>E.

Se definió entonces, un ciclo de 0.25 seg. y los siguientes factores  $k$  multitasa de 2, 4 y 8, lo que da los siguientes pasos para el subsistema rápido: 0.125, 0.0625 y 0.03125. No se interpola de los subsistemas lentos al rápido y se usa la estrategia de información adelantada (algoritmos III.5.2 y III.5.5).

Se hicieron tres experimentos para cada uno de los factores multitasa, en cada uno de estos se realizo una simulación de 0 a 400 seg. con el transitorio descrito anteriormente.

Para comparar el desempeño del MIM implantado al modelo simplificado de la UT se tomaron muestras cada segundo de las 27 variables de estado y 42 variables de salida obteniéndose los máximos errores relativos en porcentaje de cada una de las variables en toda la simulación, respecto al patrón de comparación que se definió: el

modelo integrado en forma convencional con un Runge-Kutta de orden cuarto y paso de integración  $h=0.1$ . Además se obtuvieron los errores máximos relativos en porcentaje alcanzados en el estado estacionario para las variables de estado. Las fórmulas empleadas son las siguientes

$$Er\_Rel\_Var_{m\acute{a}x} = \underset{ij}{m\acute{a}x} \left( \frac{|v_{iex}^j - v_{iap}^j|}{|v_{iex}^j|} \right) 100, \quad i=1,\dots,400, \quad j=1,\dots,69, \quad (19)$$

$$Er\_Rel\_Estc_{m\acute{a}x} = \underset{j}{m\acute{a}x} \left( \frac{|x_{fex}^j - x_{fap}^j|}{|x_{fex}^j|} \right) 100, \quad j=1,\dots,27. \quad (20)$$

donde

$v_{iex}$  variables de estado y salidas del modelo de la UT obtenidas por medio del método integrador Runge-Kutta de orden 4 programado en forma convencional.

$v_{iap}$  variables de estado y salidas del modelo de la UT obtenidas por medio del **MIM**.

$x_{fex}$  valor final de las variables de estado de la UT obtenidas por medio del método integrador Runge-Kutta de orden 4 programado en forma convencional.

$x_{fap}$  valor final de las variables de estado de la UT obtenidas por medio del **MIM**.

En la sección VI.4 se encuentran tres tablas con los resultados que se obtuvieron para los tres experimentos. En la tabla 2 se presentan los máximos errores relativos en



porcentaje, obtenidos para cada uno de las variables de estado, la tabla 3 presenta los máximos errores relativos en porcentaje de cada uno de las variables de estado respecto al estado final alcanzado (nuevo estado estacionario), finalmente en la tabla 4 se presentan los máximos errores relativos en porcentaje para cada una de las variables de salidas del modelo de la UT durante el tiempo simulado.

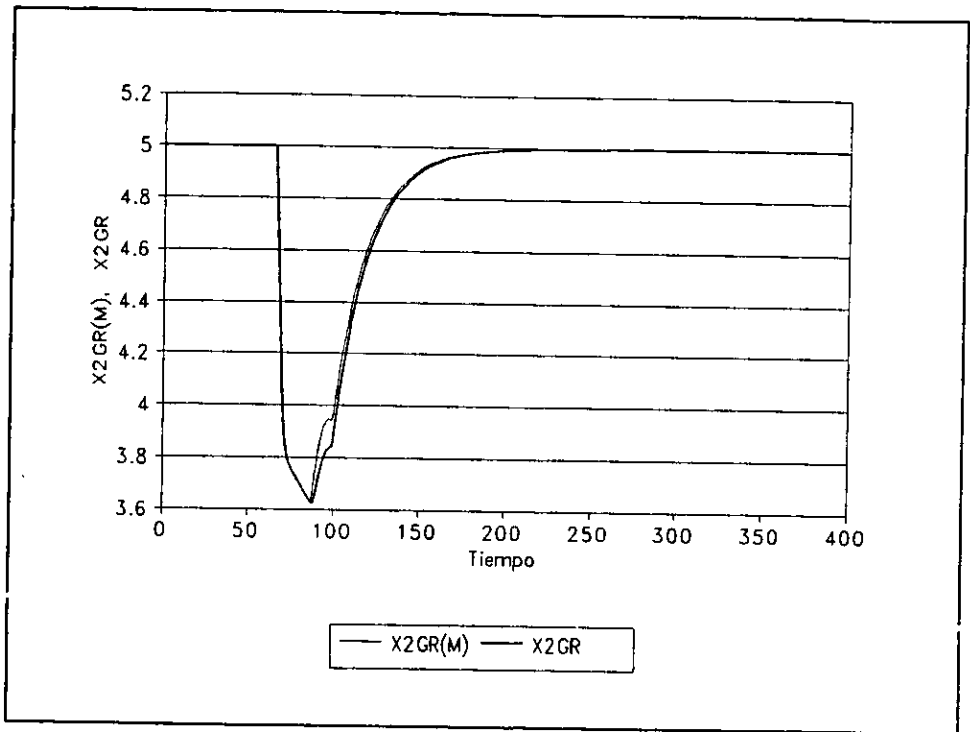
Existen tres variables que presentan los mayores errores relativos en cada uno de los experimentos, estas son la variable de estado **X2GR** con error aproximado del 3%, y las variables de salida **QWWGM** y **QSH** con errores aproximados de 6% y 8%.

No se encontraron diferencias significativas en los errores relativos para las variables de estado y variables de salida para cada uno de los experimentos, como se puede ver de las tablas 2 y 4.

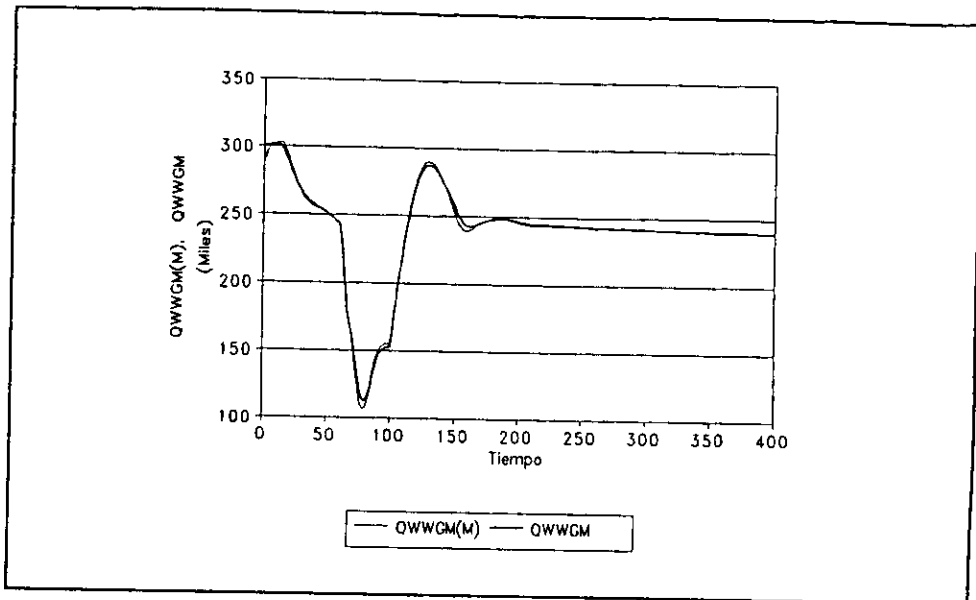
Esto se explica en parte debido a que en los tres experimentos la longitud del ciclo permanece constante (0.25 de seg.) y no se está interpolando, sino que se toma la información adelantada del subsistema lento. Lo que se varió fue el factor multitasa  $k$  (2, 4, y 8), lo que hace que la parte rápida se integre más veces por ciclo, pero la información adelantada con la que se actualiza el subsistema rápido sigue siendo la misma.

De acuerdo a lo anterior, se escogió el primer experimento para realizar las comparaciones ya que en cuanto error no existen diferencias significativas respecto a los otros experimentos, y además es el que menos tiempo de la Unidad de Procesamiento Central (UPC) consume debido a que la parte rápida es ejecutada el menor número de veces por ciclo por su factor multitasa  $k=2$ .

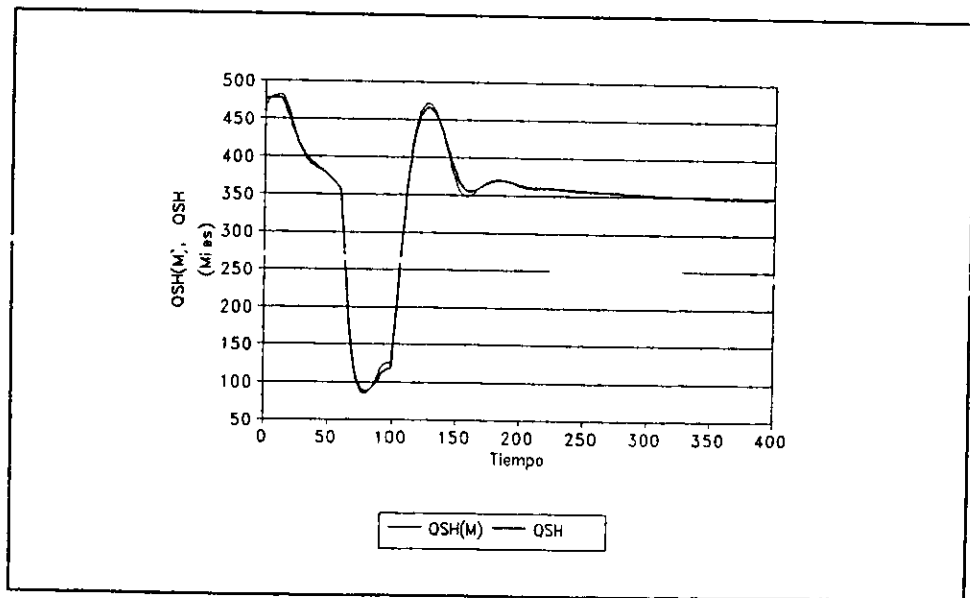
La representación gráfica de las variables del experimento 1 que tienen el mayor error se presentan en las gráficas 3, 4 y 5 respectivamente, y se comparan respecto a las "exactas", las obtenidas integrando en forma convencional la UT por medio de un Runge Kutta 4. Las variables aproximadas por el MIM tienen pospuesto (M).



Gráfica 3



Gráfica 4



Gráfica 5

Estos errores satisfacen los denominados criterios de fidelidad establecidos en la norma ANSI / ANS-3.5-1981 editados por la "American Nuclear Society" para la construcción de simuladores de entrenamiento.

El ahorro en el costo computacional de un **MIM** respecto a un método integrador convencional se manifiesta en el menor número de evaluaciones realizadas en las ecuaciones algebraicas y las ecuaciones diferenciales ordinarias. A continuación se presenta el número de evaluaciones realizadas con el método integrador convencional y el **MIM** aplicados al modelo simplificado de la UT.

El modelo simplificado de la UT tiene un total de

- \* 134 ecuaciones algebraicas,
- \* 27 ecuaciones diferenciales ordinarias.

Un método integrador Runge-Kutta de orden 4 a paso fijo  $h=0.1$ , implantado en forma convencional hace:

No. de evaluaciones de ecuaciones algebraicas por segundo	5,360
No. de evaluaciones de ec. diferenciales ordinarias por seg.	1,080
<b>Total de evaluaciones (algebraicas+derivadas) por segundo</b>	<b>6,440</b>

En el caso del **MIM**, el subsistema rápido esta constituido por:

- \* 18 ecuaciones algebraicas,
- \* 5 ecuaciones diferenciales ordinarias,

el subsistema lento tiene las ecuaciones restantes, es decir

- \* 116 ecuaciones algebraicas,
- \* 22 ecuaciones diferenciales ordinarias.

El MIM realiza el siguiente número de evaluaciones:

<b>Subsistema Lento</b>	
No. de evaluaciones de ecuaciones algebraicas por segundo	464
No. de evaluaciones de ec. diferenciales ordinarias por seg.	88
Total sub. lento de evaluaciones (algebraicas+derivadas)	552
<b>Subsistema Rápido</b>	
No. de evaluaciones de ecuaciones algebraicas por segundo	144
No. de evaluaciones de ec. diferenciales ordinarias por seg.	40
Total sub. rápido de evaluaciones (algebraicas+derivadas)	284
<b>Gran total MIM (algebraicas+derivadas) por segundo</b>	<b>836</b>

De los datos presentados en las tablas anteriores se concluye que el MIM reduce el costo computacional, respecto al número de evaluaciones de ecuaciones algebraicas y diferenciales ordinarias en el 87%.

En lo que respecta a los tiempos de la Unidad de Procesamiento Central (UPC), ya

que las pruebas se realizaron en una computadora personal 486 de 66 Mhz con 8 Mb en RAM, el reloj de ésta no tiene la resolución para obtener el tiempo de UPC por paso de integración, por lo que se midió únicamente el tiempo empleado por la UPC en la ejecución de toda la simulación del transitorio. Los resultados obtenidos son los siguientes:

<b>Método integrador convencional Runge-Kutta 4, <math>h = 0.125</math></b>	
Tiempo de UPC en 400 seg. simulados	9.35 seg
<b>Método integrador multitasa, <math>h_1=0.25</math>, <math>h_2=0.125</math>, <math>k=2</math></b>	
Tiempo de UPC en 400 seg. simulados	2.75 seg

Se obtiene entonces, un ahorro del 70.59% en el tiempo de UPC al usar un **MIM** en el caso del ejemplo estudiado.

## § VI.3 TABLAS

Variables Estado	Error Rel. exp. 1	Error Rel. Exp. 2	Error Rel. Exp. 3
X1SC	0.000%	0.000%	0.000%
X2SC	0.000%	0.000%	0.000%
X1TR	2.393%	2.427%	2.451%
X2TR	0.371%	0.374%	0.377%
X3TR	0.368%	0.374%	0.377%
X1DM	0.628%	0.620%	0.620%
X1AR	0.606%	0.596%	0.594%
X2AR	2.334%	2.329%	2.329%
X1CB	0.453%	0.450%	0.450%
X2CB	2.569%	2.569%	2.569%
X1GR	0.000%	0.000%	0.000%
X2GR	3.077%	3.074%	3.077%
X1RC	0.000%	0.000%	0.000%
X2RC	0.000%	0.000%	0.000%
X3RC	0.000%	0.000%	0.000%
HECO	0.067%	0.066%	0.066%
RDRS	0.336%	0.328%	0.323%
TWWM	0.026%	0.018%	0.018%
RSSO	0.220%	0.217%	0.216%
HSSO	0.094%	0.094%	0.094%
RRHO	0.363%	0.370%	0.374%
HRHO	0.080%	0.080%	0.080%
NTR	0.000%	0.000%	0.000%
DELTA	0.294%	0.286%	0.294%
H1HO	0.011%	0.011%	0.011%
HDEW	0.008%	0.008%	0.008%
HHO	0.240%	0.240%	0.240%

Tabla 2

Variables Estado	Error Rel. Exp. 1	Error Rel. Exp. 2	Error Rel. Exp. 3
X1SC	0.000%	0.000%	0.000%
X2SC	0.000%	0.000%	0.000%
X1TR	0.034%	0.033%	0.033%
X2TR	0.003%	0.000%	0.000%
X3TR	0.000%	0.000%	0.000%
X1DM	0.104%	0.101%	0.101%
X1AR	0.098%	0.095%	0.095%
X2AR	0.098%	0.098%	0.098%
X1CB	0.101%	0.101%	0.101%
X2CB	0.098%	0.098%	0.098%
X1GR	0.000%	0.000%	0.000%
X2GR	0.002%	0.002%	0.002%
X1RC	0.000%	0.000%	0.000%
X2RC	0.000%	0.000%	0.000%
X3RC	0.000%	0.000%	0.000%
HECO	0.026%	0.026%	0.026%
RDRS	0.002%	0.002%	0.002%
TWWM	0.000%	0.000%	0.000%
RSSO	0.031%	0.031%	0.031%
HSSO	0.013%	0.013%	0.013%
RRHO	0.021%	0.021%	0.021%
HRHO	0.007%	0.007%	0.007%
NTR	0.000%	0.000%	0.000%
DELT	0.005%	0.005%	0.023%
H1HO	0.000%	0.000%	0.000%
HDEW	0.008%	0.008%	0.008%
HHO	0.005%	0.005%	0.005%

Tabla 3



Variables Salida	Error Rel. Exp. 1	Error Rel. Exp. 2	Error Rel. Exp. 3
WG	4.148%	4.141%	4.146%
TFN1	2.883%	2.772%	2.724%
TWNGO	2.209%	2.127%	2.087%
QWNGM	6.149%	5.920%	5.809%
HDRS	0.036%	0.036%	0.036%
TDRS	0.018%	0.018%	0.018%
PDRS	0.189%	0.183%	0.181%
HDRW	0.074%	0.072%	0.071%
RDRW	0.108%	0.106%	0.106%
WDRS	0.411%	0.387%	0.389%
PSSO	0.201%	0.201%	0.201%
SSO	0.058%	0.058%	0.058%
TSSO	0.147%	0.147%	0.147%
QWMMW	1.905%	1.813%	1.770%
WFW	0.401%	0.427%	0.439%
WSSO	0.209%	0.210%	0.211%
WSY	0.000%	0.000%	0.000%
ACV	0.367%	0.372%	0.377%
WRY	0.000%	0.000%	0.000%
WAR	4.177%	4.170%	4.177%
WGR	4.178%	4.172%	4.175%
WFL	4.638%	4.458%	4.371%
XGG	0.000%	0.000%	0.000%
QSH	7.907%	7.870%	7.890%
QRH	4.935%	4.933%	4.940%
QEC	4.278%	4.112%	4.035%
PHPO	0.219%	0.221%	0.224%
PRHO	0.229%	0.231%	0.234%
TRHO	0.169%	0.169%	0.169%
WHPO	0.208%	0.210%	0.211%
HHPO	0.064%	0.064%	0.064%
THPO	0.152%	0.149%	0.149%
MWHP	0.140%	0.122%	0.122%

WRHO	0.230%	0.231%	0.234%
MWLP	0.225%	0.219%	0.218%
HFPO	0.008%	0.008%	0.008%
HCPO	0.000%	0.000%	0.000%
MWGN	0.280%	0.271%	0.280%
HLHD	0.030%	0.030%	0.030%
HDED	0.039%	0.039%	0.039%
HHHD	0.046%	0.046%	0.046%
MWRTO	0.117%	0.105%	0.099%

Tabla 4

ESTA TESIS NO DEBE  
SALIR DE LA BIBLIOTECA

# CONCLUSIONES

Se presentó un conjunto de algoritmos de integración multitasa de bajo orden con paso de integración constante, cuyo desempeño es superior a los convencionales en ambientes de tiempo real en el que las necesidades de aproximación no son tan exigentes, éstos superan las restricciones que imponen a los métodos de integración numérica los elementos modelados de los procesos en ambientes de simulación en tiempo real.

Se desarrollo una teoría para el error de truncamiento local que demuestra que éste tiende a cero cuando  $h$  tiende a cero, y se caracterizó las regiones de estabilidad absoluta de los MIM's únicamente en el caso cuando la componente lenta del sistema modelado es independiente de la rápida (o viceversa). Sin embargo, no se estableció ningún criterio simple para caracterizar las regiones de estabilidad absoluta cuando existen acoplamientos fuertes entre las componentes lentas y rápidas.

Se demostró que los algoritmos que tienen el menor error de truncamiento local son los que utilizan interpolación de los subsistemas lentos a los rápidos y los algoritmos que utilizan información adelantada o atrasada su error de truncamiento local está fuertemente afectado por el grado de acoplamiento del sistema rápido con el lento.

Se encontraron los polinomios de estabilidad de cada uno de los algoritmos presentados.

En cuanto al uso práctico de los algoritmos analizados se demostró que estos se deben aplicar únicamente en los casos en que existe un acoplamiento muy débil entre las componentes rápidas y lentas, en caso contrario los algoritmos pueden presentar problemas de estabilidad con pasos de integración que no sean lo suficientemente pequeños.

Se demostró por medio de la matriz de participación, que la característica de acoplamiento débil la presenta el modelo simplificado de la planta termoeléctrica analizado, el cual representa adecuadamente la estructura de los modelos de procesos de plantas generadoras de energía eléctrica.

Por otra parte, por las características que presentan en cuanto aproximación, estabilidad absoluta, menor complejidad en la lógica de programación, menores requerimientos de memoria y flexibilidad para cambiar los métodos de integración asociados a los subsistemas en los que se divide el proceso modelado, los algoritmos de integración multitasa que utilizan la estrategia de información adelantada, presentan un desempeño eficiente para ambientes de simulación de plantas generadoras de energía eléctrica en tiempo real.

# BIBLIOGRAFIA

- [1] Anderson L., *Decoupling of Time-Scale Linear Systems*, Proc. JACC, Philadelphia, 1978, pp. 153-163.
- [2] Andrus J. F., *Numerical Solution of Systems of Ordinary Differential Equations Separated Into Subsystems*, SIAM J. Numer. Anal., 16 (1979), pp.605-611.
- [3] Blum H., *Numerical Integration of Large Scale Systems Using Separate Step Sizes*, en A. W. BEMMETT and R. VICHNEVTSKY, Eds., Numerical Methods for Ordinary Differential Equations and Simulation J, North-Holland, Amsterdam 1978, pp. 19-23.
- [4] Castelazo I. A., *Simulador de Plantas Generadoras para Entrenamiento de Operadores*, Rpt. Interno del Instituto de Investigaciones Eléctricas, 1983.
- [5] Corripio A. B. y Smith C. A., *Control Automático de Procesos*, Ed. Limusa 1991.
- [6] Gallardo V. F., *Implementación de la fórmula BDF de orden 1 a paso fijo sin control de error para la aproximación numérica de modelos matemáticos de procesos de plantas generadoras de energía eléctrica para simulación en tiempo real*, tesis de licenciatura en Actuaría, en preparación, 1996.
- [7] Gear C. W., *Numerical Initial Value Problems in Ordinary Differential Equations*, ed., Prentice-Hall, 1971.

- [8] Gear C. W., *Multirate Methods for Ordinary Differential Equations*, Rpt. No. UIUCDCS-F-74-880 Dept. of Computer Sci., Illinois, 1974.
- [9] Gear C. W., *Simulation: Conflicts Between Real-Time and Software*, en *Mathematical Software III*, J. R. Rice, ed., Academic Press, New York, 1977, pp. 121-138.
- [10] Gear C. W., *Automatic Multirate Methods for Ordinary Differential Equations*, Rpt. No. UIUCDCS-R-80-1000, Dept. of Computer Sci., Illinois, 1980.
- [11] Gear C. W., *Numerical Solution of Ordinary Differential Equations : Is There Anything Left To Do?*, *SIAM Rev.*, 23 (1981), pp. 10-24.
- [12] Gear C. W., *Stiff Software: What do we have and what do we need?*, Rpt. No. UIUCDCS-R-82-1109 Dept. of Computer Sci., Illinois, 1982.
- [13] Gear C. W., Wells D. R., *Multirate Linear Multistep Methods*, *BIT*, 24 (1984), pp. 484-502.
- [14] Gomm W., *Stability of Explicit Multirate Methods*, *Mathematics and Computers in Simulation XXIII* (1981), pp. 34-50.
- [15] Henrici P., *Discrete Variable Methods in Ordinary Differential Equations*, John Wiley & Sons, Inc., New York, 1962.
- [16] Himmelblau D. M., *Principios y Cálculos Básicos de Ingeniería Química*, CECSA, 1992.
- [17] Hirsh M. W., Smale S., *Differential Equations, Dynamical Systems, and Linear Algebra*, Academic Press Inc., 1974.

- [18] Keepin W. N., *Solution Methods for Simulation of Nuclear Power Systems, Vol. 2., Multirate Integration of Two-Time Scale Dynamic Systems*, EPRI-NP-2341-LD, 1982
- [19] Lambert J. D., *Computational Methods in Ordinary Differential Equations*, ed., J. Wiley, New York, 1979.
- [20] Lang S., *Linear Algebra*, Addison-Wesley Publishing Company, 1969.
- [21] Ledet W. P., Himmelblau D. M., *Decomposition Procedures for the Solving of Large Scale Systems*, Adv. Chem. Eng. 8, pp. 185-255, 1970.
- [22] Luyben W. L., *Process Modeling, Simulation, and Control for Chemical Engineers*, McGraw-Hill Inc., 1981.
- [23] Ojeda R. C., Rodríguez G. G., *Ordenamiento de un Sistema de Ecuaciones Algebraicas No Lineales para Reducir el Grado de Acoplamiento entre Ellas, Aplicado a Simulación Dinámica de Procesos*, Proc. del IV Coloquio de Control Automático del AMCA, pp. 188-197, 1985.
- [24] Orailoglu A., *A Multirate Ordinary Differential Equation Integrator*, Rpt. No. UIUCDCS-R-79-959 Dept. of Computer Sci., Illinois, 1979.
- [25] Pérez A. I. J., Verghese G. C., Schweppe F. C., *Selective Modal Analysis with Applications to Electric Power Systems, Part I: Heuristic Introduction*, IEE Trans. Power App. & Systems, PAS-101, 1982, PP. 3117-3125.
- [26] Pérez A. I. J., Verghese G. C., Schweppe F. C., *Selective Modal Analysis with Applications to Electric Power Systems, Part II: The Dynamic Stability Problem*, IEE Trans. Power App. & Systems, PAS-101, 1982, PP. 3126-3134.

- [27] Rudin W., *Principles of Mathematical Analysis*, MacGraw-Hill Kogakusha, Third Edition, 1976.
- [28] Shampine L. F., *Computer Solution of Ordinary Differential Equations*, W. H. Freeman and Company, 1975.
- [29] Stoer J., Burlish R., *Introduction to Numerical Analysis*, Springer-Verlag New York Inc., 1980.
- [30] Usoro P. B., *Modeling and Simulation of a Drum Boiler-Turbine Power Plant Under Emergency State Control*, M. S. thesis, Department of Mechanical Engineering, M. I. T., 1977.
- [31] Wells D. R., *Multirate Linear Multistep Methods for the Solution of Systems of Ordinary Differential Equations*, Rept. No. UIUCDCS-R-82-1093, Dept. Computer Sci., Illinois, 1982.