



Métodos Exponential Fitting y Adaptados para Problemas Stiff

Autor: Jesús Martín Vaquero

Director: Jesús Vigo Aguiar

Tesis presentada para optar al Grado de Doctor en Ciencias
Matemáticas

Departamento de Matemática y Física Aplicadas y Ciencias de
la Naturaleza

Septiembre 2005

Autorización del director y del tutor de la tesis
(Art. 8.1 del R.D. 778/98 de 30 de Abril)

D. Jesús Vigo Aguiar, Profesor Titular del Departamento de Matemática Aplicada de la Universidad de Salamanca, como Director de la Tesis Doctoral titulada “Métodos Exponential Fitting y Adaptados para problemas stiff ”, realizada por D. Jesús Martín Vaquero, autoriza la presentación de la misma para optar al grado de Doctor en Ciencias Matemáticas.

Salamanca, Septiembre de 2005

Fdo: Jesús Vigo Aguiar

D. Regino Criado Herrero, Catedrático del Departamento de Matemáticas y Física Aplicadas y CC.NN. de la Universidad Rey Juan Carlos, como Tutor de la Tesis Doctoral titulada “Métodos Exponential Fitting y Adaptados para problemas stiff ”, realizada por D. Jesús Martín Vaquero, autoriza la presentación de la misma para optar al grado de Doctor en Ciencias Matemáticas.

Madrid, Septiembre de 2005

Fdo: Regino Criado Herrero

Agradecimientos

Quiero expresar mi más sincero agradecimiento a todas las personas que han hecho la realización de este trabajo.

A mi director de tesis, el profesor Jesús Vigo Aguiar, por su mucho esfuerzo y por la dedicación que ha mostrado en mi formación académica, así como al resto del grupo investigador al que pertenezco y que tanto me ha enseñado.

A mi tutor en la Universidad Rey Juan Carlos, el profesor Regino Criado, que siempre ha tenido un momento cuando lo he necesitado.

A mis padres y hermana por su apoyo y paciencia al escucharme y saberme aconsejar.

A los demás amigos, familiares y compañeros de trabajo por su ayuda y palabras de ánimo.

Índice general

1. Introducción.	1
2. Métodos EF-I-k-δ o A-I-k-δ.	11
2.1. Introducción.	11
2.2. Construcción de los algoritmos A-I-k- δ para problemas escalares.	13
2.3. Extensión de los BDF adaptados a problemas vectoriales.	18
2.4. Construcción de algoritmos EF-I-k- δ para integrar problemas vectoriales.	19
2.5. Convergencia.	21
3. Métodos A-I-r-γ o EF-I-r-γ.	29
3.1. Introducción.	29
3.2. Construcción de algoritmos EF-I-r- γ para problemas escalares.	30
3.3. Extensión de los EF-I-r- γ a problemas vectoriales.	35
3.4. Construcción de los A-I-r- γ para integrar problemas vectoriales.	36
3.5. Convergencia.	36
4. Otros métodos EF-γ-k-δ-r o A-γ-k-δ-r.	45
4.1. Introducción.	45
4.2. Construcción de los algoritmos EF- γ -k- δ -r o A- γ -k- δ -r.	47
4.3. Convergencia.	58

5. Métodos en paso variable y transición a métodos Runge-Kutta.	67
5.1. Introducción.	67
5.2. Construcción de métodos <i>exponential fitting</i> o adaptados de paso variable.	69
5.2.1. Construcción de métodos EF-I-k- δ o A-I-k- δ de paso variable.	69
5.2.2. Ejemplos: los métodos de paso variable de 2, 3 y 6 pasos. 73	
5.2.3. Métodos EF-2-k-1-r de paso variable.	77
5.3. Métodos <i>exponential fitting</i> de tipo BDF-Runge-Kutta.	85
5.3.1. Construcción de métodos BDF-Runge-Kutta.	86
5.3.2. Construcción de métodos BDF-Runge-Kutta de tipo <i>exponential fitting</i>	88
5.3.3. Propiedades de los métodos BDF-Runge-Kutta de tipo <i>exponential fitting</i>	92
6. Cálculo del parámetro y métodos con dos parámetros.	97
6.1. Introducción.	97
6.2. Métodos de obtención del parámetro introducido en el algoritmo.	98
6.3. Error de truncado global en problemas derivados de EDP's.	114
6.4. Construcción de métodos exponenciales con dos parámetros.	119
6.4.1. Construcción de métodos exponenciales con dos parámetros de dos pasos.	120
6.4.2. Convergencia de los métodos exponenciales con dos parámetros de dos pasos.	121
7. Ejemplos numéricos.	123
7.1. Introducción.	123
7.2. Ecuaciones en derivadas parciales.	132
7.3. Valores propios del jacobiano positivos.	143
7.4. Problemas <i>stiff</i> oscilatorios o altamente oscilatorios.	146
7.5. Otras clases de problemas.	151
7.6. Ejemplos resueltos en paso variable y métodos BDF-Runge-Kutta.	165

<i>ÍNDICE GENERAL</i>	III
8. Conclusiones.	171
A. 0-estabilidad de los métodos de orden exponencial 1.	175
B. Estabilidad absoluta de los métodos de orden exponencial 1.	185
C. 0-estabilidad de los métodos de orden algebraico 1.	203
D. Estabilidad absoluta de los métodos de orden algebraico 1.	215
E. 0-estabilidad de otros métodos EF- γ -k- δ -r o A- γ -k- δ -r.	231
F. Estabilidad absoluta de otros métodos EF- γ -k- δ -r o A- γ -k- δ -r.	243
G. 0-estabilidad de los métodos exponenciales con dos parámetros.	259
H. Estabilidad absoluta de los métodos exponenciales con dos parámetros.	267
I. Estabilidad absoluta de los métodos BDF-Runge-Kutta.	273
Bibliografía	281

Índice de figuras

5.1. Regiones de estabilidad (en gris) del método explícito EF-2-k-1-r de paso variable para parámetros $\lambda h_n \in R^-$ y $\lambda h_{n-1} \in R^-$. Los ejes horizontal y vertical representan respectivamente $\ \lambda h_n\ $ y $\ \lambda h_{n-1}\ $	82
5.2. Regiones de estabilidad (en gris) del método implícito EF-2-k-1-r de paso variable para parámetros $\lambda h_n \in R^-$ y $\lambda h_{n-1} \in R^-$. Los ejes horizontal y vertical representan respectivamente $\ \lambda h_n\ $ y $\ \lambda h_{n-1}\ $	83
7.1. Errores en la integración numérica del problema 1.1 con $N = 10$	135
7.2. Errores en la integración numérica del problema 1.1 con $N = 100$	135
7.3. Errores en la integración numérica en el problema 1.4.	141
7.4. Errores en la integración numérica en el problema 1.5.	142
7.5. Errores en la integración numérica del problema 2.3.	146
7.6. Resultados numéricos al integrar el problema 3.1.	147
7.7. Resultados numéricos al integrar el problema 3.1. Comparación del error cometido con el tiempo CPU empleado.	148
7.8. Error en la integración numérica del problema 3.2.	149
7.9. Error en la integración numérica del problema 3.3.	150
7.10. Error en la integración numérica del problema 3.4.	151
7.11. Error en la integración del problema 4.1.	152
7.12. Error en la integración numérica del problema 4.2 al utilizar métodos de orden alto.	153
7.13. Error en la integración numérica del problema 4.2 al utilizar métodos de orden 3.	153

7.14. Error en la integración numérica del problema 4.8.	159
7.15. Error en la integración numérica del problema 4.9.	161
7.16. Error en la integración numérica del problema 4.13. Comparación del error cometido con el tiempo CPU empleado.	165
7.17. Resultados al integrar numéricamente el problema 5.1.	167
7.18. Resultados al integrar numéricamente el problema 5.1.	167
7.19. Resultados al integrar numéricamente el problema 5.2.	168
7.20. Resultados al integrar numéricamente el problema 5.2.	168
A.1. Región de 0-estabilidad (en gris) de la segunda raíz del método implícito de dos pasos. El eje horizontal representa $Re(\lambda h)$, mientras que el vertical representa $Im(\lambda h)$	176
A.2. Región de 0-estabilidad (en gris) del método implícito de tres pasos. El eje horizontal representa $Re(\lambda h)$ y el vertical representa $Im(\lambda h)$	178
A.3. Regiones de 0-estabilidad (en gris) de los métodos implícitos adaptados. Los ejes horizontal y vertical representan $Re(\lambda h)$ y $Im(\lambda h)$ respectivamente.	179
A.4. Detalle de la 0-estabilidad (en gris) del método implícito adaptado de siete pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	180
A.5. Regiones de 0-estabilidad (en gris) de los métodos implícitos adaptados de ocho y nueve pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	181
A.6. Regiones de 0-estabilidad (en gris) de los métodos explícitos de tipo <i>exponential fitting</i> de dos y tres pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	182
A.7. Regiones de 0-estabilidad (en gris) de los métodos EF-I-k implícitos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	183
B.1. Región de estabilidad absoluta (en gris) del algoritmo adaptado implícito de dos pasos. El parámetro del método está cerca de la zona de 0-inestabilidad. Los ejes horizontal y vertical representan $Re(\nu h)$ e $Im(\nu h)$ respectivamente.	186
B.2. Región de inestabilidad absoluta (en blanco) del algoritmo adaptado implícito de dos paso. Parámetro del método lejos de la zona de 0-inestabilidad. Los ejes horizontal y vertical representan $Re(\nu h)$ e $Im(\nu h)$ respectivamente.	187

B.3. Regiones de estabilidad absoluta (en gris) de los métodos explícitos EF-I-k-2 y A-I-k-2. En los dibujos correspondientes al método EF-I-k-2 los ejes horizontal y vertical representan las partes $Re((\lambda + \nu)h)$ e $Im((\lambda + \nu)h)$, mientras que en los dibujos correspondientes al método A-I-k-2 los ejes horizontal y vertical representan las partes $Re(\nu h)$ e $Im(\nu h)$ 188

B.4. Región de estabilidad absoluta (en gris) del método explícito de dos pasos. El parámetro del método es $\lambda h = -1$. Los ejes horizontal y vertical representan $Re((\lambda + \nu)h)$ e $Im((\lambda + \nu)h)$ respectivamente. 192

B.5. Regiones de estabilidad absoluta (en gris) del método implícito de dos pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 195

B.6. Regiones de estabilidad absoluta (en gris) del método implícito de tres pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 196

B.7. Regiones de estabilidad absoluta (en gris) del método implícito de seis pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 197

B.8. Regiones de estabilidad absoluta (en gris) del método implícito de siete pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 198

B.9. Regiones de estabilidad absoluta (en gris) del método implícito de tres pasos. Parámetro imaginario del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 199

B.10. Regiones de estabilidad absoluta (en gris) del método implícito de cuatro pasos. Parámetro imaginario del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 200

B.11. Regiones de estabilidad absoluta (en gris) del método implícito de seis pasos. El parámetro λh del método es imaginario. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 201

C.1.	Regiones de 0-estabilidad (en gris) de los métodos EF-I-r y A-I-r explícitos de dos y tres pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	204
C.2.	Regiones de 0-estabilidad (en gris) de los métodos EF-I-r y A-I-r explícitos de cuatro y cinco pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	206
C.3.	Regiones de 0-estabilidad (en gris) de los métodos EF-I-r y A-I-r explícitos de número elevado de pasos (diez y quince pasos). Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	207
C.4.	Regiones de 0-estabilidad (en gris) de los métodos EF-I-r y A-I-r implícitos de orden bajo (de dos y tres pasos). Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	208
C.5.	Regiones de 0-estabilidad (en gris) de los métodos EF-I-r y A-I-r implícitos de orden medio (de cuatro y cinco pasos). Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	209
C.6.	Regiones de 0-estabilidad (en gris) de los métodos EF-I-r y A-I-r implícitos de orden alto (de seis y siete pasos). Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	210
C.7.	Regiones de 0-estabilidad (en gris) de los métodos EF-I-r y A-I-r implícitos de orden muy alto (de ocho y nueve pasos). Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	211
C.8.	Detalle de la 0-estabilidad (en gris) del método implícito de siete pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	212
C.9.	Detalle de las regiones de 0-estabilidad (en gris) de los algoritmos implícitos de ocho pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	213
C.10.	Detalle de las regiones de 0-estabilidad (en gris) de los algoritmos implícitos de diez y quince pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	214
D.1.	Detalle de las regiones de 0-estabilidad de los algoritmos EF-I-r-3 y A-I-r-3. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	216

D.2. Evolución de la estabilidad absoluta en función de la cercanía del parámetro λh a las regiones de 0-inestabilidad. 217

D.3. Regiones de estabilidad absoluta (en gris) del método explícito de dos pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 218

D.4. Regiones de estabilidad absoluta (en gris) del método explícito de dos pasos. El parámetro imaginario del método es λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 219

D.5. Regiones de estabilidad absoluta (en gris) del método explícito de tres pasos. El parámetro real del método es λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 221

D.6. Regiones de estabilidad absoluta (en gris) del método explícito de cinco pasos. El parámetro real del método es λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 222

D.7. Regiones de estabilidad absoluta (en gris) del método implícito de dos pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 224

D.8. Regiones de estabilidad absoluta (en gris) del método implícito de cuatro pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 225

D.9. Regiones de estabilidad absoluta (en gris) del método implícito de seis pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 226

D.10. Regiones de estabilidad absoluta (en gris) del método implícito de ocho pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 227

D.11. Regiones de estabilidad absoluta (en gris) del método implícito de cuatro pasos. Parámetro imaginario del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 228

D.12.	Regiones de estabilidad absoluta (en gris) del método implícito de cinco pasos. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$	229
E.1.	Regiones de 0-estabilidad (en gris) de los algoritmos adaptados de 4 pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	233
E.2.	Regiones de 0-estabilidad (en gris) de los algoritmos <i>exponential fitting</i> de 4 pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	234
E.3.	Regiones de 0-estabilidad (en gris) de algoritmos explícitos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	235
E.4.	Regiones de 0-estabilidad (en gris) de los algoritmos EF-2-k-2-r y A-2-k-2-r. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	236
E.5.	Regiones de 0-estabilidad (en gris) de los algoritmos de cinco pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	237
E.6.	Regiones de 0-estabilidad (en gris) de los algoritmos de seis pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	238
E.7.	Regiones de 0-estabilidad (en gris) de los algoritmos de seis pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	239
E.8.	Regiones de 0-estabilidad (en gris) de los algoritmos de siete pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	240
E.9.	Regiones de 0-estabilidad (en gris) de los algoritmos de siete pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.	241
F.1.	Detalle de la región de 0-estabilidad del método implícito A-2-k-2-r. El eje horizontal representa $Re(\lambda h)$, mientras que el vertical representa $Im(\lambda h)$	245
F.2.	Regiones de estabilidad absoluta (en gris) del método EF-2-k-2-r, con un parámetro más cercano a la zona de 0-inestabilidad ($\lambda h = -1 + 7i$) y el otro más alejado ($\lambda h = -1 + 8i$). Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$	246

F.3. Detalle de la región de 0-estabilidad del método implícito A-2-k-3-r. El eje horizontal representa $Re(\lambda h)$, mientras que el vertical representa $Im(\lambda h)$ 246

F.4. Regiones de estabilidad absoluta (en gris) del método EF-2-k-3-r, con un parámetro más cercano a la zona de 0-inestabilidad ($\lambda h = -1 + 4,9i$) y el otro más alejado ($\lambda h = -1 + 3i$). Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 247

F.5. Regiones de estabilidad absoluta (en gris) de los métodos implícitos de cuatro pasos. El parámetro real del método es $\lambda h = -0,01$. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 248

F.6. Regiones de estabilidad absoluta (en gris) de los métodos implícitos de cuatro pasos. El parámetro real del método es $\lambda h = -1$. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 249

F.7. Regiones de estabilidad absoluta (en gris) de los métodos implícitos de cuatro pasos. El parámetro real del método es $\lambda h = -10$. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 250

F.8. Regiones de estabilidad absoluta (en gris) de los métodos implícitos de seis pasos. El parámetro real del método es $\lambda h = -0,1$. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 251

F.9. Regiones de estabilidad absoluta (en gris) de los métodos implícitos de seis pasos. El parámetro real del método es $\lambda h = -1$. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 252

F.10. Regiones de estabilidad absoluta (en gris) de los métodos implícitos de seis pasos. El parámetro real del método es $\lambda h = -10$. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 253

F.11. Regiones de estabilidad absoluta (en gris) del método implícito de cinco pasos EF-2-k-4-r. El parámetro del método es imaginario. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 255

F.12. Regiones de estabilidad absoluta (en gris) del método implícito de cinco pasos EF-3-k-3-r. El parámetro del método es imaginario. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 256

- F.13. Regiones de estabilidad absoluta (en gris) del método implícito de cinco pasos EF-4-k-2-r. El parámetro del método es imaginario. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$ 257
- G.1. Regiones de 0-estabilidad (en gris) de los métodos exponenciales con dos parámetros (parámetros reales). Los ejes horizontal y vertical representan λh y τh respectivamente. 260
- G.2. Regiones de 0-estabilidad (en gris) del método explícito exponencial con dos parámetros, para ciertos valores de τh . Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente. 262
- G.3. Regiones de 0-estabilidad (en gris) del método explícito exponencial con dos parámetros, para ciertos valores de τh . Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente. 263
- G.4. Regiones de 0-estabilidad (en gris) del método explícito exponencial con dos parámetros, para ciertos valores de $t = \frac{\tau}{\lambda}$. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente. 264
- G.5. Regiones de 0-estabilidad (en gris) del método implícito exponencial con dos parámetros, para ciertos valores de τh . Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente. 265
- G.6. Regiones de 0-estabilidad (en gris) del método implícito exponencial con dos parámetros, para ciertos valores de $t = \frac{\tau}{\lambda}$. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente. 266
- H.1. Regiones de estabilidad absoluta (en gris) del método explícito exponencial con dos parámetros, para ciertos valores de λ y τ reales. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente. 269
- H.2. Regiones de estabilidad absoluta (en gris) del método explícito exponencial con dos parámetros, para ciertos valores de λ y τ complejos. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente. 270
- H.3. Regiones de estabilidad absoluta (en gris) del método implícito exponencial con dos parámetros, para ciertos valores de λ y τ reales. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente. 271

H.4. Regiones de estabilidad absoluta (en gris) del método implícito exponencial con dos parámetros, para ciertos valores de λ y τ reales. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente. 272

I.1. Regiones de estabilidad absoluta (en gris) del método BDF-RK con $s = 2$. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente. 274

I.2. Regiones de estabilidad absoluta (en gris) del método BDF-RK con $s = 2$. El parámetro del método es imaginario. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente. 275

I.3. Regiones de estabilidad absoluta (en gris) del método BDF-RK con $s = 3$. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente. 277

I.4. Regiones de estabilidad absoluta (en gris) del método BDF-RK con $s = 3$. El parámetro del método es imaginario. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente. 278

I.5. Regiones de estabilidad absoluta (en gris) del método BDF-RK con $s = 4$. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente. 279

I.6. Regiones de estabilidad absoluta (en gris) del método BDF-RK con $s = 4$. El parámetro del método es imaginario. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente. 280

Capítulo 1

Introducción.

Las ecuaciones diferenciales ordinarias (EDO's) están presentes en numerosos campos de la ciencia, ya que permiten modelizar gran variedad de situaciones habituales en la vida cotidiana. En muchas ocasiones las ecuaciones diferenciales aparecen junto con unas condiciones iniciales, es lo que se conoce como problema de valor inicial (PVI)

$$y^{(m)}(x) = f(x, y(x), y'(x), \dots, y^{(m-1)}(x)),$$

$$y(x_0) = y_0, \quad y'(x_0) = y'_0, \quad \dots, \quad y^{(m-1)}(x_0) = y_0^{(m-1)}.$$

Una ecuación diferencial (de orden m) se puede escribir como un sistema de m ecuaciones diferenciales de primer orden

$$y'(x) = f(x, y(x)), \quad y(x_0) = y_0, \tag{1.0.1}$$

donde $y = (y^1, \dots, y^m)$, $f = (f^1, \dots, f^m)$ e $y_0 = (y_0^1, \dots, y_0^m)$, son vectores de dimensión m , mientras que x y x_0 son escalares, la pretensión es conocer el comportamiento de $y(x)$ en un intervalo $[x_0, x_f] \in R$.

Este es el tipo de problemas que se va a estudiar en este trabajo. Sin embargo, tan sólo consideraremos problemas bajo ciertas condiciones que nos

permitan asegurar que existe solución única de nuestro problema y que $y(x)$ sea continua y diferenciable en un dominio $[x_0, x_f] \times R^m \subset \mathcal{D}$ (ver [65]).

Dichas hipótesis son:

i) $f(x, y)$ está definida y es continua para $x_0 \leq x \leq x_f$, $-\infty \leq y^i \leq \infty$, para $i = 1, \dots, m$,

ii) existe una constante finita L , tal que

$$\| f(x, y) - f(x, y^*) \| \leq L \| y - y^* \|$$

para cualquier $(x, y), (x, y^*)$ de D .

Desgraciadamente, la solución analítica de dichas ecuaciones sólo puede ser obtenida en unos pocos casos. Así, es bien conocido que la función $\exp(-x^2)$ no tiene una función elemental simple cuya derivada sea dicha función. De la misma forma ecuaciones con expresiones tan simples como

$$y'(x) = x^2 + y^2(x), \quad \text{ó} \quad y''(x) = 6y^2(x) + x$$

no pueden resolverse en términos de funciones elementales.

En estos casos, sólo es posible conocer una solución numérica que aproxime la curva cuyas condiciones iniciales y ecuación diferencial sean los datos dados por el PVI. Esto es, se logra una tabla de pares de números (x_i, y_i) , siendo por ejemplo $x_i = x_0 + ih$, e $y_i \approx y(x_i)$, de forma que $x_N = x_0 + Nh = x_f$ (en algunos manuales y aquí también utilizaremos la notación $[x_0, x_f] = [a, b]$ y $b - a = Nh$, h será llamada longitud de paso cuando este valor sea constante).

Sin embargo existen muchos métodos numéricos que no resultan interesantes para resolver un PVI dado, pues lo que queremos es que cuando $h \rightarrow 0$ y

para $x_n = x$, se cumpla que si nh es constante

$$\lim_{h \rightarrow 0} y_n = y(x).$$

Pero además es importante cómo se comporta este límite, ya que si para valores de $h \ll 1$, se sigue dando que la diferencia entre el valor exacto $y(x)$ y el valor estimado y_n , es todavía bastante elevado, entonces estamos en el caso de que el método numérico es muy costoso. Para explicarlo de otra forma, es necesario hacer un número muy elevado de operaciones para hallar una buena aproximación de la solución verdadera.

Con ese fin definimos que un método numérico es **convergente** si, para todo problema de valor inicial, con las hipótesis necesarias para que exista solución única del problema y que ésta sea continua y diferenciable en un dominio \mathcal{D} , entonces tenemos que

$$\max_{0 \leq n \leq N} \|y(x_n) - y_n\| \rightarrow 0 \quad \text{cuando} \quad h \rightarrow 0.$$

De entre los métodos convergentes, aquí se van a tratar métodos de tipo BDF (una clase de métodos multipaso, las iniciales BDF son el acrónimo de *backward differentiation formulae*, ver [64] ó [87]). En general todos los métodos multipaso se pueden escribir como

$$\alpha_0 y_n + \alpha_1 y_{n-1} + \dots + \alpha_k y_{n-k} = h(\beta_0 f_n + \beta_1 f_{n-1} + \dots + \beta_k f_{n-k})$$

donde k es un entero fijo mayor o igual que uno (que da el número de pasos del método), $f_j = f(x_j, y_j)$, y las α_j, β_j , son constantes que no dependen de n ; y_n es el valor que se quiere calcular en cada iteración, mientras que los y_{n-i} , con $0 < i \leq k$ son valores que se han obtenido previamente.

Si reemplazamos y_{n-i} , por $y(x - ih)$, y $hf_{n-i} = hy'_{n-i}$, por $hy'(x_{n-i})$, de esta forma obtenemos

$$\mathcal{L}(y(x_n), h) = \alpha_0 y(x_n) + \alpha_1 y(x_{n-1}) + \dots + \alpha_k y(x_{n-k}) - \\ h(\beta_0 y'(x_n) + \beta_1 y'(x_{n-1}) + \dots + \beta_k y'(x_{n-k})).$$

Utilizando el desarrollo de Taylor de las funciones $y(x - ih)$, e $y'(x - ih)$ en un entorno de $x = x_n$ en la expresión anterior resulta que podemos poner el error como una expresión del tipo

$$\mathcal{L}(y(x_n), h) = C_0 y(x) + C_1 h y'(x) + \dots + C_k h^k y^{(k)}(x) + \dots$$

en ese caso se dice que el método tiene **orden** $k - 1$ cuando se obtiene que $C_0 = C_1 = \dots = C_{k-1} = 0$ y que $C_k \neq 0$.

En ese caso el error producido con dicho método es $C_k h^k y^{(k)}(x) + O(h^{k+1})$, que para $0 < h \ll 1$, es aproximadamente $C_k h^k y^{(k)}(x)$, lo que se conoce como **error de truncado local**.

Diversos estudios (ver, por ejemplo [87]) probaron que **la condición necesaria y suficiente para que el método sea convergente es que sea a la vez consistente y cero-estable**:

i) Si consideramos que el método se obtiene de la fórmula general

$$\alpha_0 y_n + \alpha_1 y_{n-1} + \dots + \alpha_k y_{n-k} = h \phi_f(y_n, \dots, y_{n-k}, x_n; h)$$

y definimos el residuo como

$$R_n := \sum_{j=0}^k \alpha_j y(x_{n-j}) - h \phi_f(y(x_n), \dots, y(x_{n-k}), x_n; h),$$

si

$$\lim_{h \rightarrow 0} \frac{R_n}{h} = 0,$$

en ese caso se dice que el método es **consistente**.

ii) Mientras que diremos que un método es **cero-estable** si dadas dos perturbaciones $\{\delta_n, n = 0, \dots, N\}$ y $\{\delta_n^*, n = 0, \dots, N\}$, y siendo $\{z_n, n = 0, \dots, N\}$ y $\{z_n^*, n = 0, \dots, N\}$ las respectivas soluciones perturbadas definidas por el sistema de diferencias perturbado

$$\sum_{j=0}^k \alpha_j y_{n-j} = h (\phi_f(y_n, \dots, y_{n-k}, x_n; h) + \delta_n),$$

$$y_i = a_i(h) + \delta_i, \quad i = 0, -1, \dots, -k + 1,$$

entonces existen constantes S y h_0 tales que $\forall h \in (0, h_0]$, se cumple que

$$\|z_n - z_n^*\| \leq S\varepsilon, \quad 0 \leq n \leq N$$

siempre que

$$\|\delta_n - \delta_n^*\| \leq \varepsilon \quad 0 \leq n \leq N.$$

En el capítulo dedicado a los ejemplos numéricos veremos que hay muchas deficiones diferentes para explicar lo que es un problema *stiff* (ver [134]), pero la mayoría incluye alguna referencia a que en este tipo de problemas para la mayoría de los métodos explícitos la mayor longitud de paso h_n^* que garantiza la estabilidad numérica es mucho menor que la mayor longitud numérica que nos permite obtener un error de discretización normal (una definición similar aparece en [38] ó [87]).

Esta estabilidad numérica que se asocia al problema y a la longitud de paso se conoce como estabilidad absoluta y es fundamental su estudio en los problemas *stiff*.

Por esta razón en los capítulos dos, tres, cuatro, cinco y seis, después de la construcción de los correspondientes métodos, se estudiará la convergencia de los mismos, y aquí se incluirá consistencia y 0-estabilidad y también estabilidad absoluta.

Tradicionalmente con los problemas *stiff* se han utilizado sobre todo métodos Runge-Kutta (STRIDE, DIRK, Gauss, Rosembrock, Radau, ...) y los de tipo BDF implícitos (LSODE y sus variantes, DASSL, MEBDF, ...). Pero desde los años 70 se han venido utilizando otro tipo de métodos conocidos como *exponential fitting* o métodos adaptados. La idea básica de estos métodos (al menos recientemente) es que integren de forma exacta el problema (1.0.1) cuando $y(x)$ pertenece a un espacio donde e^{Ax} y las constantes están incluidas.

En este trabajo diferenciaremos entre algoritmos de tipo *exponential fitting* que estarán aplicados al problema cuando lo tengamos escrito como

$$y'(x) = G(x, y(x)), \quad y(x_0) = y_0 \quad (1.0.2)$$

y algoritmos adaptados que se aplicarán a problemas del tipo

$$y'(x) - Ay(x) = F(x, y(x)), \quad y(x_0) = y_0, \quad (1.0.3)$$

ya que, aunque en realidad van a originar los mismos métodos, los algoritmos tendrán diferentes propiedades de estabilidad, e incluso en problemas bastante concretos (problemas lineales) puedan tener diferentes resultados, ya que en esos casos los métodos implícitos no funcionarán como tales, pues $F(x, y(x)) = F(x)$ sólo dependerá de la variable temporal x y no habrá que resolver ningún sistema de ecuaciones. Los algoritmos de tipo *exponential*

fitting los denotaremos con las siglas EF, mientras que los adaptados irán representados con la inicial A.

Los aquí presentados serán métodos BDF de tipo *exponential fitting* o adaptados que integran el PVI (empezaremos construyendo el caso escalar y este nos servirá para resolver el vectorial) cuando $y(x)$ pertenece al subespacio generado por $\langle 1, \dots, x^{\delta-1}, e^{\lambda x}, \dots, x^{\gamma-1}e^{\lambda x} \rangle$. Precisamente si hemos construido los algoritmos de forma que integren este subespacio dichos algoritmos se denotarán EF- γ -k- δ -r o A- γ -k- δ -r. En ese caso diremos que el método tiene **orden exponencial γ y orden algebraico δ** .

Los métodos de orden exponencial 1 y orden algebraico δ (integran el PVI cuando $y(x)$ pertenece al subespacio generado por $\langle 1, \dots, x^{\delta-1}, e^{\lambda x} \rangle$), que también denotaremos como EF-I-k- δ o A-I-k- δ , serán estudiados en profundidad en el capítulo 2. En primer lugar construiremos los algoritmos adaptados para resolver problemas escalares, los transformaremos para poder resolver problemas vectoriales y, a partir de ellos, obtendremos los algoritmos *exponential fitting*. Finalizaremos este capítulo examinando la convergencia de dichos métodos, para ello obtendremos su consistencia y se expondrán sus propiedades más importantes de 0-estabilidad y estabilidad absoluta (que se encuentran desarrollados en los apéndices A y B).

El capítulo 3 será dedicado a los métodos de orden algebraico 1 y orden exponencial γ (que integran el PVI cuando $y(x)$ pertenece al subespacio generado por $\langle 1, e^{\lambda x}, \dots, x^{\gamma-1}e^{\lambda x} \rangle$) y que también denotaremos con A-I-r- γ o EF-I-r- γ . El esquema del capítulo será similar al del anterior: en primer lugar se construyen los métodos, esta vez primero los algoritmos *exponential fitting* y luego los adaptados y después se estudia la convergencia de los

métodos: consistencia, 0-estabilidad (que se completa en el apéndice C) y estabilidad absoluta (que se estudia con más detalle en el apéndice D).

En el capítulo 4 construiremos el resto de métodos de orden algebraico δ y orden exponencial γ y que denotaremos con EF- γ -k- δ -r o A- γ -k- δ -r. También calcularemos la consistencia de dichos algoritmos y podremos observar sus propiedades de estabilidad que estarán más desarrolladas en los apéndices E y F.

Todos estos métodos tienen longitud de paso constante. En el capítulo 5 se presentan métodos en paso variable; muchos de los problemas llamados *stiff* presentan zonas claramente diferenciadas, en unas la solución del PVI es estable, mientras en otras varía muy bruscamente, por eso es conveniente utilizar métodos en paso variable en estos problemas. Posteriormente los métodos en paso variable nos servirán también para construir métodos de tipo Runge-Kutta que procederán de los mismos métodos *exponential fitting* de tipo BDF antes presentados. Se podrá observar que es un campo de investigación muy amplio y que por tanto deben permitir en un futuro cercano seguir obteniendo nuevos y buenos resultados.

Todos los métodos hasta ese momento dependen de un sólo parámetro (en algunos casos se puede hablar de frecuencia) Ah , son métodos, como veremos, muy potentes sobre un área muy amplia de tests, sin embargo, para ello es necesario escoger un parámetro del método “adecuado”. En el capítulo 6 se presentarán diferentes formas de obtener Ah y cuya conveniencia está (como parece lógico) en función del tipo de problemas a los que nos enfrentemos. Por supuesto no es tarea nada sencilla y hasta ahora no se han presentado ideas “mágicas” que resuelvan todo tipo de problemas. Una clase de problemas donde el jacobiano suele ser más fácil de hallar es la de las ecuaciones

diferenciales ordinarias procedentes de discretizar en alguna de las variables una ecuación en derivadas parciales, en este caso hemos profundizado en el estudio del análisis del error de truncado global en este tipo de problemas. En este capítulo también mostraremos la forma de obtener métodos que dependen de dos parámetros y cómo varía la estabilidad en estos casos.

Se ha dedicado el capítulo 7 a los ejemplos numéricos, en este capítulo se pretende llegar a la conclusión de que los nuevos métodos son muy eficaces sobre un conjunto amplio de tests que se encuentran clasificados en función del tipo de problemas y las dificultades que presenten: problemas que proceden de ecuaciones en derivadas parciales; problemas donde el jacobiano de la función tiene algún valor propio positivo, pero la solución de los problemas no contiene ninguna exponencial positiva; problemas *stiff* oscilatorios o altamente oscilatorios; y también veremos qué resultados se obtienen con métodos con dos parámetros, fórmulas en paso variable y algoritmos del tipo BDF-Runge-Kutta.

Todas las demostraciones y desarrollos son originales, o al menos no tenemos conocimiento de que sea de otra forma. No sucede lo mismo con la mayoría de los tests numéricos que se presentan, ya que nos parecía conveniente probar los nuevos métodos con otros ya contrastados sobre problemas muy considerados.

Capítulo 2

Métodos EF-I-k- δ o A-I-k- δ .

2.1. Introducción.

Consideremos en el intervalo cerrado $[a,b]$ el siguiente problema de valor inicial

$$y'(x) = \lambda y(x) + f(x, y(x)), \quad y(x_0) = y_0 \quad (2.1.1)$$

donde λ es un escalar y $f(x, y(x))$ es una función que satisface los teoremas de existencia y unicidad de la solución del problema de valor inicial.

En este tipo de problemas ya se han propuesto muchas clases de métodos, siempre buscando las mejores propiedades de estabilidad y de forma que el orden de los métodos sea máximo. Un ejemplo básico es el que resulta de conocer la solución exacta del problema

$$y'(x) - \lambda y(x) = c, \quad y(0) = y_0.$$

Efectivamente, ya sabemos que

$$y(x) = e^{\lambda x} y_0 + \int_0^x e^{\lambda t} c dt,$$

con lo que rápidamente llegamos a la conclusión de que un buen método explícito de orden 1 es

$$\alpha_0 y_n = \alpha_0 e^{\lambda h} y_{n-1} + h f_{n-1},$$

siendo $\alpha_0 = -\lambda h(1 - e^{\lambda h})^{-1}$.

Este método, por tanto integra el problema (2.1.1) sin error de truncamiento cuando f es constante (este método se conoce con diferentes nombres, el más utilizado es el de método de Euler generalizado o Euler exponencial, ver [31], [32], [69], [70], [156] y [157], también se conoce como método de linealización local, ver [7], [51], [78] ó [118]). Si $\lambda \rightarrow 0$, este método se reduce al BDF clásico de orden 1.

En este capítulo, expondremos primero como obtener métodos de paso fijo con un orden mayor. Aunque trabajaremos en un primer instante únicamente con problemas escalares, luego se puede generalizar a problemas vectoriales; con el principal problema de que para calcular los pesos de los métodos será necesario calcular la exponencial de matrices, lo cual si el problema de valor inicial es de una dimensión grande supone un problema nada sencillo que se encuentra abierto. Aquí también se explicarán cuáles son los métodos más aceptados en esta cuestión.

En estos primeros métodos las propiedades de estabilidad de los métodos explícitos son más pobres cómo se podrá observar (salvo el caso del BDF adaptado de orden 1), sin embargo los métodos implícitos trabajan mucho mejor hasta orden 6.

2.2. Construcción de los algoritmos A-I-k- δ para problemas escalares.

En [149] se encuentra la teoría general para construir de una forma rápida métodos multipaso adaptados en general, y en [96] podemos ver la teoría para construir esta clase de métodos del tipo BDF que integren sin error de truncamiento los problemas

$$y'(x) - \lambda y(x) = f(x)$$

cuando $f(x)$ pertenece al espacio generado por las combinaciones lineales de las funciones $1, x, x^2, \dots, x^{r-1}$.

La idea es partir de la conocida fórmula

$$\nabla_{D-\lambda} \sum_{j=0}^r \beta_j^s \nabla^j y_n = h f_{n-s} \quad (2.2.1)$$

donde $s = 0$ en el caso implícito y $s = 1$ en el explícito, $\nabla_{D-\lambda} = Id - e^{\lambda h} E^{-1} = \nabla_P$, Id es la identidad, E es el operador siguiente: $E(f_n) = f_{n+1}$, y ∇ es el siguiente operador: $\nabla^0 f_n = f_n$, $\nabla^{j+1} f_n = \nabla^j f_n - \nabla^j f_{n-1}$.

Los coeficientes β_j se pueden obtener de imponer las distintas condiciones para que el método (2.2.1) sea exacto para las funciones $e^{\lambda x}, 1, x, x^2, \dots, x^{r-1}$.

Sin embargo en [96], ya se expone que estos coeficientes coinciden con los que se obtienen del desarrollo en series de McLaurin de diferentes funciones generatrices.

Para ello, como hemos indicado partimos de que queremos escribir los métodos en la forma que aparece en la fórmula (2.2.1), dado que

$$h f(x) = h(D - \lambda)y(x) = \nabla_P \frac{hD - \lambda h}{\nabla_P} y(x),$$

entonces se verifica que

$$hf(x - sh) = hE^{-s}f(x) = \nabla_P E^{-s} \frac{hD - \lambda h}{\nabla_P} y(x).$$

Ahora, procediendo simbólicamente escribiendo que $hD = \ln(E) = -\ln(Id - \nabla)$, se verifica

$$hf(x - sh) = \nabla_P E^{-s} \frac{hD - \lambda h}{\nabla_P} y(x) = \nabla_P (Id - \nabla)^s \frac{-\ln(Id - \nabla) - \lambda h}{\nabla_P} y(x),$$

y por tanto podemos escribir los métodos como en (2.2.1), siendo los coeficientes β_j^s aquellos que son obtenidos a partir del desarrollo de McLaurin en torno al punto $\xi = 0$ de las funciones generatrices

$$G_0(\xi, \lambda h) = \frac{-\ln(1 - \xi) - \lambda h}{1 - e^{\lambda h}(1 - \xi)}, \text{ para los métodos implícitos,} \quad (2.2.2)$$

$$G_1(\xi, \lambda h) = (1 - \xi) \frac{-\ln(1 - \xi) - \lambda h}{1 - e^{\lambda h}(1 - \xi)}, \text{ para los métodos explícitos,} \quad (2.2.3)$$

ya que $\nabla = Id - E^{-1}$ y $\nabla_P = Id - e^{\lambda h} E^{-1}$, con lo cual hacemos el cambio $\nabla_P = Id - e^{\lambda h}(Id - \nabla)$.

Entonces, si ponemos $G_0(\xi, \lambda h) = \sum_{i=0}^{\infty} \beta_i^0 \xi^i$ los coeficientes β_i^0 (del método implícito), con $i = 0, \dots, 8$ son los siguientes:

$$\begin{aligned} \beta_0^0 &= \frac{-\lambda h}{1 - e^{\lambda h}}, \\ \beta_1^0 &= \frac{1 + e^{\lambda h}(-1 + h\lambda)}{(1 - e^{\lambda h})^2}, \\ \beta_2^0 &= \frac{-1 + 4e^{\lambda h} + e^{2\lambda h}(-3 + 2\lambda h)}{2(-1 + e^{\lambda h})^3}, \\ \beta_3^0 &= \frac{2 - 9e^{\lambda h} + 18e^{2\lambda h} + e^{3\lambda h}(-11 + 6\lambda h)}{6(1 - e^{\lambda h})^4}, \\ \beta_4^0 &= \frac{-3 + 16e^{\lambda h} - 36e^{2\lambda h} + 48e^{3\lambda h} + e^{4\lambda h}(-25 + 12\lambda h)}{12(-1 + e^{\lambda h})^5}, \end{aligned}$$

$$\beta_5^0 = \frac{12 - 75e^{\lambda h} + 200e^{2\lambda h} - 300e^{3\lambda h} + 300e^{4\lambda h} + e^{5\lambda h}(-137 + 60\lambda h)}{60(-1 + e^{\lambda h})^6},$$

$$\beta_6 = \frac{-10 + 72e^{\lambda h} - 225e^{2\lambda h} + 400e^{3\lambda h} - 450e^{4\lambda h} + C_6}{60(-1 + e^{\lambda h})^7},$$

$$C_6 = 360e^{5\lambda h} + 3e^{6\lambda h}(-49 + 20\lambda h),$$

$$\beta_7 = \frac{60 - 490e^{\lambda h} + 1764e^{2\lambda h} - 3675e^{3\lambda h} + 4900e^{4\lambda h} + C_7}{420(-1 + e^{\lambda h})^8},$$

$$C_7 = -4410e^{5\lambda h} + 2940e^{6\lambda h} + 3e^{7\lambda h}(-363 + 140\lambda h),$$

$$\beta_8 = \frac{-105 + 960e^{\lambda h} - 3920e^{2\lambda h} + 9408e^{3\lambda h} - 14700e^{4\lambda h} + C_8}{840(-1 + e^{\lambda h})^9},$$

$$C_8 = 15680e^{5\lambda h} - 11760e^{6\lambda h} + 6720e^{7\lambda h} + 3e^{8\lambda h}(-761 + 280\lambda h).$$

En cuanto a los primeros coeficientes del método explícito escribiendo $G_1(\xi, \lambda h) =$

$\sum_{i=0}^{\infty} \beta_i^1 \xi^i$ son:

$$\beta_0^1 = \frac{-\lambda h}{1 - e^{\lambda h}},$$

$$\beta_1^1 = \frac{1 - e^{\lambda h} + h\lambda}{(1 - e^{\lambda h})^2},$$

$$\beta_2^1 = -\frac{-1 + e^{2\lambda h} - 2e^{\lambda h}\lambda h}{2(-1 + e^{\lambda h})^3},$$

$$\beta_3^1 = \frac{-1 + 6e^{\lambda h} - 2e^{3\lambda h} + e^{2\lambda h}(-3 + 6\lambda h)}{6(1 - e^{\lambda h})^4}.$$

Otra forma de obtener los coeficientes del método explícito es, una vez se conocen los del implícito, a partir de las funciones generatrices: como

$G_1(\xi, \lambda h) = (1 - \xi)G_0(\xi, \lambda h)$, se verifica que

$$\sum_{i=0}^{\infty} \beta_i^1 \xi^i = G_1(\xi, \lambda h) = (1 - \xi)G_0(\xi, \lambda h) = (1 - \xi) \sum_{i=0}^{\infty} \beta_i^0 \xi^i,$$

de lo que se obtiene la siguiente recurrencia

$$\beta_0^1 = \beta_0^0,$$

$$\beta_i^1 = \beta_i^0 - \beta_{i-1}^0 \text{ para } i \geq 1.$$

Ahora, si tomamos la expresión (2.2.1) y desarrollamos la parte izquierda de dicha igualdad

$$\nabla_P \sum_{j=0}^r \beta_j^s \nabla^j y_n,$$

siendo, como ya se ha dicho $\nabla_P = Id - e^{\lambda h} E^{-1}$ y considerando que

$$\nabla^j = \binom{j}{0} Id^j E^0 + \binom{j}{1} Id^{j-1} (-E^{-1})^1 + \dots + \binom{j}{j} Id^0 (-E^{-1})^j$$

obtenemos la siguiente proposición que nos da de forma explícita los coeficientes de los métodos:

Proposición 2.2.1. *Sea $\rho_r^s(z)$ el polinomio $\beta_0^s + \beta_1^s z + \beta_2^s z^2 + \dots + \beta_r^s z^r$, donde los β_i^s son los coeficientes antes hallados. Entonces el método de tipo (2.2.1) con paso constante de $r + 1$ pasos se puede escribir como*

$$\rho_r^s(1)y_n = \left(\sum_{j=1}^{r+1} (-1)^{j+1} \left\{ \frac{(\rho_r^s)^j(1)}{j!} + \frac{(\rho_r^s)^{j-1}(1)}{(j-1)!} e^{\lambda h} \right\} y_{n-j} \right) + h f_{n-s}. \quad (2.2.4)$$

Demostración: Empecemos por el caso $r = 0$, entonces

$$\nabla_P \beta_0^s y_n = \beta_0^s y_n - e^{\lambda h} \beta_0^s y_{n-1} = h f_{n-s},$$

por lo que el método resulta ser

$$\beta_0^s y_n = e^{\lambda h} \beta_0^s y_{n-1} + h f_{n-s}.$$

Como $\rho_0^s(z) = \beta_0^s$, su derivada ya es nula y $\rho_0^s(1) = \beta_0^s$, por lo que

$$\rho_0^s(1)y_n = \beta_0^s y_n = e^{\lambda h} \beta_0^s y_{n-1} + h f_{n-s} = \left\{ \frac{\rho_0^s(1)}{0!} e^{\lambda h} \right\} y_{n-1} + h f_{n-s},$$

verificándose el teorema para $r = 0$.

Para $r = 1$

$$\nabla_P(\beta_0^s y_n + \beta_1^s \nabla y_n) = \nabla_P((\beta_0^s + \beta_1^s) y_n - \beta_1^s y_{n-1}) =$$

$$= (\beta_0^s + \beta_1^s)y_n - (e^{\lambda h}(\beta_0^s + \beta_1^s) + \beta_1^s)y_{n-1} + e^{\lambda h}\beta_1^s y_{n-2} = hf_{n-s},$$

de nuevo y_n está multiplicado por $\rho_1^s(1)$, mientras que y_{n-1} está multiplicado por $-(e^{\lambda h}\rho_1^s(1) + (\rho_1^s)'(1))$, el coeficiente de y_{n-2} es $e^{\lambda h}(\rho_1^s)'(1)$.

En general ya se puede demostrar que la propiedad se verifica para todo r ,

i) Como $\nabla^j = 1 + \binom{j}{1}Id^{j-1}(-E^{-1})^1 + \dots + \binom{j}{j}Id^0(-E^{-1})^j$,

$$\nabla^j y_n = y_n - \binom{j}{1}y_{n-1} + \dots + (-1)^j y_{n-j},$$

por lo que el coeficiente de y_n es $\beta_0^s + \beta_1^s + \beta_2^s + \dots + \beta_r^s = \rho_r^s(1)$.

ii) Debido a la parte $-e^{\lambda h}E^{-1}$ del operador ∇_P , y_{n-1} queda multiplicado por $-\rho_r^s(1)e^{\lambda h}$ y si $r \geq 1$ tiene otro factor, que es el que resulta de aplicar la identidad a

$$\sum_{j=1}^r \left(\beta_j^s \binom{j}{1} (-E^{-1}) \right) y_n,$$

pero como

$$\sum_{j=1}^r (j\beta_j^s) = (\rho_r^s)'(1),$$

entonces el coeficiente de y_{n-1} es $-(\rho_r^s)'(1) + \rho_r^s(1)e^{\lambda h}$

iii) Por tanto, el coeficiente de y_{n-i} para un $i \leq r$ proviene de los términos $\binom{j}{i-1}(-E^{-1})^{i-1} + \binom{j}{i}(-E^{-1})^i$ de cada ∇^j para $j \geq i$.

Al aplicar la identidad sobre

$$\sum_{j=i}^r \left(\beta_j^s \binom{j}{i} (-E^{-1})^i \right) y_n = (-1)^i \sum_{j=i}^r \left(\beta_j^s \binom{j}{i} \right) y_{n-i} = (-1)^i \frac{(\rho_r^s)^i(1)}{i!} y_{n-i}.$$

Mientras que al aplicar $-e^{\lambda h}E^{-1}$ sobre

$$\begin{aligned} \sum_{j=i-1}^r \left(\beta_j^s \binom{j}{i-1} (-E^{-1})^{i-1} \right) y_n = \\ (-1)^{i-1} \frac{(\rho_r^s)^{i-1}(1)}{(i-1)!} y_{n-i+1} \end{aligned}$$

nos queda la otra parte del coeficiente del término en y_{n-i} .

iv) Por último el coeficiente de y_{n-r-1} se obtiene de aplicar $-e^{\lambda h} E^{-1}$ a

$$\left(\beta_r^s \binom{r}{r} (-E^{-1})^r \right) y_n = (-1)^r \frac{(\rho_r^s)^r (1)}{r!} y_{n-r},$$

concluyendo pues $(\rho_r^s)^{r+1} (1) = 0$. □

2.3. Extensión de los BDF adaptados a problemas vectoriales.

Sin embargo, mucho más interesante que los problemas escalares son los vectoriales: consideramos en el intervalo cerrado $[a,b]$ el problema de valor inicial (1.0.3), donde A es una matriz $m \times m$ constante y $F(x, y(x)) = (f_1(x, y), \dots, f_m(x, y))$ es un vector de dimensión m , cuyas componentes satisfacen los teoremas de existencia y unicidad de la solución del problema de valor inicial.

En esas condiciones vemos que podemos utilizar la misma técnica que en el caso escalar, sin más que cambiar en las funciones generatrices, primero, y luego en los coeficientes, λ por A y 1 por la correspondiente matriz identidad de dimensión $m \times m$ y considerando que $\frac{B}{C} = BC^{-1}$.

De tal forma que nuestro método se escribirá como

$$hf_{n-s} = \nabla_{D-A} \sum_{j=0}^r \beta_j^s \nabla^j y_n. \quad (2.3.1)$$

Antes de continuar es preciso recordar que ahora los coeficientes van a depender del cálculo de exponenciales de matrices.

Este cálculo no es un tema ni mucho menos banal. Al contrario, es un campo de investigación abierto, donde ahora mismo se sigue trabajando y donde se sigue publicando.

Grandes resultados se han venido dando en los últimos años, y no es fácil quedarse con un único trabajo. No obstante, como referencia proponemos acudir a las páginas web <http://na.uni-tuebingen.de/na/software.shtml> (desde donde se puede descargar el programa exp4, el código está tanto en MATLAB como en C, ver también [70]) o <http://www.maths.uq.edu.au/expokit/> (donde está disponible el programa expokit cuyo código está escrito en FORTRAN y MATLAB, ver también [132]).

Aunque muchas vías se han utilizado para calcular la exponencial de la matriz, en los últimos años las publicaciones aparecidas se centran en ideas basadas en subespacios de Krylov y el método de Arnoldi. La bibliografía sobre este tema es cada vez más extensa, una muestra de estas publicaciones podría ser [13], [26], [27], [28], [43], [55], [56], [69], [83], [97], [99], [103], [114] ó [126].

2.4. Construcción de algoritmos EF-I-k- δ para integrar problemas vectoriales.

Consideramos en el intervalo cerrado $[a,b]$ el siguiente problema de valor inicial

$$y'(x) = Ay + F(x, y(x)) = G(x, y(x)), \quad y(t_0) = y_0, \quad (2.4.1)$$

donde A es una matriz $m \times m$ constante y $G(x, y(x)) = (g_1(x, y), \dots, g_m(x, y))$ es un vector de dimensión m , cuyas componentes satisfacen los teoremas de

existencia y unicidad de la solución del problema de valor inicial.

Una vez construidos los algoritmos adaptados para integrar problemas vectoriales del tipo de la ecuación (1.0.3), podemos construir los algoritmos *exponential fitting* que resuelvan (2.4.1) cuando $y(x)$ son vectores cuyas componentes pertenecen a combinaciones lineales del espacio generado por $\langle e^{\lambda_i x}, 1, x, \dots, x^r \rangle$.

Supongamos que tenemos un algoritmo implícito de tipo *exponential fitting* (EFM)

$$c_0 y_n + c_1 y_{n-1} + \dots + c_m y_{n-m} = h G_n, \quad (2.4.2)$$

de m pasos y que denotaremos como EF-I-k-m. El algoritmo implícito de tipo adaptado (AM) que integra de forma exacta el mismo espacio de funciones y que denotaremos como A-I-k-m se escribe como

$$c_0^* y_n + c_1 y_{n-1} + \dots + c_m y_{n-m} = h F_n, \quad (2.4.3)$$

donde las únicas diferencias son que $c_0 - c_0^* = Ah$, y que $G_n = G(x_n, y_n) = F(x_n, y_n) + Ay_n = F_n + Ay_n$. Es decir, el algoritmo *exponential fitting* y el adaptado son básicamente el mismo, aunque los coeficientes cambien y por tanto las regiones de 0-estabilidad y de estabilidad absoluta serán diferentes. La explicación a esta aparente contradicción se dará más adelante.

De la misma forma si el algoritmo explícito de tipo EFM se escribe como

$$c_0 y_n + c_1 y_{n-1} + \dots + c_m y_{n-m} = h G_{n-1}, \quad (2.4.4)$$

el algoritmo explícito de tipo AM que integra de forma exacta el mismo espacio de funciones se escribe de la siguiente manera

$$c_0 y_n + c_1^* y_{n-1} + \dots + c_m y_{n-m} = h F_{n-1}, \quad (2.4.5)$$

donde esta vez el único coeficiente que varía es ahora el segundo: $c_1 - c_1^* = Ah$. De nuevo las regiones de estabilidad cambiarán, por lo que será preciso estudiar tanto los EFM como los AM, aunque ya de antemano sabemos que los resultados numéricos al integrar problemas vectoriales del tipo de la ecuación (1.0.3) con un algoritmo AM que integra de forma exacta el problema cuando la solución es un vector cuyas componentes pertenecen al espacio generado por $\langle e^{\lambda_i x}, 1, x, \dots, x^r \rangle$, van a ser los mismos que al integrar los mismos problemas vectoriales, pero esta vez integrando la ecuación escrita de la forma que aparece en (2.4.1) con un algoritmo EFM que integre el mismo espacio generado por $\langle e^{\lambda_i x}, 1, x, \dots, x^r \rangle$.

2.5. Convergencia.

Como ya se ha comentado la convergencia equivale a consistencia y estabilidad. Para la primera estudiaremos el error de truncado local, mientras que para la segunda propiedad no nos quedaremos únicamente con la 0-estabilidad, también nos extenderemos en la estabilidad absoluta, si bien en esta sección sólo enunciaremos los resultados que se desarrollarán en los apéndices A y B.

Para la consistencia de los nuevos métodos nos guiaremos de los siguientes teoremas.

Teorema 2.5.1. *El método implícito*

$$\nabla_{D-A} \sum_{j=0}^{r-1} \beta_j^0 \nabla^j y_n = h f_n \quad (2.5.1)$$

de r pasos es convergente de orden r . Su error de truncado local se puede expresar como:

$$-h^{r+1}\beta_r^0(D - A)D^r y(x) + O(h^{r+2}).$$

Demostración: Sea $P(x) = x - Ah$, considerando que $hD = \ln(E)$, se verifica

$$hf(x, y(x)) = P(hD)y(x) = P(\ln(E)) \frac{(Id - \nabla)^{-1} \nabla_P}{L(Id/(Id - \nabla))} y(x),$$

donde $L(E) = (E - e^{Ah})$.

Entonces, obtenemos que

$$\nabla_P \frac{P(-\ln(Id - \nabla))}{(Id - \nabla)L(Id/(Id - \nabla))} y(x) = hf(x, y(x)), \quad (2.5.2)$$

y consecuentemente el método implícito es como en (2.5.1) donde los coeficientes β_j^0 se obtienen como ya se ha dicho del desarrollo en series en un entorno de $\xi = 0$ de la función generatriz (2.2.2) cambiando λh por Ah , 1 por la matriz identidad y multiplicando por la inversa por la derecha en vez de dividir.

A partir de aquí podemos obtener el error de truncado local de forma similar a como se procedió en [149] (Teorema 2.1).

Por tanto la expresión que nos da el error de truncado local del método implícito (2.5.1) de r pasos sale de restar (2.5.1) menos (2.5.2) y es

$$-\nabla_P \beta_r^0 \nabla^r y(x) + O(h^{r+2}), \quad (2.5.3)$$

pero ahora considerando que las diferencias satisfacen que $\nabla^r y(x) = h^r D^r y(x) + O(h^{r+1})$, y $\nabla_P y(x) = h(D - A)y(x) + O(h^2)$, entonces el error de truncado local de (2.5.1) se escribe como

$$-h^{r+1}\beta_r^0(D - A)D^r y(x) + O(h^{r+2}). \quad \square$$

Teorema 2.5.2. *El método explícito*

$$\nabla_{D-A} \sum_{j=0}^{r-1} \beta_j^1 \nabla^j y_n = h f_{n-1}$$

de r pasos es convergente de orden r . Su error de truncado local puede expresarse como:

$$-h^{r+1} \beta_k^1 (D - A) D^r y(x) + O(h^{r+2}).$$

Demostración: Sólo debemos recordar que

$$y(x) = E^{-1} y(x+h) = (Id - \nabla) y(x+h),$$

con lo cual

$$\nabla_P \frac{P(-\ln(Id - \nabla))}{L(Id/(Id - \nabla))} y(x+h) = h f(x, y(x)),$$

y se procede de forma similar a como hemos hecho en el teorema 2.5.1. \square

Es bastante común pensar que los métodos adaptados o exponential fitting como los aquí propuestos son 0-estables; Hochbruck, Lubich y Selhofer llegan a decir (ver [70], p.6) “When the method is exact for linear differential equations, it is trivially A-stable”. Sin embargo, hemos comprobado (ver [151], por ejemplo) que en general no es cierto entre los BDF, como vamos a demostrar a continuación. Sí veremos que entre los implícitos de orden hasta 6 será bastante sencillo elegir un tamaño de paso para el cuál el método sea estable.

Para comprobar la 0-estabilidad, consideraremos el sistema lineal

$$y'(x) = Ay(x) \tag{2.5.4}$$

donde los valores propios $\lambda_i, i = 1, \dots, m$ de la matriz A de dimensión $m \times m$ satisfacen que $Re(\lambda_i) \leq 0$.

Es fácil observar (ver [87]) que las soluciones del método numérico (2.3.1) cuando se aplican a la ecuación (2.5.4) han de satisfacer que

$$\|y(x)\| \rightarrow 0 \text{ cuando } x \rightarrow \infty.$$

Por tanto, si los valores propios de la matriz A son todos distintos entre sí, sólo tendremos que comprobar en que condiciones, para cada valor propio λ las raíces r_k del polinomio característico asociado al método

$$p(r) = \sum_{i=0}^m \alpha_i r^{m-i},$$

se verifica que $|r_k| < 1$, y en aquellas raíces que tengan módulo 1 ver cuándo son simples.

El apéndice A dedicado a la 0-estabilidad de los métodos de orden exponencial 1 se extiende en la demostración empírica de las propiedades de estos nuevos métodos. Las principales conclusiones son que los métodos explícitos no serán 0-estables (salvo el de 1 paso), mientras que los implícitos hasta orden 6 muestran regiones de 0-estabilidad mucho mejores (incluso en casos concretos hasta orden 7).

Otra importante conclusión es que a pesar de que en realidad son el mismo método, sin embargo, los algoritmos EF-I-k- δ y los A-I-k- δ no muestran las mismas regiones de 0-estabilidad, la razón se puede observar en el apéndice B dedicado a la estabilidad absoluta de los métodos de orden exponencial 1.

Por supuesto hay una gran relación entre 0-estabilidad y estabilidad absoluta, pero las definiciones clásicas de estabilidad absoluta y A-estabilidad que aparecen en multitud de libros y artículos (ver, por ejemplo [64] ó [87]) se diseñaron para métodos lineales multipaso con coeficientes constantes, aquí se van a considerar otras definiciones más apropiadas (ver [29] ó [151]) para el

caso de los métodos multipaso adaptados o de tipo *exponential fitting*. Ya que, si consideramos el problema (2.5.4), donde la matriz A es la que aparece en el método, está claro que estos métodos nuevos serían estables absolutamente (con las definiciones clásicas) siempre que fueran 0-estables.

Por tanto para los algoritmos adaptados vamos a tratar el problema

$$y'(x) - Ay(x) = By(x),$$

siendo Ah el parámetro introducido en el algoritmo adaptado. Esto nos permite comprender el tipo de error que podemos cometer al elegir el parámetro del método.

Es fácil comprender que, en realidad basta con estudiar el caso escalar, ya que el caso vectorial haciendo un cambio de variable se va a reducir a varios problemas escalares del tipo

$$y'(x) - \lambda y(x) = \nu y(x)$$

donde $Re(\lambda + \nu) < 0$.

De la misma forma podemos estudiar lo que sucede con el correspondiente algoritmo *exponential fitting* para el problema escalar del tipo

$$y'(x) = \lambda y(x) + \nu y(x)$$

donde $Re(\lambda + \nu) < 0$, siendo λh el parámetro introducido en el algoritmo.

Como se puede deducir de lo dicho anteriormente, hemos introducido λh en el algoritmo, sin embargo, la verdadera solución depende de la exponencial de $\lambda + \nu$.

Para decirlo de otra forma, la pregunta que nos hacemos es hasta qué punto nos podemos equivocar al introducir el parámetro en el método. No es una

pregunta banal, ya que, en muchos problemas en condiciones normales no conocemos la solución del mismo, y por tanto no sabemos cuál es el parámetro ideal para el método.

Los problemas más interesantes son los PVI (problemas de valor inicial) no lineales donde el parámetro ideal para el método puede ir variando a lo largo del tiempo. A nosotros nos puede interesar, sin embargo, que cada varias iteraciones sólo tengamos que calcular una vez el parámetro y por tanto los coeficientes del método, por tanto será necesario un buen estudio de la estabilidad absoluta de los métodos.

Sin embargo, sí podemos considerar que ν va a ser un valor pequeño en comparación con λ , es decir, ν el error que se cometa al calcular el parámetro deseable para el método, podemos esperar que sea cercano a cero.

Por tanto, si aplicamos el método (ó algoritmo)

$$c_0 y_n + c_1 y_{n-1} + \dots + c_m y_{n-m} = h \text{func}_{n-s}, \quad (2.5.5)$$

donde $\text{func}(t) = \mu y(t)$ será la función utilizada en cada método (ó algoritmo), obtenemos el siguiente polinomio de estabilidad

$$\sum_{i=0}^m c_i (\lambda h) r^{m-i} = \mu h r^{m-s}, \quad (2.5.6)$$

por tanto nuestras definiciones de estabilidad absoluta quedarán como siguen:

Definición 2.5.1. Diremos que el método (ó algoritmo) lineal (2.5.5) es estable absolutamente para el par dado $u = (u_1 = \lambda h, u_2 = \mu h)$, si para dicho u todas las raíces del polinomio de estabilidad (2.5.6) satisfacen $|r_k| \leq 1$ (y las de módulo 1 son simples), y será inestable absolutamente en caso contrario.

Definición 2.5.2. Diremos que el método (ó algoritmo) multipaso (2.5.5) tiene una región \mathfrak{R}_A de estabilidad absoluta, donde \mathfrak{R}_A es una región del espacio complejo $C \times C$, si es estable absolutamente para todo $u \in \mathfrak{R}_A$.

Obsérvese que en los algoritmos adaptados $\mu = \nu$, mientras que en los *exponential fitting* $\mu = \nu + \lambda$.

Por supuesto, a partir de la definición dada es elemental que vamos a tener una dificultad añadida que con las definiciones clásicas de estabilidad absoluta. Dado que las regiones de estabilidad absoluta \mathfrak{R}_A son regiones de $C \times C$, es obvio que no vamos a poder dibujar las regiones de estabilidad absoluta.

Para solucionar este problema, vamos a fijar la primera componente de u , esto es, vamos a considerar ya establecido el valor $u_1 = \lambda h$ que introducimos en el método, donde u_1 será un valor para el cual el método es 0-estable y dibujaremos en el plano complejo los valores de la segunda componente $u_2 = \mu h$ para los cuales el método es absolutamente estable.

Las principales conclusiones que se observan en el apéndice B son que hay una gran relación entre el valor u_1 que usamos en el método y el tamaño de la región de inestabilidad absoluta, si u_1 está cerca de la zona de 0-inestabilidad, entonces la región de inestabilidad puede llegar a ser muy grande.

También se puede observar que las regiones de estabilidad absoluta de los algoritmos EF-I-k- δ y las de los A-I-k- δ sí que están íntimamente relacionadas, de hecho son las mismas salvo una traslación de λh .

Otro resultado que se obtiene es el que reflejamos en el siguiente teorema:

Teorema 2.5.3. *Sea $\lambda h \in R^-$ la región de estabilidad absoluta del método es simétrica respecto del eje de abcisas.*

Demostración: tomemos un $\lambda h \in R^-$ genérico y un método de orden exponencial 1 cualquiera con su polinomio característico asociado a la estabilidad absoluta $P(\xi, \lambda h)$ y se resuelve el polinomio en la variable μh . Haciendo el cambio $\xi = \cos(t) + i \sin(t)$ se logra la frontera de la región de estabilidad tal como se explica por ejemplo en [87]. Y para estos métodos se obtiene que los valores de μh para t y $-t$ son complejos conjugados. Basta sumar ambos valores y se obtienen expresiones que dependen únicamente de constantes reales, $e^{\lambda h}$ y λh . □

De las figuras del apéndice B también se observa que dado un $\lambda h \in R^-$, cuanto mayor sea este número en valor absoluto, más pequeña es la región de estabilidad absoluta.

Todos estos resultados que aparecen más desarrollados en los apéndices A y B son claves para entender el comportamiento de los métodos de este capítulo al aplicarlos sobre los distintos ejemplos numéricos que se mostrarán en otros capítulos.

Capítulo 3

Métodos A-I-r- γ o EF-I-r- γ .

3.1. Introducción.

De nuevo vamos a considerar en el intervalo cerrado $[a,b]$ el siguiente problema de valor inicial

$$y'(x) = \lambda y(x) + f(x, y(x)) = g(x, y(x)), \quad y(t_0) = y_0 \quad (3.1.1)$$

donde λ es un escalar y $g(x, y(x))$ es una función que satisface los teoremas de existencia y unicidad de la solución del problema de valor inicial.

De la misma forma que en el capítulo anterior hemos construido métodos que integran dicho problema de forma exacta cuando $y(x)$ pertenece al subespacio generado por $\langle 1, \dots, x^{k-1}, e^{\lambda x} \rangle$, ahora vamos a construir métodos numéricos que integren el problema de valor inicial de forma exacta cuando $y(x)$ pertenezca al subespacio generado por $\langle 1, e^{\lambda x}, \dots, x^{k-1}e^{\lambda x} \rangle$.

De nuevo, los problemas que nos van a interesar son sobre todo los problemas vectoriales, por lo que después haremos la extensión de dichos métodos para que integren problemas vectoriales. Si bien a la hora de estudiar la estabilidad de los métodos, como ya se ha visto, nos centraremos en el problema escalar.

En estos nuevos métodos veremos que las propiedades de estabilidad de los métodos explícitos son bastante mejores que las de los métodos del capítulo anterior, destacando el método explícito de 2 pasos (ver teorema C.0.6), mientras que los métodos implícitos vuelven a trabajar muy bien hasta orden 6 y conservan buenas propiedades de estabilidad incluso para un número de pasos muy elevado.

Por tanto el esquema de este capítulo, es similar al del anterior.

3.2. Construcción de algoritmos EF-I-r- γ para problemas escalares.

La idea, ahora, es partir de la nueva fórmula

$$\nabla \sum_{j=0}^k \overline{\beta}_j^s \nabla_{D-\lambda}^j y_n = h g_{n-s} \quad (3.2.1)$$

siendo como en el capítulo anterior $s = 0$ en el caso implícito y $s = 1$ en el explícito, $\nabla_{D-\lambda} = \nabla_P, Id, E$ y ∇ los operadores descritos en el capítulo anterior.

Los coeficientes $\overline{\beta}_j^s$ se pueden obtener de imponer las distintas condiciones para que el método (3.2.1) sea exacto para las funciones $1, e^{\lambda x}, \dots, x^{k-1} e^{\lambda x}$.

La otra posibilidad es tratar de hallar unas funciones generatrices que nos permitan calcular los coeficientes que nos den el método.

Dado que

$$hg(x) = hDy(x) = \nabla \frac{hD}{\nabla} y(x),$$

entonces se verifica que

$$hg(x - sh) = hE^{-s}g(x) = \nabla E^{-s} \frac{hD}{\nabla} y(x).$$

Ahora, procediendo simbólicamente puesto que $hD = \ln(E) = -\ln(Id - \nabla)$, y que $\nabla = Id - E^{-1}$ y $\nabla_P = Id - e^{\lambda h} E^{-1}$, con lo cual $\nabla = \nabla_P e^{-\lambda h} + Id - e^{-\lambda h}$, se verifica

$$hg(x - sh) = \nabla(e^{-\lambda h} - e^{-\lambda h} \nabla_P)^s \frac{-\ln(e^{-\lambda h} - \nabla_P e^{-\lambda h})}{\nabla_P e^{-\lambda h} + Id - e^{-\lambda h}} y(x),$$

y por tanto podemos escribir los métodos como en la fórmula (3.2.1) siendo los coeficientes $\overline{\beta}_j^s$ aquellos que son obtenidos a partir del desarrollo de McLaurin en torno al punto $\xi = 0$ de las funciones generatrices

$$\overline{G}_0(\xi, \lambda h) = \frac{-\ln(1 - \xi) - \lambda h}{\xi e^{-\lambda h} + 1 - e^{-\lambda h}}, \text{ para los métodos implícitos,} \quad (3.2.2)$$

$$\overline{G}_1(\xi, \lambda h) = (e^{-\lambda h} - e^{-\lambda h} \xi) \frac{-\ln(1 - \xi) - \lambda h}{\xi e^{-\lambda h} + 1 - e^{-\lambda h}}, \text{ métodos explícitos.} \quad (3.2.3)$$

Esto es, siendo $\overline{G}_0(\xi, \lambda h) = \sum_{i=0}^{\infty} \overline{\beta}_i^0 \xi^i$ una representación de los coeficientes $\overline{\beta}_i^0$ (del método implícito, vamos a dar los nueve primeros) son los siguientes:

$$\begin{aligned} \overline{\beta}_0^0 &= \frac{\lambda h}{1 - e^{-\lambda h}}, \\ \overline{\beta}_1^0 &= \frac{e^{\lambda h}(-1 + e^{\lambda h} - \lambda h)}{(1 - e^{\lambda h})^2}, \\ \overline{\beta}_2^0 &= \frac{e^{\lambda h}(3 - 4e^{\lambda h} + e^{2\lambda h} + 2\lambda h)}{2(-1 + e^{\lambda h})^3}, \\ \overline{\beta}_3^0 &= \frac{e^{\lambda h}(-11 + 18e^{\lambda h} - 9e^{2\lambda h} + 2e^{3\lambda h} - 6\lambda h)}{6(1 - e^{\lambda h})^4}, \\ \overline{\beta}_4^0 &= \frac{e^{\lambda h}(25 - 48e^{\lambda h} + 36e^{2\lambda h} - 16e^{3\lambda h} + 3e^{4\lambda h} + 12\lambda h)}{12(-1 + e^{\lambda h})^5}, \\ \overline{\beta}_5^0 &= \frac{e^{\lambda h}(-137 + 300e^{\lambda h} - 300e^{2\lambda h} + 200e^{3\lambda h} - 75e^{4\lambda h} + 12e^{5\lambda h} - 60\lambda h)}{60(-1 + e^{\lambda h})^6}, \\ \overline{\beta}_6^0 &= \frac{e^{\lambda h}(147 - 360e^{\lambda h} + 450e^{2\lambda h} - 400e^{3\lambda h} + \overline{C}_6)}{60(-1 + e^{\lambda h})^7}, \\ \overline{C}_6 &= 225e^{4\lambda h} - 72e^{5\lambda h} + 10e^{6\lambda h} + 60\lambda h, \end{aligned}$$

$$\begin{aligned}\overline{\beta}_7^0 &= \frac{e^{\lambda h}(-1089 + 2940e^{\lambda h} - 4410e^{2\lambda h} + 4900e^{3\lambda h} + \overline{C}_7)}{420(-1 + e^{\lambda h})^8}, \\ \overline{C}_7 &= -3675e^{4\lambda h} + 1764e^{5\lambda h} - 490e^{6\lambda h} + 60e^{7\lambda h} - 420\lambda h, \\ \overline{\beta}_8^0 &= \frac{e^{\lambda h}(2283 - 6720e^{\lambda h} + 11760e^{2\lambda h} - 15680e^{3\lambda h} + 14700e^{4\lambda h} + \overline{C}_8)}{840(-1 + e^{\lambda h})^9}, \\ \overline{C}_8 &= -9408e^{5\lambda h} + 3920e^{6\lambda h} - 960e^{7\lambda h} + 105e^{8\lambda h} + 840\lambda h, \\ \overline{\beta}_9^0 &= \frac{e^{\lambda h}(-7129 + 22680e^{\lambda h} - 45360e^{2\lambda h} + 70560e^{3\lambda h} - 79380e^{4\lambda h} + \overline{C}_9)}{2520(-1 + e^{\lambda h})^{10}}, \\ \overline{C}_9 &= 63504e^{5\lambda h} - 35280e^{6\lambda h} + 12960e^{7\lambda h} - 2835e^{8\lambda h} + 280e^{9\lambda h} - 2520\lambda h.\end{aligned}$$

Mientras que si escribimos $\overline{G}_1(\xi, \lambda h) = \sum_{i=0}^{\infty} \overline{\beta}_i^1 \xi^i$ una muestra de los tres primeros coeficientes de los métodos explícitos es

$$\begin{aligned}\overline{\beta}_0^1 &= \frac{-\lambda h}{1 - e^{\lambda h}}, \\ \overline{\beta}_1^1 &= \frac{-1 + e^{\lambda h} - h e^{\lambda h} \lambda}{(1 - e^{\lambda h})^2}, \\ \overline{\beta}_2^1 &= -\frac{-1 + e^{2\lambda h} - 2e^{\lambda h} \lambda h}{2(-1 + e^{\lambda h})^3}, \\ \overline{\beta}_3^1 &= -\frac{2 + 3e^{\lambda h} - 6e^{2\lambda h} + e^{3\lambda h} + 6\lambda e^{\lambda h} h}{6(1 - e^{\lambda h})^4}.\end{aligned}$$

Otra posibilidad es obtener los coeficientes de los métodos explícitos utilizando que conocemos los coeficientes de los implícitos y la relación que hay entre unos y otros (de manera similar a como hicimos en el capítulo anterior).

Ahora si tomamos la expresión (3.2.1) y desarrollamos la parte de la izquierda de dicha igualdad

$$\nabla \sum_{j=0}^k \overline{\beta}_j^s \nabla_P^j y_n,$$

siendo, como ya se ha dicho $\nabla_P = Id - E^{-1}e^{\lambda h}$ y considerando que

$$\begin{aligned}\nabla_P^j &= \binom{j}{0} Id^j (E^{-1}e^{\lambda h})^0 + \binom{j}{1} Id^{j-1} (-E^{-1}e^{\lambda h})^1 + \dots + \\ &+ \binom{j}{j-1} Id^1 (-E^{-1}e^{\lambda h})^{j-1} + \binom{j}{j} Id^0 (-E^{-1}e^{\lambda h})^j,\end{aligned}\quad (3.2.4)$$

obtenemos la siguiente proposición que nos da de forma explícita los coeficientes de los métodos:

Proposición 3.2.1. Sea $\overline{\rho_k^s(z)}$ el polinomio $\overline{\beta_0^s} + \overline{\beta_1^s}z + \overline{\beta_2^s}z^2 + \dots + \overline{\beta_k^s}z^k$, donde los $\overline{\beta_i^s}$ son los coeficientes antes hallados. Entonces el método de tipo (3.2.1) con paso constante de $k + 1$ pasos se puede escribir como

$$\overline{\rho_k^s(1)}y_n = \left(\sum_{j=1}^{k+1} (-1)^{j+1} \left\{ \frac{(\overline{\rho_k^s})^{(j)}(1)}{j!} e^{j\lambda h} + \frac{(\overline{\rho_k^s})^{(j-1)}(1)}{(j-1)!} e^{(j-1)\lambda h} \right\} y_{n-j} \right) + hg_{n-s} \quad (3.2.5)$$

Demostración: Aunque se puede demostrar directamente el caso general, igual que en el capítulo anterior comenzaremos con los casos $k = 0, 1$, con el fin de que el lector pueda familiarizarse con la notación empleada.

Si $k = 0$, entonces

$$\nabla \overline{\beta_0^s} y_n = \overline{\beta_0^s} y_n - \overline{\beta_0^s} y_{n-1} = hg_{n-s},$$

por lo que el método resulta ser

$$\overline{\beta_0^s} y_n = \overline{\beta_0^s} y_{n-1} + hg_{n-s}.$$

Como $\overline{\rho_0^s(z)} = \overline{\beta_0^s}$, su derivada ya es nula y $\overline{\rho_0^s(1)} = \overline{\beta_0^s}$, por lo que

$$\overline{\rho_0^s(1)}y_n = \overline{\beta_0^s}y_n = \overline{\beta_0^s}y_{n-1} + hg_{n-s} = \left\{ \frac{\overline{\rho_0^s(1)}}{0!} e^{0\lambda h} \right\} y_{n-1} + hg_{n-s},$$

verificándose el teorema para $k = 0$.

Para $k = 1$

$$\begin{aligned} \nabla(\overline{\beta_0^s}y_n + \overline{\beta_1^s}\nabla_P y_n) &= \nabla((\overline{\beta_0^s} + \overline{\beta_1^s})y_n - \overline{\beta_1^s}e^{\lambda h}y_{n-1}) = \\ &= (\overline{\beta_0^s} + \overline{\beta_1^s})y_n - ((\overline{\beta_0^s} + \overline{\beta_1^s}) + e^{\lambda h}\overline{\beta_1^s})y_{n-1} + e^{\lambda h}\overline{\beta_1^s}y_{n-2} = hf_{n-s}, \end{aligned}$$

por tanto el coeficiente de y_n es $\overline{\rho_1^s(1)}$, mientras que y_{n-1} está multiplicado por $-(\overline{\rho_1^s(1)} + e^{\lambda h}\overline{(\rho_1^s)'(1)})$ y el coeficiente de y_{n-2} es $e^{\lambda h}\overline{(\rho_1^s)'(1)}$.

Ahora demostraremos que es cierto para todo k ,

i) De (3.2.4) obtenemos que

$$\nabla_P^j y_n = y_n - e^{\lambda h} y_{n-1} + \dots + (-1)^{j-1} \binom{j}{j-1} e^{(j-1)\lambda h} y_{n-j+1} + (-1)^j e^{j\lambda h} y_{n-j},$$

por lo que el coeficiente de y_n es $\overline{\beta_0^s} + \overline{\beta_1^s} + \overline{\beta_2^s} + \dots + \overline{\beta_k^s} = \overline{\rho_k^s(1)}$.

ii) Si $k \geq 1$, además de estar multiplicado por $-\overline{\rho_k^s(1)}$, y_{n-1} tiene otro factor, que es el que resulta de aplicar la identidad a

$$\sum_{j=1}^k \left(\overline{\beta_j^s} \binom{j}{1} (-E^{-1} e^{\lambda h}) \right) y_n,$$

y como

$$\sum_{j=1}^k (j \overline{\beta_j^s}) = \overline{(\rho_k^s)'(1)},$$

entonces el coeficiente de y_{n-1} es $-(e^{\lambda h} \overline{(\rho_k^s)'(1)} + \overline{\rho_k^s(1)})$

iii) El coeficiente de y_{n-i} para un $i \leq k$ proviene de los términos $\binom{j}{i-1} (-E^{-1} e^{\lambda h})^{i-1} + \binom{j}{i} (-E^{-1} e^{\lambda h})^i$ de cada ∇_P^j para $j \geq i$.

Al aplicar la identidad sobre

$$\sum_{j=i}^k \left(\overline{\beta_j^s} \binom{j}{i} (-E^{-1} e^{\lambda h})^i \right) y_n = (-1)^i e^{i\lambda h} \sum_{j=i}^k \left(\overline{\beta_j^s} \binom{j}{i} \right) y_{n-i} = (-1)^i e^{i\lambda h} \frac{\overline{(\rho_k^s)^{(i)}(1)}}{i!} y_{n-i}.$$

Mientras que al aplicar $-E^{-1}$ sobre

$$\sum_{j=i-1}^k \left(\overline{\beta_j^s} \binom{j}{i-1} (-E^{-1} e^{\lambda h})^{i-1} \right) y_n = (-1)^{i-1} e^{(i-1)\lambda h} \frac{\overline{(\rho_k^s)^{(i-1)}(1)}}{(i-1)!} y_{n-i+1}$$

obtenemos la otra parte del coeficiente del término en y_{n-i} .

iv) Por último el coeficiente de y_{n-k-1} se obtiene de aplicar $-E^{-1}$ sobre

$$\left(\overline{\beta_k^s} \binom{k}{k} (-E^{-1} e^{\lambda h})^k \right) y_n = (-1)^k e^{\lambda k h} \frac{\overline{(\rho_k^s)^k(1)}}{k!} y_{n-k},$$

concluyendo, ya que $\overline{(\rho_k^s)^{k+1}(1)} = 0$. \square

3.3. Extensión de los EF-I-r- γ a problemas vectoriales.

Sin embargo, como ya hemos explicado en el capítulo anterior, debido a que los problemas de valor inicial vectoriales son mucho más abundantes, vamos a realizar la extensión a este tipo de problema, para ello consideramos en el intervalo cerrado $[a,b]$ el siguiente problema de valor inicial

$$y'(x) = Ay(x) + F(x, y(x)) = G(x, y(x)), \quad y(t_0) = y_0 \quad (3.3.1)$$

donde A es una matriz $m \times m$ y $G(x, y(x)) = (g_1(x, y), \dots, g_m(x, y))$ es un vector de dimensión m , cuyas componentes satisfacen los teoremas de existencia y unicidad de la solución del problema de valor inicial.

En esas condiciones vemos que podemos utilizar la misma técnica que en el caso escalar, sin más que cambiar en las funciones generatrices, primero, y luego en los coeficientes λ por A y 1 por la correspondiente matriz identidad de dimensión $m \times m$.

De tal forma que nuestro método se escribirá como

$$hg_{n-s} = \nabla \sum_{j=0}^k \overline{\beta}_j^s \nabla_{D-A}^j y_n, \quad (3.3.2)$$

donde, como ya sucedió en el capítulo anterior, los $\overline{\beta}_j^s$ son los coeficientes hallados en la sección anterior, pero cambiando 1 por la matriz identidad de dimensión $m \times m$, λ por la matriz A y considerando que $\frac{B}{C} = BC^{-1}$.

De nuevo los coeficientes van a depender del cálculo de exponenciales de matrices, que, como ya se ha explicado, puede ser un cálculo laborioso si la matriz tiene una dimensión elevada.

3.4. Construcción de los A-I-r- γ para integrar problemas vectoriales.

De nuevo consideramos en el intervalo cerrado $[a, b]$ el problema de valor inicial (2.4.1) donde A ahora es una matriz $m \times m$ y $F(x, y(x)) = (f_1(x, y), \dots, f_m(x, y))$ es un vector de dimensión m , cuyas componentes satisfacen los teoremas de existencia y unicidad de la solución del problema de valor inicial.

Como ya se explicó en el capítulo anterior es fácil pasar de un algoritmo de tipo *exponential fitting* a un algoritmo adaptado, sin más que cambiar un coeficiente en el algoritmo.

Si el algoritmo implícito de tipo *exponential fitting* se escribe como en (2.4.2) entonces el algoritmo implícito de tipo adaptado que integra de forma exacta el mismo espacio de funciones se escribe como en (2.4.3) donde las únicas diferencias son que $c_0 - c_0^* = Ah$, y que $G_n = G(x_n, y_n) = F(x_n, y_n) + Ay_n = F_n + Ay_n$.

Y de la misma forma si el algoritmo explícito de tipo EFM se escribe como (2.4.4), el algoritmo explícito de tipo AM que integra de forma exacta el mismo espacio de funciones se escribe como en (2.4.5) donde esta vez el único coeficiente que varía es ahora el segundo: $c_1 - c_1^* = Ah$.

3.5. Convergencia.

Al igual que en el capítulo anterior aquí vamos a ver por una parte el error de truncado local de los diferentes métodos expuestos y posteriormente se citarán las propiedades de estabilidad que pueden resultar más interesantes

para saber como funcionarán los métodos, pero las propiedades se estudiarán con más detalle en los apéndices C y D.

Teorema 3.5.1. *El método implícito*

$$\nabla \sum_{j=0}^{k-1} \overline{\beta}_j^0 \nabla_{D-A}^j y_n = h g_n, \quad (3.5.1)$$

de k pasos es convergente de orden k . Su error de truncado local se puede expresar como :

$$-h^{k+1} \overline{\beta}_k^0 (D - A)^k D y(x) + O(h^{k+2}).$$

Demostración: De nuevo procediendo de forma similar a como se demostró el error de truncado local de los métodos adaptados de orden exponencial 1 (ver también [149]), ó bien procediendo como se realizó para obtener las funciones generatrices de los coeficientes de estos nuevos métodos (cambiando λ por A y 1 por Id), se demuestra que el método implícito se puede hallar a partir de la fórmula (3.5.1) donde los coeficientes $\overline{\beta}_j^0$ se obtienen como ya se ha dicho del desarrollo en series en un entorno de $\xi = 0$ de la función generatriz que aparece en (3.2.2) cambiando λh por Ah , 1 por la matriz identidad y multiplicando por la inversa por la derecha en vez de dividir.

A partir de aquí podemos obtener el error de truncado local de forma similar a como se procedió en [149] (Teorema 2.1), o como ya hicimos en el capítulo anterior.

Una vez que la expresión que nos da el error de truncado local del método (3.2.5) de k pasos es

$$-\nabla \overline{\beta}_k^0 \nabla_P^k y(x) + O(h^{k+2}), \quad (3.5.2)$$

pero podemos considerar que los operadores satisfacen que $\nabla y(x) = hDy(x) + O(h^2)$, y $\nabla_P^k y(x) = h^k(D-A)^k y(x) + O(h^{k+1})$, con lo que el error de truncado local de (3.2.5) se escribe como

$$-h^{k+1}\overline{\beta}_k^0(D-A)^k Dy(x) + O(h^{k+2}). \quad \square$$

Teorema 3.5.2. *El método explícito procedente de la fórmula*

$$\nabla \sum_{j=0}^{k-1} \overline{\beta}_j^1 \nabla_{D-A}^j y_n = hg_{n-1},$$

de k pasos es convergente de orden k . Su error de truncado local puede expresarse como:

$$-h^{k+1}\overline{\beta}_k^1(D-A)^k Dy(x) + O(h^{k+2}).$$

Demostración: Sólo debemos recordar que

$$y(x) = E^{-1}y(x+h) = (Id - \nabla)y(x+h),$$

y que $(Id - \nabla) = e^{-Ah}(Id - \nabla_P)$ y se procede de forma similar al Teorema 3.5.1. \square

En teoría el número de pasos y el orden del método coinciden, sin embargo en la práctica es todo algo más complejo. Para comprobar lo que sucede pongamos una serie de resultados numéricos.

Consideremos

$$y'(x) = -10^7(y(x) - 2x - \sin(x)) + 2 + \cos(x), \quad y(0) = 0, \quad (3.5.3)$$

en el intervalo $[0, 1]$. La solución del problema de valor inicial es $y(x) = 2x + \sin(x)$, y la de los problemas perturbados (cuando $y(0) = c$) será del tipo $y(x) = 2x + \sin(x) + ce^{-10000000x}$.

Para integrar dicho problema tomamos el método explícito de 10 pasos, con paso constante $h = \frac{1}{100}, \dots, \frac{1}{10^6}$. En la tabla 3.1 mostramos los resultados en el punto final de integración.

Paso	Error
$h = \frac{1}{100}$	-2,57159
$h = \frac{1}{1000}$	-2,81447
$h = \frac{1}{10^4}$	-2,83877
$h = \frac{1}{10^5}$	-2,8412
$h = \frac{1}{10^6}$	-2,84144

Cuadro 3.1: Errores al integrar el problema (3.5.3) con el método EF-I-r-10 explícito.

Es decir, no sólo no parece tener orden 10, sino que no parece siquiera consistente. Para un paso más pequeño de $h = \frac{1}{10^7}$ el método no es estable, como se deduce en el Apéndice C, por lo que no se puede apreciar bien el comportamiento del método en referencia al orden.

Aquí se están juntando dos circunstancias:

- i) Por una parte estamos tomando $\lambda = -10^7$, es decir, estamos considerando que $y' \approx -10^7 y$, cuando en realidad esto no sucede en absoluto.
- ii) $\|\lambda h\|$ necesitamos que sea relativamente grande para que el método sea estable (como se puede observar en la figura C.3).

Por eso en la mayoría de las iteraciones $\nabla_P y(x) = y(x) - e^{\lambda h} y(x-h) \approx y(x)$.

De esta forma en estos casos la expresión de truncado local es

$$-\nabla \overline{\beta}_k^0 \nabla_P^k y(x) + O(h^{k+2}) \approx O(h).$$

Por tanto necesitamos, bien un parámetro más “adecuado”, o bien un integrador que nos permita dar pasos más pequeños.

Vamos ahora a comprobarlo con un par de ejemplos. En primer lugar consideremos el problema

$$y'(x) = -10^5(y(x) - 2x - \sin(x)) + 2 + \cos(x), \quad y(0) = 0, \quad (3.5.4)$$

en el intervalo $[0, 1]$. La solución del problema de valor inicial es $y(x) = 2x + \sin(x)$, y la de los problemas perturbados (cuando $y(0) = c$) será del tipo $y(x) = 2x + \sin(x) + ce^{-100000x}$. Pero esta vez lo integramos con el método explícito de dos pasos, que es estable incluso para pasos pequeños (como se observa en la figura C.1), los resultados para $x = 1$ se pueden observar en el cuadro 3.2.

Paso	Error
$h = \frac{1}{10}$	-0,258091
$h = \frac{1}{100}$	-0,025394
$h = \frac{1}{1000}$	-0,0024899
$h = \frac{1}{10^4}$	-0,00020325
$h = \frac{1}{10^5}$	$-4,16492 \times 10^{-6}$
$h = \frac{1}{10^6}$	$-4,303 \times 10^{-8}$

Cuadro 3.2: Errores al integrar el problema (3.5.4) con el método EF-I-r-2 explícito.

Aunque al principio parece que tiene orden 1, cuando $\lambda h \rightarrow 0$, el método se comporta como uno de orden 2.

Por último vamos a comprobar el comportamiento de este mismo método

explícito de orden 2 para el problema no lineal que aparece en [4]

$$y'(x) = -y(x)\sqrt{y(x)^2 + 10000}, \quad y(0) = -100. \quad (3.5.5)$$

La solución exacta de dicho problema es

$$y(x) = \frac{-20000(100 + \sqrt{20000})e^{100x}}{(100 + \sqrt{20000})^2 e^{200x} - 10000}.$$

El jacobiano de $g(x, y(x))$ respecto de $y(x)$ es

$$g_y = \frac{-2(y^2 + 5000)}{\sqrt{y^2 + 10000}} \approx -100,$$

por tanto vamos a considerar $\lambda = -100$.

En la tabla 3.3 mostramos los resultados en el punto $x = 1$.

Paso	Error
$h = \frac{1}{10}$	$-6,00177 \times 10^{-12}$
$h = \frac{1}{20}$	$-5,40166 \times 10^{-13}$
$h = \frac{1}{40}$	$-1,04904 \times 10^{-13}$
$h = \frac{1}{80}$	$-5,91917 \times 10^{-14}$

Cuadro 3.3: Errores al integrar el problema (3.5.5) con el método EF-I-r-2 explícito.

Mientras el paso se ha dividido por 8, el error se ha dividido por 101 aproximadamente, el método se comporta como uno de orden 2 incluso para longitudes de paso relativamente altas debido a que el jacobiano de la función es relativamente constante y por tanto el parámetro tomado es “eficiente”.

En el capítulo dedicado a los ejemplos numéricos se observarán mejor estos comportamientos, no obstante, ya hemos observado que la obtención óptima

del parámetro nos va a permitir una convergencia más rápida a la solución. Este hecho no sólo se da en los métodos que aparecen en este capítulo, sino en todos los que se presentan en esta tesis. Pero los métodos de orden algebraico 1 van a ser los más necesitados de un “buen” parámetro como veremos más adelante.

La primera observación que tenemos que hacer con respecto a la estabilidad es que en este caso el método explícito de orden 2 sí tiene una buena estabilidad (como se puede ver en las figuras C.1 y D.3). Sin embargo podemos observar que salvo el método explícito de orden 2, el resto de los métodos explícitos sólo van a funcionar bien en un número bastante limitado de problemas y utilizando un parámetro muy adecuado. Esto es debido a las propiedades de estabilidad que poseen estos métodos.

En cambio los métodos implícitos funcionan mucho mejor, y de hecho con bastantes problemas se podrán utilizar métodos de elevado número de pasos. Aunque la estabilidad de estos métodos de orden algebraico 1 es mejor que la de los métodos de orden exponencial 1, ya podemos darnos cuenta que en gran cantidad de ejemplos numéricos van a funcionar peor y necesitarán mayor precisión a la hora de elegir el parámetro adecuado.

Al igual que en el capítulo dedicado a los métodos de orden exponencial 1, a la hora de estudiar la estabilidad absoluta, consideraremos el problema

$$y'(t) = (A + B)y(t),$$

siendo Ah la matriz introducida en el método *exponential fitting* o adaptado. Esto nos permite comprender el tipo de error que podemos cometer al elegir la matriz parámetro del método.

De nuevo vamos a reducirnos a estudiar el caso escalar, ya que el caso vectorial haciendo un cambio de variable se va a reducir a varios problemas escalares del tipo

$$y'(t) = (\lambda + \nu)y(t),$$

donde $Re(\lambda + \nu) < 0$.

Sin embargo, de nuevo podemos considerar que ν va a ser un valor pequeño en comparación con λ , podemos esperar que ν sea cercano a cero.

Utilizaremos las definiciones de método (ó algoritmo) absolutamente estable y región de estabilidad de un método (ó algoritmo) dadas en el capítulo anterior.

Al igual que en el otro capítulo las regiones de estabilidad absoluta \mathfrak{R}_A son regiones de $C \times C$ y por tanto vamos a fijar la primera componente de u , esto es vamos a considerar el valor $u_1 = \lambda h$ que introducimos en el método, donde u_1 será un valor para el cual el método es 0-estable y dibujaremos en el plano complejo los valores de la segunda componente $u_2 = \nu h$ (ó $u_2 = (\lambda + \nu)h$, según cuál sea el algoritmo elegido) para los cuales el método es absolutamente estable.

Las propiedades son similares a las de los métodos de orden exponencial 1. Igualmente es conveniente que el parámetro u_1 que usamos en el método esté cerca de la zona de 0-estabilidad.

Pero ahora las zonas de inestabilidad absoluta decrecen más rápido que con los métodos anteriores cuándo u_1 se distancia de la zona de 0-inestabilidad.

Todas estas propiedades están más desarrolladas en el apéndice D.

Capítulo 4

Otros métodos EF- γ -k- δ -r o A- γ -k- δ -r.

4.1. Introducción.

Comencemos considerando en el intervalo cerrado $[a,b]$ el problema de valor inicial (2.1.1) siendo λ un escalar y $g(x, y(x))$ (ó bien $f(x, y(x))$) una función que satisface los teoremas de existencia y unicidad de la solución del problema de valor inicial.

Esta vez vamos a construir métodos que integren dicho problema de forma exacta cuando $y(x)$ pertenece al subespacio generado por $\langle 1, \dots, x^{r-1}, e^{\lambda x}, \dots, x^{k-1}e^{\lambda x} \rangle$. Esto es, vamos a construir de forma general los métodos *exponential fitting* o adaptados de orden algebraico r y orden exponencial k .

Los métodos de orden algebraico 1 ó los métodos de orden exponencial 1 que se han visto en capítulos anteriores no son más que casos particulares de estos.

De hecho si denominamos EF- γ -k- δ -r y A- γ -k- δ -r a los algoritmos que integran de forma exacta el problema (2.1.1) cuando $y(x)$ pertenece al subespacio

generado por $\langle 1, \dots, x^{\delta-1}, e^{\lambda x}, \dots, x^{\gamma-1}e^{\lambda x} \rangle$, entonces se verifica que los algoritmos que hemos denominado EF-I-k- δ (o A-I-k- δ) en el capítulo dedicado a los métodos de orden exponencial 1, se corresponden con los que llamamos ahora EF-1-k- δ -r (o A-1-k- δ -r).

De la misma manera se verifica que los algoritmos que hemos denominado A-I-r- γ (o EF-I-r- γ) en el capítulo dedicado a los métodos de orden algebraico 1, se corresponden con los que llamamos ahora A- γ -k-1-r (o EF- γ -k-1-r).

Por tanto el lector se puede hacer una idea de las diversas formas que hay de construir los métodos:

- 1) Imponer las condiciones para que el método integre de forma exacta el problema (2.1.1) cuando $y(x)$ pertenece al subespacio generado por $\langle 1, \dots, x^{r-1}, e^{\lambda x}, \dots, x^{k-1}e^{\lambda x} \rangle$.
- 2) Es ampliamente conocida la función generatriz que nos da los coeficientes del método BDF tradicional que integra de forma exacta el problema (2.1.1) cuando $y(x)$ pertenece al subespacio generado por $\langle 1, \dots, x^{r-1} \rangle$. Bastará con encontrar la forma de añadir $\langle e^{\lambda x}, \dots, x^{k-1}e^{\lambda x} \rangle$ al subespacio que el método BDF integra de forma exacta.
- 3) El mismo proceso, pero al revés. Primero encontramos la función generatriz que nos permite hallar el método BDF que integra el problema (2.1.1) cuando $y(x)$ pertenece al subespacio generado por $\langle e^{\lambda x}, \dots, x^{k-1}e^{\lambda x} \rangle$ y luego se añade a este subespacio el generado por $\langle 1, \dots, x^{r-1} \rangle$.
- 4) Hay otras vías procedentes de las fórmulas de capítulos anteriores y que se mencionarán brevemente más adelante.

Después de construir los métodos seguiremos el mismo guión de capítulos anteriores. Ya se han dado las definiciones de 0-estabilidad y estabilidad ab-

soluta que vamos a seguir, además como veremos más adelante, las formas que van a seguir las regiones 0-estabilidad y estabilidad absoluta de los diferentes algoritmos quedarán dentro de unos márgenes que nos dan los métodos de orden algebraico 1 y los de orden exponencial 1 que ya han sido expuestos. También se desarrollarán los teoremas que nos darán la consistencia de los BDF, con todo ello tendremos las propiedades necesarias para sacar las conclusiones necesarias acerca del comportamiento de los nuevos métodos.

4.2. Construcción de los algoritmos EF- γ -k- δ -r o A- γ -k- δ -r.

Ya hemos comentado que vamos a construir métodos que integren el problema (2.1.1) de forma exacta cuando $y(x)$ pertenece al subespacio generado por $\langle 1, \dots, x^{r-1}, e^{\lambda x}, \dots, x^{k-1}e^{\lambda x} \rangle$, y que para ello hay varias vías:

1) Como ya se ha explicado la primera forma en la que podemos obtener los coeficientes de los métodos proviene de resolver el sistema de ecuaciones no lineales que se obtiene de imponer que:

$$c_0y(x) + \dots + c_ny(x - nh) = hy'(x - sh),$$

para $y(x) = 1, y(x) = e^{\lambda x}, \dots$

Es un método bastante rápido si el orden del método es pequeño, y tiene la ventaja de que los coeficientes c_0, \dots, c_n se obtienen de forma explícita. Sin embargo, para los algoritmos de orden medio, alto o muy alto, este mecanismo es bastante (o muy) lento y costoso, y es necesario utilizar software matemático que realice cálculo simbólico bastante potente.

2) Las formulas en diferencias regresivas tradicionales son ampliamente conocidas. Sabemos que considerando el polinomio interpolador de la función $y(x)$ en el conjunto de puntos $\{x_{n-q}, \dots, x_n\}$, derivando este y evaluando en el punto x_{n-s} , obtenemos la fórmula

$$\sum_{i=0}^{r-1} \theta_{s,i} \nabla^i y_n = h g_{n-s},$$

donde

$$\theta_{s,i} = (-1)^i h \frac{d}{dx} \binom{-t}{i} \Big|_{x=x_{n-s}} = (-1)^i \frac{d}{dt} \binom{-t}{i} \Big|_{t=-s},$$

siendo $t = \frac{x-x_n}{h}$. Haciendo el sumatorio infinito de los términos obtenemos la función generatriz de los coeficientes $\theta_{s,i}$, que no es otra que la ya conocida

$$-\ln(1-\xi)(1-\xi)^s. \quad (4.2.1)$$

Por tanto si desarrollamos dicha función en torno al punto $\xi = 0$, obtenemos los coeficientes del BDF tradicional.

Esta fórmula nos permite integrar de forma exacta el problema (2.1.1) cuando $y(x)$ pertenece al subespacio generado por $\langle 1, \dots, x^{r-1} \rangle$. Luego, obviamente la fórmula

$$\left(\sum_{i=0}^{r-1} \theta_{s,i} \nabla^i y_n \right) + b_0^{s,k,r} \nabla^r y_n + \dots + b_{k-1}^{s,k,r} \nabla^{r+k-1} y_n = h g_{n-s}, \quad (4.2.2)$$

sigue integrando de forma exacta el problema (2.1.1) cuando $y(x)$ pertenece al subespacio generado por $\langle 1, \dots, x^{r-1} \rangle$, ya sólo nos queda por imponer que el algoritmo *exponential fitting* integre de forma exacta el problema (2.1.1) cuando $y(x)$ pertenece al subespacio generado por $\langle e^{\lambda x}, \dots, x^{k-1} e^{\lambda x} \rangle$ y resolver el sistema de ecuaciones en función de los $b_0^{s,k,r}, \dots, b_{k-1}^{s,k,r}$.

Obsérvese que respecto a la vía anterior de construir los métodos ganamos que en la mayoría de los métodos hay un número significativamente menor de ecuaciones e incógnitas que hay que resolver.

Obviamente los métodos *exponential fitting* de orden exponencial 1 y orden algebraico 1 son más fáciles de construir como ya se ha explicado previamente. Pero para el resto de métodos daremos a continuación los coeficientes $b_i^{s,k,r}$ de los métodos *exponential fitting* con los que vamos a trabajar más adelante en este capítulo.

Empezaremos con los explícitos, si bien ya podemos decir que fuera de los métodos de orden algebraico 1, el resto de ellos tienen malas propiedades de estabilidad como veremos más adelante. Sólo vamos a necesitar algunos métodos de orden algebraico 2 para demostrar que sus propiedades de estabilidad son malas.

En concreto vamos a dar los coeficientes de los métodos explícitos EF-2-k-2-r (hay que dar dos coeficientes), EF-3-k-2-r (tres) y EF-4-k-2-r (cuatro). Los coeficientes son diferentes para cada método, los podremos distinguir mediante los subíndices y superíndices. Así cuando se mencione el coeficiente $b_i^{1,j,l}$ nos estaremos refiriendo al coeficiente del método EF- j -k- l -r explícito, mientras que, como ya se ha comentado el coeficiente $b_i^{0,j,l}$ será el correspondiente al método EF- j -k- l -r implícito.

Los coeficientes del EF-2-k-2-r son:

$$b_0^{1,2,2} = \frac{e^{\lambda h}(3 + 2\lambda h + e^{\lambda h}(-3 + \lambda h))}{(-1 + e^{\lambda h})^2},$$

$$b_1^{1,2,2} = -\frac{e^{2\lambda h}(2 + \lambda h + e^{\lambda h}(-2 + \lambda h))}{(-1 + e^{\lambda h})^3}.$$

En cambio los coeficientes del EF-3-k-2-r son:

$$b_0^{1,3,2} = -\frac{e^{\lambda h}(-11 + e^{2\lambda h} - 6\lambda h + e^{\lambda h}(10 - 6\lambda h))}{2(-1 + e^{\lambda h})^2},$$

$$b_1^{1,3,2} = \frac{e^{2\lambda h}(-7 + e^{2\lambda h} - 3\lambda h + e^{\lambda h}(6 - 5\lambda h))}{(-1 + e^{\lambda h})^3},$$

$$b_2^{1,3,2} = -\frac{e^{3\lambda h}(-5 + e^{2\lambda h} - 2\lambda h + e^{\lambda h}(4 - 4\lambda h))}{2(-1 + e^{\lambda h})^4}.$$

Por último los coeficientes del EF-4-k-2-r son:

$$b_0^{1,4,2} = \frac{e^{\lambda h}(50 - 12e^{2\lambda h} + e^{3\lambda h} + 24\lambda h + e^{\lambda h}(-39 + 36\lambda h))}{6(-1 + e^{\lambda h})^2},$$

$$b_1^{1,4,2} = -\frac{e^{2\lambda h}(31 - 11e^{2\lambda h} + e^{3\lambda h} + 12\lambda h + 7e^{\lambda h}(-3 + 4\lambda h))}{2(-1 + e^{\lambda h})^3},$$

$$b_2^{1,4,2} = \frac{e^{3\lambda h}(22 - 10e^{2\lambda h} + e^{3\lambda h} + 8\lambda h + e^{\lambda h}(-13 + 22\lambda h))}{2(-1 + e^{\lambda h})^4},$$

$$b_3^{1,4,2} = -\frac{e^{4\lambda h}(17 - 9e^{2\lambda h} + e^{3\lambda h} + 6\lambda h + 9e^{\lambda h}(-1 + 2\lambda h))}{6(-1 + e^{\lambda h})^5}.$$

Los implícitos, en cambio, tienen mejores propiedades de estabilidad. Por eso vamos a dar los coeficientes de los métodos que no hayan sido descritos hasta ahora hasta orden 7.

Empezamos con los métodos de orden exponencial 2 (EF-2-k-2-r, EF-2-k-3-r, EF-2-k-4-r, EF-2-k-5-r, EF-2-k-6-r), cada uno de ellos necesita de dos coeficientes $b_i^{0,k,r}$, los coeficientes, claro, son diferentes para cada método, y por tanto para distinguirlos utilizamos los subíndices y superíndices como ya se ha comentado anteriormente:

$$b_0^{0,2,2} = -\frac{e^{\lambda h}(-2 + e^{2\lambda h} + e^{\lambda h}(1 - 3\lambda h))}{(-1 + e^{\lambda h})^2},$$

$$b_1^{0,2,2} = \frac{e^{2\lambda h}(-1 + e^{2\lambda h} - 2e^{\lambda h}\lambda h)}{(-1 + e^{\lambda h})^3},$$

$$b_0^{0,2,3} = -\frac{e^{\lambda h}(1 - 5e^{\lambda h} + 3e^{2\lambda h} + e^{3\lambda h} - 4e^{2\lambda h}\lambda h)}{(-1 + e^{\lambda h})^3},$$

$$b_1^{0,2,3} = \frac{e^{2\lambda h}(1 - 6e^{\lambda h} + 3e^{2\lambda h} + 2e^{3\lambda h} - 6\lambda h e^{2\lambda h})}{2(-1 + e^{\lambda h})^4},$$

$$\begin{aligned}
 b_0^{0,2,4} &= \frac{e^{\lambda h}(4 - 21e^{\lambda h} + 54e^{2\lambda h} - 6e^{4\lambda h} + e^{3\lambda h}(-31 + 30\lambda h))}{6(-1 + e^{\lambda h})^4}, \\
 b_1^{0,2,4} &= \frac{e^{2\lambda h}(-1 + 6e^{\lambda h} - 18e^{2\lambda h} + 3e^{4\lambda h} - 2e^{3\lambda h}(-5 + 6\lambda h))}{3(-1 + e^{\lambda h})^5}, \\
 b_0^{0,2,5} &= \frac{e^{\lambda h}(-1 + 6e^{\lambda h} - 16e^{2\lambda h} + 28e^{3\lambda h} - 2e^{5\lambda h} + 3e^{4\lambda h}(-5 + 4\lambda h))}{2(-1 + e^{\lambda h})^5}, \\
 b_1^{0,2,5} &= \frac{e^{2\lambda h}(3 - 20e^{\lambda h} + 60e^{2\lambda h} - 120e^{3\lambda h} + 12e^{5\lambda h} + e^{4\lambda h}(65 - 60\lambda h))}{12(-1 + e^{\lambda h})^6}, \\
 b_0^{0,2,6} &= \frac{e^{\lambda h}(24 - 165e^{\lambda h} + 500e^{2\lambda h} - 900e^{3\lambda h} + C_0^{0,2,6})}{60(-1 + e^{\lambda h})^6}, \\
 C_0^{0,2,6} &= 1200e^{4\lambda h} - 60e^{6\lambda h} + e^{5\lambda h}(-599 + 420\lambda h), \\
 b_1^{0,2,6} &= \frac{e^{2\lambda h}(-2 + 15e^{\lambda h} - 50e^{2\lambda h} + 100e^{3\lambda h} - 150e^{4\lambda h} + C_1^{0,2,6})}{10(-1 + e^{\lambda h})^7}, \\
 C_1^{0,2,6} &= 10e^{6\lambda h} + e^{5\lambda h}(77 - 60\lambda h).
 \end{aligned}$$

Tan sólo los métodos implícitos de orden hasta 6 serán 0-estables en un entorno del origen. Aunque algunos métodos de orden superior a 6 pueden dar buenos resultados en algunos problemas concretos, en nuestras explicaciones llegaremos únicamente al de orden 7.

No sólo sucede esto con los métodos de orden exponencial 2, sino también con los de orden exponencial 3, 4, 5, 6 ó 7.

A continuación damos los coeficientes de los métodos implícitos de orden exponencial 3 ($EF-3-k-2-r$, $EF-3-k-3-r$, $EF-3-k-4-r$, $EF-3-k-5-r$), cada uno de ellos necesita de tres coeficientes $b_i^{0,k,r}$.

$$\begin{aligned}
 b_0^{0,3,2} &= \frac{e^{\lambda h}(6 - 8e^{2\lambda h} + e^{3\lambda h} + e^{\lambda h}(1 + 12\lambda h))}{2(-1 + e^{\lambda h})^2}, \\
 b_1^{0,3,2} &= -\frac{e^{2\lambda h}(3 - 7e^{2\lambda h} + e^{3\lambda h} + e^{\lambda h}(3 + 8\lambda h))}{(-1 + e^{\lambda h})^3}, \\
 b_2^{0,3,2} &= \frac{e^{3\lambda h}(2 - 6e^{2\lambda h} + e^{3\lambda h} + e^{\lambda h}(3 + 6\lambda h))}{2(-1 + e^{\lambda h})^4},
 \end{aligned}$$

$$\begin{aligned}
b_0^{0,3,3} &= \frac{e^{\lambda h}(-3 + 18e^{\lambda h} - 10e^{3\lambda h} + e^{4\lambda h} + e^{2\lambda h}(-6 + 20\lambda h))}{2(-1 + e^{\lambda h})^3}, \\
b_1^{0,3,3} &= \frac{e^{2\lambda h}(3 - 22e^{\lambda h} + 18e^{3\lambda h} - 2e^{4\lambda h} + e^{2\lambda h}(3 - 30\lambda h))}{2(-1 + e^{\lambda h})^4}, \\
b_2^{0,3,3} &= \frac{e^{3\lambda h}(-1 + 8e^{\lambda h} - 8e^{3\lambda h} + e^{4\lambda h} + 12e^{2\lambda h}\lambda h)}{2(-1 + e^{\lambda h})^5}, \\
b_0^{0,3,4} &= \frac{e^{\lambda h}(2 - 12e^{\lambda h} + 38e^{2\lambda h} - 12e^{4\lambda h} + C_0^{0,3,4})}{2(-1 + e^{\lambda h})^4}, \tag{4.2.3}
\end{aligned}$$

$$\begin{aligned}
C_0^{0,3,4} &= e^{5\lambda h} + e^{3\lambda h}(-17 + 30\lambda h), \\
b_1^{0,3,4} &= -\frac{e^{2\lambda h}(1 - 7e^{\lambda h} + 26e^{2\lambda h} - 11e^{4\lambda h} + C_1^{0,3,4})}{(-1 + e^{\lambda h})^5}, \tag{4.2.4}
\end{aligned}$$

$$\begin{aligned}
C_1^{0,3,4} &= e^{5\lambda h} + 2e^{3\lambda h}(-5 + 12\lambda h) \\
b_2^{0,3,4} &= \frac{e^{3\lambda h}(2 - 15e^{\lambda h} + 60e^{2\lambda h} - 30e^{4\lambda h} + 3e^{5\lambda h} + C_2^{0,3,4})}{6(-1 + e^{\lambda h})^6}, \tag{4.2.5}
\end{aligned}$$

$$\begin{aligned}
C_2^{0,3,4} &= 20e^{3\lambda h}(-1 + 3\lambda h), \\
b_0^{0,3,5} &= \frac{e^{\lambda h}(-3 + 20e^{\lambda h} - 62e^{2\lambda h} + 136e^{3\lambda h} - 28e^{5\lambda h} + C_0^{0,3,5})}{4(-1 + e^{\lambda h})^5},
\end{aligned}$$

$$\begin{aligned}
C_0^{0,3,5} &= 2e^{6\lambda h} + e^{4\lambda h}(-65 + 84\lambda h), \\
b_1^{0,3,5} &= \frac{e^{2\lambda h}(9 - 68e^{\lambda h} + 240e^{2\lambda h} - 600e^{3\lambda h} + C_1^{0,3,5})}{12(-1 + e^{\lambda h})^6},
\end{aligned}$$

$$\begin{aligned}
C_1^{0,3,5} &= 156e^{5\lambda h} - 12e^{6\lambda h} - 5e^{4\lambda h}(-55 + 84\lambda h), \\
b_2^{0,3,5} &= \frac{e^{3\lambda h}(-1 + 8e^{\lambda h} - 30e^{2\lambda h} + 80e^{3\lambda h} - 24e^{5\lambda h} + C_2^{0,3,5})}{4(-1 + e^{\lambda h})^7},
\end{aligned}$$

$$C_2^{0,3,5} = 2e^{6\lambda h} + 5e^{4\lambda h}(-7 + 12\lambda h).$$

Ahora es el turno de los métodos implícitos de orden exponencial 4 (EF-4-k-2-r, EF-4-k-3-r, EF-4-k-4-r), en cada uno de ellos necesitamos resolver un sistema de cuatro ecuaciones para obtener los cuatro coeficientes $b_i^{0,k,r}$, estos cálculos pueden resultar bastante complicados y costosos. Es cierto que hay otras formas de obtener estos métodos menos complicadas, por ejemplo la que se expone en el punto 3), o en vez de partir de los BDF clásicos, se podría

partir de las fórmulas de los métodos de orden exponencial de orden 1. Sin embargo, vamos a acabar de dar todos los coeficientes de esta forma para que el lector tenga un sistema de escribir todos los métodos que se expliquen en este capítulo.

$$\begin{aligned}
 b_0^{0,4,2} &= \frac{e^{\lambda h}(24 - 60e^{2\lambda h} + 15e^{3\lambda h} - 2e^{4\lambda h} + e^{\lambda h}(23 + 60\lambda h))}{6(-1 + e^{\lambda h})^2}, \\
 b_1^{0,4,2} &= \frac{e^{2\lambda h}(-6 + 25e^{2\lambda h} - 7e^{3\lambda h} + e^{4\lambda h} - e^{\lambda h}(13 + 20\lambda h))}{(-1 + e^{\lambda h})^3}, \\
 b_2^{0,4,2} &= \frac{e^{3\lambda h}(8 - 42e^{2\lambda h} + 13e^{3\lambda h} - 2e^{4\lambda h} + e^{\lambda h}(23 + 30\lambda h))}{2(-1 + e^{\lambda h})^4}, \\
 b_3^{0,4,2} &= \frac{e^{4\lambda h}(-3 + 18e^{2\lambda h} - 6e^{3\lambda h} + e^{4\lambda h} - 2e^{\lambda h}(5 + 6\lambda h))}{3(-1 + e^{\lambda h})^5}, \\
 b_0^{0,4,3} &= \frac{e^{\lambda h}(-6 + 42e^{\lambda h} - 45e^{3\lambda h} + 9e^{4\lambda h} - e^{5\lambda h} + e^{2\lambda h}(1 + 60\lambda h))}{3(-1 + e^{\lambda h})^3}, \\
 b_1^{0,4,3} &= \frac{e^{2\lambda h}(6 - 52e^{\lambda h} + 78e^{3\lambda h} - 17e^{4\lambda h} + 2e^{5\lambda h} - e^{2\lambda h}(17 + 90\lambda h))}{(-1 + e^{\lambda h})^4}, \\
 b_2^{0,4,3} &= -\frac{e^{3\lambda h}(2 - 19e^{\lambda h} + 34e^{3\lambda h} - 8e^{4\lambda h} + e^{5\lambda h} - 2e^{\lambda h}(5 + 18\lambda h))}{(-1 + e^{\lambda h})^5}, \\
 b_3^{0,4,3} &= \frac{e^{4\lambda h}(3 - 30e^{\lambda h} + 60e^{3\lambda h} - 15e^{4\lambda h} + 2e^{5\lambda h} - 20e^{2\lambda h}(1 + 3\lambda h))}{6(-1 + e^{\lambda h})^6}, \\
 b_0^{0,4,4} &= \frac{e^{\lambda h}(8 - 54e^{\lambda h} + 204e^{2\lambda h} - 126e^{4\lambda h} + C_0^{0,4,4})}{6(-1 + e^{\lambda h})^4}, \\
 C_0^{0,4,4} &= 21e^{5\lambda h} - 2e^{6\lambda h} + 3e^{3\lambda h}(-17 + 70\lambda h), \\
 b_1^{0,4,4} &= \frac{e^{2\lambda h}(-2 + 16e^{\lambda h} - 71e^{2\lambda h} + 56e^{4\lambda h} - 10e^{5\lambda h} + C_1^{0,4,4})}{(-1 + e^{\lambda h})^5}, \\
 C_1^{0,4,4} &= e^{6\lambda h} + e^{3\lambda h}(10 - 84\lambda h), \\
 b_2^{0,4,4} &= \frac{e^{3\lambda h}(8 - 69e^{\lambda h} + 330e^{2\lambda h} - 300e^{4\lambda h} + 57e^{5\lambda h} - 6e^{6\lambda h} + C_2^{0,4,4})}{6(-1 + e^{\lambda h})^6}, \\
 C_2^{0,4,4} &= 20e^{3\lambda h}(-1 + 21\lambda h) \\
 b_3^{0,4,4} &= \frac{e^{4\lambda h}(-1 + 9e^{\lambda h} - 45e^{2\lambda h} + 45e^{4\lambda h} - 9e^{5\lambda h} + C_3^{0,4,4})}{3(-1 + e^{\lambda h})^7},
 \end{aligned}$$

$$C_3^{0,4,4} = e^{6\lambda h} - 60e^{3\lambda h}\lambda h.$$

Por último vamos a enumerar los coeficientes de los métodos de orden exponencial 5 (EF-5-k-2-r, EF-5-k-3-r), y el de orden exponencial 6 (EF-6-k-2-r), en cada uno de los primeros necesitamos resolver un sistema de cinco ecuaciones y en el último un sistema de seis ecuaciones, aquí se hace obvio que es conveniente encontrar otra vía para escribir estos métodos.

$$b_0^{0,5,2} = \frac{e^{\lambda h}(20 - 80e^{2\lambda h} + 30e^{3\lambda h} - 8e^{4\lambda h} + e^{5\lambda h} + e^{\lambda h}(37 + 60\lambda h))}{4(-1 + e^{\lambda h})^2},$$

$$b_1^{0,5,2} = -\frac{e^{2\lambda h}(30 - 195e^{2\lambda h} + 81e^{3\lambda h} - 23e^{4\lambda h} + 3e^{5\lambda h} + 8e^{\lambda h}(13 + 15\lambda h))}{3(-1 + e^{\lambda h})^3},$$

$$b_2^{0,5,2} = \frac{e^{3\lambda h}(20 - 162e^{2\lambda h} + 73e^{3\lambda h} - 22e^{4\lambda h} + 3e^{5\lambda h} + e^{\lambda h}(88 + 90\lambda h))}{2(-1 + e^{\lambda h})^4},$$

$$b_3^{0,5,2} = -\frac{e^{4\lambda h}(5 - 46e^{2\lambda h} + 22e^{3\lambda h} - 7e^{4\lambda h} + e^{5\lambda h} + e^{\lambda h}(25 + 24\lambda h))}{(-1 + e^{\lambda h})^5},$$

$$b_4^{0,5,2} = \frac{e^{5\lambda h}(12 - 120e^{2\lambda h} + 60e^{3\lambda h} - 20e^{4\lambda h} + 3e^{5\lambda h} + e^{\lambda h}(65 + 60\lambda h))}{12(-1 + e^{\lambda h})^6},$$

$$b_0^{0,5,3} = \frac{e^{\lambda h}(-30 + 240e^{\lambda h} - 420e^{3\lambda h} + 126e^{4\lambda h} - 28e^{5\lambda h} + C_0^{0,5,3})}{12(-1 + e^{\lambda h})^3},$$

$$C_0^{0,5,3} = 3e^{6\lambda h} + e^{2\lambda h}(109 + 420\lambda h),$$

$$b_1^{0,5,3} = \frac{e^{2\lambda h}(130 - 100e^{\lambda h} + 238e^{3\lambda h} - 77e^{4\lambda h} + C_1^{0,5,3})}{2(-1 + e^{\lambda h})^4},$$

$$C_1^{0,5,3} = 18e^{5\lambda h} - 2e^{6\lambda h} - 3e^{2\lambda h}(29 + 70\lambda h),$$

$$b_2^{0,5,3} = \frac{e^{3\lambda h}(-10 + 110e^{\lambda h} - 308e^{3\lambda h} + 106e^{4\lambda h} - 26e^{5\lambda h} + C_2^{0,5,3})}{2(-1 + e^{\lambda h})^5},$$

$$C_2^{0,5,3} = e^{6\lambda h} + e^{2\lambda h}(125 + 252\lambda h),$$

$$b_3^{0,5,3} = \frac{e^{4\lambda h}(15 - 174e^{\lambda h} + 540e^{3\lambda h} - 195e^{4\lambda h} + C_3^{0,5,3})}{6(-1 + e^{\lambda h})^6},$$

$$C_3^{0,5,3} = 50e^{5\lambda h} - 6e^{6\lambda h} - 10e^{2\lambda h}(23 + 42\lambda h),$$

$$b_4^{0,5,3} = \frac{e^{5\lambda h}(-2 + 24e^{\lambda h} - 80e^{3\lambda h} + 30e^{4\lambda h} - 8e^{5\lambda h} + C_4^{0,5,3})}{4(-1 + e^{\lambda h})^7},$$

$$\begin{aligned}
 C_4^{0,5,3} &= e^{6\lambda h} + e^{2\lambda h}(7 + 12\lambda h), \\
 b_0^{0,6,2} &= \frac{e^{\lambda h}(120 - 700e^{2\lambda h} + 350e^{3\lambda h} - 140e^{4\lambda h} + C_0^{0,6,2})}{20(-1 + e^{\lambda h})^2}, \\
 C_0^{0,6,2} &= 35e^{5\lambda h} - 4e^{6\lambda h} + e^{\lambda h}(339 + 420\lambda h), \\
 b_1^{0,6,2} &= \frac{e^{2\lambda h}(-90 + 840e^{2\lambda h} - 462e^{3\lambda h} + 196e^{4\lambda h} - 51e^{5\lambda h} + C_1^{0,6,2})}{6(-1 + e^{\lambda h})^3}, \\
 C_1^{0,6,2} &= 6e^{6\lambda h} - e^{\lambda h}(439 + 420\lambda h), \\
 b_2^{0,6,2} &= \frac{e^{3\lambda h}(40 - 462e^{2\lambda h} + 273e^{3\lambda h} - 122e^{4\lambda h} + C_2^{0,6,2})}{2(-1 + e^{\lambda h})^4}, \\
 C_2^{0,6,2} &= 33e^{5\lambda h} - 4e^{6\lambda h} + e^{\lambda h}(242 + 210\lambda h), \\
 b_3^{0,6,2} &= \frac{e^{4\lambda h}(-15 + 196e^{2\lambda h} - 122e^{3\lambda h} + 57e^{4\lambda h} - 16e^{5\lambda h} + C_3^{0,6,2})}{(-1 + e^{\lambda h})^5}, \\
 C_3^{0,6,2} &= 2e^{6\lambda h} - 6e^{\lambda h}(17 + 14\lambda h), \\
 b_4^{0,6,2} &= \frac{e^{5\lambda h}(72 - 1020e^{2\lambda h} + 660e^{3\lambda h} - 320e^{4\lambda h} + C_4^{0,6,2})}{12(-1 + e^{\lambda h})^6}, \\
 C_4^{0,6,2} &= 93e^{5\lambda h} - 12e^{6\lambda h} + e^{\lambda h}(527 + 420\lambda h), \\
 b_5^{0,6,2} &= \frac{e^{6\lambda h}(-10 + 150e^{2\lambda h} - 100e^{3\lambda h} + 50e^{4\lambda h} - 15e^{5\lambda h} + C_5^{0,6,2})}{10(-1 + e^{\lambda h})^7}, \\
 C_5^{0,6,2} &= 2e^{6\lambda h} - e^{\lambda h}(77 + 60\lambda h).
 \end{aligned}$$

3) Como ya hemos explicado otra posibilidad es calcular la función generatriz que nos permite hallar el BDF que integra el problema (2.1.1) cuando $y(x)$ pertenece al subespacio generado por $\langle e^{\lambda x}, \dots, x^{k-1}e^{\lambda x} \rangle$ y luego realizar el mismo proceso que en el caso 2), pero ahora habría que añadir a este subespacio el generado por $\langle 1, \dots, x^{r-1} \rangle$.

Dado que como ya hemos explicado los métodos explícitos de este capítulo tienen malas propiedades y apenas los vamos a utilizar nos vamos a centrar en el caso implícito.

Ya hemos visto en capítulos anteriores que $Id - \nabla = E^{-1} = (Id - \nabla_P)e^{-\lambda h}$, por tanto

$$\begin{aligned} hf(x) &= h(D - \lambda)y(x) = \\ &= (-\ln((Id - \nabla_P)e^{-\lambda h}) - \lambda h)y(x) = -\ln(Id - \nabla_P)y(x), \end{aligned}$$

por lo que podemos escribir los BDF que integran el problema (2.1.1) cuando $y(x)$ pertenece al subespacio generado por $\langle e^{\lambda x}, \dots, x^{k-1}e^{\lambda x} \rangle$ a partir de la fórmula

$$\left(\sum_{i=0}^{k-1} \theta_{0,i} \nabla_P^i y_n \right) = hf_n,$$

donde los $\theta_{0,i}$ coinciden con los coeficientes del BDF tradicional, ya que sus funciones generatrices son las mismas.

Por tanto las fórmulas

$$\left(\sum_{i=0}^{k-1} \theta_{0,i} \nabla_P^i y_n \right) + \overline{b_0^{s,k,r}} \nabla_P^r y_n + \dots + \overline{b_{r-1}^{s,k,r}} \nabla_P^{r+k-1} y_n = hf_{n-s},$$

siguen integrando el problema (2.1.1) cuando $y(x)$ pertenece al subespacio generado por $\langle e^{\lambda x}, \dots, x^{k-1}e^{\lambda x} \rangle$, ya sólo nos queda imponer las condiciones para que integren el problema (2.1.1) cuando $y(x)$ pertenezca a $\langle 1, \dots, x^{r-1} \rangle$.

La ventaja de esta forma de obtener los métodos respecto a la anterior es que en algunos métodos el número de ecuaciones a resolver es menor. Por ejemplo, el método EF-6-k-2-r antes presentaba un sistema de 6 ecuaciones que no era nada sencillo de resolver, con la nueva vía, sólo habría que resolver un sistema de dos ecuaciones para obtener los coeficientes del algoritmo A-6-k-2-r. Por el contrario si antes en el método EF-2-k-6-r, sólo había que resolver un sistema de dos ecuaciones, ahora serán seis las que haya que resolver para obtener los coeficientes del A-2-k-6-r.

Aquí sólo se han propuesto tres formas de desarrollar los métodos pero claramente existen más. Un ejemplo ya fue descrito, y es el de utilizar en vez de los BDF clásicos, las fórmulas de los métodos de orden exponencial 1 y añadir nuevas condiciones hasta ampliar el subespacio que el método integra de forma exacta. O utilizar los métodos de orden algebraico 1, de forma similar a como se acaba de hacer cuando los métodos integran el problema si $y(x)$ pertenece al subespacio generado por $\langle e^{\lambda x}, \dots, x^{k-1}e^{\lambda x} \rangle$.

Obviamente así es como se construyen los algoritmos para los problemas escalares, si por el contrario nuestro problema fuera vectorial, esto es, si consideramos en el intervalo cerrado $[a,b]$ el P.V.I. (1.0.3) donde A es una matriz $m \times m$ y $F(x, y(x)) = (f_1(x, y), \dots, f_m(x, y))$ es un vector de dimensión m , cuyas componentes satisfacen los teoremas de existencia y unicidad de la solución del problema de valor inicial; entonces, las vías de construcción de los algoritmos son las ya descritas sin más que cambiar λ por A y 1 por la correspondiente matriz identidad de dimensión $m \times m$, y considerando que $\frac{B}{C} = BC^{-1}$.

La forma de construir los algoritmos *exponential fitting* a partir de los adaptados o viceversa es obviamente la ya explicada: si el método implícito de tipo *exponential fitting* es como en (2.4.2), el método implícito de tipo adaptado que integra de forma exacta el mismo espacio de funciones se escribe como en (2.4.3), donde las únicas diferencias son que $c_0 - c_0^* = Ah$, y que $g_n = G(x_n, y_n) = F(x_n, y_n) + Ay_n = f_n + Ay_n$, y de la misma forma si el método explícito de tipo EFM se escribe como en (2.4.4), el método explícito de tipo AM que integra de forma exacta el mismo espacio de funciones se escribe al igual que en (2.4.5), donde esta vez el único coeficiente que varía

es ahora el segundo: $c_1 - c_1^* = Ah$.

4.3. Convergencia.

Como en otros capítulos, en esta sección se va a estudiar la estabilidad y consistencia de los nuevos métodos.

En los apéndices E y F, queda bien claro que las propiedades de estabilidad de los métodos explícitos que aparecen por primera vez en este capítulo presentan malas propiedades de estabilidad, por esa razón nos vamos a centrar especialmente en dar la consistencia de los métodos implícitos que no se hayan dado en capítulos anteriores.

Para estudiar la consistencia del método implícito $EF-\gamma-k-\delta-r$ basta hacer el desarrollo en series de Taylor de la expresión (4.2.2), cambiando y_n por $y(x)$ y considerando que $s = 0$ y que por tanto $g_{n-s} = y'(x)$.

Para hacer el desarrollo de Taylor, vamos a partir del conocimiento que tenemos de la expresión de truncado local del BDF de δ pasos hasta un orden $\gamma + \delta$. Para ello basta con recordar que los coeficientes $\theta_{0,i}$ provienen del desarrollo en series de la función generatriz $-\ln(1-t)$ en torno a $t = 0$, por tanto hasta el término 8 los coeficientes del BDF clásicos son los mostrados en el cuadro 4.1.

0	1	2	3	4	5	6	7	8
0	1	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{5}$	$\frac{1}{6}$	$\frac{1}{7}$	$\frac{1}{8}$

Cuadro 4.1 : Coeficientes de los métodos BDF implícitos clásicos.

Para hacer el desarrollo en series de Taylor de los coeficientes $b_i^{0,\gamma,\delta}$ tan sólo

es preciso recordar cuál es el desarrollo de $e^{\lambda h}$ en torno a $h = 0$:

$$e^{\lambda h} \approx 1 + \frac{\lambda h}{1!} + \dots + \frac{(\lambda h)^n}{n!}.$$

Por tanto ya sólo queda hacer el desarrollo de Taylor de las diferencias regresivas $\nabla^i y(x)$ y posteriormente habrá que simplificar.

Denotemos

$$(\nabla y_n) = (\nabla^0 y_n, \dots, \nabla^{\gamma+\delta-1} y_n),$$

si hacemos el desarrollo en series de Taylor del producto alcanzamos que

$$(\nabla y_n)^t = M_{\gamma+\delta} \cdot D_{\gamma+\delta}^t + O(h^{\gamma+\delta}), \quad (4.3.1)$$

(denotaremos la traspuesta del vector v con v^t) donde

$$D_{\gamma+\delta} = \left(y(x_n), \frac{y'(x_n)}{1!}, \dots, \frac{y^{\gamma+\delta-1}(x_n)}{(\gamma+\delta-1)!} \right) \quad (4.3.2)$$

y $M_{\gamma+\delta}$ representa la matriz de dimensión $(\gamma+\delta) \times (\gamma+\delta)$ cuyos elementos son

$$(M_{\gamma+\delta})_{ij} = \sum_{m=0}^{i-1} (-1)^{m+j-1} \binom{i-1}{m} (mh)^{j-1}. \quad (4.3.3)$$

Ahora si los desarrollos de Taylor de los coeficientes $b_i^{0,\gamma,\delta}$ los denotamos como $d_i^{\gamma,\delta}$ y denotando

$$\left(\theta_{0,0}, \dots, \theta_{0,\delta-1}, d_0^{\gamma,r}, \dots, d_{\gamma-1}^{\gamma,\delta} \right) = (C^{\gamma,\delta}),$$

se verifica que el error de truncado local se logra a partir de la fórmula

$$(C^{\gamma,\delta}) \cdot (\nabla y_n) - h y'(x_n) + O(h^{\gamma+\delta+1}) =$$

$$(C^{\gamma,\delta}) \cdot M_{\gamma+\delta} \cdot D_{\gamma+\delta}^t - \theta_{0,\gamma+\delta} h^{\gamma+\delta} y^{\gamma+\delta} - h y'(x_n) + O(h^{\gamma+\delta+1}).$$

Así, por ejemplo, el error de truncado local del método implícito EF-3-k-4-r se calcularía de la siguiente manera.

Sabemos que $\theta_{0,0} = 0$, $\theta_{0,1} = 1$, $\theta_{0,2} = \frac{1}{2}$ y $\theta_{0,3} = \frac{1}{3}$, y $b_0^{0,3,4}$, $b_1^{0,3,4}$ y $b_2^{0,3,4}$ son los dados en (4.2.3), (4.2.4) y (4.2.5), respectivamente.

Dado que sabemos que $\nabla^i y(x_n) = y^{(i)} h^i + O(h^{i+1})$, tendremos que hacer el desarrollo de Taylor de orden 3 del coeficiente (4.2.3), el de orden 2 de (4.2.4) y el de orden 1 de (4.2.5). De esta forma obtenemos que

$$(C^{3,4}) = \left(0, 1, \frac{1}{2}, \frac{1}{3}, \frac{1}{4} + \frac{(\lambda h)^3}{7}, \frac{1}{5} - \frac{3(\lambda h)^2}{7}, \frac{1}{6} + \frac{3(\lambda h)}{7} \right).$$

Ahora, como

$$M_7 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & h & -h^2 & h^3 & -h^4 & h^5 & -h^6 \\ 0 & 0 & 2h^2 & -6h^3 & 14h^4 & -30h^5 & 62h^6 \\ 0 & 0 & 0 & 6h^3 & -36h^4 & 150h^5 & -540h^6 \\ 0 & 0 & 0 & 0 & 24h^4 & -240h^5 & 910h^6 \\ 0 & 0 & 0 & 0 & 0 & 120h^5 & -1800h^6 \\ 0 & 0 & 0 & 0 & 0 & 0 & 720h^6 \end{pmatrix}, \quad (4.3.4)$$

entonces

$$(C^{3,4}) \cdot M_7 \cdot D_7^t = hy'(x) - \frac{h^7((D-A)^3 D^4 y(x) - y^{(7)}(x))}{7},$$

con lo que el error de truncado local es

$$(C^{3,4}) \cdot M_7 \cdot D_7^t - hy'(x) - \frac{h^7 y^{(7)}(x)}{7} = -\frac{h^7 (D-A)^3 D^4 y(x)}{7}.$$

Ahora bien, es claro como es el desarrollo en series de Taylor de los coeficientes $b_i^{0,\gamma,\delta}$, ya que si $\lambda h \rightarrow 0$, entonces los métodos nuevos tienden a los BDF clásicos, por lo que si $\lambda h \rightarrow 0$, entonces $b_i^{0,\gamma,\delta} \rightarrow \theta_{i+\delta}$ y sabemos cuál es el error de los BDF clásicos, por lo que si denotamos

$$(\theta^{\gamma+\delta}) = (\theta_{0,0}, \dots, \theta_{0,\gamma+\delta-1})$$

resulta mucho más fácil calcular el error de truncado local a partir de la fórmula

$$((C^{\gamma,\delta}) - (\theta^{\gamma+\delta})) \cdot M_{\gamma+\delta} \cdot D_{\gamma+\delta}^t - \theta_{0,\gamma+\delta} h^{\gamma+\delta} y^{\gamma+\delta}(x),$$

ya que los términos de

$$((C^{\gamma,\delta}) - (\theta^{\gamma+\delta}))$$

ó son 0 ó son monomios, con lo que es más fácil operar con ellos.

Operando de esta forma obtenemos los siguientes teoremas de truncado local:

Teorema 4.3.1. *El método implícito EF-2-k-2-r obtenido a partir de la fórmula (4.2.2) de 3 pasos es convergente de orden 3. Su error de truncado local se puede expresar como:*

$$-\frac{h^4(D-A)^2D^2y(x)}{4} + O(h^5).$$

Teorema 4.3.2. *El método implícito EF-2-k-3-r obtenido a partir de la fórmula (4.2.2) de 4 pasos es convergente de orden 4. Su error de truncado local se puede expresar como:*

$$-\frac{h^5(D-A)^2D^3y(x)}{5} + O(h^6).$$

Teorema 4.3.3. *El método implícito EF-2-k-4-r obtenido a partir de la fórmula (4.2.2) de 5 pasos es convergente de orden 5. Su error de truncado local se puede expresar como:*

$$-\frac{h^6(D-A)^2D^4y(x)}{6} + O(h^7).$$

Teorema 4.3.4. *El método implícito EF-2-k-5-r obtenido a partir de la fórmula (4.2.2) de 6 pasos es convergente de orden 6. Su error de truncado local se puede expresar como:*

$$-\frac{h^7(D-A)^2D^5y(x)}{7} + O(h^8).$$

Teorema 4.3.5. *El método implícito EF-2-k-6-r obtenido a partir de la fórmula (4.2.2) de 7 pasos es convergente de orden 7. Su error de truncado*

local se puede expresar como:

$$-\frac{h^8(D-A)^2D^6y(x)}{8} + O(h^9).$$

Teorema 4.3.6. El método implícito $EF-3-k-2-r$ obtenido a partir de la fórmula (4.2.2) de 4 pasos es convergente de orden 4. Su error de truncado local se puede expresar como:

$$-\frac{h^5(D-A)^3D^2y(x)}{5} + O(h^6).$$

Teorema 4.3.7. El método implícito $EF-3-k-3-r$ obtenido a partir de la fórmula (4.2.2) de 5 pasos es convergente de orden 5. Su error de truncado local se puede expresar como:

$$-\frac{h^6(D-A)^3D^3y(x)}{6} + O(h^7).$$

Teorema 4.3.8. El método implícito $EF-3-k-4-r$ obtenido a partir de la fórmula (4.2.2) de 6 pasos es convergente de orden 6. Su error de truncado local como ya se ha demostrado es:

$$-\frac{h^7(D-A)^3D^4y(x)}{7} + O(h^8).$$

Teorema 4.3.9. El método implícito $EF-3-k-5-r$ obtenido a partir de la fórmula (4.2.2) de 7 pasos es convergente de orden 7. Su error de truncado local se puede expresar como:

$$-\frac{h^8(D-A)^3D^5y(x)}{8} + O(h^9).$$

Teorema 4.3.10. El método implícito $EF-4-k-2-r$ obtenido a partir de la fórmula (4.2.2) de 5 pasos es convergente de orden 5. Su error de truncado local se puede expresar como:

$$-\frac{h^6(D-A)^4D^2y(x)}{6} + O(h^7).$$

Teorema 4.3.11. *El método implícito EF-4-k-3-r obtenido a partir de la fórmula (4.2.2) de 6 pasos es convergente de orden 6. Su error de truncado local se puede expresar como:*

$$-\frac{h^7(D-A)^4 D^3 y(x)}{7} + O(h^8).$$

Teorema 4.3.12. *El método implícito EF-4-k-4-r obtenido a partir de la fórmula (4.2.2) de 7 pasos es convergente de orden 7. Su error de truncado local se puede expresar como:*

$$-\frac{h^8(D-A)^4 D^4 y(x)}{8} + O(h^9).$$

Teorema 4.3.13. *El método implícito EF-5-k-2-r obtenido a partir de la fórmula (4.2.2) de 6 pasos es convergente de orden 6. Su error de truncado local se puede expresar como:*

$$-\frac{h^7(D-A)^5 D^2 y(x)}{7} + O(h^8).$$

Teorema 4.3.14. *El método implícito EF-5-k-3-r obtenido a partir de la fórmula (4.2.2) de 7 pasos es convergente de orden 7. Su error de truncado local se puede expresar como:*

$$-\frac{h^8(D-A)^5 D^3 y(x)}{8} + O(h^9).$$

Teorema 4.3.15. *El método implícito EF-6-k-2-r obtenido a partir de la fórmula (4.2.2) de 7 pasos es convergente de orden 7. Su error de truncado local se puede expresar como:*

$$-\frac{h^8(D-A)^6 D^2 y(x)}{8} + O(h^9).$$

Y como ya se ha dicho antes que $\theta_{0,i} = \frac{1}{i}$, entonces todos estos métodos verifican el siguiente teorema:

Teorema 4.3.16. *El método implícito EF- γ - k - δ - r (con $\gamma + \delta \leq 8$) obtenido a partir de la fórmula (4.2.2) de $\gamma + \delta - 1$ pasos es convergente de orden $\gamma + \delta - 1$. Su error de truncado local se puede expresar como:*

$$-\theta_{0,\gamma+\delta}h^{\gamma+\delta}(D - A)^\gamma D^\delta y(x) + O(h^{\gamma+\delta+1}).$$

Podemos recordar que tanto los métodos de orden exponencial 1, como los métodos de orden algebraico 1, verificaban este teorema, ya que la expresión de truncado local del método de orden exponencial 1 y algebraico r era

$$-h^{r+1}\beta_r^0(D - A)D^r y(x) + O(h^{r+2})$$

pero ya hemos comentado que cuando $h \rightarrow 0$, entonces $\beta_r^0 \rightarrow \theta_{0,r+1}$, mientras la expresión de truncado local de los métodos de orden algebraico 1 y exponencial k era

$$-h^{k+1}\overline{\beta}_k^0(D - A)D^k y(x) + O(h^{k+2})$$

y de nuevo cuando $h \rightarrow 0$, entonces $\overline{\beta}_k^0 \rightarrow \delta_{0,r+1}$.

De forma análoga se comportan los métodos explícitos con orden menor o igual a 4, así como todos los métodos explícitos de orden exponencial 1, y todos los métodos explícitos de orden algebraico 1, cuya expresión de truncado local es

$$-\theta_{1,\gamma+\delta}h^{\gamma+\delta}(D - A)^\gamma D^\delta y(x) + O(h^{\gamma+\delta+1}),$$

donde los $\theta_{1,i}$ son los coeficientes del BDF tradicional y que se obtienen a partir de la función generatriz

$$-\ln(1 - \xi)(1 - \xi),$$

de esta forma los seis primeros son los que aparecen en el cuadro 4.2.

0	1	2	3	4	5	6
0	1	$-\frac{1}{2}$	$-\frac{1}{6}$	$-\frac{1}{12}$	$-\frac{1}{20}$	$-\frac{1}{30}$

Cuadro 4.2 : Coeficientes de los métodos BDF explícitos clásicos .

Vista la consistencia de los nuevos métodos comentaremos brevemente la estabilidad de los nuevos métodos, propiedad que queda más desarrollada en los apéndices E y F.

Como ya se ha comentado, los métodos explícitos muestran malas regiones de estabilidad, no así los implícitos que trabajan muy bien hasta orden 6, e incluso es posible encontrar métodos de orden 7 u 8 que logren buenos resultados numéricos con problemas concretos para longitudes de paso relativamente grandes.

En cuanto a la estabilidad absoluta de los métodos mostrados en este capítulo podemos obtener que a partir de las definiciones dadas en el capítulo de los métodos de orden exponencial 1, se siguen verificando las propiedades que hemos podido contemplar en los dos capítulos anteriores a este (y que se encuentran desarrolladas en los apéndices A, B, C y D), dichas propiedades se pueden resumir en:

- 1.- Observamos que las regiones de inestabilidad absoluta de un método dado pueden ser muy grandes si el parámetro λh está muy cerca de las regiones de 0-inestabilidad de los algoritmos correspondientes a dicho método.
- 2.- Observamos que las regiones de estabilidad absoluta crecen cuando $\lambda h \rightarrow -\infty$, siendo $\lambda h \in R^-$, esto es, cuando el parámetro $|\lambda h|$ es muy grande, el error al elegir dicho parámetro puede ser considerablemente grande y sin embargo, el método continúa siendo estable. Aunque previamente ya se ha

explicado que es muy conveniente que, si $|\lambda h|$ es muy grande, el parámetro del método sea adecuado a la solución, o de lo contrario el error al aproximar la solución puede ser considerable.

3.- Observamos que, si $\lambda h \in C^-$, las regiones de estabilidad absoluta crecen para λht cuando $t \rightarrow \infty$ (considerando que los λht pertenecen a las regiones de 0-estabilidad).

4.- También podemos observar que, si $\lambda_1 h = a + ib \in C^-$, $a, b \in R$ y $\lambda_2 h = a - ib$ las regiones de estabilidad absoluta de las dos figuras son simétricas entre sí.

Además, ya hemos mencionado que entre dos métodos del mismo número de pasos, el de mayor orden exponencial tiene mejores propiedades de estabilidad, como queda claro en el apéndice F.

Capítulo 5

Métodos en paso variable y transición a métodos Runge-Kutta.

5.1. Introducción.

En este capítulo y el siguiente se pretenden mostrar nuevas vías de investigación que se originan a partir de la tesis. La idea de los métodos *exponential fitting* o adaptados que se muestra en esta tesis, pensamos, es muy amplia y diversa, y aún quedan muchos puntos importantes que pueden y deben estudiarse en el futuro. En los dos próximos capítulos pretendemos afrontar vías de expansión (nuevos temas de investigación) que se originan a partir de los métodos que se han mostrado en anteriores capítulos.

1) Una característica bastante común entre los problemas considerados *stiff* es que habitualmente poseen zonas claramente diferenciadas, en unas la solución de la ecuación cambia bruscamente en un corto espacio de tiempo, en esas zonas la solución suele ser bastante inestable, por contra en otras zonas la solución tiene una variación más moderada en relación a la variable temporal.

68 *Métodos en paso variable y transición a métodos Runge-Kutta.*

Por tanto, los integradores que más frecuentemente se utilizan en problemas *stiff* pueden cambiar la longitud de paso. El tema de cambio de longitud de paso es un tema que no se ha tratado todavía y que es bastante complejo según se va a mostrar en este capítulo.

En primer lugar daremos alguna idea para desarrollar algunos métodos *exponential fitting* o adaptados de paso variable.

Después analizaremos las propiedades de dos de los métodos de dos pasos de paso variable. Con este ejemplo, se podrá adivinar que estudiar en profundidad la estabilidad de los métodos de varios pasos con longitud de paso variable es un campo de investigación muy grande que se abre según aumenta el número de pasos.

2) Actualmente (ver [117] y [152]) se están utilizando métodos multipaso (con paso variable) para construir métodos de tipo Runge-Kutta muy potentes. En este capítulo aprovecharemos que ya sabremos construir métodos en paso variable para construir métodos *exponential fitting* de tipo Runge-Kutta.

3) Los métodos que aparecen en este trabajo dependen de un parámetro que en muchas ocasiones no se logra de forma inmediata. Lógicamente este parámetro depende del problema que se trata y por tanto la forma de hallarla será diferente en muchos casos. En este aspecto se profundizará en el próximo capítulo.

4) Otro campo de investigación del que todavía no hemos hablado es el de las funciones que deben integrar de forma exacta los métodos. Hasta ahora nos hemos limitado a pedir que los métodos integraran de forma exacta el problema cuando cada una de las componentes de la solución pertenecía al subespacio generado por $\langle 1, e^{\lambda_i x} \rangle$. A este subespacio le hemos añadido poli-

nomios y polinomios por la exponencial. Es decir la base de los métodos hasta ahora es que integran de forma exacta el problema cuando la solución son constantes más la matriz exponencial e^{Ax} . Sería interesante preguntarnos que sucede si añadimos otra (u otras) exponencial(es) diferente(s) e^{Bx} (e^{Cx}, \dots) al subespacio que el método integra de forma exacta.

De momento en este capítulo vamos a centrarnos en los dos primeros puntos, mientras que los apartados 3) y 4) se desarrollarán en el siguiente capítulo.

5.2. Construcción de métodos *exponential fitting* o adaptados de paso variable.

Esta sección la vamos a dividir en dos partes: en la primera de ellas vamos a dar un método de construcción de métodos adaptados de orden exponencial 1 con paso variable.

Posteriormente nos vamos a centrar en los dos métodos (uno explícito y el otro implícito) de orden exponencial 2 y orden algebraico 1, es decir en los dos EF-2-k-1-r, y vamos a estudiar sus propiedades de estabilidad.

5.2.1. Construcción de métodos EF-I-k- δ o A-I-k- δ de paso variable.

Para ello empezamos escribiendo los métodos de paso fijo con el algoritmo adaptado

$$(\beta) \cdot (\nabla y_{n+1})^t - e^{\lambda h} (\beta) \cdot (\nabla y_n)^t = h_n f_{n+s},$$

70 Métodos en paso variable y transición a métodos Runge-Kutta.

donde $s = 0, 1$ representa los casos explícito e implícito respectivamente,

$h_n = x_{n+1} - x_n$ y donde

$$(\beta) = (\beta_0, \dots, \beta_{k-1}),$$

$$(\nabla y_n) = (\nabla^0 y_n, \dots, \nabla^{k-1} y_n).$$

Ya hemos visto que si hacemos el desarrollo en series de las diferencias regresivas (prescindiendo del término $O(h^k)$) alcanzamos que

$$(\beta) \cdot (\nabla y_n)^t = (\beta) \cdot M_k \cdot D_k^t,$$

donde

$$D_k = \left(y(x_n), \frac{y'(x_n)}{1!}, \dots, \frac{y^{(k-1)}(x_n)}{(k-1)!} \right)$$

y M_k representa la matriz de dimensión $k \times k$ cuyos elementos son los mencionados en (4.3.3).

En este punto recordamos la noción de diferencia dividida ceroésima de una función f en un punto x_i como el valor de la función en ese punto, y recursivamente se logran las diferencias divididas de orden mayor a partir de la igualdad

$$y[x_i, \dots, x_{i+k}] = \frac{y[x_i, \dots, x_{i+k-1}] - y[x_{i+1}, \dots, x_{i+k}]}{x_i - x_{i+k}}$$

de esta forma si asumimos que f es una función analítica tenemos la identidad

$$y[x_n, \dots, x_{n-(k-1)}] = \sum_{j=0}^{\infty} h_{k,j-(k-1)} \frac{y^{(j)}(x_n)}{(j)!}$$

donde los polinomios simétricos completos $h_{k,j-(k-1)}$ se expresan en términos de los valores $H_n = x_n - x_n$, $H_{n-1} = x_{n-1} - x_n$, \dots , $H_{n-(k-1)} = x_{n-(k-1)} - x_n$,

definiéndose el polinomio simétrico completo $h_{n,k}$ en las variables z_1, \dots, z_n como la suma de todos los monomios de grado k , es decir

$$\begin{aligned} h_{n,0} &= 1, \\ h_{n,k} &= \sum_{1 \leq i_1 \leq i_2 \leq \dots \leq i_k \leq n} \prod_{m=1}^k z_{i_m}, \quad k > 0. \end{aligned} \quad (5.2.1)$$

De esta forma

$$\begin{pmatrix} y[x_n] \\ y[x_n, x_{n-1}] \\ \vdots \\ y[x_n, \dots, x_{n-(k-1)}] \end{pmatrix} = P_k \cdot D_k^t + \overrightarrow{O(H^k)}$$

donde P_k es la matriz $k \times k$

$$P_k = \begin{pmatrix} h_{1,0} & h_{1,1} & \cdots & h_{1,k-1} \\ 0 & h_{2,0} & \cdots & h_{2,k-2} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & h_{k,0} \end{pmatrix},$$

$$H = \max\{|H_n|, |H_{n-1}|, \dots, |H_{n-(k-1)}|\} \text{ y } \overrightarrow{O(H^k)} = (O(H^k), \dots, O(H^k)).$$

Por tanto, si llamamos S_k , a la inversa de la matriz P_k , (ver Vigo-Aguar [150]) obtenemos

$$D_k^t = S_k \cdot \begin{pmatrix} y[x_n] \\ y[x_n, x_{n-1}] \\ \vdots \\ y[x_n, \dots, x_{n-(k-1)}] \end{pmatrix} + \overrightarrow{O(H^k)}.$$

Obsérvese que es una aproximación de D_k de orden k .

La matriz S_k se puede expresar en forma de los polinomios simétricos elementales en las variables $H_n, \dots, H_{n-(k-1)}$,

$$S_k = \begin{pmatrix} s_{0,0} & -s_{1,1} & s_{2,2} & -s_{3,3} & \cdots & (-1)^{k-1} s_{(k-1),(k-1)} \\ 0 & s_{1,0} & -s_{2,1} & s_{3,2} & \cdots & (-1)^{k-2} s_{(k-1),(k-2)} \\ 0 & 0 & s_{2,0} & -s_{3,1} & \cdots & (-1)^{k-3} s_{(k-1),(k-3)} \\ 0 & 0 & 0 & s_{3,0} & \cdots & (-1)^{k-4} s_{(k-1),(k-4)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 & s_{(k-1),0} \end{pmatrix},$$

72 Métodos en paso variable y transición a métodos Runge-Kutta.

definiéndose el polinomio elemental de grado k en las variables z_1, \dots, z_n como

$$\begin{aligned} s_{n,0} &= 1, \\ s_{n,k} &= \sum_{1 < i_1 < i_2 < \dots < i_k < n} \prod_{m=1}^k z_{i_m}, \quad k > 0. \end{aligned} \quad (5.2.2)$$

De esta manera tenemos que

$$(\beta) \cdot (\nabla y_n)^t \approx (\beta) \cdot M_k \cdot S_k \cdot \begin{pmatrix} y[x_n] \\ y[x_n, x_{n-1}] \\ \vdots \\ y[x_n, \dots, x_{n-(k-1)}] \end{pmatrix}$$

con lo cual obtenemos una aproximación de los algoritmos implícitos adaptados de paso variable de tipo BDF

$$\begin{aligned} & (\beta) \cdot M_k \cdot S_k \cdot \begin{pmatrix} y[x_n] \\ y[x_n, x_{n-1}] \\ \vdots \\ y[x_n, \dots, x_{n-(k-1)}] \end{pmatrix} - \\ & - e^{\lambda h} (\beta) \cdot M_k \cdot S_k^* \cdot \begin{pmatrix} y[x_{n-1}] \\ y[x_{n-1}, x_{n-2}] \\ \vdots \\ y[x_{n-1}, \dots, x_{n-k}] \end{pmatrix} = h_n f_n, \end{aligned}$$

donde S_k^* es la matriz expresada en términos de los nuevos polinomios simétricos elementales en las variables $H_n^* = 0, H_{n-1}^* = x_{n-2} - x_{n-1}, \dots, H_{n-(k-1)}^* = x_{n-k} - x_{n-1}$.

Obsérvese que dichos métodos no tienen por qué integrar de forma exacta el problema

$$y'(x) - \lambda y(x) = p(x)$$

cuando la solución del problema $y(x)$ pertenece al subespacio generado por $\langle e^{\lambda x}, 1, x, \dots, x^k \rangle$, a pesar de ello la aproximación es buena y muy recomendable ante problemas que combinan zonas donde la solución varía rápidamente con zonas donde varíe más lentamente, y en el capítulo dedicado a

ejemplos numéricos veremos que los métodos en paso variable así logrados tendrán buenos resultados numéricos.

5.2.2. Ejemplos: los métodos de paso variable de 2, 3 y 6 pasos.

Por tanto de esta forma podemos construir el método de dos pasos implícito a partir de la fórmula y de los coeficientes dados en el capítulo de los métodos de orden exponencial 1:

$$(\beta_0^0, \beta_1^0) \cdot M_2 \cdot S_2 \cdot \left(\frac{y(x_{n+1})}{\frac{y(x_{n+1})-y(x_n)}{x_{n+1}-x_n}} \right) - e^{\lambda h} (\beta_0^0, \beta_1^0) \cdot M_2 \cdot S_2^* \cdot \left(\frac{y(x_n)}{\frac{y(x_n)-y(x_{n-1})}{x_n-x_{n-1}}} \right) = h_n f_{n+1},$$

donde

$$M_2 = \begin{pmatrix} 1 & 0 \\ 0 & h_n \end{pmatrix}, \quad S_2 = S_2^* = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

Así el método quedaría como

$$y_{n+1} = c_1 y_n + c_2 y_{n-1} + h_n c_0 f_{n+1},$$

donde los coeficientes dados de una forma explícita serían

$$c_0 = \frac{(-1 + e^{\lambda h_n})^2}{1 - e^{\lambda h_n} + \lambda(-1 + 2e^{\lambda h_n})h_n},$$

$$c_1 = -\frac{\lambda e^{2\lambda h_n} h_n^2 + h_{n-1} - e^{\lambda h_n} h_{n-1}}{(-1 + e^{\lambda h_n} + \lambda h_n - 2\lambda e^{\lambda h_n} h_n)h_{n-1}} - \frac{e^{\lambda h_n} h_n (1 - e^{\lambda h_n} + \lambda e^{\lambda h_n} h_{n-1})}{(-1 + e^{\lambda h_n} + \lambda h_n - 2\lambda e^{\lambda h_n} h_n)h_{n-1}},$$

y

$$c_2 = -\frac{e^{\lambda h_n} h_n (1 - e^{\lambda h_n} + \lambda e^{\lambda h_n} h_n)}{(1 - e^{\lambda h_n} + \lambda(-1 + 2e^{\lambda h_n})h_n)h_{n-1}}.$$

74 **Métodos en paso variable y transición a métodos Runge-Kutta.**

De forma muy similar se puede construir el método de tres pasos implícito

$$(\beta_0^0, \beta_1^0, \beta_2^0) \cdot M_3 \cdot S_3 \cdot \left(\begin{array}{c} y(x_{n+1}) \\ \frac{y(x_{n+1})-y(x_n)}{h_n} \\ \frac{h_{n-1}(y(x_{n+1})-y(x_n))-h_n(y(x_n)-y(x_{n-1}))}{h_n h_{n-1}} \end{array} \right) -$$

$$-e^{\lambda h}(\beta_0^0, \beta_1^0, \beta_3^0) \cdot M_3 \cdot S_3^* \cdot$$

$$\cdot \left(\begin{array}{c} y(x_n) \\ \frac{y(x_n)-y(x_{n-1})}{h_{n-1}} \\ \frac{h_{n-2}(y(x_n)-y(x_{n-1}))-h_{n-1}(y(x_{n-1})-y(x_{n-2}))}{h_{n-1} h_{n-2}} \end{array} \right) = h_n f_{n+1},$$

donde

$$M_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & h_n & -h_n^2 \\ 0 & 0 & 2h_n^2 \end{pmatrix}, \quad S_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & h_n \\ 0 & 0 & 1 \end{pmatrix}, \quad S_3^* = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & h_{n-1} \\ 0 & 0 & 1 \end{pmatrix}.$$

De esta forma el método sería

$$y_{n+1} = c_1 y_n + c_2 y_{n-1} + c_3 y_{n-2} + h_n c_0 f_{n+1},$$

siendo los coeficientes

$$c_0 = \frac{(-1 + e^{\lambda h_n})^3 (h_n + h_{n-1})}{C_H},$$

$$c_1 = \frac{H_{n-2}(H_{n-2}H_{n-2}^* + e^{h\lambda}(3h_n(h_{n-1} + h_{n-2}) - h_n^2))}{(h_{n-1} + h_{n-2})h_{n-1}C_H} +$$

$$+ \frac{(h_n + h_{n-1})(e^{h_n\lambda}(2h_{n-1}(h_{n-1} + h_{n-2})) + e^{3h_n\lambda}h_n(2h_n^2\lambda))}{(h_{n-1} + h_{n-2})h_{n-1}C_H} -$$

$$- \frac{H_{n-2}(e^{3h_n\lambda}h_n(-H_{n-2}^*(h_{n-1}\lambda - 1) + h_n(h_{n-1}\lambda + h_{n-2}\lambda - 3)))}{(h_{n-1} + h_{n-2})h_{n-1}C_H} +$$

$$+ \frac{(h_n + h_{n-1})e^{2h_n\lambda}(-h_{n-1}(h_{n-1} + h_{n-2}) - h_n(h_{n-1} + h_{n-2})(1 + h_{n-1}\lambda))}{h_{n-1}(h_{n-1} + h_{n-2})C_H} +$$

$$+ \frac{(h_n + h_{n-1})e^{2h_n\lambda}h_n^2(4 + h_{n-1}\lambda + h_{n-2}\lambda)}{(h_{n-1} + h_{n-2})h_{n-1}C_H},$$

siendo

$$H_{n-2} = -h_n - h_{n-1},$$

$$H_{n-2}^* = -h_{n-1} - h_{n-2},$$

$$c_2 = \frac{-h_n(-h_n h_{n-2} - e^{h_n \lambda}(h_n^2 + h_n(h_{n-1} - 3h_{n-2}) + h_{n-1}h_{n-2}))}{h_{n-2}h_{n-1}C_H} +$$

$$+ \frac{h_n e^{3h_n \lambda}(h_n + h_{n-1})(-h_{n-2} + 2h_n^2 \lambda + h_n(-3 + h_{n-2} \lambda))}{h_{n-2}h_{n-1}C_H} +$$

$$+ \frac{h_n e^{2h_n \lambda}(2h_{n-1}h_{n-2} + h_n^2(4 + h_{n-2} \lambda) - h_n(h_{n-2} + h_{n-1}(-4 + h_{n-2} \lambda)))}{h_{n-2}h_{n-1}C_H},$$

y

$$c_3 = \frac{(e^{h_n \lambda} h_n^2 (h_n + h_{n-1})(-1 + 4e^{h_n \lambda} + e^{2h_n \lambda}(-3 + 2h_n \lambda)))}{(h_{n-1} + h_{n-2})h_{n-2}C_H},$$

con

$$C_H = -(-1 + e^{h_n \lambda})^2 h_{n-1} + (1 - 3e^{h_n \lambda} + 4e^{2h_n \lambda})h_n^2 \lambda +$$

$$(1 - 3e^{h_n \lambda} + 2e^{2h_n \lambda})h_n(-2 + h_{n-1} \lambda).$$

De forma muy parecida se puede construir el resto de fórmulas implícitas en paso variable. Más adelante se mostrarán algunos resultados con el método de paso variable de 6 pasos, el cual se construye a partir de la fórmula

$$(\beta_0, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5) \cdot M_6 \cdot S_6 \cdot$$

$$\cdot \left(\begin{array}{c} y(x_{n+1}) \\ y[x_{n+1}, x_n] \\ y[x_{n+1}, x_n, x_{n-1}] \\ y[x_{n+1}, x_n, x_{n-1}, x_{n-2}] \\ y[x_{n+1}, x_n, x_{n-1}, x_{n-2}, x_{n-3}] \\ y[x_{n+1}, x_n, x_{n-1}, x_{n-2}, x_{n-3}, x_{n-4}] \end{array} \right) -$$

$$- e^{\lambda h} (\beta_0, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5) \cdot M_6 \cdot S_6^* \cdot$$

$$\cdot \left(\begin{array}{c} y(x_n) \\ y[x_n, x_{n-1}] \\ y[x_n, x_{n-1}, x_{n-2}] \\ y[x_n, x_{n-1}, x_{n-2}, x_{n-3}] \\ y[x_n, x_{n-1}, x_{n-2}, x_{n-3}, x_{n-4}] \\ y[x_n, x_{n-1}, x_{n-2}, x_{n-3}, x_{n-4}, x_{n-5}] \end{array} \right) = h_n f_{n+1},$$

76 *Métodos en paso variable y transición a métodos Runge-Kutta.*

donde

$$M_6 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & h_n & -h_n^2 & h_n^3 & -h_n^4 & h_n^5 \\ 0 & 0 & 2h_n^2 & -6h_n^3 & 14h_n^4 & -30h_n^5 \\ 0 & 0 & 0 & 6h_n^3 & -36h_n^4 & 150h_n^5 \\ 0 & 0 & 0 & 0 & 24h_n^4 & -240h_n^5 \\ 0 & 0 & 0 & 0 & 0 & 120h_n^5 \end{pmatrix},$$

$$S_6 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & h_n & h_n(h_n + h_{n-1}) & -s_{4,3} & s_{5,4} \\ 0 & 0 & 1 & 2h_n + h_{n-1} & s_{4,2} & -s_{5,3} \\ 0 & 0 & 0 & 1 & 3h_n + 2h_{n-1} + h_{n-2} & s_{5,2} \\ 0 & 0 & 0 & 0 & 1 & -s_{5,1} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

siendo

$$-s_{4,3} = h_n(h_n + h_{n-1})(h_n + h_{n-1} + h_{n-2}),$$

$$s_{5,4} = h_n(h_n + h_{n-1}) \dots (h_n + h_{n-1} + h_{n-2})(h_n + h_{n-1} + h_{n-2} + h_{n-3}),$$

$$s_{4,2} = h_n(h_n + h_{n-1}) + h_n(h_n + h_{n-1} + h_{n-2}) +$$

$$+ (h_n + h_{n-1})(h_n + h_{n-1} + h_{n-2}),$$

$$-s_{5,3} = h_n(h_n + h_{n-1})(h_n + h_{n-1} + h_{n-2}) +$$

$$+ h_n(h_n + h_{n-1})(h_n + h_{n-1} + h_{n-2} + h_{n-3}) +$$

$$+ h_n(h_n + h_{n-1} + h_{n-2})(h_n + h_{n-1} + h_{n-2} + h_{n-3}) +$$

$$+ (h_n + h_{n-1})(h_n + h_{n-1} + h_{n-2})(h_n + h_{n-1} + h_{n-2} + h_{n-3}),$$

$$s_{5,2} = h_n(h_n + h_{n-1}) + h_n(h_n + h_{n-1} + h_{n-2}) +$$

$$+ h_n(h_n + h_{n-1} + h_{n-2} + h_{n-3}) +$$

$$+ (h_n + h_{n-1})(h_n + h_{n-1} + h_{n-2}) +$$

$$+ (h_n + h_{n-1})(h_n + h_{n-1} + h_{n-2} + h_{n-3}) +$$

$$+ (h_n + h_{n-1} + h_{n-2})(h_n + h_{n-1} + h_{n-2} + h_{n-3}),$$

$$-s_{5,1} = 4h_n + 3h_{n-1} + 2h_{n-2} + h_{n-3},$$

y donde S_6^* se constuye de igual forma que S_6 , pero ahora en términos de los nuevos polinomios simétricos elementales en las variables $H_n^* = 0$, $H_{n-1}^* = x_{n-2} - x_{n-1} = -h_{n-1}$, \dots , $H_{n-5}^* = x_{n-6} - x_{n-1} = -(h_{n-1} + h_{n-2} + h_{n-3} + h_{n-4} + h_{n-5})$.

5.2.3. Métodos EF-2-k-1-r de paso variable.

Los métodos dados hasta ahora en este capítulo no tienen porqué integrar de forma exacta el PVI general que nos preocupa a nosotros, sin embargo, sí es posible construir métodos de paso variable que integren de forma exacta el problema (1.0.2) cuando $y(x)$ pertenece al subespacio generado por $\langle \vec{1}, e^{Ax}, \dots \rangle$. A continuación mostramos otra forma de obtener métodos en paso variable mucho más laboriosa, pero más precisa.

En los métodos de paso variable nos vamos a encontrar con bastantes dificultades para estudiar con detalle las propiedades de estabilidad de los diferentes métodos. De hecho pronto nos vamos a dar cuenta de que a mayor número de pasos más complicado será profundizar en las propiedades de estabilidad.

Para hacernos una buena idea vamos a mostrar cuál es el comportamiento de los EF-2-k-1-r explícito e implícito de paso variable.

Empezaremos construyendo los métodos que integren de forma exacta el problema cuando la solución esté generada por el subespacio $\langle \vec{1}, e^{Ax}, xe^{Ax} \rangle$, primero el explícito y luego el implícito.

Cuando el último y el penúltimo paso no sean iguales el método se obtiene directamente de resolver las ecuaciones que salen del problema

$$y'(x) = f(x, y(x))$$

78 **Métodos en paso variable y transición a métodos Runge-Kutta.**

cuando $y(x) = 1$, $y(x) = e^{\lambda x}$ y $y(x) = xe^{\lambda x}$, las cuales se pueden simplificar a las siguientes ecuaciones:

$$c_0 + c_1 + c_2 = 0,$$

$$c_0 e^{\lambda h_n} + c_1 + c_2 e^{-\lambda h_{n-1}} = \lambda h_n,$$

$$c_0 h_n e^{\lambda h_n} - (c_2 h_{n-1}) e^{\lambda h_{n-1}} = h_n,$$

donde h_n es la longitud $x_{n+1} - x_n$ y $h_{n-1} = x_n - x_{n-1}$.

De esta forma el método de paso variable se convierte en

$$c_0 y_{n+1} + c_1 y_n + c_2 y_{n-1} = h_n f(x_n, y_n), \quad (5.2.3)$$

pero siendo ahora los pesos

$$c_0 = \frac{h_n(-1 + e^{\lambda h_{n-1}} - \lambda h_{n-1})}{e^{\lambda h_n}(-1 + e^{\lambda h_{n-1}})h_n + h_{n-1} - e^{\lambda h_n}h_{n-1}},$$

$$c_1 = \frac{h_n(1 - e^{\lambda(h_n+h_{n-1})}) + \lambda(e^{\lambda(h_n+h_{n-1})}h_n + h_{n-1})}{e^{\lambda h_n}(-1 + e^{\lambda h_{n-1}})h_n + h_{n-1} - e^{\lambda h_n}h_{n-1}},$$

$$c_2 = \frac{-e^{\lambda h_{n-1}}h_n(1 - e^{\lambda h_n} + \lambda e^{\lambda h_n}h_n)}{e^{\lambda h_n}(-1 + e^{\lambda h_{n-1}})h_n + h_{n-1} - e^{\lambda h_n}h_{n-1}}.$$

Si por el contrario el problema es vectorial, los coeficientes serían

$$c_0 = \frac{h_n(-Id + e^{Ah_{n-1}} - Ah_{n-1})}{e^{Ah_n}(-Id + e^{Ah_{n-1}})h_n + h_{n-1}Id - e^{Ah_n}h_{n-1}}, \quad (5.2.4)$$

$$c_1 = \frac{h_n(Id - e^{A(h_n+h_{n-1})}) + A(e^{A(h_n+h_{n-1})}h_n + h_{n-1}Id)}{e^{Ah_n}(-Id + e^{Ah_{n-1}})h_n + h_{n-1}Id - e^{Ah_n}h_{n-1}} \quad (5.2.5)$$

$$c_2 = \frac{-e^{Ah_{n-1}}h_n(Id - e^{Ah_n} + Ae^{Ah_n}h_n)}{e^{Ah_n}(-Id + e^{Ah_{n-1}})h_n + h_{n-1}Id - e^{Ah_n}h_{n-1}}. \quad (5.2.6)$$

De forma similar se obtiene el método implícito, pero ahora nuestro método es el que se obtiene de resolver las ecuaciones

$$c_0^* + c_1^* + c_2^* = 0,$$

$$c_0^* e^{\lambda h_n} + c_1^* + c_2^* e^{-\lambda h_{n-1}} = \lambda e^{\lambda h_n} h_n,$$

$$c_0^* h_n e^{\lambda h_n} - c_2^* h_{n-1} e^{\lambda h_{n-1}} = h_n e^{\lambda h_n} (1 + \lambda h_n),$$

por tanto los coeficientes ahora son, en el caso del problema escalar

$$c_0^* = \frac{e^{\lambda h_n} h_n (-1 + e^{\lambda h_{n-1}} (1 + \lambda h_n) - \lambda (h_n + h_{n-1}))}{e^{\lambda (h_n + h_{n-1})} h_n + h_{n-1} - e^{\lambda h_n} (h_n + h_{n-1})},$$

$$c_1^* = \frac{e^{\lambda h_n} h_n (1 - e^{\lambda (h_n + h_{n-1})} + \lambda (h_n + h_{n-1}))}{e^{\lambda (h_n + h_{n-1})} h_n + h_{n-1} - e^{\lambda h_n} (h_n + h_{n-1})},$$

$$c_2^* = \frac{-e^{\lambda (h_n + h_{n-1})} h_n (1 - e^{\lambda h_n} + \lambda h_n)}{e^{\lambda (h_n + h_{n-1})} h_n + h_{n-1} - e^{\lambda h_n} (h_n + h_{n-1})},$$

o bien

$$c_0^* = \frac{e^{Ah_n} h_n (-Id + e^{Ah_{n-1}} (Id + Ah_n) - A(h_n + h_{n-1}))}{e^{A(h_n + h_{n-1})} h_n + h_{n-1} Id - e^{Ah_n} (h_n + h_{n-1})},$$

$$c_1^* = \frac{e^{Ah_n} h_n (Id - e^{A(h_n + h_{n-1})} + A(h_n + h_{n-1}))}{e^{A(h_n + h_{n-1})} h_n + h_{n-1} Id - e^{Ah_n} (h_n + h_{n-1})},$$

$$c_2^* = \frac{-e^{A(h_n + h_{n-1})} h_n (Id - e^{Ah_n} + Ah_n)}{e^{A(h_n + h_{n-1})} h_n + h_{n-1} Id - e^{Ah_n} (h_n + h_{n-1})},$$

si el problema es vectorial.

Claramente el mismo sistema puede ser utilizado para conseguir los diferentes métodos de paso variable que integran exactamente el problema

$$y'(x) = f(x, y(x))$$

cuando $y(x)$ pertenece al subespacio generado por las funciones $\langle 1, x, \dots, x^{r-1}, e^{\lambda x}, \dots, x^{k-1} e^{\lambda x} \rangle$.

Dados ya los coeficientes, podemos estudiar la estabilidad de los métodos de paso variable.

80 Métodos en paso variable y transición a métodos Runge-Kutta.

En primer lugar es preciso recordar que tanto el explícito como el implícito eran estables en todo C^- , tanto los algoritmos adaptados como los *exponential fitting*.

Por tanto si consideramos un método multipaso general de longitud de paso variable:

$$y_{n+k} + \sum_{j=0}^{k-1} \alpha_{jn} y_{n+j} = h_{n+k-s} \sum_{j=0}^k \beta_{jn} f_{n+j}, \quad (5.2.7)$$

donde los coeficientes α_{jn} y β_{jn} dependen de los cocientes $w_i = h_i/h_{i-1}$, $i = n+1, \dots, n+k-1$, podemos aplicar el siguiente teorema (ver [63]):

Teorema 5.2.1. *Un método del tipo (5.2.7) cumpliendo:*

- a) *El método es consistente.*
- b) *Los coeficientes $\alpha_{jn} = \alpha_j(w_{n+1}, \dots, w_{n+k-1})$, son continuos en un entorno del $(1, \dots, 1)$.*
- c) *La correspondiente fórmula de paso fijo es fuertemente estable, es decir, todas las raíces de*

$$\zeta^k + \sum_{j=0}^{k-1} \alpha_j(1, \dots, 1) \zeta^j = 0$$

se encuentran en el disco unitario abierto, con excepción de la raíz 1.

Entonces, con esas condiciones, existen números reales, σ y ρ ($\sigma < 1 < \rho$), tales que el método es estable si

$$\sigma < \frac{h_n}{h_{n-1}} < \rho.$$

Dado que los métodos son consistentes de orden 2 como veremos a continuación, que además los coeficientes α_{jn} son continuos siempre que $h_n, h_{n-1} \neq 0$ y los valores propios del jacobiano sean distintos de 0 y que los métodos de paso fijo son fuertemente estables, existen constantes que hacen que los

métodos variables son estables. Sin embargo dichas constantes van a depender del λ elegido.

Si escribimos los métodos de paso variable y obtenemos las raíces de los polinomios asociados a los métodos, es fácil observar que los métodos variables son estables siempre que h_n/h_{n-1} verifiquen que para toda λ valor propio de A

$$\left| \frac{e^{\lambda h_{n-1}}(1 + e^{\lambda h_n}(-1 + \lambda h_n))}{-1 + e^{\lambda h_{n-1}} - \lambda h_{n-1}} \right| < 1$$

para el método explícito y

$$\left| \frac{e^{\lambda h_{n-1}}(-1 + e^{\lambda h_n} - \lambda h_n)}{-1 + e^{\lambda h_{n-1}}(1 + \lambda h_n) - \lambda(h_n + h_{n-1})} \right| < 1$$

para el implícito.

Obsérvese aquí que las raíces en realidad sólo dependen de dos parámetros λh_n y λh_{n-1} , pero $\lambda \in C$ y $h_n, h_{n-1} \in R^+$, por lo que no es posible dibujar las zonas de 0-estabilidad para todos los valores de λ, h_n, h_{n-1} .

Es preciso, bien fijar el valor de λh_{n-1} o bien el de λ , para obtener regiones de 0-estabilidad.

Por ejemplo, si nos restringimos al caso en que $\lambda \in R^-$, en la figura 5.1 podemos ver que cuanto más se de que $\lambda h_{n-1} \ll 0$ más podremos aumentar el paso sin perder estabilidad.

Esto es el explícito de paso variable, por ejemplo, es estable si se verifica alguna de estas condiciones (caso en que los valores propios de A sean todos reales).

a) Si para todo λ_i , valor propio de A , se verifica que $\lambda_i h_{n-1} \leq -1$, entonces basta con que $5 > h_n/h_{n-1} > 0$.

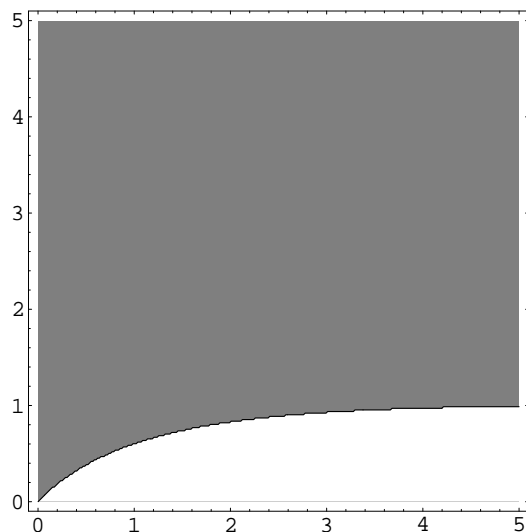


Figura 5.1: Regiones de estabilidad (en gris) del método explícito EF-2-k-1-r de paso variable para parámetros $\lambda h_n \in R^-$ y $\lambda h_{n-1} \in R^-$. Los ejes horizontal y vertical representan respectivamente $\|\lambda h_n\|$ y $\|\lambda h_{n-1}\|$.

b) Si para todo λ_i , valor propio de A , se verifica que $0 > \lambda_i h_{n-1}$, entonces basta con que $1 \geq h_n/h_{n-1} > 0$, es decir, basta que el paso se mantenga o decrezca.

Hagamos lo mismo con el implícito, consideramos $\lambda h_n, \in R^-, \lambda h_{n-1}, \in R^-$ y comprobamos cuanto puede aumentar el paso. Los resultados se observan en la figura 5.2.

Ahora podemos afirmar que el implícito de paso variable es aún mejor en este apartado que el explícito. Por ejemplo, es estable si se verifican alguna de estas condiciones (siendo los valores propios de A reales):

a) Si para todo λ_i , valor propio de A , se verifica que $\lambda_i h_{n-1} \leq -1$, entonces basta que $10 > h_n/h_{n-1} > 0$.

b) Si para todo λ_i , valor propio de A , se verifica que $10^{-7} > \lambda_i h_{n-1}$, entonces basta que $2,4 \geq h_n/h_{n-1} > 0$, es decir, el paso se puede incluso duplicar. De

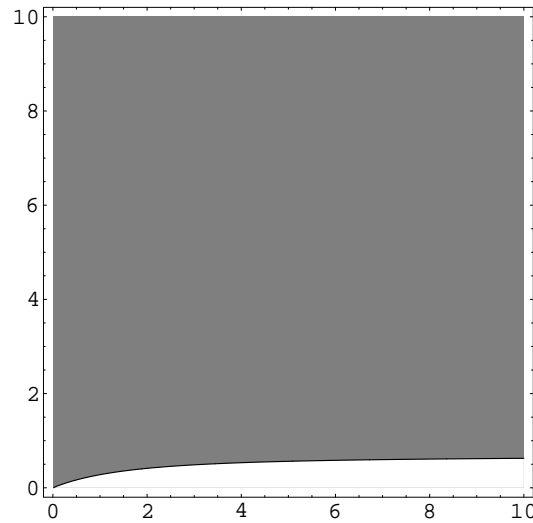


Figura 5.2: Regiones de estabilidad (en gris) del método implícito EF-2-k-1-r de paso variable para parámetros $\lambda h_n \in R^-$ y $\lambda h_{n-1} \in R^-$. Los ejes horizontal y vertical representan respectivamente $\|\lambda h_n\|$ y $\|\lambda h_{n-1}\|$.

hecho cuanto menor es $\lambda_i h_{n-1}$ (o mayor en valor absoluto, pues consideraremos $h > 0$, y en general $\lambda_i < 0$) la cota superior de h_n/h_{n-1} aumenta.

Por tanto ya sabemos cuándo los métodos son 0-estables, si son consistentes ya sabremos para qué valores de λh_n y λh_{n-1} los métodos serán convergentes. Y efectivamente ambos métodos lo son.

Teorema 5.2.2. *El método explícito (5.2.3) cuyos coeficientes son los que aparecen en (5.2.4), (5.2.5) y (5.2.6), de dos pasos con paso variable es convergente de orden 2. Su error de truncado local se puede expresar como:*

$$\frac{h_n^2 h_{n-1} (A^2 y'(x) - 2A y''(x) + y'''(x))}{6}.$$

Teorema 5.2.3. *El método implícito de dos pasos con paso variable es convergente de orden 2. Su error de truncado local se puede expresar como:*

$$\frac{h_n^2 (h_{n-1} + h_n) (A^2 y'(x) - 2A y''(x) + y'''(x))}{6}.$$

84 Métodos en paso variable y transición a métodos Runge-Kutta.

Nos queda estudiar la estabilidad absoluta, lo cual no es tarea sencilla ya que si consideramos el problema test (2.5.6), las raíces del polinomio característico asociado al método explícito son:

$$r_1 = \frac{b_1 + \sqrt{b_2 - b_1^2}}{b_3},$$

$$r_2 = \frac{b_1 - \sqrt{b_2 - b_1^2}}{b_3},$$

donde

$$b_1 = -1 - \lambda h_{n-1} + h_{n-1}\mu - e^{h_n}(h_n + h_{n-1})\mu + e^{\lambda(h_{n-1}+h_n)}(1 - \lambda h_n + h_n\mu),$$

$$b_2 = 4e^{\lambda h_{n-1}}(1 + e^{\lambda h_n}(-1 + \lambda h_n))(-1 + e^{\lambda h_{n-1}} - \lambda h_{n-1}),$$

$$b_3 = 2(-1 + e^{\lambda h_{n-1}} - \lambda h_{n-1}).$$

Mientras que las del implícito son:

$$r_1^* = \frac{b_1^* + \sqrt{b_2^* - b_3^*}}{2b_4^*},$$

$$r_2^* = \frac{b_1^* - \sqrt{b_2^* - b_3^*}}{2b_4^*},$$

donde

$$b_1^* = -e^{\lambda(2h_n+h_{n-1})} + e^{\lambda h_n}(1 + \lambda(h_n + h_{n-1})),$$

$$b_2^* = e^{2\lambda h_n}(1 - e^{\lambda(h_n+h_{n-1})} + \lambda(h_n + h_{n-1}))^2,$$

$$b_3^* = 4e^{\lambda(h_n+h_{n-1})}(-1 + e^{\lambda h_n} - \lambda h_n)b_4^*,$$

$$b_4^* = (-h_{n-1}\mu + e^{\lambda(h_n+h_{n-1})}(1 + \lambda h_n - h_n\mu) + e^{\lambda h_n}(-1 - \lambda(h_n + h_{n-1}) + h_n\mu + h_{n-1}\mu)).$$

Obsérvese que ahora las regiones de estabilidad absoluta dependerán de $\lambda, \mu \in C^-$, $h_n, h_{n-1} \in R^+$ (o si se prefiere de $\lambda h_n, \lambda h_{n-1} \in C^-$ y de $\mu h_n, \mu h_{n-1} \in C^-$). Para dar una buena imagen de cómo van a cambiar las regiones de estabilidad del método de paso variable se necesitarían multitud de dibujos.

Además no tiene sentido fijar los valores de $\lambda h_n, \lambda h_{n-1} \in C^-$, ó $\mu h_n, \mu h_{n-1} \in C^-$, ya que son métodos de paso variable por lo que es previsible que de una iteración a otra cambien tanto h_n como h_{n-1} .

Por supuesto al estudiar métodos de mayor número de pasos el estudio de los métodos de paso variable se complica aún más, especialmente la estabilidad absoluta como ya se puede deducir de los resultados obtenidos de los métodos de dos pasos con paso variable. Por esa circunstancia nos hemos detenido en estudiar con detalle únicamente los EF-2-k-1-r, que nos permiten hacer un estudio más completo que otros métodos con mayor número de pasos.

5.3. Métodos *exponential fitting* de tipo BDF-Runge-Kutta.

Junto con los métodos BDF otro tipo de métodos muy utilizado para integrar problemas *stiff* son los Runge-Kutta (implícitos). Incluso, recientemente, se han utilizado métodos BDF clásicos en paso variable para construir métodos Runge-Kutta con resultados y propiedades muy interesantes incluso para métodos de orden alto (ver [117] y [152], en este último se construyen métodos de orden hasta 12 con propiedades de estabilidad muy apreciables). En esta sección vamos a hacer un trabajo similar, pero utilizando los BDF de tipo *exponential fitting* que hemos venido construyendo. Debido a que el estudio de como hallar los coeficientes y las regiones de estabilidad es algo más complicado nos vamos a quedar en un orden más bajo.

Dado que [117] y [152] son trabajos todavía no publicados, aquí se hará un resumen breve de la idea original para los métodos clásicos antes de incorporar

la idea a los métodos de tipo *exponential fitting*.

5.3.1. Construcción de métodos BDF-Runge-Kutta.

Partamos, como ya hemos hecho del PVI vectorial

$$y'(x) = g(x, y(x)), \quad x \in [x_0, x_f], \quad y(x_0) = y_0, \quad (5.3.1)$$

donde g satisface todos los requerimientos necesarios para la existencia de una única solución.

Supongamos que hemos obtenido una buena aproximación y_n de $y(x)$ en el punto $x = x_n$ y que estamos ahora interesados en la aproximación en el punto $x = x_{n+1} = x_n + h$.

Si reescribimos la solución $y(x)$ en el intervalo $[x_n, x_{n+1}]$ en términos de una variable intermedia α

$$x = x_n + \frac{h}{2}(1 + \alpha),$$

con $\alpha \in [-1, 1]$, entonces podemos aproximar la solución de la EDO como

$$y(x) = y\left(x_n + \frac{h}{2}(1 + \alpha)\right) = \bar{y}(\alpha)$$

por una suma finita de la forma (ver [52])

$$\bar{y}(\alpha) = \sum_{k=0}^s a_k T_k(\alpha), \quad (5.3.2)$$

donde $T_k(\alpha)$ es el polinomio de Chebyshev de primera clase y grado k , definidos por la relación trigonométrica

$$T_k(x) = \cos(k\theta), \quad x = \cos(\theta), \quad \theta \in [0, \pi],$$

mientras que los a_k se obtienen a partir de la fórmula

$$a_k = \frac{2}{s} \sum_{j=0}^s{}'' y(x_n + \xi_j h) T_k(\cos(\theta_j)), \quad (5.3.3)$$

con

$$\theta_j = (s - j) \frac{\pi}{s}, \quad \xi_j = \frac{1 + \cos(\theta_j)}{2}, \quad j = 0, \dots, s, \quad (5.3.4)$$

y donde '' simboliza que el primero y el último término del sumatorio se dividen por 2.

Por tanto, a partir de (5.3.2), si sustituimos en (5.3.1) se obtiene

$$\begin{aligned} g(x, y(x)) &= \frac{dy(x)}{dx} = \frac{d\bar{y}(\alpha)}{d\alpha} \frac{d\alpha}{dx} \\ &= \frac{d}{d\alpha} \left(\sum_{k=0}^s{}'' a_k T_k(\alpha) \right) \frac{2}{h} = \frac{2}{h} \sum_{k=0}^s{}'' a_k T_k'(\alpha). \end{aligned} \quad (5.3.5)$$

Si introducimos los valores dados en (5.3.3) en la fórmula (5.3.5) obtenemos

$$hg(x, \bar{y}(\alpha)) = \sum_{j=0}^s{}'' \left(\frac{4}{s} \sum_{k=0}^s{}'' T_k(\cos(\theta_j)) T_k'(\alpha) \right) y(x_n + \xi_j h). \quad (5.3.6)$$

Y evaluando la fórmula (5.3.6) en los puntos de Chebyshev-Gauss-Lobatto

$$\alpha_i = \cos(\theta_i) \quad i = 1, \dots, n, \quad (5.3.7)$$

obtenemos un sistema implícito de ecuaciones implícitas,

$$hg(x_i, y_i) = \sum_{j=0}^s m_{i,j} y_j \quad i = 1, \dots, s,$$

donde los coeficientes $m_{i,j}$ vienen dados por

$$m_{i,j} = \begin{cases} \frac{4}{s} \sum_{k=0}^s{}'' T_k(\alpha_j) T_k'(\alpha_i), & j \neq 0, s, \\ \frac{2}{s} \sum_{k=0}^s{}'' T_k(\alpha_j) T_k'(\alpha_i), & j = 0, s. \end{cases}$$

Resolviendo el sistema se obtiene el valor para $x = x_{n+1}$.

Si por el contrario resolvemos el sistema para una función g genérica en los puntos intermedios ξ_j con $j = 1, \dots, s$, se obtienen los métodos Runge-Kutta implícitos.

5.3.2. Construcción de métodos BDF-Runge-Kutta de tipo *exponential fitting*.

Obviamente se puede realizar el mismo trabajo, pero en vez de utilizar métodos de tipo BDF clásico se pueden usar de tipo *exponential fitting*. La idea, por tanto, es tomar los puntos de Chebyshev-Gauss-Lobatto, pero, si en [152] se exigía que los BDF integraran de forma exacta el problema cuando la solución fuera un polinomio (BDF clásicos), ahora (se considerará en primer lugar el problema escalar) vamos a pedir que los BDF integren de forma exacta el problema cuando la solución pertenezca al subespacio generado por $\langle e^{\lambda x}, 1, \dots, x^\delta \rangle$ (es decir utilizaremos básicamente los métodos EF-I-k- δ de paso variable, pero evaluados en puntos intermedios de $[x_n, x_{n+1}]$).

Por tanto los coeficientes $v_{i,j}$ los vamos a obtener de imponer que

$$hg(x_n + \xi_i h, y(x_n + \xi_i h)) = \sum_{j=0}^s v_{i,s-j} y(x_n + \xi_j h), \quad i = 1, \dots, s,$$

integre de forma exacta el problema de valor inicial escalar cuando $y(x) = 1$, $y(x) = x$, ..., $y(x) = x^{s-1}$, $y(x) = e^{\lambda x}$, con $i = 1, \dots, s$.

De esta forma si $s = 2$, puesto que los puntos de Chebyshev están igualmente espaciados: $\xi_0 = 0$, $\xi_1 = 1/2$, $\xi_2 = 1$, los $v_{1,j}$ son los coeficientes del método explícito EF-I-k-2, sin más que cambiar h por $h/2$; mientras que los $v_{2,j}$ son los coeficientes del método implícito EF-I-k-2, cambiando h por $h/2$.

Por contra si $s = 3$, los puntos de Chebyshev no están igualmente espaciados: $\xi_0 = 0$, $\xi_1 = 1/4$, $\xi_2 = 3/4$, $\xi_3 = 1$ y es preciso calcular los $v_{i,j}$ a partir de los métodos en paso variable.

De esta forma los coeficientes serían:

$$v_{1,0} = \frac{8 - 2e^{3\lambda h/4} + 3e^{\lambda h/4}(-2 + \lambda h)}{3(-1 + e^{\lambda h/4})^3(1 + e^{\lambda h/4})},$$

$$v_{1,1} = \frac{2(-9 + e^{\lambda h} + e^{\lambda h/4}(8 - 3\lambda h))}{3(-1 + e^{\lambda h/4})^3(1 + e^{\lambda h/4})},$$

$$v_{1,2} = \frac{2(5 - 8e^{3\lambda h/4} + 3e^{\lambda h} + 3\lambda e^{\lambda h/4}h)}{3(-1 + e^{\lambda h/4})^3(1 + e^{\lambda h/4})},$$

$$v_{1,3} = -\frac{e^{\lambda h/4}(10 - 18e^{\lambda h/2} + 8e^{3\lambda h/4} + 3\lambda h)}{3(-1 + e^{\lambda h/4})^3(1 + e^{\lambda h/4})},$$

$$v_{2,0} = \frac{-8 + 18e^{\lambda h/4} + e^{3\lambda h/4}(-10 + 3\lambda h)}{3(-1 + e^{\lambda h/4})^3(1 + e^{\lambda h/4})},$$

$$v_{2,1} = \frac{2(3 - 8e^{\lambda h/4} + 5e^{\lambda h} - 3\lambda e^{3\lambda h/4}h)}{3(-1 + e^{\lambda h/4})^3(1 + e^{\lambda h/4})},$$

$$v_{2,2} = \frac{2 - 18e^{\lambda h} + e^{3\lambda h/4}(16 + 6\lambda h)}{3(-1 + e^{\lambda h/4})^3(1 + e^{\lambda h/4})},$$

$$v_{2,3} = \frac{-2e^{\lambda h/4} + 8e^{\lambda h} - 3e^{3\lambda h/4}(2 + \lambda h)}{3(-1 + e^{\lambda h/4})^3(1 + e^{\lambda h/4})},$$

$$v_{3,0} = \frac{-16 + 30e^{\lambda h/4} - 14e^{3\lambda h/4} + 3\lambda e^{\lambda h}}{3(-1 + e^{\lambda h/4})^3(1 + e^{\lambda h/4})},$$

$$v_{3,1} = \frac{18 - 32e^{\lambda h/4} + e^{\lambda h}(14 - 6\lambda h)}{3(-1 + e^{\lambda h/4})^3(1 + e^{\lambda h/4})},$$

$$v_{3,2} = \frac{-2 + 32e^{3\lambda h/4} + 6e^{\lambda h}(-5 + \lambda h)}{3(-1 + e^{\lambda h/4})^3(1 + e^{\lambda h/4})},$$

$$v_{3,3} = \frac{2e^{\lambda h/4} - 18e^{3\lambda h/4} + e^{\lambda h}(16 - 3\lambda h)}{3(-1 + e^{\lambda h/4})^3(1 + e^{\lambda h/4})}.$$

90 **Métodos en paso variable y transición a métodos Runge-Kutta.**

Por último daremos los coeficientes del método para $s = 4$. En este caso $\xi_0 = 0$, $\xi_1 = (2 - \sqrt{2})/4$, $\xi_2 = 1/2$, $\xi_3 = (2 + \sqrt{2})/4$, $\xi_4 = 1$, entonces si denominamos $p_i = e^{\xi_i \lambda h}$, los $v_{i,j}$ se obtienen de:

$$\begin{aligned} v_{1,4} &= -v_{1,0} - v_{1,1} - v_{1,2} - v_{1,3}, \\ v_{1,3} &= \frac{-4 + 4v_{1,0} + (2 + \sqrt{2})v_{1,1} + 2v_{1,2}}{-2 + \sqrt{2}}, \\ v_{1,2} &= -\sqrt{2}(-2 + \sqrt{2} + (2 + \sqrt{2})v_{1,0} + (1 + \sqrt{2})v_{1,1}), \\ v_{1,1} &= -\frac{2(-1 + \sqrt{2} + (2 + \sqrt{2})v_{1,0})}{2 + \sqrt{2}}, \\ v_{1,0} &= \frac{4 - 4(-1 + \sqrt{2})p_2 + (-4 + 3\sqrt{2})p_3 + (-4 + \sqrt{2} + \lambda h)p_1}{1 + 2p_2 + p_4 - 2p_1 - 2p_3}, \end{aligned}$$

$$\begin{aligned} v_{2,4} &= -v_{2,0} - v_{2,1} - v_{2,2} - v_{2,3}, \\ v_{2,3} &= -\frac{4 - 4v_{2,0} - (2 + \sqrt{2})v_{2,1} - 2v_{2,2}}{-2 + \sqrt{2}}, \\ v_{2,2} &= -\sqrt{2}(-2 - \sqrt{2} + (2 + \sqrt{2})v_{2,0} + (1 + \sqrt{2})v_{2,1}), \\ v_{2,1} &= -\frac{2(-\sqrt{2} + (2 + \sqrt{2})v_{2,0})}{2 + \sqrt{2}}, \\ v_{2,0} &= -\frac{4 - 4(1 + \sqrt{2})p_1 + 4(-1 + \sqrt{2})p_3 + (4 - 2\lambda h)p_2}{2 + 4p_2 + 2p_4 - 4p_1 - 4p_3}, \end{aligned}$$

$$\begin{aligned} v_{3,4} &= -v_{3,0} - v_{3,1} - v_{3,2} - v_{3,3}, \\ v_{3,3} &= -\frac{4 - 4v_{3,0} - (2 + \sqrt{2})v_{3,1} - 2v_{3,2}}{-2 + \sqrt{2}}, \\ v_{3,2} &= -\sqrt{2}(-2 - 3\sqrt{2} + (2 + \sqrt{2})v_{3,0} + (1 + \sqrt{2})v_{3,1}), \\ v_{3,1} &= -\frac{2(-5 - 3\sqrt{2} + (2 + \sqrt{2})v_{3,0})}{2 + \sqrt{2}}, \\ v_{3,0} &= -\frac{-4 - 4(1 + \sqrt{2})p_2 + (4 + 3\sqrt{2})p_1 + (4\sqrt{2} - \lambda h)p_3}{1 + 2p_2 + p_4 - 2p_1 - 2p_3}, \end{aligned}$$

$$\begin{aligned}
v_{4,4} &= -v_{4,0} - v_{4,1} - v_{4,2} - v_{4,3}, \\
v_{4,3} &= -\frac{4 - 4v_{4,0} - (2 + \sqrt{2})v_{4,1} - 2v_{4,2}}{-2 + \sqrt{2}}, \\
v_{4,2} &= -\sqrt{2}(-6 - \sqrt{2} + (2 + \sqrt{2})v_{4,0} + (1 + \sqrt{2})v_{4,1}), \\
v_{4,1} &= -\frac{2(-10 - 3\sqrt{2} + (2 + \sqrt{2})v_{4,0})}{2 + \sqrt{2}}, \\
v_{4,0} &= -\frac{-10 - 18p_2 + 2(7 + 2\sqrt{2})p_1 - 2(-7 + 2\sqrt{2})p_3 - \lambda hp_4}{1 + p_4 + 2p_2 - 2p_1 - 2p_3}.
\end{aligned}$$

Así tendríamos un sistema de ecuaciones implícitas y dada cada $g(x, y(x))$ a partir de cada y_n podríamos obtener y_{n+1} resolviendo el sistema.

La otra posibilidad es construir los métodos BDF-Runge-Kutta, para ello se resuelve el sistema quedando $y_{n+\xi_j}$ ($j = 1, \dots, s$), en función de $g(x_n + \xi_j h, y(x_n + \xi_j h))$.

Si denotamos los métodos de la forma

$$\begin{aligned}
y_{n+1} &= y_n + h \sum_{i=1}^s b_i g(x_n + c_i h, u_i), \\
u_i &= y_n + h \sum_{j=1}^s a_{ij} g(x_n + c_j h, u_j),
\end{aligned}$$

y los colocamos en caja de forma tradicional:

$$\begin{array}{c|ccc}
c_1 & a_{11} & \dots & a_{1s} \\
c_2 & a_{21} & \dots & a_{2s} \\
\vdots & \vdots & \ddots & \vdots \\
c_s & a_{s1} & \dots & a_{ss} \\
\hline
& b_1 & \dots & b_s
\end{array}$$

tenemos que si $s = 2$ el Runge-Kutta viene dado por

$$\begin{array}{c|cc}
\frac{1}{2} & \frac{2-2e^{-\lambda h/2}-\lambda h e^{\lambda h/2}}{2\lambda h(1-e^{\lambda h/2})} & \frac{1}{2} - a_{11} \\
1 & \frac{1+e^{\lambda h}(-1+\lambda h)}{\lambda h(-e^{\lambda h/2}+e^{\lambda h})} & 1 - a_{21} \\
\hline
& a_{21} & a_{22}
\end{array}$$

92 Métodos en paso variable y transición a métodos Runge-Kutta.

mientras que si $s = 3$ los coeficientes del Runge-Kutta son

$\frac{1}{4}$	$\frac{8-8e^{-\lambda h/4}-7\lambda h e^{\lambda h/2}+5\lambda h e^{3\lambda h/4}}{8\lambda h-24\lambda h e^{\lambda h/2}+16\lambda h e^{3\lambda h/4}}$	$\frac{-24+24e^{-\lambda h/4}+7\lambda h-\lambda h e^{3\lambda h/4}}{8\lambda h-24\lambda h e^{\lambda h/2}+16\lambda h e^{3\lambda h/4}}$	$\frac{1}{4} - a_{11} - a_{12}$
$\frac{3}{4}$	$\frac{-8+9\lambda h e^{\lambda h}+e^{3\lambda h/4}(8-15\lambda h)}{8\lambda h(e^{\lambda h/4}-3e^{3\lambda h/4}+2e^{\lambda h})}$	$\frac{3(8-8e^{3\lambda h/4}+5\lambda h e^{\lambda h/4}+\lambda h e^{\lambda h})}{8\lambda h(e^{\lambda h/4}-3e^{3\lambda h/4}+2e^{\lambda h})}$	$\frac{3}{4} - a_{21} - a_{22}$
1	$\frac{-1-2\lambda h e^{3\lambda h/4}+e^{\lambda h}(1+\lambda h)}{\lambda h(e^{\lambda h/4}-3e^{3\lambda h/4}+2e^{\lambda h})}$	$\frac{3+2\lambda h e^{\lambda h/4}+e^{\lambda h}(-3+\lambda h)}{\lambda h(e^{\lambda h/4}-3e^{3\lambda h/4}+2e^{\lambda h})}$	$1 - a_{31} - a_{32}$
	a_{31}	a_{32}	a_{33}

Y en general se construyen así el resto de métodos *exponential fitting* BDF-Runge-Kutta de orden más alto.

Hasta ahora sólo hemos trabajado pensando en problemas escalares, para problemas vectoriales tan sólo hay que cambiar λ por la matriz A , 1 por la matriz identidad y considerar $\frac{B}{C} = BC^{-1}$.

5.3.3. Propiedades de los métodos BDF-Runge-Kutta de tipo *exponential fitting*.

Dado que en definitiva los métodos se pueden escribir como Runge-Kutta por tanto son siempre 0-estables, ya que el polinomio característico asociado a un Runge-Kutta es $x - 1$.

Para estudiar el error de truncado local de los métodos procederemos como en [152] escribiendo los métodos de la forma

$$V Y_{n+1} = R Y_n + h F_{n+1},$$

donde

$$Y_{n+1} = (y(x_n + \xi_1 h), \dots, y(x_n + \xi_s h))^t,$$

$$Y_n = (y(x_{n-1} + \xi_1 h), \dots, y(x_{n-1} + \xi_s h))^t,$$

$$F_{n+1} = (f(x_n + \xi_1 h, y(x_n + \xi_1 h)), \dots, f(x_n + \xi_s h, y(x_n + \xi_s h)))^t,$$

siendo

$$V = \begin{pmatrix} v_{1,s-1} & \dots & v_{1,1} & v_{1,0} \\ v_{2,s-1} & \dots & v_{2,1} & v_{2,0} \\ \vdots & & \vdots & \vdots \\ v_{s,s-1} & \dots & v_{s,1} & v_{s,0} \end{pmatrix},$$

$$R = \begin{pmatrix} 0 & \dots & 0 & -v_{1,s} \\ 0 & \dots & 0 & -v_{2,s} \\ \vdots & & \vdots & \vdots \\ 0 & \dots & 0 & -v_{n,s} \end{pmatrix},$$

siendo los $v_{i,j}$ los dados anteriormente (y que dependen del número de etapas s).

De esta forma podemos calcular $T_s = \| V Y_{n+1} - R Y_n - h F_{n+1} \|_\infty$.

Por ejemplo para $s = 2$ podemos aprovechar los cálculos de la expresión del error de los dos métodos (explícito e implícito) EF-I-k-2, pero teniendo en cuenta que ahora h queda dividida por 2, y quedarnos con la cota superior (la del implícito). De esta forma obtenemos que

$$T_2 = \frac{(\lambda y''(x) - y'''(x))h^3}{12}.$$

Para $s = 3$ utilizando cálculo simbólico alcanzamos que

$$T_3 = \frac{(\lambda y^{(3)}(x) - y^{(4)}(x))h^4}{128}.$$

Mientras que si $s = 4$ tenemos que

$$T_4 = \frac{(\lambda y^{(4)}(x) - y^{(5)}(x))h^5}{1920}.$$

Para calcular las regiones de estabilidad absoluta de los métodos consideraremos el famoso test de Dahlquist, pero ahora, si hemos introducido como parámetro en el método $h\lambda$, el test es

$$y' = \mu y, \quad y_0 = 1, \quad z = h\mu,$$

94 **Métodos en paso variable y transición a métodos Runge-Kutta.**

y calcularemos el conjunto

$$S = \{z \in C; |R(z)| \leq 1\},$$

llamado dominio de estabilidad del método, y donde $R(z)$ como proponen Hairer y Wanner en [64] se puede calcular como

$$R(z) = 1 + zb^t(Id - zA)^{-1}\mathbf{1},$$

donde $b^t = (b_1, \dots, b_s)$, $A = (a_{ij})_{i,j=1}^s$ y $\mathbf{1} = (1, \dots, 1)^t$, o bien a partir de la fórmula (ver [64], [128] ó [135], son equivalentes como se demuestra en la proposición 3.2, p. 41 de [64])

$$R(z) = \frac{\det(Id - zA + z\mathbf{1}b^t)}{\det(Id - zA)}.$$

De esta forma si llamamos $u = h\lambda$, para $s = 2$ obtenemos la función de estabilidad

$$R(z) = \frac{2(-e^{u/2} + e^u)z + u(2e^{u/2} - e^u(2 + z))}{ue^{u/2}(2 + e^{u/2}(-2 + z) - 2z) - z(e^u(-2 + z) - 2e^{u/2}(-1 + z) + z)},$$

que cuando $u \rightarrow 0$, $R(z) \rightarrow \frac{4+z}{4-3z+z^2}$ (que resulta ser la función de estabilidad del método de orden 2 en [152]).

Mientras que si $s = 3$ la función de estabilidad es

$$R(z) = \frac{e^{u/4}(C_1 + C_2)}{C_3 + C_4},$$

con

$$C_1 = 2z(-8 - z - 8e^{3u/4}(2 + z) + 3e^{u/2}(8 + 3z)),$$

$$C_2 = u(2(8 + z) - 6e^{u/2}(8 + 3z) + e^{3u/4}(32 + 16z + 3z^2)),$$

$$C_3 = ue^{u/4}(16 - 14z + 6z^2 - 6e^{u/2}(8 - 5z + z^2) + e^{3u/4}(32 - 16z + 3z^2)),$$

$$C_4 = z(3z^2 + e^u(-32 + 16z - 3z^2) + 6e^{3u/4}(8 - 5z + z^2) - 2e^{u/4}(8 - 7z + 3z^2)),$$

que cuando $u \rightarrow 0$, $R(z) \rightarrow \frac{-96-32z-3z^2}{-96+64z-19z^2+3z^3}$, como se preveía del método de orden 3 en [152].

De forma análoga se construyen las funciones de estabilidad para $s > 3$.

De nuevo hay que tener en cuenta que las regiones de estabilidad $\mathfrak{R}_A \in C \times C$ y por tanto tenemos que fijar uno de los parámetros (como antes, fijaremos $u = \lambda h$). En el apéndice I mostramos una serie de regiones de estabilidad absoluta de los métodos BDF-Runge-Kutta de orden 2, 3 y 4.

96 *Métodos en paso variable y transición a métodos Runge-Kutta.*

Capítulo 6

Cálculo del parámetro y métodos con dos parámetros.

6.1. Introducción.

Una duda que es fácil plantearse es cuál es el método de obtención del parámetro Ah (ó de los parámetros) que vamos a utilizar en los métodos. Esta es probablemente la mayor dificultad con la que nos hemos encontrado, y obviamente es un tema muy importante. La segunda sección de este capítulo se centrará en dar referencias de cómo ha evolucionado el tema y después haremos nuevas aportaciones más en esta cuestión ya que resulta fundamental para conseguir buenos resultados numéricos.

Dado que es muy abundante el número de ecuaciones parciales (EDP's) que al ser discretizadas en una de las variables se convierten en ecuaciones diferenciales ordinarias de tipo *stiff* y que los nuevos métodos van a aportar importantes mejoras, se va a profundizar en el cálculo del error de truncado global en la sección tercera de este capítulo.

En la cuarta sección se presentará la forma de obtener métodos que integren de forma exacta el PVI cuando la solución exacta pertenece a un espacio

donde esté incluido el subespacio generado por e^{Ax}, e^{Bx} , de esta forma se puede favorecer la integración de problemas con alta variabilidad o con problemas que den complicaciones con la estabilidad, pero cuanto mayor sea el número de parámetros más complicado será hacer un estudio detallado de las propiedades del método.

6.2. Métodos de obtención del parámetro introducido en el algoritmo.

La idea de utilizar métodos que integren de forma exacta soluciones de tipo exponencial o polinomios por funciones trigonométricas no es ni mucho menos nueva (por ejemplo en [10], [39], [57] ó en [60] ya se observa la idea). En [19], Cash explica que si $x'(t) = f(t, x)$ y la matriz $\frac{df(t, p(t))}{dx}$, donde $p(t)$ es la solución del problema de valor inicial, varía lentamente entonces las componentes de la solución $p(t)$ dependen de las exponenciales de los valores propios del jacobiano de la función. En ese caso se proponía utilizar métodos exponential fitting con problemas que tuvieran un único valor propio dominante. Es decir el parámetro que utilizaba en esas fórmulas de tipo *exponential fitting* era un valor propio del jacobiano multiplicado por la longitud de paso.

No es la primera vez que se utilizó este valor como parámetro (ver [77] ó [90]), pero limitaba bastante la clase de problemas donde se podía utilizar este tipo de algoritmos con buenos resultados (ver [19] y [154]). Los coeficientes dependen de un único número real y ya se vislumbra la idea de utilizar el jacobiano para calcular el parámetro del método.

La idea de utilizar la función exponencial del jacobiano evaluado en el valor

obtenido en el último paso para la integración numérica es muy corriente desde mediados de los 80 (ver, por ejemplo, [43], [55], [56], [85], [101], [106], [122], ó [132]). Sin embargo, Lambert (ver [87]) ó Dekker y Verwer (ver [38]) por ejemplo, han mostrado numerosos ejemplos numéricos en los cuales los autovalores del jacobiano son positivos (y por tanto son muy complejos para los métodos tradicionales) y a pesar de todo las soluciones del problema no dependen de ninguna exponencial positiva. En ese caso se pueden provocar problemas de estabilidad a métodos *exponential fitting* si se toma como parámetro del método el jacobiano de la función, aquí se demostrará que es posible obtener buenos resultados tomando otros parámetros.

Recientemente (ver [72], [73] ó [74]) han surgido otros métodos para obtener el parámetro del método (aquí lo llaman frecuencia) basados en anular la parte principal del error de truncado local. Como veremos más adelante en el caso del método Euler exponencial obtendríamos el jacobiano como parámetro del método, sin embargo, con métodos de orden más elevado el parámetro hallado puede ser muy diferente.

En esta sección pretendemos dar formas adicionales de hallar el parámetro del método, pero es un tema bastante complejo y en el que presumiblemente se seguirá escribiendo mucho en los próximos años.

La vía de encontrar el parámetro antes de saber cómo se comporta la solución del problema de valor inicial es una pregunta muy importante aunque veremos que hallado uno “adecuado ” al problema los métodos de tipo *exponential fitting* aquí presentados son un instrumento muy potente.

Por supuesto el parámetro ideal depende del PVI y se puede entender que la mejor forma de encontrar el Ah dependerá del tipo de problema que ten-

gamos.

En primer lugar vamos a considerar una serie de problemas del tipo $y'(x) = B(x)y(x)$, el primero lo podemos encontrar en [87]):

$$\begin{cases} y_1'(x) = y_2(x), \\ y_2'(x) = \frac{1-x}{x}y_1(x) - \frac{1-2x}{x}y_2(x), \\ y_1(0'1) = e^{-0'1}, \quad y_2(0'1) = -e^{-0'1}, \end{cases} \quad (6.2.1)$$

cuya solución es

$$y_1(x) = e^{-x}, \quad y_2(x) = -e^{-x}.$$

Los valores propios del jacobiano $B(x)$ son $\lambda_1 = -1$ y $\lambda_2 = \frac{1-x}{x}$, por tanto en $x \in (0, 1)$, $\lambda_2 > 0$, este, por tanto, es uno de los muchos ejemplos en los que el jacobiano no informa adecuadamente del tipo de solución del problema que tenemos, ya que, mientras que las soluciones de los PVI que surgen de perturbar nuestro ejemplo numérico no dependen de exponenciales positivas, uno de los valores propios del jacobiano sí que es positivo.

En el capítulo dedicado a ejemplos numéricos hemos integrado dicho problema en el intervalo $[0'1, 3]$ y como mostraremos la mayoría de los métodos utilizados tienen numerosos problemas para dar una solución adecuada en el punto $x = 3$.

Sin embargo, sí es posible encontrar algoritmos numéricos que den una buena solución, por ejemplo si consideramos el método explícito Euler exponencial:

$$Ah(e^{Ah} - Id)^{-1}y_{n+1} - Ah(e^{Ah} - Id)^{-1}y_n = hf_n,$$

nos encontramos con que los dos primeros términos del error de truncado local son $\frac{-y''(x)+Ay'(x)}{2}h^2 + \frac{A^2y'(x)-3Ay''(x)+2y'''(x)}{12}h^3 + O(h^4)$.

Por tanto si consideramos

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}, \quad (6.2.2)$$

y procuramos resolver $(y''(x) - Ay'(x))|_{(x_0, y_0)} = 0$, esto es $(B'(x) + (B(x) - A).B(x)).y_0 = 0$, siendo $(y_0 = y(0')$ y $x = 0')$, obtenemos que $a = -1 + b$ y $c = 1 + d$, estos mismos parámetros ya nos permiten obtener que $(A^2y'(x) - 3Ay''(x) + 2y'''(x))|_{(x_0, y_0)} = 0$. Entonces ya podemos pensar que estos son unos buenos parámetros y de hecho si elegimos b y d de forma que los valores propios de A sean los dos negativos, obtendremos buenos resultados numéricos, como se puede comprobar en el capítulo dedicado a los ejemplos numéricos.

Estudiemos otro ejemplo de la misma clase $y'(x) = B(x)y(x)$ y sacado de la misma referencia [87].

$$\begin{cases} y_1'(x) = y_2(x), \\ y_2'(x) = \frac{\cos(x) - \sin(x)}{2 + \cos(x) + \sin(x)}y_1(x) - \frac{2(1 + \sin(x))}{2 + \cos(x) + \sin(x)}y_2(x), \\ y_1(0) = 2, \quad y_2(0) = 1, \end{cases} \quad (6.2.3)$$

cuya solución es

$$y_1(x) = 2 + \sin(x), \quad y_2(x) = \cos(x).$$

Los valores propios de $B(x)$ son ahora $\lambda_1 = -1$ y $\lambda_2 = \frac{\cos(x) - \sin(x)}{2 + \cos(x) + \sin(x)}$, esto significa que hay muchos momentos en $[0, 100]$ (el intervalo donde vamos a probar a integrar el problema) donde $\lambda_2 > 0$. Sin embargo, de nuevo sucede que las soluciones de los problemas perturbados permiten que haya métodos numéricos con buenos resultados.

Estos son ejemplos de cómo el jacobiano no siempre es útil para calcular el parámetro del método.

Si en el problema (6.2.3) tomamos $x = 0$ y $y_0 = (2, 1)^t$ y resolvemos $(y''(x) - Ay'(x))|_{(x_0, y_0)} = (B'(x) + (B(x) - A).B(x)).y_0 = 0$ (donde A vuelve a tener la forma de (6.2.2)), llegamos a que $c = -1$, $a = 0$, con dichos valores resolvemos

$(A^2y'(x) - 3Ay''(x) + 2y'''(x)) |_{(x_0, y_0)} = 0$ obteniendo $b = 1$, $d = 0$. Con estos valores veremos en el siguiente capítulo que se obtienen unos resultados óptimos con el método de Euler exponencial en un problema donde la mayor parte de los integradores tienen grandes problemas.

Enunciemos de una forma más genérica la forma en la que obtendremos los parámetros con los diferentes métodos de orden exponencial 1:

i) Los dos primeros términos del error de truncado local del método Euler exponencial explícito son $\frac{-y''(x)+Ay'(x)}{2}h^2 + \frac{A^2y'(x)-3Ay''(x)+2y'''(x)}{12}h^3$, pero si $y''(x)-Ay'(x) = 0$, entonces $A^2y'(x)-3Ay''(x)+2y'''(x) = Ay''(x)-3Ay''(x)+2y'''(x) = 2(y'''(x) - Ay''(x))$.

Por tanto si el sistema de EDO's tiene dimensión 2 nuestras condiciones serían $(y''(x) - Ay'(x)) |_{(x_n, y_n)} = 0$ y $(y'''(x) - Ay''(x)) |_{(x_n, y_n)} = 0$.

ii) Los dos primeros términos del error de truncado local del método Euler exponencial implícito son $\frac{y''(x)-Ay'(x)}{2}h^2 + \frac{A^2y'(x)+3Ay''(x)-4y'''(x)}{12}h^3$, pero si $y''(x)-Ay'(x) = 0$, entonces $A^2y'(x) + 3Ay''(x) - 4y'''(x) = Ay''(x) + 3Ay''(x) - 4y'''(x) = -4(y'''(x) - Ay''(x))$.

Por tanto si el sistema de EDO's tiene dimensión 2 nuestras condiciones serían $(y''(x) - Ay'(x)) |_{(x_n, y_n)} = 0$ y $(y'''(x) - Ay''(x)) |_{(x_n, y_n)} = 0$.

iii) Los dos primeros términos del error de truncado local del método EF-I-k-2 implícito son $\frac{-y'''(x)+Ay''(x)}{3}h^3 + \frac{A^2y''(x)-y^4(x)}{12}h^4$, pero si $y'''(x) - Ay''(x) = 0$, entonces $A^2y''(x) - y^4(x) = Ay'''(x) - y^4(x)$.

Por tanto si el sistema de EDO's tiene dimensión 2 nuestras condiciones serían $(y'''(x) - Ay''(x)) |_{(x_n, y_n)} = 0$ y $(y^4(x) - Ay'''(x)) |_{(x_n, y_n)} = 0$.

iv) Los dos primeros términos del error de truncado local del método EF-I-k-3 implícito son $\frac{-y^4(x)+Ay'''(x)}{4}h^4 + \frac{3A^2y'''(x)-5Ay^4(x)+2y^5(x)}{40}h^5$, pero si $y^4(x) -$

$Ay'''(x) = 0$, entonces $3A^2y'''(x) - 5Ay^4(x) + 2y^5(x) = 3Ay'''(x) - 5Ay^4(x) + 2y^5(x) = 2(y^5(x) - Ay^4(x))$.

Por tanto si el sistema de EDO's tiene dimensión 2 nuestras condiciones serían $(y^4(x) - Ay'''(x))|_{(x_n, y_n)} = 0$ y $(y^5(x) - Ay^4(x))|_{(x_n, y_n)} = 0$.

v) Los dos primeros términos del error de truncado local del método EF-I-k-4 implícito son $\frac{-y^5(x) + Ay^4(x)}{5}h^5 + \frac{A^2y^4(x) - 3Ay^5(x) + 2y^6(x)}{15}h^6$.

Por tanto si el sistema de EDO's tiene dimensión 2 nuestras condiciones serían $(y^5(x) - Ay^4(x))|_{(x_n, y_n)} = 0$ y $(y^6(x) - Ay^5(x))|_{(x_n, y_n)} = 0$.

vi) Los dos primeros términos del error de truncado local del método EF-I-k-5 implícito son $\frac{-y^6(x) + Ay^5(x)}{6}h^6 + \frac{5A^2y^5(x) - 21Ay^6(x) + 16y^7(x)}{84}h^7$.

Por tanto si el sistema de EDO's tiene dimensión 2 nuestras condiciones serían $(y^6(x) - Ay^5(x))|_{(x_n, y_n)} = 0$ y $(y^7(x) - Ay^6(x))|_{(x_n, y_n)} = 0$.

vii) Los dos primeros términos del error de truncado local del método EF-I-k-6 implícito son $\frac{-y^7(x) + Ay^6(x)}{7}h^7 + \frac{3A^2y^6(x) - 16Ay^7(x) + 13y^8(x)}{56}h^8$.

Por tanto si el sistema de EDO's tiene dimensión 2 nuestras condiciones serían $(y^7(x) - Ay^6(x))|_{(x_n, y_n)} = 0$ y $(y^8(x) - Ay^7(x))|_{(x_n, y_n)} = 0$.

Por supuesto si la dimensión del sistema de EDO's no fuera 2 sino superior tomaríamos tantos términos del error de truncado local como la dimensión del sistema de EDO's y calcularíamos el parámetro del método de forma que anulara cada uno de los términos del error de truncado local consiguiendo así una mayor precisión.

Las condiciones que hay que imponer en un problema dado para calcular un buen parámetro elegido para el método EF-I-k-r (con $r \leq 6$) salen del siguiente teorema.

Teorema 6.2.1. *Sea nuestro problema de valor inicial elegido del tipo de la ecuación (1.0.3) de dimensión $m \times m$ y el método elegido para integrarla es el EF-I-k-r de orden y número de pasos r .*

Un buen parámetro para el método es Ah , donde A es la matriz de dimensión $m \times m$ que resulta de resolver en cada paso n el sistema de ecuaciones $(y^{r+1})(x) - Ay^r(x) |_{(x_{n-1}, y_{n-1})} = 0, \dots, (y^{r+m})(x) - Ay^{r+m-1}(x) |_{(x_{n-1}, y_{n-1})} = 0$.

Demostración: Como ya vimos en el capítulo dedicado a los métodos de orden exponencial 1, los primeros $m + 1$ términos del error de truncado local provienen de la expresión

$$\nabla_P \sum_{j=r}^{r+m} \beta_j^0 \nabla^j y(x_n) + O(h^{r+m+2}),$$

dado que $\nabla_P \nabla^j s(x) = \nabla^j \nabla_P s(x)$ (pues $\nabla_P \nabla s(x) = \nabla \nabla_P s(x)$), obtenemos que el error de truncado local es

$$\sum_{j=r}^{r+m} \beta_j^0 \nabla^j \nabla_P y(x_n) + O(h^{r+m+2}),$$

pero ya sabemos desarrollar $\nabla^j \nabla_P y(x_n)$ con $j = r, \dots, r + m - 1$:

$$\begin{aligned} \nabla^j \nabla_P y(x_n) &= \nabla^j y(x_n) - e^{Ah} \nabla^j y(x_{n-1}) = \\ &c_{j,j} h^j (y^{(j)}(x_n) - e^{Ah} y^{(j)}(x_{n-1})) + c_{j,j+1} h^{j+1} (y^{(j+1)}(x_n) - e^{Ah} y^{(j+1)}(x_{n-1})) + \dots, \end{aligned}$$

de esta forma el error de truncado local es

$$\begin{aligned} \nabla_P \sum_{j=r}^{r+m} \beta_j^0 \nabla^j y(x_n) + O(h^{r+m+2}) &= C_r h^r (y^r(x_n) - e^{Ah} y^r(x_{n-1})) + \dots + \\ &+ C_{r+m-1} h^{r+m-1} (y^{r+m-1}(x_n) - e^{Ah} y^{r+m-1}(x_{n-1})) + \\ &+ (\nabla_P \beta_j^{r+m} \nabla^{r+m} y(x_n) + C_{r+m} h^{r+m} (y^{r+m}(x_n) - e^{Ah} y^{r+m}(x_{n-1}))) + \end{aligned}$$

$$+O(h^{r+m+2}).$$

Si pudiéramos elegir A una matriz de dimensión $m \times m$, de forma que $(y^{r+1}(x) - Ay^r(x)) = 0, \dots, (y^{r+m}(x) - Ay^{r+m-1}(x)) = 0, \forall x \in [x_{n-1}, x_n]$, tendríamos que en un intervalo que incluye el intervalo $[x_{n-1}, x_n]$, se verificaría que $y^j(x) = e^{A(x-x_{n-1})}y^j(x_{n-1})$ para $j = r, \dots, r+m-1$.

Con lo que $y^j(x) - e^{Ah}y^j(x_{n-1}) = 0$, para $j = r, \dots, r+m-1$, de esta forma el error de truncado local sería

$$(\nabla_P \beta_j^{r+m} \nabla^{r+m} y(x_n) + C_{r+m} h^{r+m} (y^{r+m}(x_n) - e^{Ah} y^{r+m}(x_{n-1}))) + O(h^{r+m+2}),$$

y cuando $h \rightarrow 0$, ya hemos comentado que

$$y^{r+m}(x_n) - e^{Ah} y^{r+m}(x_{n-1}) \rightarrow h (y^{r+m+1}(x_n) - Ay^{r+m}(x_n)) + O(h^2).$$

Con lo que si pudiéramos elegir A una matriz de dimensión $m \times m$, de forma que $(y^{r+1}(x) - Ay^r(x)) = 0, \dots, (y^{r+m}(x) - Ay^{r+m-1}(x)) = 0, \forall x \in [x_{n-1}, x_n]$, tendríamos que nuestro error de truncado local sería un $O(h^{r+m+1})$, claramente no podemos imponer esa condición $\forall x \in [x_{n-1}, x_n]$ (ya que sólo tenemos el valor en x_{n-1} , y de una forma aproximada). Por tanto lo que haremos será imponer que A sea la matriz de dimensión $m \times m$ que resulta de resolver en cada paso el sistema de ecuaciones $(y^{r+1}(x) - Ay^r(x))|_{(x_{n-1}, y_{n-1})} = 0, \dots, (y^{r+m}(x) - Ay^{r+m-1}(x))|_{(x_{n-1}, y_{n-1})} = 0$. \square

Este es un procedimiento que se ha mostrado altamente efectivo en muchos de los problemas de valor inicial donde la EDO es del tipo $y'(x) = B(x)y(x)$, incluso cuando alguno de los valores propios del jacobiano $B(x)$ es positivo, pero las soluciones no contienen exponenciales positivas.

Debemos reseñar que en los dos ejemplos antes mencionados se tomó un único parámetro Ah en todo el intervalo de integración, eso era posible porque éste

apenas variaba en cada paso. Sin embargo, no siempre es posible y debemos ser cuidadosos con el intervalo donde podemos elegir la misma matriz A . En muchos de los casos es necesario comprobar si debemos cambiar, o no, el parámetro después de una serie de pasos.

i) Una conclusión que podemos sacar a partir de los ejemplos es la relación entre la solución del problema de valor inicial y los valores propios del parámetro elegido: en el primer ejemplo numérico las soluciones del PVI eran $\alpha_1 e^{-x} + \alpha_2$ y uno de los valores propios de A era $\lambda_1 = -1$, mientras que en el segundo ejemplo las soluciones del PVI eran $\alpha_1 \sin(x) + \alpha_2 \cos(x) + \alpha_3$ y por tanto los valores propios de un buen parámetro A eran $\lambda_1 = i$, $\lambda_2 = -i$.

Es decir, a posteriori una vez se conoce la solución del PVI es posible dar un buen parámetro para el método. Sin embargo nos enfrentamos a la integración del problema, por tanto no podemos suponer conocida la solución del PVI, es preciso dar un procedimiento previo para el cálculo de un buen parámetro.

ii) Conviene recordar que hasta el momento hemos tratado dicha pregunta con los métodos de orden exponencial 1. Con otros métodos de orden exponencial mayor que 1, claramente no se verifican las mismas condiciones si imponemos que los primeros términos del error de truncado local deben anularse. Es más habría que solucionar ecuaciones cuadráticas, o incluso de mayor orden, dependiendo de la dimensión del sistema y del orden exponencial del método, razón por la que se complica incluso saber si es posible conseguir solución al sistema y por tanto tener una única matriz A , parámetro del método.

Pero el método EF- γ -k- δ -r debemos recordar es el que integra de forma exacta el problema (2.1.1) cuando $y(x)$ (la solución del PVI) pertenece al subespacio generado por $\langle 1, \dots, x^{\delta-1}, e^{\lambda x}, \dots, x^{\gamma-1} e^{\lambda x} \rangle$. Por tanto el método integra de forma exacta el subespacio generado por $\langle 1, \dots, x^{\delta-1}, e^{\lambda x} \rangle$, es decir, un buen parámetro será el obtenido para el método EF-I-k- δ .

Estas dos últimas apreciaciones (las hemos denotado con i) y ii)) nos van a permitir hallar otra forma de obtener “buenos” parámetros:

Vamos a partir del caso vectorial más sencillo, supongamos que tenemos una EDO de dimensión 2 y que tenemos una idea bastante general del comportamiento de la solución del PVI, con esto queremos indicar que conocemos si las componentes de la solución dependen en gran medida de funciones trigonométricas (senos y cosenos), en cuyo caso vamos a elegir una matriz del tipo

$$A = \begin{pmatrix} 0 & \lambda_1 \\ \lambda_2 & 0 \end{pmatrix},$$

(con $\lambda_1 \lambda_2 < 0$), o bien dependen de exponenciales negativas, en cuyo caso tomaremos una matriz del tipo

$$A = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$$

(con $\lambda_1 < 0, \lambda_2 < 0$).

Un ejemplo de esta clase de PVI es el ejemplo 1 que aparece en [62]:

$$\begin{cases} y_1'(x) = -1002y_1(x) + 1000y_2^2(x), \\ y_2'(x) = y_1(x) - y_2(x)(1 + y_2(x)), \\ y_1(0) = 1, \quad y_2(0) = 1, \end{cases} \quad (6.2.4)$$

cuya solución es

$$y_1(x) = e^{-2x}, \quad y_2(x) = e^{-x}.$$

Vamos a partir del hecho de que no conocemos de la solución exacta del PVI, pero sí sabemos que las componentes dependen de exponenciales negativas más polinomios, por tanto el parámetro será del tipo

$$A = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$$

En numerosas publicaciones se hace referencia a los llamados métodos de las series de Taylor (ver, por ejemplo [63]), que como la mayoría de los métodos explícitos no son apropiados para integrar problemas *stiff* (ver [79]), sin embargo ciertas modificaciones se han propuesto para adaptarse a estos problemas (ver [8], [34], [75] ó [81]).

En [62] se propone como obtener de forma rápida que cerca de $x = 0$, $y_1(x) \approx 1 - 2x + 2x^2 - \frac{4}{3}x^3 + \frac{2}{3}x^4$, mientras que $y_2(x) \approx 1 - x + \frac{1}{2}x^2 - \frac{1}{6}x^3 + \frac{1}{24}x^4$.

Supongamos que queremos utilizar un método de orden algebraico 3, en ese caso básicamente buscamos λ_1, λ_2 , tales que $y_1(x) \approx \kappa_{1,0} + \kappa_{1,1}x + \kappa_{1,2}x^2 + \sigma_1 e^{\lambda_1 x}$, $y_2(x) \approx \kappa_{2,0} + \kappa_{2,1}x + \kappa_{2,2}x^2 + \sigma_2 e^{\lambda_2 x}$.

Por tanto, utilizando el desarrollo de Taylor de la función exponencial, nuestras condiciones en la primera variable son:

$$y_1(x) \approx 1 - 2x + 2x^2 - \frac{4}{3}x^3 + \frac{2}{3}x^4 = \kappa_{1,0} + \kappa_{1,1}x + \kappa_{1,2}x^2 + \sigma_1 \left(1 + \lambda_1 x + \frac{\lambda_1^2}{2}x^2 + \frac{\lambda_1^3}{6}x^3 + \frac{\lambda_1^4}{24}x^4 \right),$$

y dado que $\kappa_{1,i}$ son valores reales cualesquiera, estamos imponiendo que

$$-8 = \sigma_1 \lambda_1^3, \quad 16 = \sigma_1 \lambda_1^4,$$

y resolviendo obtenemos que $\lambda_1 = -2$. De igual forma de las condiciones de la segunda variable se obtiene que $\lambda_2 = -1$. Evidentemente con este

método y parámetro se integra de forma exacta el PVI (6.2.4), pero es más, con el mismo parámetro otros métodos (como el Euler exponencial explícito) también son exactos.

Para estudiar lo que sucede en el caso de que la solución del PVI dependa fundamentalmente de funciones trigonométricas tomaremos el ejemplo 2 de [143]:

$$y''(x) + 10^8(y(x) - \cos(x))^3 = -\cos(x), \quad y(0) = 1, \quad y'(0) = 0,$$

que nosotros transformamos en

$$\begin{cases} y_1'(x) = y_2(x), \\ y_2'(x) = -10^8(y_1(x) - \cos(x))^3 - \cos(x), \\ y_1(0) = 1, \quad y_2(0) = 0, \end{cases} \quad (6.2.5)$$

y cuya solución es

$$y_1(x) = \cos(x), \quad y_2(x) = -\sin(x).$$

En este caso supongamos que vamos a utilizar un método de orden algebraico 2, entonces el primer paso al igual que anteriormente es calcular el desarrollo en series de Taylor de cada una de las componentes, en este caso hasta un orden 3 (uno más del orden algebraico del método que vamos a utilizar, de esta forma coinciden el número de ecuaciones y el número de incógnitas del sistema que vamos a resolver). Se puede comprobar fácilmente que obtenemos (lógicamente) que $y_1(x) \approx 1 - \frac{x^2}{2}$, $y_2(x) \approx -x + \frac{x^3}{6}$, mientras que si decimos que nuestro parámetro es del tipo

$$A = \begin{pmatrix} 0 & \lambda_1 \\ \lambda_2 & 0 \end{pmatrix},$$

(con $\lambda_1\lambda_2 < 0$), entonces suponemos que las soluciones son $y_1(x) \approx \kappa_{1,0} + \kappa_{1,1}x + \sigma_1 \cos(\sqrt{-\lambda_1\lambda_2}x) + \sigma_2 \sqrt{\frac{-\lambda_1}{\lambda_2}} \sin(\sqrt{-\lambda_1\lambda_2}x)$, $y_2(x) \approx \kappa_{2,0} + \kappa_{2,1}x + \sigma_2 \cos(\sqrt{-\lambda_1\lambda_2}x) - \sigma_1 \sqrt{\frac{-\lambda_2}{\lambda_1}} \sin(\sqrt{-\lambda_1\lambda_2}x)$.

Haciendo el desarrollo en series de Taylor de las funciones trigonométricas llegamos a que nuestras ecuaciones a resolver son

$$\sigma_1 \lambda_1 \lambda_2 = -1,$$

$$\sigma_2 \sqrt{\lambda_1^4 \lambda_2^2} = 0,$$

$$\sigma_2 \lambda_1 \lambda_2 = 0,$$

$$\sigma_1 \sqrt{\lambda_2^4 \lambda_1^2} = 1.$$

Rápidamente se concluye que $\sigma_2 = 0$ (primera y tercera ecuación) y como $y_1'(x) = y_2(x)$, entonces la única posibilidad es que $\lambda_1 = 1$, $\sigma_1 = 1$, $\lambda_2 = -1$. De nuevo este parámetro es óptimo para los distintos métodos propuestos en este trabajo.

Lógicamente estas dos formas de obtener el parámetro del método son muy eficaces cuando el paso h es relativamente pequeño, se podría pensar que estas vías de obtención de Ah no van a ser muy útiles cuando la longitud de paso es mayor. Por ese motivo hemos decidido integrar el problema conocido como E5 (el nombre se debe a Enright, Hull y Lindeberg, [47]), inicialmente el problema se debe a una reacción química estudiada por Datta en 1967 y describe una reacción química con 6 sustancias, posteriormente (ver [1]) se redujo al problema

$$\begin{pmatrix} y_1'(x) \\ y_2'(x) \\ y_3'(x) \\ y_4'(x) \end{pmatrix} = \begin{pmatrix} -Ay_1(x) - By_1(x)y_3(x) \\ A - MCy_2(x)y_3(x) \\ Ay_1(x) - By_1(x)y_3(x) - MCy_2(x)y_3(x) + Cy_4(x) \\ By_1(x)y_3(x) - Cy_4(x) \end{pmatrix}, \quad (6.2.6)$$

siendo $A = 7,89 \times 10^{-10}$, $B = 1,1 \times 10^7$, $C = 1,13 \times 10^3$, $M = 10^6$, con condiciones iniciales $(1,76 \times 10^{-3}, 0, 0, 0)$. Se verifica además que $y_2(x) -$

$y_3(x) - y_4(x) = 0$, con lo que se puede reducir a un sistema de tres ecuaciones diferenciales.

Este conocido PVI se suele resolver en el intervalo $[0, 1000]$, sin embargo, Alexander (1997) descubrió que la solución posee interesantes propiedades en un intervalo mucho mayor. Hairer y Wanner (ver [64]) proponen diferentes valores finales $t = 10, 10^3, 10^5, 10^7, \dots, 10^{13}$. En el capítulo dedicado a ejemplos numéricos se dan los resultados en un punto intermedio ($t = 10^7$) ya que se han utilizado métodos de paso fijo para integrar este problema, por lo que era necesario un coste muy alto para integrar hasta $t = 10^{13}$.

Dado que un dibujo aproximado de la solución la podemos encontrar por ejemplo en

<http://www.pitagora.dm.uniba.it/~testset/report/e5.pdf>

podemos intuir que la solución dependerá de exponenciales negativas más polinomios. Y utilizando la forma de obtener el parámetro ya descrita podemos llegar a buenos resultados, como veremos en el próximo capítulo, incluso aunque utilicemos longitudes de paso grandes (de hecho necesitaremos utilizar pasos $h > 1$, para que el coste computacional en el intervalo $[0, 10^7]$ no sea demasiado elevado).

No son estos los únicos procedimientos para obtener el parámetro del método que se han utilizado en este trabajo.

Otro tipo de ecuación muy general donde la mayoría de los métodos clásicos pueden tener problemas es

$$y''(x) + \lambda^2 y(x) + \epsilon F(y(x), y'(x)) = 0, \quad \lambda \ll 0. \quad (6.2.7)$$

En [9] se muestra una forma en la que se puede obtener la ecuación linealizada equivalente al problema no lineal:

$$y''(x) + \lambda \left(\frac{s'}{s} \right) y(x) + (\lambda^2 + \lambda\phi') y(x) = 0, \quad (6.2.8)$$

donde

$$s' = -\frac{\epsilon}{2\lambda} s_0,$$

$$\phi' = \frac{\epsilon}{2s\lambda} s'_0$$

siendo s_0, s'_0 los coeficientes obtenidos de las series de Fourier:

$$F(s \sin(\xi), s\lambda \cos(\xi)) \cos(\xi) = \frac{s_0}{2} + \sum_{n=1}^{\infty} (s_n \cos(n\xi) + r_n \sin(n\xi)),$$

$$F(s \sin(\xi), s\lambda \cos(\xi)) \sin(\xi) = \frac{s'_0}{2} + \sum_{n=1}^{\infty} (s'_n \cos(n\xi) + r'_n \sin(n\xi)).$$

El problema a resolver sigue siendo el problema no lineal (6.2.7), pero la solución de este es muy parecida a la de (6.2.8) y por tanto se puede tomar como matriz A del problema no lineal la resultante del problema equivalente linealizado.

Finalmente, se va a considerar otra clase especial de problemas muy amplia. Pero antes estudiemos el siguiente problema stiff (ver [87], y posteriormente [73], [74], [143] ó [147]):

$$\begin{cases} y'_1(x) = -2y_1(x) + y_2(x) + 2 \sin(x), \\ y'_2(x) = 998y_1(x) - 999y_2(x) + 999(\cos(x) - \sin(x)), \\ y_1(0) = 2, \quad y_2(0) = 3, \end{cases} \quad (6.2.9)$$

con solución

$$y_1(x) = 2e^{-x} + \sin(x), \quad y_2(x) = 2e^{-x} + \cos(x),$$

y donde los valores propios del jacobiano de la función son $\lambda_1 = -1, \lambda_2 = -1000$.

(En varios de los artículos antes mencionados el problema aparece como

$$\begin{cases} y_1'(x) = -2y_1(x) + y_2(x) + 2\sin(x), \\ y_2'(x) = -(\beta + 2)y_1(x) + (\beta + 1)(y_2(x) - \cos(x) + \sin(x)), \\ y_1(0) = 2, \quad y_2(0) = 3, \end{cases}$$

y la solución es independiente del valor β no así los valores propios del jacobiano de la función que pasan a ser $\lambda_1 = -1$, $\lambda_2 = \beta$, con lo que el problema puede hacerse aún más stiff.)

Pero si se añaden dos nuevas variables: $y_3 = \sin(x)$, $y_4 = \cos(x)$, entonces el problema pasa a ser

$$\begin{cases} y_1'(x) = -2y_1(x) + y_2(x) + 2y_3, \\ y_2'(x) = 998y_1(x) - 999y_2(x) + 999(y_4(x) - y_3(x)), \\ y_3'(x) = y_4(x), \\ y_4'(x) = -y_3(x) \\ y_1(0) = 2, \quad y_2(0) = 3, \quad y_3(0) = 0, \quad y_4(0) = 1. \end{cases}$$

Hemos introducido dos nuevas variable y por tanto la dimensión del problema crece, pero los resultados claramente mejoran sin que el coste computacional crezca de la misma forma.

Esta idea se puede utilizar en muchos ejemplos, pero especialmente es útil en aquellos del tipo

$$y'(x) = Ay(x) + F(x) + G(x) + P(x) + \varepsilon(x, y(x)),$$

donde $F(x) = (f_1(x), \dots, f_m(x))$, y cada $f_i(x)$ pertenece al espacio generado por las funciones $\langle \sin(\lambda_{i,1}x), \cos(\lambda_{i,1}x) \rangle$, $G(x) = (g_1(x), \dots, g_m(x))$, donde cada $g_i(x)$ pertenece al espacio generado por $\langle e^{\lambda_{i,2}x} \rangle$, $P(x) = (p_1(x), \dots, p_m(x))$, siendo cada $p_i(x)$ un polinomio del subespacio generado por $\langle 1, x, \dots, x^k \rangle$ y $\|\varepsilon(x, y(x))\| \ll 1$.

6.3. Error de truncado global en problemas derivados de EDP's.

Dado que es muy abundante el número de ecuaciones parciales que al ser discretizadas en una de las variables se convierten en ecuaciones diferenciales ordinarias de tipo *stiff* y que como se ha podido observar los nuevos métodos pueden aportar importantes mejoras al respecto, este va a ser un punto en el que se va a prestar atención en los ejemplos numéricos que se traten a continuación. Los enunciados y las demostraciones que aparecen en esta sección están basados de los que aparecen en [153].

En concreto el primer tipo de problemas que se tratarán serán ecuaciones parabólicas en un espacio \mathcal{H} de Hilbert, que al ser discretizadas en la variable espacial x se transforman en EDO's del tipo

$$u'(t) - Au(t) = f(t), \quad u(0) = u_0, \quad (6.3.1)$$

siendo $-A$ un operador definido positivo y f una función de t con valores en \mathcal{H} .

Consideremos los métodos exponential fitting de orden exponencial 1 de q pasos, si el problema fuera escalar y $A = \mu \in R$ y si llamamos

$$\nabla_P \sum_{j=0}^{q-1} \beta_j^0 \nabla^j U^n + h\mu U^n = \alpha_0 U^n + \dots + \alpha_q U^{n-q},$$

donde U^n es la aproximación a la solución $u(t)$ en la iteración n y si denominamos $h\mu = \lambda$, entonces el polinomio característico

$$P(\xi, \lambda) = (\alpha_0 - \lambda)\xi^q + \dots + \alpha_q,$$

que resulta ser el polinomio característico asociado al algoritmo adaptado implícito A-I-k- q verifica el siguiente lema:

Lema 6.3.1. *Sea $q \leq 6$ y sea $P(\xi, \lambda)$ el polinomio antes mencionado.*

Entonces $P(\xi, 0)$ tiene una raíz simple para $\xi = 1$ y el resto de ceros están en el interior del disco unitario. Es más, para $\lambda < 0$ todas las raíces de $P(\xi, \lambda)$ están en el interior del disco unitario y tienden a cero si $\lambda \rightarrow -\infty$.

Demostración: si consideramos que cuando $\lambda = 0$, los métodos BDF adaptados son los BDF clásicos, entonces es ampliamente conocido (ver [141], con software matemático son simples cuentas) que $P(\xi, 0)$ tiene una raíz simple para $\xi = 1$ y el resto de ceros están en el interior del disco unitario.

Debido a que el polinomio $P(\xi, \lambda)$ es el polinomio característico asociado a los algoritmos A-I-k-q se observará (en el apéndice dedicado a la 0-estabilidad del capítulo dedicado a los algoritmos de orden exponencial 1) que para $\lambda < 0$ todas las raíces de $P(\xi, \lambda)$ están en el interior del disco unitario.

Nos quedaría probar que para $q \leq 6$, si $\lambda \rightarrow -\infty$ las raíces de $P(\xi, \lambda)$ tienden a 0. De nuevo basta con utilizar un programa informático como MATLAB, Mathematica o Maple para obtener las raíces de dichos polinomios y después tomando límite cuando $\lambda \rightarrow -\infty$ fácilmente se puede comprobar el lema.

Dado que las raíces de los polinomios de orden 4, 5 ó 6 son bastante largas sólo vamos a ofrecer las de orden bajo.

Por supuesto una de ellas es siempre $r_1 = e^{\lambda h}$, que cuando $\lambda \rightarrow -\infty$, es obvio que $r_1 \rightarrow 0$.

Esta es la única raíz del algoritmo de 1 paso, mientras que las raíces del polinomio asociado al algoritmo de 2 pasos son

$$r_1 = e^u, \quad r_2 = \frac{1 - e^u + e^u u}{1 - e^u - u + 2e^u u}$$

donde $u = \lambda h$, tal como se comentó en el capítulo dedicado a los métodos de orden exponencial 1. Si $u \rightarrow -\infty$ el numerador tiende a 1, pero en el denominador manda $-u \rightarrow \infty$.

Las raíces del polinomio asociado al algoritmo de 3 pasos son

$$r_1 = e^u, \quad r_2 = \frac{C_1 + \sqrt{C_2}}{C_3}, \quad r_3 = \frac{C_1 - \sqrt{C_2}}{C_3},$$

donde

$$C_1 = -2 + e^u(6 - u) + e^{2u}(-4 + 3u),$$

$$C_2 = -(-1 + e^u)^2(-1 - 2u + e^u(2 + 6u) + e^{2u}(-1 - 4u + 3u^2)),$$

$$C_3 = -3 + 2u + e^u(8 - 6u) + e^{2u}(-5 + 6u),$$

en los numeradores de r_2 y r_3 , dominan $\pm\sqrt{2u}$, pero el denominador crece mucho más rápido a $-\infty$, por lo que ambas raíces van a 0.

De forma similar se comportan las raíces de los polinomios asociados a los algoritmos adaptados de orden 4,5 y 6, si no las mostramos es porque salvo r_1 el resto de raíces son expresiones muy largas y que se pueden calcular fácilmente con el programa adecuado. \square

Lema 6.3.2. *Las soluciones del método EF-I-k-q para el problema escalar (6.3.1) podrían escribirse como*

$$U^n = h \sum_{j=1}^n \delta_{n-j} f^j + \sum_{s=0}^q \delta_{n,s} U^s, \quad n \geq q, \quad (6.3.2)$$

donde $f^j = f(jh)$,

$$\tilde{\delta}(\xi) = \sum_{j=0}^{\infty} \delta_j(\lambda) \xi^j := \left(\sum_{j=0}^q \alpha_j \xi^j - \lambda \right)^{-1},$$

y

$$\delta_{n,s} = - \sum_{j=q-s}^q \delta_{n-s-j}(\lambda) \alpha_j.$$

Además si $q \leq 6$ existen constantes positivas c, C y una constante negativa λ_0 tales que

$$|\beta_j(\lambda)| \leq \begin{cases} Ce^{cj\lambda} & \text{si } \lambda_0 \leq \lambda < 0 \\ -C\lambda^{-1}e^{-cj} & \text{si } \lambda \leq \lambda_0. \end{cases} \quad (6.3.3)$$

Demostración: la demostración es muy similar a la que aparece en el libro de V. Thomée [141] (lema 10.3) para los BDF clásicos.

En primer lugar si representamos U^n de la forma

$$U^n = \sum_{q+1}^n \gamma_{n-j} f^j + \sum_{s=0}^q \gamma_{n,s} U^s, \quad n \geq q,$$

y se toma que $f^j = 1$ para $j = q+1$, $f^j = 0$ para $j > q+1$ y $U^s = 0$ para $s < q$ y se multiplica el método adaptado por ξ^n sumando los términos con $n \geq q+1$ se obtiene por una parte que

$$\tilde{U}(\xi) = \sum_{j=0}^{\infty} U^j \xi^j = \xi^{q+1} \tilde{\delta}(\xi),$$

y por otra que

$$\tilde{U}(\xi) = \sum_{n=q+1}^{\infty} \gamma_{n-q-1} \xi^n = \xi^{q+1} \tilde{\gamma}(\xi).$$

por lo que $\gamma_j = \delta_j$ para $j \geq 0$.

Si ahora tomamos $f^j = 0$ para $j > q$ y $U^j = 1$ para $j = s$, y $U^j = 0$ para $0 \leq j \leq q, j \neq s$, y se procede como en [141] (p. 149), se obtiene que $\gamma_{ns} = \delta_{ns}$ con lo que se demuestra (6.3.2).

La demostración de (6.3.3) es parecida a la que aparece en el libro de Thomée (ver [141] p. 150) basta utilizar el lema 6.3.1 y tener en cuenta que cuando $\lambda = 0$ los métodos adaptados coinciden con los BDF clásicos, que los coeficientes son los mismos y que por tanto el polinomio $P(\xi; 0)$ aquí es el mismo que utiliza Thomée. \square

Lema 6.3.3. *Sea $q \leq 6$, entonces existe una constante C , independiente del operador A definido negativo, tales que la solución del problema parabólico (6.3.1) al aplicar el algoritmo adaptado A-I-k-q verifica*

$$\| U^n \| \leq C \sum_{j=0}^{q-1} \| U^j \| + Ch \sum_{j=q}^n \| f^j \|, \quad n \geq q. \quad (6.3.4)$$

Demostración: la demostración es similar a la que aparece en el libro de V. Thomée [141] (lema 10.1) para los BDF clásicos. Para esta demostración tal como explica Thomée se puede reducir la demostración al caso escalar con $\lambda = h\mu$.

Al descomponer utilizando la lista de valores $(\{\mu_l\}_{l=1}^{\infty})$ y funciones propias $(\{\varphi_l\}_{l=1}^{\infty})$ asociados al problema (6.3.1) tenemos que si $U^j = 0$ para $j \leq q-1$ y $f^j = 0$ si $j \neq s$, $q \leq s \leq n$, entonces se verifica que

$$U^n = h \sum_{l=1}^{\infty} \delta_{n-s}(h\mu_l)(f^s, \varphi_l)\varphi_l,$$

y por tanto

$$\| U^n \| \leq h \sup_{\lambda < 0} | \delta_{n-s}(\lambda) | \| f^s \| \leq Ch \| f^s \|.$$

De forma similar se procede con los valores discretos iniciales. □

Teorema 6.3.4. *Sea $q \leq 6$, entonces existe una constante C , independiente del operador A definido negativo, tal que si u^n y U^n son respectivamente la solución exacta del problema parabólico (6.3.1) y la aproximación numérica a dicha solución al aplicar el algoritmo adaptado A-I-k-q, y u es suficientemente suave, entonces se verifica que*

$$\| U^n - u^n \| \leq C \sum_{j=0}^{q-1} \| U^j - u^j \| + Ch^q \int_0^{nh} \| f^q \|,$$

Demostración: Sea $e^n = U^n - u^n$, entonces se verifica que

$$\frac{1}{h} \nabla_P \sum_{j=0}^{q-1} \beta_j^0 \nabla^j (U^n - u^n) = -\tau^n, \quad n \geq q$$

donde $\tau^n = \frac{1}{h} \nabla_P \sum_{j=0}^{q-1} \beta_j^0 \nabla^j u^n - f^n$.

De esta forma aplicando el lema 6.3.3 a e^n obtenemos que

$$\| e^n \| \leq C \sum_{j=0}^{q-1} \| e^j \| + Ch \sum_{j=q}^n \| \tau^j \| .$$

Considerando una interpolación mixta (pues podemos considerar la solución como $u(t) = e^{At} + Q(t) + R(t)$, donde $Q(t)$ es un polinomio de grado $q - 1$ y $R(t)$ es el resto), a partir de las propiedades de ∇_P (si $u(t) = e^{At}u(0) + e^{At} \int_0^t e^{-Ay} f(y) dy$, hay que tener en cuenta que $\nabla_P e^{At_n} u(0) = 0$ y que $\nabla_P e^{At_n} \int_0^{t_n} e^{-Ay} f(y) dy = e^{At_n} \int_{t_{n-1}}^{t_n} e^{-Ay} f(y) dy$) se concluye con

$$h \| \tau^j \| \leq Ch^q \int_{t_{j-q}}^{t_j} \| f^q \| ds. \quad \square$$

6.4. Construcción de métodos exponenciales con dos parámetros.

Consideremos en el intervalo cerrado $[a, b]$ el problema escalar de valor inicial que aparece en (3.1.1), donde $g(x, y(x))$ es una función que satisface los teoremas de existencia y unicidad de la solución del problema de valor inicial.

Definición 6.4.1. Diremos que un método es exponencial con dos parámetros si integra el problema (3.1.1) de forma exacta cuando $y(x)$ pertenece a un espacio donde se incluye el subespacio generado por $\langle 1, e^{\lambda x}, e^{\tau x} \rangle$.

Veremos a continuación que los coeficientes de los nuevos métodos van a depender de los parámetros λh y τh , siendo $\lambda \in C^-$, $\tau \in C$.

Consideraremos que la 0-estabilidad se estudiará a partir del problema lineal

$$y'(x) = \lambda y(x), \quad (6.4.1)$$

y que la estabilidad absoluta saldrá a partir del problema

$$y'(x) = \mu y(x), \quad (6.4.2)$$

ya nos podemos dar cuenta que vamos a tener problemas similares a los que teníamos anteriormente con el paso variable. Esta es la razón por la que de nuevo sólo vamos a estudiar una pareja de métodos: los métodos explícito e implícito de dos pasos que integran de forma exacta el problema (3.1.1) cuando $y(x)$ pertenece al subespacio generado por $\langle 1, e^{\lambda x}, e^{\tau x} \rangle$.

El procedimiento será similar al de anteriores capítulos: primero calcularemos los coeficientes de los métodos para el problema escalar, el paso a problemas vectoriales es ya conocido, daremos los errores de truncado local de los métodos y después las principales propiedades de 0-estabilidad y estabilidad absoluta de los métodos, propiedades que se desarrollarán en los apéndices G y H.

6.4.1. Construcción de métodos exponenciales con dos parámetros de dos pasos.

Al igual que en la sección anterior podemos obtener los coeficientes directamente de imponer que los métodos integren de forma exacta el problema (3.1.1) cuando $y(x) = 1$, $y(x) = e^{\lambda x}$ e $y(x) = e^{\tau x}$.

De esta forma, si escribimos el método explícito como

$$c_0 y_{n+1} + c_1 y_n + c_2 y_{n-1} = h f(x_n, y_n), \quad (6.4.3)$$

los pesos son

$$\begin{aligned} c_0 &= \frac{(-\tau e^{\tau h}(-1 + e^{\lambda h}) + \lambda e^{\lambda h}(-1 + e^{\tau h}))h}{(-1 + e^{\lambda h})(e^{\lambda h} - e^{\tau h})(-1 + e^{\tau h})}, \\ c_1 &= \frac{(\tau e^{\tau h}(-1 + e^{2\lambda h}) - \lambda e^{\lambda h}(-1 + e^{2\tau h}))h}{(-1 + e^{\lambda h})(e^{\lambda h} - e^{\tau h})(-1 + e^{\tau h})}, \\ c_2 &= \frac{e^{(\lambda+\tau)h}(\tau - \tau e^{\lambda h} - \lambda + \lambda e^{\tau h})h}{(-1 + e^{\lambda h})(e^{\lambda h} - e^{\tau h})(-1 + e^{\tau h})}, \end{aligned}$$

mientras que el método implícito es

$$c_0^* y_{n+1} + c_1^* y_n + c_2^* y_{n-1} = h f(x_{n+1}, y_{n+1}), \quad (6.4.4)$$

siendo los pesos

$$\begin{aligned} c_0^* &= \frac{(-\tau e^{\tau h}(-1 + e^{\lambda h}) + \lambda e^{\lambda h}(-1 + e^{\tau h}))h}{(-1 + e^{\lambda h})(e^{\lambda h} - e^{\tau h})(-1 + e^{\tau h})}, \\ c_1^* &= \frac{(\tau e^{\tau h}(-1 + e^{2\lambda h}) - \lambda e^{\lambda h}(-1 + e^{2\tau h}))h}{(-1 + e^{\lambda h})(e^{\lambda h} - e^{\tau h})(-1 + e^{\tau h})}, \\ c_2^* &= \frac{e^{(\lambda+\tau)h}(\tau(1 - e^{\lambda h}) - \lambda(1 - e^{\tau h}))h}{(-1 + e^{\lambda h})(e^{\lambda h} - e^{\tau h})(-1 + e^{\tau h})}. \end{aligned}$$

Por supuesto si el problema es vectorial se afronta como anteriores veces cambiando λ por la matriz A , 1 por la matriz identidad y considerando la división como la multiplicación por la derecha por la matriz inversa.

6.4.2. Convergencia de los métodos exponenciales con dos parámetros de dos pasos.

Dados los coeficientes del caso escalar basta realizar el desarrollo en series de Taylor y simplificar y obtenemos la consistencia de ambos métodos en el caso escalar.

Teorema 6.4.1. *El método explícito (6.4.3) de dos pasos es convergente de orden 2. Su error de truncado local se puede expresar como:*

$$\frac{(\lambda\tau y'(x) - (\lambda + \tau)y''(x) + y'''(x))}{6}.$$

Teorema 6.4.2. *El método implícito (6.4.4) de dos pasos es convergente de orden 2. Su error de truncado local se puede expresar como:*

$$-\frac{(\lambda\tau y'(x) - (\lambda + \tau)y''(x) + y'''(x))}{3}. \quad (6.4.5)$$

Para el caso vectorial se procede cambiando λ y τ por las matrices elegidas en el método, A y B .

Las propiedades de 0-estabilidad y estabilidad absoluta de los métodos exponenciales con dos parámetros se encuentran desarrolladas en los apéndices G y H. Las conclusiones más importantes que se observan son:

- 1) Es posible obtener más métodos explícitos de orden 2 con buena estabilidad, de hecho en el capítulo dedicado a los ejemplos numéricos se muestran diferentes resultados con diferentes problemas, basta considerar una matriz Ah "adecuada" y $B = tA$, con $t > 0$.
- 2) Los métodos implícitos siguen manteniendo mejores propiedades de estabilidad, como se observa sobre todo a partir del apéndice H dedicado a la estabilidad absoluta de los métodos exponenciales con dos parámetros.
- 3) Si $B \rightarrow 0$, los métodos exponenciales con dos parámetros tienden a los métodos EF-I-k-2, mientras que si $B \rightarrow A$, los métodos exponenciales con dos parámetros tienden a los métodos EF-I-r-2 (tiene fácil demostración ya que al hacer límites los coeficientes coinciden).

Capítulo 7

Ejemplos numéricos.

7.1. Introducción.

Hasta ahora se han presentado los métodos nuevos aportados en este trabajo y se han estudiado sus propiedades, sin embargo, en este capítulo nos vamos a concentrar en el problema a resolver. Ya hemos comentado que nuestros métodos van a funcionar bastante bien (debido sobre todo a las propiedades de estabilidad) con problemas del tipo de (2.4.1), donde $y = [y^1, \dots, y^m]$, y $G = [g^1, \dots, g^m]$ del tipo *stiff* o altamente oscilatorios. Pero también proporcionan buenos resultados incluso ante problemas donde el jacobiano de $G(x, y(x))$ tiene valores propios positivos, pero las soluciones de los problemas no tienen exponenciales positivas.

Antes de continuar conviene dar una definición de *stiffness* (en algunos manuales en español se habla del término problemas rígidos), lo cual no es nada sencillo, debido a que cada autor parece tener la suya (o suyas) propia (propias), por ejemplo, en [134] se explica bien cómo a veces las definiciones de problema *stiff* pueden ser muy diferentes.

La más común de las definiciones de *stiffness* (ver [134]) es: “*Stiffness occurs*

if for most explicit methods, the largest step size h_n^ , guaranteeing numerical stability is much smaller than the largest step h_n for which the local discretization error is still sufficiently small (in norm), i.e., $h_n^* \ll h_n$* ” (una definición similar aparece en [1], [38], [58], [87] ó [130], tal como se recoge en [134]). En esta definición no se menciona explícitamente que problemas escalares (por ejemplo los problemas del tipo Prothero-Robinson) no sean problemas *stiff*, de hecho este tipo de ejemplos numéricos se suelen utilizar frecuentemente para demostrar propiedades o los resultados numéricos que son capaces de alcanzar los diferentes métodos que se presentan habitualmente (ver, por ejemplo, [6], [110], [144] ó [146]). Sin embargo, el criterio que se da en la misma referencia bibliográfica para comprobar si un problema es *stiff* con esta definición es que dados los λ_i valores propios del jacobiano se verifique que $\frac{\max|Re\lambda_i|}{\min|Re\lambda_i|} \gg 1$ y que $\max Re\lambda_i < 0$ (ver también [87]).

Claramente con este criterio los problemas del tipo Prothero-Robinson u otros problemas escalares similares no serían considerados problemas *stiff*.

Otra definición muy frecuente que se da para explicar lo que es un problema *stiff* es (ver de nuevo [134]): “*Initial value problems are stiff if they are (exceedingly) difficult to solve by ordinary explicit step-by-step methods, whereas certain implicit methods perform quite well*” (ver también [36], [64] ó [98]), sin embargo, esta definición probablemente se puede considerar ya un poco obsoleta, debido a que últimamente han aparecido métodos explícitos que presentan buenos resultados con problemas ampliamente considerados *stiff* (mismamente en este trabajo aparecen métodos explícitos que tendrán buenos resultados).

En [38] la definición de problema lineal *stiff* que se enuncia es: “*We next consider the general linear problem $y'(t) = Ay(t) + r(t), t \geq 0, y(0) = y_0,$*

where A is a constant $m \times m$ matrix and r a time dependent forcing term. The most obvious way of defining stiffness for this linear system is to impose conditions on the eigenvalues of A . Let $\lambda_1, \dots, \lambda_m$ denote these eigenvalues. Then this problem may be called stiff if

1° λ_i exist with $Re(\lambda_i) \ll 0$.

2° λ_i exist of moderate size, i.e., $|\lambda_i|$ small when compared with the modulus of the eigenvalues satisfying 1°.

3° no λ_i exist with a large positive real part.

4° no λ_i exist with a large imaginary part, unless $Re(\lambda_i) \ll 0$."

Sin embargo, hay numerosas referencias de problemas donde los autovalores del jacobiano son únicamente imaginarios puros y que son considerados problemas *stiff* oscilatorios (ver [53], [54]). Lo cual a su vez entra en contradicción con otras muchas referencias donde se define problema *stiff* oscilatorio como un problema "... of the general form (2.4.1) where the eigenvalues λ_j of the Jacobian matrix $\frac{dG}{dy}$ can be written as $\lambda_j = \mu_j + i\nu_j$, where $\mu_j < 0$ for all j , $\max_{1 \leq j \leq m} |\mu_j| \gg \min_{1 \leq j \leq m} |\mu_j|$, and $|\mu_j| \ll |\nu_j|$ for at least one pair of eigenvalues of large modulus ..." (ver [24] o [25]), y se diferencia entre problema *stiff* oscilatorio y problema altamente oscilatorio.

Por tanto no es fácil quedarse con una única definición de *stiffness* y mucho menos con una de problema *stiff* oscilatorio. En este trabajo no se pretende polemizar sobre estas cuestiones, sino ofrecer una serie de métodos que nos permitan obtener buenos resultados frente a un gran número de problemas.

Por todas estas cuestiones en este capítulo no nos quedaremos con unas definiciones determinadas, sino que ofrecemos una variedad bastante amplia de las que se pueden leer en diferentes manuales y artículos científicos.

Compararemos nuestros métodos con otros ya descritos tanto con problemas escalares, como en problemas donde los autovalores del jacobiano son únicamente imaginarios puros, o incluso tienen parte real positiva, sin que esto signifique que nos pronunciemos a favor ni en contra de que sean considerados problemas *stiff*. También mostraremos los resultados que proporcionan los nuevos métodos frente a problemas *no-stiff*, debido a que en muchos problemas se combinan zonas donde el problema es *stiff* con otras donde no lo es, por lo que es necesario comparar los resultados frente a integradores para problemas *stiff*. Sin embargo, en este grupo de problemas los nuevos métodos aquí expuestos no son tan prometedores.

El tipo de problemas en los que nos vamos a centrar, los problemas *stiff* (y también los problemas *stiff* oscilatorios o altamente oscilatorios) son muy comunes en una gran cantidad de áreas de las ciencias aplicadas. Dos buenos manuales donde se recogen las aplicaciones de estos problemas son [1] y [38], aquí haremos una breve recopilación sobre cuáles son los campos donde este tipo de problemas son más frecuentes según estos y otros manuales y artículos científicos.

- a) **Atmósfera:** los fenómenos atmosféricos requieren de ecuaciones de transporte con reacciones químicas, dichas ecuaciones químicas-cinéticas son habitualmente *stiff*.
- b) **Biología:** en numerosas reacciones cinéticas de enzimas y modelos farmacológicos las ecuaciones resultantes son *stiff*.
- c) **Combustión:** de nuevo este área requiere de reacciones químicas y ecuaciones de transporte, la rigidez del problema puede derivarse del sistema cinético.

- d) Dinámica de la guía de misiles.
- e) Dinámica molecular: muy común al modelar matemáticamente experimentos computacionales que no puedan o sean muy costosos de realizar en el laboratorio.
- f) Fases de dispersión: la rigidez se puede derivar de modelos donde sea necesaria una transferencia de calor o de masa.
- g) Fluidos: numerosas ecuaciones en derivadas parciales procedentes de problemas de fluidos homogéneos con cambios muy bruscos en sus propiedades se pueden discretizar para ser estudiadas como EDO's de tipo *stiff*.
- h) Láser: problemas *stiff* y altamente oscilatorios son bastante habituales debido a que es frecuente tener partículas con un alto estado de excitación, ver [11] ó [12].
- i) Mecánica: muy comunes en el análisis de estructuras dinámicas o también en la mecánica celeste. Uno de los problemas más conocidos es el "*stiff beam*", ver [24] ó [64], por ejemplo.
- j) Medicina: como el problema llamado Medical Akzo Nobel, ver [66], [71], [91], ó [94], también en

<http://www.dm.uniba.it~testset/>
- k) Nuclear: no sólo al modelar la dinámica molecular, también al estudiar el diseño o el control de procesos enteros.
- l) Procesos en reactores: debido a que muchos de ellos dependen de altísimas temperaturas que a menudo cambian bruscamente.
- m) Procesos industriales: este área puede fácilmente necesitar transferencias de calor o de masa con ecuaciones de transporte o de reacciones que puedan

ser consideras *stiff*. Una ecuación clásica de este tipo es la ecuación de Burger's, ver [1], [41], [64] ó [105].

n) Reacciones químico-cinéticas: de lejos es el área donde son más comunes los problemas *stiff*, en una gran cantidad de los casos, la rigidez se produce debido a que las constantes que miden la reacción son muy diferentes entre sí. Desde mediados de los años 70 ([48] ó [123], por ejemplo) se han propuesto numerosos ejemplos de esta clase. Muchos de los ejemplos más conocidos (problema de HIREs, ver [59], [64], [94], [127], [138] ó [148]; reacción de Belousov, ver [48], [61], [64], [82], [137], [155], [158] ó [159]; ó la reacción de Robertson, ver [14], [64] ó [123]; por mencionar algunos) pueden ser considerados de esta clase de problemas.

o) Teoría de circuitos electrónicos: a menudo la dinámica de las máquinas y sus sistemas de control requieren la solución de EDO's o de ecuaciones algebraicas con una dimensión del problema muy elevada.

p) Teoría de control: hay muchos problemas perturbados singulares que se resuelven con métodos habituales de los problemas *stiff*.

q) Transferencias de calor: son muy habituales, la rigidez suele originarse de cambios muy bruscos en el ambiente o a partir de grandes diferencias en las capacidades o propiedades de los componentes en la transferencia. Mención aparte merece la abundante clase de problemas originados a partir de discretizar problemas en EDP's.

Los problemas *stiff*, por tanto, consituyen una clase de PVI muy frecuentes y un área muy interesante donde resulta muy complicado incluir todas las contribuciones.

Gran cantidad de métodos se han estudiado con el fin de aportar buenos resultados frente a problemas *stiff*, sin embargo, se podría reseñar tres grandes clases de métodos donde se suelen agrupar los integradores numéricos que presentan mejores resultados con este tipo de problemas.

i) Métodos Runge-Kutta implícitos que tengan buenas propiedades de estabilidad, se suele buscar que sean A-estables (ver [37]), L-estables (ver [44]), aunque desde finales de los 70 se han añadido nuevas propiedades (como la S-estabilidad, B-estabilidad o BN-estabilidad).

Una subclase muy importante de estos integradores es la formada por los Runge-Kutta diagonalmente implícitos, de esta subclase destacan, por citar algunos los STRIDE ([2], [35] ó [102]) ó los DIRK ([2] ó [17]) y sus variantes.

Otros métodos muy utilizados son los Gauss ([64]) ó los Rosembrock ([64] ó [125]).

En la actualidad métodos muy citados son los Radau, especialmente los RadauIIA (ver [64]) de dos y tres pasos debido a que sus regiones de estabilidad absoluta (con la definición clásica) son muy grandes.

Probablemente uno de los algoritmos más mencionados últimamente de este tipo es el radau5 que se puede obtener en la página web de E. Hairer

<http://www.unige.ch/~hairer/software.html>

y que está basado en el radauIIA de tres pasos con paso variable.

ii) BDF implícitos: clásicamente han sido muy utilizados en paquetes integrados que servían para integrar problemas *stiff*. Algunos ejemplos pueden ser el LSODE (ver [68]) y sus variantes (proceden del paquete GEAR, ver [67], y del LINPACK, ver [42]) o el DASSL (ver [108] ó [109]).

Sin embargo en [24] ó en [25], se explica como el LSODE, el DASSL o el DIFSUB (ver [58]) tienen grandes dificultades al integrar problema *stiff* con valores propios del jacobiano con parte real negativa, pero muy cerca del eje imaginario.

Frente a este tipo de problemas J.R. Cash muestra un método que funciona muy bien, es un método de longitud de paso variable y de orden también variable hasta 7 conocido como MEBDF (ver [23]) que se puede obtener en la página web

http://www.ma.ic.ac.uk/~jcash/IVP_software/readme.html

iii) Métodos exponential fitting o adaptados: se ha usado este tipo de métodos para aproximar la integración de problemas *stiff* desde mediados de los años 70. Sin embargo, la idea ha tenido un mayor seguimiento en las últimas dos décadas donde ha merecido una especial atención. La idea básica de estos métodos (al menos recientemente) es que integren de forma exacta el problema (2.4.1) cuando $y(x)$ pertenece a un espacio donde e^{Ax} está incluido.

Una aportación interesante es que se muestran métodos explícitos que integran con buenos resultados una gran cantidad de problemas considerados *stiff*.

Una posible laguna que tienen muchos de los trabajos en los que se estudian dichos métodos es que habitualmente no se estudia la estabilidad de los mismos, o se dice que los métodos *exponential fitting* son siempre estables. Sin embargo, como ya se ha visto esta afirmación en general es falsa.

En este capítulo en la mayor parte de los casos hemos utilizado los ya mencionados Radau5 y MEBDF para comparar los nuevos métodos, sin embar-

go a menudo los resultados con métodos *exponential fitting* son claramente mejores, en algunos de estos casos hemos comparado los nuevos resultados frente a los resultados obtenidos con diferentes métodos *exponential fitting* de paso fijo (ver [18], [20], [73], [77] ó [139]).

Otros integradores habituales en la literatura científica y que se han utilizado aquí han sido los integradores del MATLAB (ver [131]) que son de paso variable.

A veces las comparaciones pueden parecer no muy fiables debido a que comparamos métodos de paso variable con los nuestros que frecuentemente serán de paso fijo. Para hacer la comparación más fiable también se han utilizado métodos de paso fijo como el radauIIA de dos pasos y orden tres, o el método de Gauss de dos pasos y orden cuatro.

Se ha dividido el capítulo según el origen y la clase de los problemas a tratar. Como ya se ha explicado anteriormente gran cantidad de los problemas proceden de ecuaciones en derivadas parciales, de ellos se hablará en la segunda sección del capítulo. En el tercer punto se integrarán con diferentes métodos problemas donde el jacobiano de la función tiene valores propios positivos, pero la solución de los problemas no contiene ninguna exponencial positiva. En la cuarta parte en la que se divide el capítulo compararemos los diferentes resultados obtenidos frente a problemas considerados *stiff* oscilatorios o altamente oscilatorios. En la quinta sección del capítulo mostraremos los resultados con otros tipos de problemas y procuraremos comparar nuestros métodos entre sí.

En último lugar veremos algún ejemplo de cómo funcionan los métodos de tipo BDF (caso escalar) en paso variable, así como los métodos de tipo BDF-

Runge-Kutta (vectorial).

7.2. Ecuaciones en derivadas parciales.

Como ya se ha comentado varias veces en este trabajo, los nuevos métodos son capaces de resolver de forma exacta (salvo errores de máquina) una gran cantidad de problemas que aparecen en la literatura científica. Un ejemplo claro es la ecuación de calor $u_t = u_{xx} + f(x)$ que se explica bien en [130], por ejemplo. Si discretizamos en la parte espacial, obtenemos un problema del tipo $y'(t) = Ay(t) + b$ (con autovalores de A , $\lambda_i \in R^-$) y con $b = [b_1, \dots, b_n]$, $b_i \in R$. El problema puede ser tan stiff como queramos, ya que los autovalores del jacobiano estarán en el intervalo $[-4N^2, -\pi^2]$ con $N \rightarrow \infty$, sin embargo, los nuevos métodos integrarán sin error de truncado local la discretización del problema.

Esta puede ser una característica importante de estos métodos exponential fitting. Por ejemplo, si consideramos el problema de la ecuación de calor que aparece en [50]:

$$u_t = u_{xx} + f(x), \quad u(0, x) = 0, \quad 0 \leq x \leq 1,$$

$$u_0(t) = 0, \quad u_{N+1}(t) = 0, \quad t \in [0, 1]$$

donde $f(x)$ es la aproximación de la función delta de Dirac en el punto $x = 0,5$. Al discretizar en la variable espacial obtenemos el problema

$$u'(t) = Au(t) + b, \quad u(0) = [0, \dots, 0],$$

donde A es la matriz de dimensión $N \times N$

$$A = N^2 \begin{pmatrix} -2 & 1 & 0 & \dots & 0 \\ 1 & -2 & 1 & 0 & \dots \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & 1 & -2 \end{pmatrix}$$

y $u(0)$ es un vector de dimensión N . Los nuevos métodos resuelven exactamente (sin error de truncado local) el problema. Por ejemplo, si utilizáramos el método explícito EF-I-r-2 con un paso $h = 0,1$ necesitaría 10 pasos (y por tanto 10 evaluaciones de la función), sin embargo, con $N = 100$, el método MEBDFDAE con orden variable hasta 7 y de paso también variable, utilizando como constantes $Rtol = Atol = 10^{-4}$ necesita 28835 pasos y 103024 evaluaciones de la función para integrar hasta el punto $t = 1$, donde el error es $\| y_{28835} - y(1) \|_2 \approx 1,77379 \times 10^{-4}$.

A continuación mostramos los resultados logrados con los nuevos métodos frente a problemas no triviales para métodos *exponential fitting*.

Problema 1.1, un ejemplo de este tipo es la discretización de la ecuación de difusión que muestra J.C. Butcher et al. en [15].

Al principio el problema es

$$u_t = u_{xx},$$

en el intervalo $x \in [0, 1]$, con valores iniciales $u(0, x) = a \sin(\sqrt{2}x) - \sin(x)$ y con valores de contorno $u(t, 0) = 0$, $u(t, 1) = ae^{-2t} \sin(\sqrt{2}) - e^{-t} \sin(1)$, siendo $a = \cos(\sqrt{2})/(\sqrt{2} \cos(2^{-1/2}))$.

Si discretizamos considerando $y_i(t) = u(t, i/(N + 1))$, $i = 1, \dots, N$, ahora

nuestro problema será

$$y'(t) = (N+1)^2 \begin{pmatrix} -2 & 1 & 0 & 0 & \dots & 0 & 0 \\ 1 & -2 & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & -2 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & -2 & 1 \\ 0 & 0 & 0 & 0 & \dots & 1 & -2 \end{pmatrix} y(t) + (N+1)^2 \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ \phi(t) \end{pmatrix}$$

donde $\phi(t) = ae^{-\nu t} \sin(\sqrt{2}) - e^{-\mu t} \sin(1)$ y

$$\mu = 2(N+1)^2 \left(1 - \cos\left(\frac{1}{N+1}\right) \right), \quad \nu = 2(N+1)^2 \left(1 - \cos\left(\frac{\sqrt{2}}{N+1}\right) \right).$$

La solución del problema discretizado es

$$y_i(t) = ae^{-\nu t} \sin\left(\frac{\sqrt{2}i}{N+1}\right) - e^{-\mu t} \sin\left(\frac{i}{N+1}\right),$$

y los dos problemas convergen (uno al otro) ya que si $N \rightarrow \infty$, entonces $\mu \rightarrow 1$ y $\nu \rightarrow 2$.

Al igual que en [15] hemos considerado $N = 10$, de forma que dos buenas aproximaciones para μ y ν son

$$\mu = 0,99931148446379, \quad \nu = 1,99724669642314.$$

Hemos comparado el algoritmo A-I-k-4 con los métodos de orden 4 ARK con $\lambda = 1/2$ y ARK con $\lambda = 3/5$ y el Gauss de dos pasos.

En la figura 7.1 mostramos los resultados al integrar con paso constante $h = 1/n$ y tomando n pasos.

Si en cambio tomamos $N = 100$ en este mismo problema (en ese caso μ y ν son aproximadamente $\mu = 0,9999918308928846$ y $\nu = -1,9999673236788025$), entonces las diferencias son algo mayores debido a la B-convergencia, ya que como explican Lambert [87] ó Hairer y Wanner [64], cuando $\nu h \rightarrow -\infty$

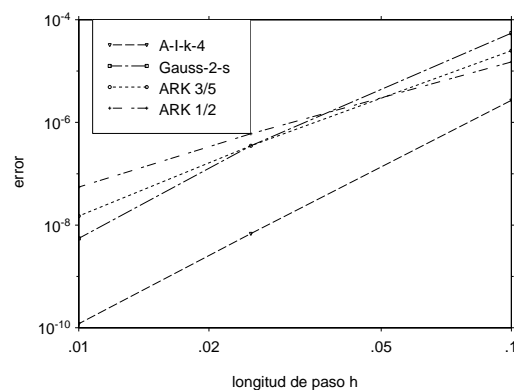


Figura 7.1: Errores en la integración numérica del problema 1.1 con $N = 10$.

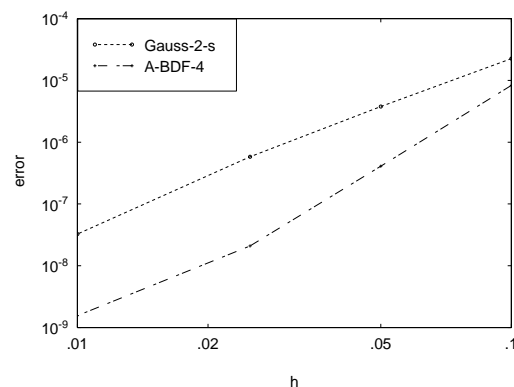


Figura 7.2: Errores en la integración numérica del problema 1.1 con $N = 100$.

(donde ν es el mayor de los valores propios del jacobiano, los autovalores del jacobiano están en el intervalo $[-40794'1, -9'86881]$, por lo que el problema puede ser considerado *stiff*) el error local del método Gauss-2-s es un $O(h^3)$, mientras que el error del A-I-k-4 es un $O(h^4)$ como podemos observar en la figura 7.2.

Si comparamos el tiempo CPU empleado (ambos métodos han sido escritos en el mismo lenguaje usando un Intel Pentium M 715 con 1.50 GHz, como siempre que comparemos el CPU empleado por los diferentes algoritmos) las diferencias son mayores, por ejemplo con $h = 0,05$ el Gauss-2-s necesita

100,064 segundos y su error es $3,76812 \times 10^{-6}$, mientras que el A-I-k-4 con $h = 0,01$ necesita 5,197 segundos y el error es $1,56289 \times 10^{-9}$. La razón es que usamos un único parámetro Ah en el problema, además los métodos adaptados funcionan como algoritmos explícitos con problemas lineales.

Problema 1.2, este problema aparece en [119], [120] y [121]. Es un problema parabólico lineal de 1 dimensión singularmente perturbado que procede de una reacción de advección-difusión

$$\frac{\partial u}{\partial t} + a(x, t) \frac{\partial u}{\partial x} + b(x, t)u = \varepsilon \frac{\partial^2 u}{\partial x^2} + f(x, t), \quad 0 \leq x \leq 1, \quad 0 \leq t,$$

tomamos como valores de contorno

$$u(0, t) = u(1, t) = 0, \quad t \geq 0,$$

$$u(x, 0) = u_0(x), \quad 0 \leq x \leq 1,$$

y consideramos

$$a(x, t) = 2 - x^2, \quad b(x, t) = x,$$

$$u_0(x) = 0, \quad f(x, t) = 10t^2 e^{-t} x(1 - x),$$

mientras que $\varepsilon = 2^0, 2^{-4}, 2^{-8}$.

Hemos discretizado en la variable espacial x , y consideramos el sistema de ecuaciones diferenciales ordinarias en t .

Este ejemplo numérico no tiene solución exacta, por eso estimamos el error a partir de

$$e_\varepsilon^{N, \Delta t}(x_i, t^n) = | u^N(x_i, t^n) - u_e(x_i, t^n) |,$$

donde $N = \frac{1}{\Delta x}$ y $t^n = nh = n\Delta t$ y u_e es la solución numérica lograda a partir del radauIIA de dos pasos con longitud de paso $h = 10^{-3}$.

Hemos comparado en las tablas los errores máximos nodales

$$E_{\varepsilon}^{N,\Delta t} = \max_{i,n} e_{\varepsilon}^{N,\Delta t}(x_i, t^n),$$

y hemos tomado como en [119], [120] ó [121] $N = 16$ y $\Delta t = 0,1$ y hemos integrado el problema para $0 \leq t \leq 3$. Hemos comparado los resultados obtenidos con los métodos adaptados de 3 (A-I-k-3) y de siete pasos (A-I-k-7) al utilizar longitud de paso constante $h = 0,1$, con los obtenidos al utilizar el radauIIA de dos pasos con longitud de paso $h = 0,1$ y $h = 0,25$. Dichos resultados se muestran en el cuadro 7.1. En este ejemplo hemos comparado también el tiempo empleado por la CPU ya que todos los métodos han sido escritos en el mismo lenguaje, utilizando la misma precisión de la máquina y utilizando el mismo procesador, un Intel Pentium M 715 a 1.50 GHz, como en todos los casos en los que se van a comparar error-tiempo empleado.

Si $\varepsilon = 1$, todos los valores de la matriz A son reales y pertenecen al intervalo $[-1013, -10]$, y si $\varepsilon = 10^{-4}$ los autovalores de la matriz siguen siendo reales pero ahora en el intervalo $[-52, -11]$, como $h = 0,1$, no hay problemas con la estabilidad y obtenemos buenos resultados con el método implícito de siete pasos.

Sin embargo con $\varepsilon = 10^{-8}$ los valores propios son complejos y por ejemplo dos de ellos son $-1,71917 \pm 28,9945i$, por eso el método de siete pasos tiene problemas de estabilidad para este ε (como se puede deducir de lo visto en el capítulo de los métodos de orden exponencial 1), no así el de 3 pasos que presenta buenos resultados.

Podemos observar que al calcular una sola vez el parámetro A y por tanto los coeficientes los métodos son “rápidos”. Otro hecho importante que ya podemos anotar es que cuando los autovalores del jacobiano tienden al eje

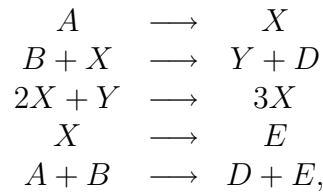
ε	Método	Error	Tiempo CPU (segundos)
10^0	A-I-k-3	$1,36076 \times 10^{-4}$	0.851
	A-I-k-7	$3,4105 \times 10^{-7}$	1.002
	radau, $h = 0,1$	$1,31096 \times 10^{-5}$	6.61
	radau, $h = 0,25$	$1,35706 \times 10^{-4}$	2.544
10^{-4}	A-I-k-3	$6,30235 \times 10^{-4}$	0.831
	A-I-k-7	$2,24621 \times 10^{-7}$	1.031
	radau, $h = 0,1$	$4,30247 \times 10^{-5}$	6.74
	radau, $h = 0,25$	$5,78259 \times 10^{-4}$	2.544
10^{-8}	A-I-k-3	$1,51375 \times 10^{-3}$	0.831
	radau, $h = 0,1$	$2,55203 \times 10^{-4}$	6.609
	radau, $h = 0,25$	$2,03278 \times 10^{-3}$	2.533

Cuadro 7.1 : Errores al integrar numéricamente el problema 1.2.

imaginario puro (desde el semieje real negativo), los errores crecen, pero algo más lentamente que con otros métodos como en este caso el radau.

Problema 1.3, hasta ahora hemos estudiado el comportamiento de los métodos con problemas lineales, sin embargo, los nuevos métodos también tienen un buen comportamiento frente a problemas no lineales. Un ejemplo se muestra en [95]: el modelo Brusselator (que aparece previamente en [5], [40], [45] [64], [93], [107], [113] ó [124]) es un modelo dinámico no lineal en dos variables, muestra un mecanismo muy simple de una reacción química que exhibe propiedades de oscilación no lineal.

El mecanismo químico es



donde A, B son las sustancias químicas iniciales que se introducen con concentraciones ciertas constantes; D, E son las sustancias de salida y X, Y son sustancias intermedias.

Entonces las ecuaciones de concentración para esas dos sustancias intermedias son

$$\begin{aligned} u_t &= A - (1 + B)u + Cu^2v + \alpha_1 u_{xx} \\ v_t &= Bu - Cu^2v + \alpha_2 v_{xx}. \end{aligned}$$

Estas son las ecuaciones que hemos considerado. De nuevo, hemos discretizado la variable $x \in [0, 1]$ y de esta forma transformamos la ecuación en derivadas parciales (EDP's) en ecuaciones diferenciales ordinarias (EDO's).

En el ejemplo que hemos tomado, hemos asignado los valores $A = 0'25$, $B = 0'01$, $C = 0'1$ y $\alpha_1 = \alpha_2 = 0'2$ y para las condiciones de contorno hemos elegido

$$\begin{aligned} u(x, 0) &= \frac{1 + \sin(2\pi x)}{2}, & v(x, 0) &= 0,5 \\ u_0(t) = 0,4 &= u_{N+1}(t), & v_0(t) = 0,4 &= v_{N+1}(t), \end{aligned}$$

donde $N = \frac{1}{\Delta x} = 100$.

De esta forma, tenemos un sistema de 200 ecuaciones diferenciales ordinarias con autovalores del jacobiano en el intervalo $[-8000, -1'9]$, de esta forma el problema puede ser considerado *stiff*.

Hemos integrado dicho problema en el intervalo $[0, 5]$, y hemos calculado el error en el punto $t = 5$ como $\sqrt{\frac{\sum_{i=1}^N (u_i^e - u_i(5))^2 + \sum_{i=1}^N (v_i^e - v_i(5))^2}{200}}$ (donde u_i^e, v_i^e son los resultados de los métodos en el punto $t = 5$, mientras que $u_i(5), v_i(5)$ son

los valores hallados con el NDSolve del Mathematica), para tener una idea del error medio en cada componente. Comparamos el error y el número de evaluaciones de la función en el cuadro 7.2.

Método	Error	N. e.f.
EF-I-r-2 explícito, orden 2	$5,22003 \times 10^{-6}$	50
MEBDF, orden variable hasta 7	$2,08744 \times 10^{-5}$	59
Radau5, orden 5	$1,11443 \times 10^{-4}$	91

Cuadro 7.2 : Errores al integrar numéricamente el problema 1.3.

Problema 1.4, hemos tomado este problema de [3]. Originalmente el problema es una ecuación de Burgers

$$u_t + uu_x = \nu u_{xx}, \quad \nu > 0,$$

donde como antes $u = u(x, t)$, y esta vez los límites son $0 \leq x \leq 1$, $0 \leq t \leq 1$ y las condiciones de contorno e iniciales son

$$u(x, 0) = (\sin(3\pi x))^2(1 - x)^{3/2}, \quad u(0, t) = u(1, t) = 0.$$

De nuevo hemos discretizado en la variable espacial x , de esta forma si denotamos $\Delta x = \frac{1}{N+1}$, $u_i(t) = u(i\Delta x, t)$, $i = 1, \dots, N$ tenemos el siguiente PVI

$$u'_i - \nu \frac{u_{i+1} - 2u_i + u_{i-1}}{(\Delta x)^2} = -\frac{u_{i+1}^2 - u_{i-1}^2}{4\Delta x}, \quad i = 1, \dots, N,$$

$$u_i(0) = (\sin(3\pi\Delta x))^2(1 - i\Delta x)^{3/2}, \quad i = 1, \dots, N.$$

Hemos considerado como en anteriores casos

$$A = \nu(N + 1)^2 \begin{pmatrix} -2 & 1 & 0 & 0 & \dots & 0 & 0 \\ 1 & -2 & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & -2 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & -2 & 1 \\ 0 & 0 & 0 & 0 & \dots & 1 & -2 \end{pmatrix}.$$

Hemos considerado como en [3] $N = 24$ y $\nu = 0,2$ y hemos comparado el método GRK de orden 3 que toma Álvarez en su tesis con el método EF-I-k-3, hemos calculado los errores numéricos en el punto $t = 1$ en la figura 7.3.

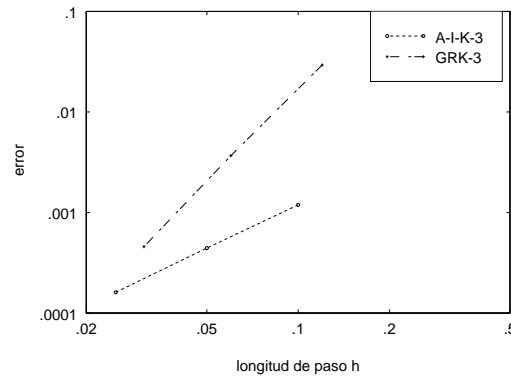


Figura 7.3 : Errores en la integración numérica en el problema 1.4.

El método GRK tiene claramente orden 3 al integrar este problema, por contra el nuevo método parece tener orden 2, es decir, parece claro que el parámetro introducido no es óptimo y por tanto hay margen de mejora, a pesar de todo los resultados son mejores para los tamaños de paso que se muestran en la figura 7.3.

Problema 1.5, hemos tomado este problema de [80]. Originalmente el problema es una ecuación de Allen-Cahn con condiciones de Dirichlet constantes

$$u_t = \varepsilon u_{xx} + u - u^3, \quad \varepsilon > 0, x \in [-1, 1],$$

$$u(x, 0) = 0,53x + 0,47 \sin(-1,5\pi x), u(-1, t) = -1, u(1, t) = 0,$$

con $\varepsilon = 0,001$ y hemos hecho los cálculos en el punto $t = 3$.

De nuevo hemos tomado como parámetro constante (al igual que en los otros

problemas)

$$A = \varepsilon(N + 1)^2 \begin{pmatrix} -2 & 1 & 0 & 0 & \dots & 0 & 0 \\ 1 & -2 & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & -2 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & -2 & 1 \\ 0 & 0 & 0 & 0 & \dots & 1 & -2 \end{pmatrix},$$

siendo esta vez $N + 1 = \frac{1}{\Delta x} = 21$.

En la figura 7.4 hemos comparado los nuevos métodos EF-I-k-3 y EF-I-k-4, con los métodos de orden 4 que se utilizan en [80]: Linearly Implicit, Split Step y el ETDRK4 (que se puede hallar previamente en [33]).

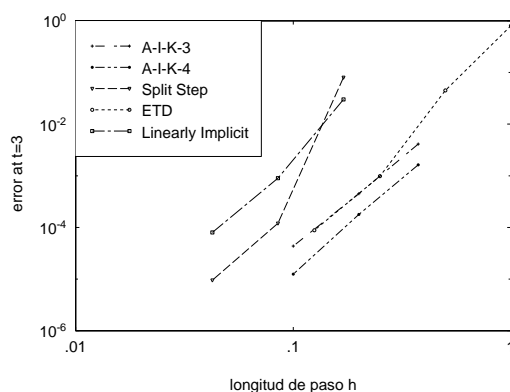


Figura 7.4 : Errores en la integración numérica en el problema 1.5.

Obsérvese que en todos los problemas de esta sección se ha tomado un único parámetro y por tanto sólo se ha calculado una vez los coeficientes del método, por esa razón la comparación error vs. número de evaluaciones de la función (o longitud de paso en el método) se puede considerar fiable siempre que los métodos sean todos implícitos.

7.3. Valores propios del jacobiano positivos.

Como ya se explicó anteriormente han aparecido varias publicaciones donde se mostraban ejemplos numéricos en los que no hay una clara correspondencia entre los autovalores que resultan de tomar el jacobiano de la función en un conjunto discreto de puntos y la solución del problema de valor inicial. También se han mostrado una serie de problemas donde existen valores propios del jacobiano positivos y sin embargo es posible obtener buenas aproximaciones con los métodos adaptados que se han construido en este trabajo.

A continuación mostramos una serie de PVI de esta clase:

Problema 2.1, el ya bien conocido problema (6.2.1) que aparece en [87] y cuya solución ya se ha comentado es $y_1(x) = e^{-x}$, $y_2(x) = -e^{-x}$.

Hemos integrado dicho problema en el intervalo $[0, 3]$ con varios de los integradores que posee el MATLAB así como con el MEBDF que se puede obtener de la página web de J. Cash. Y vamos a comparar dichos resultados con los obtenidos con el método explícito de Euler adaptado de 1 paso y orden 1. Si consideramos que el parámetro A es como en (6.2.2) ya hemos visto anteriormente que se tenía que verificar que $a = -1 + b$ y $c = 1 + d$, para el resto de parámetros hemos elegido diferentes valores:

i) Ejemplo 1: $h = 0,1$, $b = \frac{1}{3}$, $d = -\frac{1}{2}$.

ii) Ejemplo 2: $h = 0,1$, $b = 0$, $d = -1$.

Estos han sido parámetros con los que hemos obtenido los resultados que se muestran en el cuadro 7.3, los cuales han sido obtenidos en el punto $x = 3$, tomando las constantes $Rtol = Atol = 10^{-8}$ tanto en los integradores del

MATLAB como en el método MEBDF y siendo “N° de e.f.” el número de evaluaciones de la función.

Método	Error y_1	Error y_2	N° de e.f.
MATLAB ode23s	$1,2192 \times 10^{-4}$	$1,2192 \times 10^{-4}$	1802
MATLAB ode15s	0,0013	0,0013	144
MATLAB ode45	0,4859	0,4859	169
MATLAB ode113	0,0019	0,0019	69
MEBDF	0,011023	0,011023	67
Ej. 1 del Euler adaptado	$1,94289 \times 10^{-16}$	$6,245 \times 10^{-17}$	29
Ej. 2 del Euler adaptado	$> 10^{-20}$	$2,35922 \times 10^{-16}$	29

Cuadro 7.3 : Errores al integrar numéricamente el problema 2.1.

Obviamente, los resultados de los integradores clásicos no son nada buenos, la razón de estos resultados es que uno de los valores propios del jacobiano es positivo durante parte de este intervalo de integración y por eso los métodos tradicionales no siguen bien la solución del PVI.

Problema 2.2, consideramos el problema (6.2.3) (del que ya se habló anteriormente) y que aparece en [87] y cuya solución como ya se ha dicho es $y_1(x) = 2 + \sin(x)$, $y_2(x) = \cos(x)$.

Hemos integrado dicho problema en el intervalo $[0, 100]$ y hemos calculado los errores cometidos en el punto $x = 100$ en las dos componentes del problema tanto al integrar el problema con el método explícito de Euler adaptado, como con varios de los integradores del MATLAB y con el MEBDF (se ha tomado como constante $Rtol = Atol = 10^{-8}$). Estos resultados se muestran en el cuadro 7.4

Método	Error y_1	Error y_2	Nº e.f.
ode23s	$6,2458 \times 10^{-4}$	$3,6604 \times 10^{-4}$	117062
ode15s	0,0294	0,018	2539
ode45	0,3446	0,1245	6889
ode113	0,1173	0,0806	1131
MEBDF	$4,1092 \times 10^{-7}$	$2,38712 \times 10^{-7}$	1296
Euler adaptado	$4,44089 \times 10^{-15}$	$-3,77476 \times 10^{-15}$	1000

Cuadro 7.4 : Errores al integrar el problema 2.2 numéricamente.

En realidad el método Euler adaptado integra de forma exacta el problema, los errores que se cometen son los propios del redondeo al programar con el Mathematica, lo mismo sucede en el problema anterior.

Problema 2.3, se trata de un problema no lineal donde uno de los valores propios del jacobiano es positivo, este problema aparece en [38], p. 11. Se trata del ejemplo 1.1.5 donde se han tomado $d_1 = 5$, $d_2 = -20$, $w = -10$ y $\coth(\xi) = \frac{1}{5}$:

$$\begin{cases} y'(x) = \frac{5}{2}(3 + 5 \cos(20x) + \sin(20x))y(x) + (5 \cos(10x) - 5 \sin(10x))z(x), \\ z'(x) = (-5 \sin(10x)(5 \cos(10x) + \sin(10x))y(x) + \\ \quad + \frac{5}{2}(-3 + 5 \cos(20x) + \sin(20x))z(x), \\ y(0) = \frac{-13}{2}, \quad z(0) = -4, \end{cases}$$

y cuya solución es

$$\begin{cases} y(x) = 5e^{-10x}(\frac{-3}{2} \cos(10x) + \sin(10x)) + e^{-5x}(\cos(10x) - \sin(10x)), \\ z(x) = 5e^{-10x}(-\cos(10x) - \frac{3}{2} \sin(10x)) + e^{-5x}(\cos(10x) + \sin(10x)). \end{cases}$$

Los valores propios de la matriz $B(x)$ jacobiano de la función son $\lambda_1 = 5$, $\lambda_2 = -20$. Hemos resuelto este problema en el intervalo $[0, 5]$, y damos los errores para $x = 5$ según la norma $\| \cdot \|_2$, con algunos de los integradores del MATLAB ODE en la figura 7.5 (MATLAB ode45, MEBDF o Radau5 tienen malos resultados en este problema).

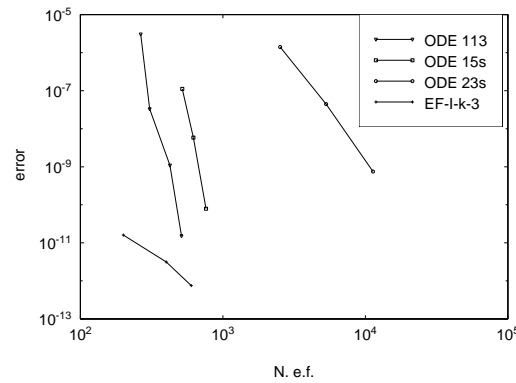


Figura 7.5 : Errores en la integración numérica del problema 2.3.

7.4. Problemas *stiff* oscilatorios o altamente oscilatorios.

Problema 3.1, hemos elegido como primer problema de este tipo el llamado B5 (que aparece en la página web de J. Cash, por ejemplo, y que es citado en una bibliografía muy extensa: [21], [22], [24], [47], [49], [100] ó [111], por citar algunos). Sin embargo nuestros métodos integran de forma exacta este problema por lo que hemos decidido cambiarlo ligeramente para tener el que llamaremos B5 modificado:

$$\begin{pmatrix} y_1'(x) \\ y_2'(x) \\ y_3'(x) \\ y_4'(x) \\ y_5'(x) \\ y_6'(x) \end{pmatrix} = \begin{pmatrix} -10y_1(x) + wy_2(x) \\ -wy_1(x) - 10y_2(x) \\ -4y_3(x) \\ -y_4(x) \\ -0,5y_5(x) \\ -0,1y_6(x) \end{pmatrix} + \begin{pmatrix} \varepsilon(\cos(x) + 10\sin(x)) \\ w\varepsilon\sin(x) \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

donde $w = 50$ y hemos considerado $\varepsilon = 10^{-3} \ll 1$ (es un problema bastante duro para bastantes métodos clásicos y sus resultados están íntimamente ligados a las propiedades de estabilidad absoluta de los métodos). Podemos observar que existen métodos que nos permiten integrar de forma exacta este nuevo problema sin más que añadir dos nuevas variables $y_7(x) = \sin(x)$,

$y_8(x) = \cos(x)$, e introducirlas en el problema, sin embargo, en este caso no lo vamos a hacer debido a que también nos interesa interpretar cuál va a ser el error frente a problemas no lineales donde no conozcamos la solución exacta y sí un parámetro Ah bastante cercano al óptimo.

La solución exacta de este problema modificado es

$$y(x) = \begin{pmatrix} e^{-10x}(\cos(wx) + \sin(wx)) + \varepsilon \sin(x) \\ e^{-10x}(\cos(wx) - \sin(wx)) \\ e^{-4x} \\ e^{-x} \\ e^{-\frac{x}{2}} \\ e^{-\frac{x}{10}} \end{pmatrix}.$$

En la figura 7.6 comparamos el mayor error cometido $\max_{i=1,\dots,n} \| y_i - y(x_i) \|_2$ con la longitud de paso (con h la longitud de paso fija en ambos) en el intervalo $x \in [0, 20]$, al utilizar el método explícito EF-I-r-2 y con el radauIIA también de dos pasos, pero de orden 3.

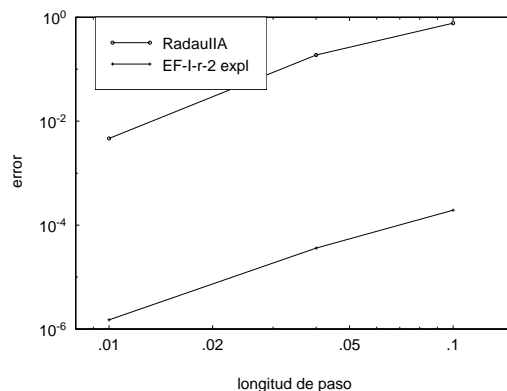


Figura 7.6 : Resultados numéricos al integrar el problema 3.1.

La ventaja del método explícito es que es muy rápido comparado con los implícitos, en la figura 7.7 comparamos el mayor error cometido $\max_{i=1,\dots,n} \| y_i - y(x_i) \|_2$ con el tiempo CPU empleado (con h la longitud de paso fija en

ambos) en el intervalo $x \in [0, 20]$, al utilizar los dos métodos recientemente citados.

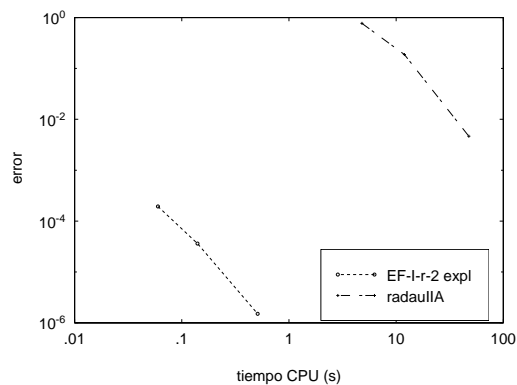


Figura 7.7: Resultados numéricos al integrar el problema 3.1. Comparación del error cometido con el tiempo CPU empleado.

Ya se puede observar que los nuevos métodos van a tener un gran comportamiento si somos capaces de elegir un buen parámetro Ah del método en problemas *stiff* donde sea necesario que los métodos tengan unas buenas propiedades de estabilidad.

Problema 3.2, se trata de un problema *stiff* oscilatorio (o altamente oscilatorio) no lineal:

$$y''(x) + 100^2 y(x) = 10^{-3} y(x)^3, \quad y(0) = 3, \quad y'(0) = 0,$$

integramos en el intervalo $x \in [0, 1]$. De esta forma nuestro sistema es

$$\begin{cases} y'(x) = z(x), & y(0) = 3, \\ z'(x) = -100^2 y(x) + 10^{-3} y(x)^3, & z(0) = 0. \end{cases}$$

Este sistema es conservativo, esto significa que la solución del mismo satisface que

$$E = \frac{y'^2}{2} + \frac{10^4 y^2}{2} + \frac{y^4}{4000} = 45000,02025.$$

En el punto $x = 0$, los valores propios del jacobiano son números imaginarios puros cercanos a $100i$ y $-100i$, lo cual supone un gran problema para muchos integradores que utilizan el método explícito de Euler en la inicialización.

En la figura 7.8 proporcionamos los errores cometidos en el punto $x = 1$ con la integral de energía al utilizar varios integradores.

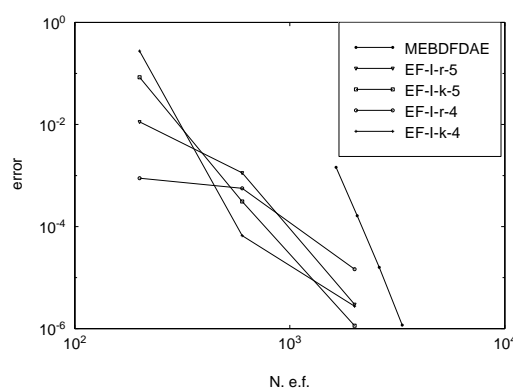


Figura 7.8 : Error en la integración numérica del problema 3.2.

Problema 3.3, el siguiente problema *stiff* oscilatorio (o altamente oscilatorio) no lineal:

$$y''(x) + 100^2 y(x) = 10^{-1} y(x)^5 y'(x)^2, \quad y(0) = 1, \quad y'(0) = 0, \quad (7.4.1)$$

con $x \in [0, 2]$. De esta forma el sistema que nosotros resolvemos sería

$$\begin{cases} y'(x) = z(x), & y(0) = 1, \\ z'(x) = -100^2 y(x) + 10^{-1} y(x)^5 z(x)^2, & z(0) = 0. \end{cases}$$

De nuevo en el punto $x = 0$, los autovalores del jacobiano son $100i$ y $-100i$. Este es el primer problema que tienen la mayor parte de los integradores (por ejemplo los que utilizan el método explícito de Euler en la inicialización). Posteriormente los valores propios del jacobiano llegan a tener parte real positiva, aunque los autovalores quedan muy cerca del eje imaginario, esta es otra razón por la que muchos de los métodos clásicos tienen graves problemas.

En la figura 7.9 mostramos el error que se produce en el punto $x = 2$, hemos considerado como valor correcto el logrado con el programa Mathematica tomando en la función NDSolve los parámetros AccuracyGoal $\rightarrow 15$, WorkingPrecision $\rightarrow 40$, que nos permiten una mayor aproximación, de esta forma obtenemos los valores $y(2) = 0,96220956268568564870546763582907518326$ y $z(2) = 27,27669809801177068508460384119524866$.

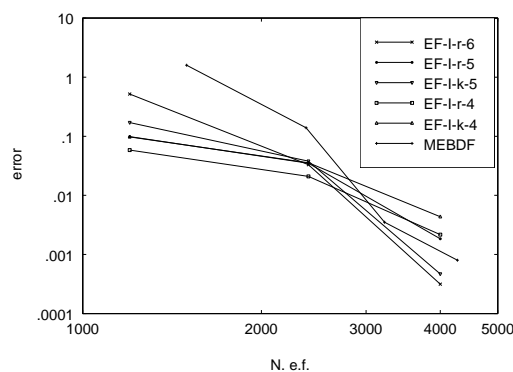


Figura 7.9 : Error en la integración numérica del problema 3.3.

Problema 3.4, el siguiente problema no lineal procede de combinar dos ecuaciones de Mathieu

$$\begin{cases} y''(x) + 2y'(x) + (101 + 0,02 \cos(2x))y(x) = 0, \\ z''(x) + 200z'(x) + (10100 + 0,02 \cos(2x))z(x) = 0, \\ y(0) = 0, \quad y'(0) = 10,00111100498693, \\ z(0) = 0, \quad z'(0) = 10,00111100498693. \end{cases}$$

Hemos integrado dicho test en el intervalo $[0, 5]$ y hemos considerado como solución en el punto $x = 5$ la proporcionada por el programa Mathematica a partir de la solución analítica que tienen cada una de las ecuaciones ya que son separables y que es $y(5) = -0,001769832951670131$, $z(5) = -1,8713875447601102 \times 10^{-218}$.

Consideramos que un buen parámetro del método podría ser

$$Ah = h \begin{pmatrix} 0 & 1 & 0 & 0 \\ -101 & -2 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -10100 & -200 \end{pmatrix},$$

y tomando estos valores nos encontramos que los resultados numéricos son los que se muestran en la figura 7.10

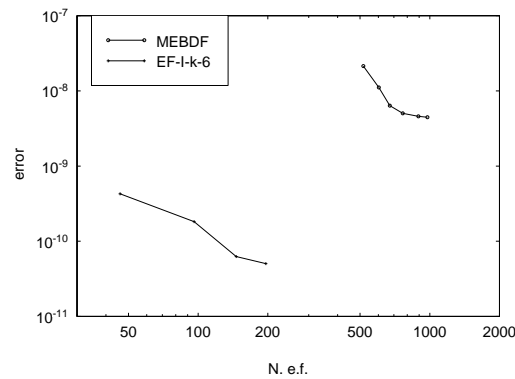


Figura 7.10 : Error en la integración numérica del problema 3.4.

7.5. Otras clases de problemas.

Problema 4.1, es un problema del tipo de Prothero-Robinson

$$y'(x) = \lambda(y(x) - g(x)) + g'(x), \quad x \in [0, 10]$$

donde $\lambda = -10^6$ y $g(t) = \sin(10x) + x$.

Este tipo de problemas son muy útiles para estudiar el orden de convergencia de nuestros métodos, en la figura 7.11 comparamos el número de evaluaciones de la función con el error en el punto $x = 10$, al utilizar los algoritmos MEBDF, A-k-1-r-6, A-k-3-r-4, A-k-5-r-2 (obsérvese que de esta forma la función $f(x) = -\lambda g(x) + g'(x)$ sólo hay que evaluarla una vez en cada iteración,

de alguna forma el algoritmo actúa como si fuera explícito), se ha tomado $\lambda h = -10^6 h$ como parámetro en el método.

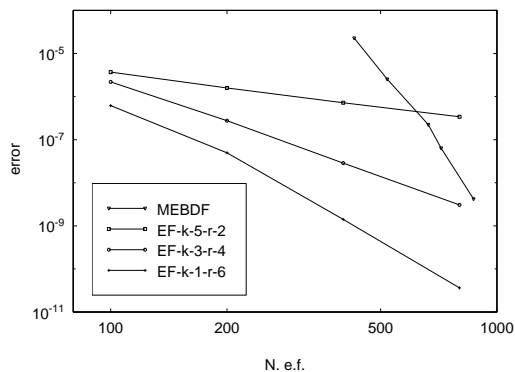


Figura 7.11 : Error en la integración del problema 4.1.

Problema 4.2, es un problema lineal que nos vuelve a servir para procurar estudiar el tipo de convergencia de nuestros métodos:

$$y'(x) = Ay(x) + f(x), \quad x \in [0, 10]$$

donde

$$A = \begin{pmatrix} -1000 & 1000 & 999 \\ -1000 & -1000 & 1000 \\ 0 & 0 & -1 \end{pmatrix},$$

$$f(x) = \begin{pmatrix} 1000 \sin(x) - 999 \cos(x) \\ 999 \sin(x) + 1000 \cos(x) \\ 0 \end{pmatrix},$$

y

$$y(0) = (2, 2, 1)^T.$$

La solución de este test es sencilla de calcular

$$y(x) = \begin{pmatrix} e^{-1000x}(\sin(1000x) + \cos(1000x)) + \sin(x) + e^{-x} \\ e^{-1000x}(\cos(1000x) - \sin(1000x)) + \cos(x) \\ e^{-x} \end{pmatrix}.$$

En la figura 7.12 compararemos los resultados en el punto $x = 10$ al utilizar los métodos A-k-1-r-6, A-k-3-r-4, A-k-5-r-2 y el MEBDF, al utilizar como

parámetro Ah (aunque es fácil observar que es posible obtener mejores resultados considerando otro método para obtener el valor de Ah , pero lo que buscamos aquí es estudiar el comportamiento del orden de los métodos).

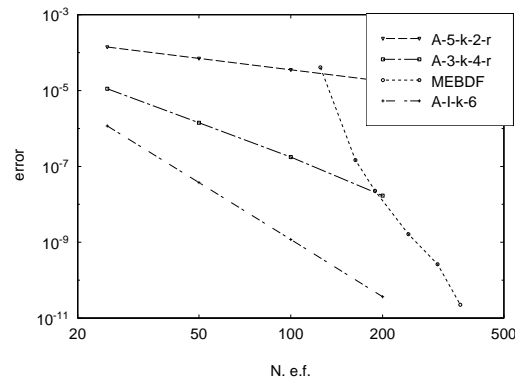


Figura 7.12: Error en la integración numérica del problema 4.2 al utilizar métodos de orden alto.

Mientras que en la figura 7.13 compararemos los resultados en el punto $t = 10$ al utilizar los métodos A-k-1-r-3, A-k-2-r-2, A-k-3-r-1 y radauIIA de dos pasos y paso constante.

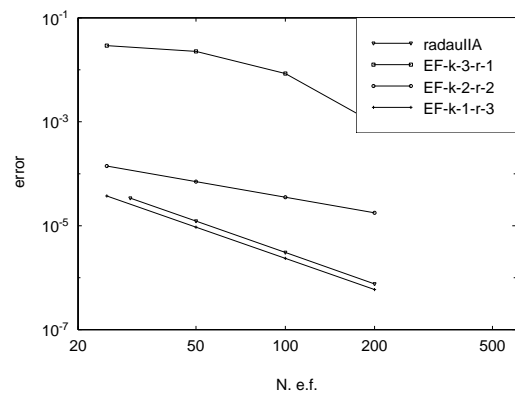


Figura 7.13: Error en la integración numérica del problema 4.2 al utilizar métodos de orden 3.

Problema 4.3, el problema *stiff* lineal (6.2.9) (ver [73] ó [87], por ejemplo).

Este problema se suele integrar en el intervalo $[0,10]$, y hemos considerado apropiado una longitud de paso $h = \frac{1}{10}$. En el cuadro 7.5 damos los errores absolutos en el punto $x = 10$ al integrarse con el nuevo método explícito de orden 2 EF-I-r-2 y los resultados obtenidos por Vanden Berghe et al. (VB3, ver [73]).

Método	Error y_1	Error y_2
VB3	$2,41 \times 10^{-4}$	$2,41 \times 10^{-4}$
EF-I-r-2 Explícito	$3,88 \times 10^{-13}$	$5,66 \times 10^{-13}$

Cuadro 7.5 : Errores al integrar numéricamente el problema 4.3.

Problema 4.4, el siguiente problema *stiff* es lineal, los valores propios del jacobiano son $\lambda_1 = -10^6 + 10^6i$, y $\lambda_2 = -10^6 - 10^6i$:

$$\begin{cases} y'(x) = -10^6 y(x) + 10^6 z(x) + f_1(x), & y(0) = 1, \\ z'(x) = 10^6 y(x) - 10^6 z(x) + f_2(x), & z(0) = 2. \end{cases}$$

donde $f_1(x) = 5(399999e^{-5x} + 200010 \cos(50x) + 200000 \sin(50x))$ y $f_2(x) = -5e^{-5x} + 10^6 \cos(50x) - 1000050 \sin(50x)$.

Dado que es lineal es sencillo obtener la solución del mismo y que resulta ser $y(x) = \sin(50x) + e^{-5x}$, $z(x) = \cos(50x) + e^{-5x}$, por esa razón es posible obtener el error con la norma $\| \cdot \|_2$. En el cuadro 7.6 comparamos los resultados obtenidos con dos de los nuevos métodos de orden 3, frente a los alcanzados con el radauIIA de 2 pasos (y también de orden 3) en el punto $x = 10$ y con paso fijo $h = 0,2$.

Problema 4.5, este ya es un problema *stiff* no lineal, que ya comentaron Liniger y Willoughby (ver [89]). Los valores propios del jacobiano en el punto $x = 0$ son $\lambda_1 = -1012$ y $\lambda_2 = -0,01$:

$$\begin{cases} y'(x) = 0,01 - (y^2(x) + 1001y(x) + 1001)p(y(x), z(x)), & y(0) = 0, \\ z'(x) = 0,01 - p(y(x), z(x))(1 + z^2), & z(0) = 0. \end{cases}$$

Método	Error	Número de pasos
radauIIA	$2,817 \times 10^{-5}$	50
EF-I-k-3 implícito	$5,551 \times 10^{-17}$	50
EF-I-r-3 implícito	$5,551 \times 10^{-17}$	50

Cuadro 7.6 : Errores al integrar numéricamente el problema 4.4.

donde $p(y(x), z(x)) = 0,01 + y(x) + z(x)$.

Se trata de un problema muy duro para los métodos clásicos como se puede comprobar a partir de los resultados obtenidos con el MEBDF y el radau5. Hemos considerado como intervalo de integración el cerrado $[0, 100]$, hemos calculado únicamente el error en el punto $x = 100$, donde hemos considerado como verdadera $y(100) = -0,99164207$ y $z(100) = 0,983336361$ (que es la obtenida con un Runge-Kutta de orden 4 y paso $h = 5 \times 10^{-4}$, ver Lambert y Sigurdson [86]).

Por ejemplo, con unas constantes $Rtol = Atol = 10^{-4}$ el MEBDF necesita nada menos que 160488 pasos y 394147 evaluaciones de la función. Además, el error con la norma $\| \cdot \|_2$ es $4,20447 \times 10^{-6}$. El Radau5 con las mismas constantes del error ($Rtol = Atol = 10^{-4}$) necesita más de 800000 evaluaciones de la función para integrar este problema tan sólo en el intervalo $[0, 40]$.

Por el contrario hemos observado que métodos de tipo *exponential fitting* sí son capaces de lograr buenos resultados con este problema.

Así, denotaremos en el cuadro 7.7, con LW3 el método de orden 3 propuesto por Liniger y Willoughby (ver [89]), mientras que e.f. significa el número de llamadas de la función.

Con JK4 denotaremos (en la tabla 7.8) el método de orden 4 que desarrollaron

Método	Error	Número de pasos
LW3 con $h = 0,0625$	$1,74974 \times 10^{-6}$	1600
EF-I-k-3 implícito con $h = 0,0625$	$9,1076 \times 10^{-8}$	1600 (4800 e.f.)
EF-I-r-3 implícito con $h = 0,0625$	$9,1087 \times 10^{-8}$	1600 (4800 e.f.)

Cuadro 7.7 : Errores al integrar numéricamente el problema 4.5.

Jackson y Kenue (ver [77]) y con C4 el método de orden 4 de Cash (ver [20]).

Método	Error	Número de pasos
JK4 con $h = 0,0625$	$4,12311 \times 10^{-8}$	1600
C4 con $h = 0,0625$	$1,45344 \times 10^{-7}$	1600
EF-I-k-4 implícito con $h = 0,0625$	$1,42176 \times 10^{-8}$	1600 (4800 e.f.)
EF-I-r-4 implícito con $h = 0,0625$	$1,42178 \times 10^{-8}$	1600 (4800 e.f.)

Cuadro 7.8 : Errores al integrar numéricamente el problema 4.5.

Mientras que llamaremos C5 al método de orden 4 de Cash (ver [18]), en el cuadro 7.9 mostramos los resultados.

Método	Error	Número de pasos
C5 con $h = 0,125$	$2,6303 \times 10^{-7}$	800
EF-I-k-5 implícito con $h = 0,125$	$1,89678 \times 10^{-8}$	800 (2400 e.f.)
EF-I-r-5 implícito con $h = 0,125$	$1,89676 \times 10^{-8}$	800 (2400 e.f.)

Cuadro 7.9 : Errores al integrar numéricamente el problema 4.5.

Problema 4.6, es un problema no lineal y no *stiff* (ver [139], por ejemplo):

$$y''(x) + y(x) - 10^{-4}y(x)^3 = 0, \quad y(0) = 1, \quad y'(0) = 0,$$

con $x \in [0, 100\pi]$. De esa forma nuestro sistema es

$$\begin{cases} y'(x) = z(x), & y(0) = 1, \\ z'(x) = -y(x) + 10^{-4}y(x)^3, & z(0) = 0. \end{cases}$$

Este sistema es conservativo, así pues las soluciones satisfacen la siguiente integral de energía

$$E = \frac{1}{2}(y^2 + y'^2) - \frac{10^{-4}}{4}y^4 = \frac{1}{2} - \frac{10^{-4}}{4} = 0,499975.$$

Para tener una buena idea del comportamiento de los nuevos métodos hemos calculado el error con la energía al integrar el problema con nuestros integradores frente a otros métodos de tipo *exponential fitting* (ver [129], [139] ó [140]) y paso fijo con orden elevado que demuestran la eficiencia que tienen los nuevos algoritmos también con problemas no *stiff* (en el cuadro 7.10). De esta forma llamaremos EF4 al esquema de 4 pasos que aparece en [139]), EF5 denotará al esquema de 5 pasos, ...

Problema 4.7, se trata de un problema *stiff* no lineal muy conocido como problema de Van der Pol (ver [14], [64], [76], [92], [145] ó [146]). Hemos elegido como constante $\mu = 5000$, en ese caso los autovalores en el punto $x = 0$ son aproximadamente $\lambda_1 = -1491,15$ y $\lambda_2 = -8,85292$:

$$\begin{cases} y'(x) = z(x), & y(0) = 2, \\ z'(x) = 5000(1 - y(x)^2)z(x) - y(x), & z(0) = -0,66. \end{cases}$$

Hemos resuelto este problema para $x \in [0, 1]$, con los códigos MEBDF y Radau5, con $Rtol = Atol = 10^{-11}$ y también con $Rtol = Atol = 10^{-12}$. Los resultados se muestran en el cuadro 7.11.

Para la solución correcta hemos considerado el valor obtenido en el punto $x = 1$ bien con el MEBDF o con el Radau5, con $Rtol = 10^{-14}$ y $Atol = 10^{-30}$.

Método	Error	N. e.f.
EF4	$1,5163 \times 10^{-6}$	3142
EF-k-1-r-4 implícito	$9,68829 \times 10^{-8}$	3142
EF-k-2-r-3 implícito	$2,36355 \times 10^{-7}$	3142
EF-k-3-r-2 implícito	$6,8005 \times 10^{-8}$	3142
EF-k-4-r-1 implícito	$5,50155 \times 10^{-8}$	3142
EF5	$7,5058 \times 10^{-7}$	3142
EF-k-1-r-5 implícito	$1,94708 \times 10^{-10}$	3142
EF-k-2-r-4 implícito	$5,41128 \times 10^{-10}$	3142
EF-k-3-r-3 implícito	$7,08154 \times 10^{-8}$	3142
EF-k-4-r-2 implícito	$1,38784 \times 10^{-7}$	3142
EF-k-5-r-1 implícito	$2,11148 \times 10^{-7}$	3142
EF6	$6,7664 \times 10^{-8}$	3142

Cuadro 7.10 : Errores al integrar numéricamente el problema 4.6.

Problema 4.8, se trata de un problema vectorial no lineal con valores propios complejos cercanos a $\lambda_1 = -1000 + i$ y $\lambda_2 = -1000 - i$:

$$\begin{cases} y'(x) = -1000y(x) + z(x) + 0,1 \sin(xy(x)), & y(0) = 1, \\ z'(x) = -y(x) - 1000z(x) + \cos(x), & z(0) = 1. \end{cases}$$

Vamos a utilizar este problema primero para comparar varios de los nuevos métodos de orden 6 y después para observar como funcionan los métodos exponenciales con dos parámetros.

Hemos tomado como solución en el punto $x = 10$ la proporcionada por el programa Mathematica tomando altos parámetros para los comandos AccuracyGoal (20 cifras) y PrecisionGoal (también 20 cifras), $y(10) = -8,40997665 \times 10^{-07}$, $z(10) = -8,39613869033 \times 10^{-04}$.

Método	Error $ y_n - y(1) $	Error $ z_n - z(1) $	N. e.f.
EF-I-k-3 implícito	$< \times 10^{-14}$	$< \times 10^{-18}$	200
MEBDF, $tol = 10^{-11}$	$1,29 \times 10^{-13}$	$1,22 \times 10^{-17}$	370
MEBDF, $tol = 10^{-12}$	$< \times 10^{-14}$	$< \times 10^{-18}$	497
Radau5, $tol = 10^{-11}$	$7,91 \times 10^{-14}$	$3,12 \times 10^{-13}$	945
Radau5, $tol = 10^{-12}$	$7,40 \times 10^{-14}$	$3,12 \times 10^{-13}$	1385

Cuadro 7.11 : Errores al integrar numéricamente el problema 4.7.

En la figura 7.14 mostramos el error con diversos métodos de orden alto en el último punto de integración ($x = 10$).

Hemos introducido como parámetro en el método

$$Ah = h \begin{pmatrix} -1000 & 1 \\ -1 & -1000 \end{pmatrix} \tag{7.5.1}$$

si bien es fácil observar que es posible obtener mejores resultados modificando ligeramente el problema o tomando otro parámetro.

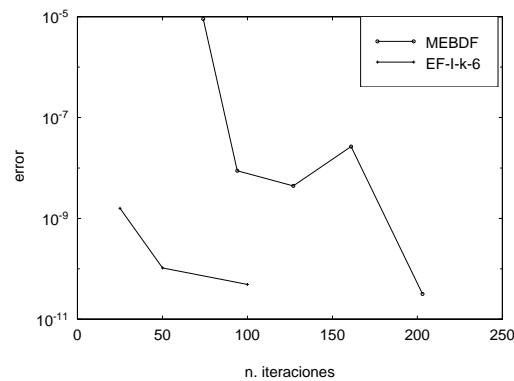


Figura 7.14 : Error en la integración numérica del problema 4.8.

Ahora vamos a tomar el método explícito exponencial con dos parámetros (Exp. 2 fr.), una de ellas se mantiene, Ah sigue siendo (7.5.1), mientras que vamos a tomar distintos valores para la otra, $B = \tau A$ (con $\tau \in R^+$). Los

resultados se muestran en el cuadro 7.12.

Método	Error $h = 0,1$	T. CPU (s)	Error $h = 0,01$	T. CPU (s)
Exp. 2 fr. $\tau = 4$	$4,96 \times 10^{-5}$	0.01	$4,86 \times 10^{-6}$	0.09
Exp. 2 fr. $\tau = 2$	$4,96 \times 10^{-5}$	0.01	$4,86 \times 10^{-6}$	0.09
EF-2-k-1-r	$4,92 \times 10^{-5}$	0.01	$4,32 \times 10^{-6}$	0.09
Exp. 2 fr. $\tau = 1/2$	$2,18 \times 10^{-5}$	0.01	$2,13 \times 10^{-6}$	0.09

Cuadro 7.12 : Errores al integrar numéricamente el problema 4.8.

Podemos observar que al ser Ah una matriz constante y como los coeficientes del método se calculan una única vez, no somos capaces de apreciar la diferencia en el tiempo empleado por la CPU entre el método EF-2-k-1-r y el método Exp. 2 fr.

Problema 4.9, consideramos el problema no lineal que cita Enright en [46], y que también aparece en [49], [100] ó [111] (es un problema de sistemas de control)

$$\begin{cases} y^4(x) = (y(x)^2 - \sin(y(x)) - \Gamma^4)y(x) + \left(\frac{y'(x)y''(x)}{1+y(x)^2} - 4\Gamma^3\right)y'(x) + \\ + (1 - 6\Gamma^2)y''(x) + (10e^{-y'''(x)^2} - 4\Gamma)y'''(x) + 1, \\ x \in [0, 1], \quad \Gamma = 100, \\ y(0) = 0, \quad y'(0) = 0, \quad y''(0) = 0, \quad y'''(0) = 0. \end{cases}$$

Hemos considerado como solución en el punto $x = 1$ la proporcionada por el MEBDFDAE con altos parámetros de tolerancia ($RTol = 10^{-20}$, $ATol = 10^{-50}$), $y(1) = 1,000000 \times 10^{-8}$, $y'(1) = -2,8 \times 10^{-23}$, $y''(1) = -6,4 \times 10^{-22}$, $y'''(1) = 1,1 \times 10^{-19}$.

Un buen parámetro para el método es

$$Ah = h \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -100^4 & -4 \times 100^3 & 1 - 6 \times 100^2 & 10 - 400 \end{pmatrix},$$

donde los valores propios (de la matriz) son $\lambda_1 = -130,923 + 68,7684i$, $\lambda_2 = -130,923 - 68,7684i$, $\lambda_3 = -64,0766 + 21,6023i$ y $\lambda_4 = -64,0766 - 21,6023i$ (por lo que puede haber dudas de si es un problema *stiff* oscilatorio).

En la figura 7.15 mostramos los resultados con métodos de orden alto.

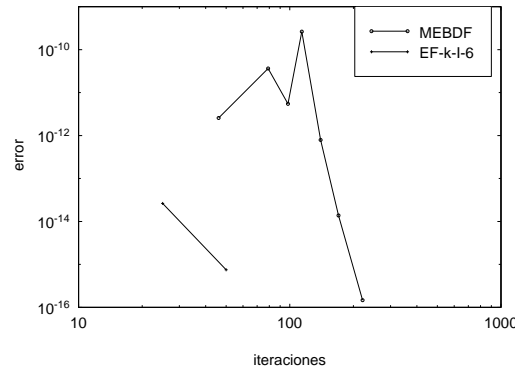


Figura 7.15 : Error en la integración numérica del problema 4.9.

De nuevo un buen test es considerar el método explícito exponencial con dos parámetros, con el mismo Ah y $B = \tau A$. Se muestran los resultados en el cuadro 7.13.

Método	Error $h = 0,04$	Tiempo CPU (s)
Doble exp. $\tau = 4$	0,00618	0.
Doble exp. $\tau = 2$	0,00425	0.
EF-2-k-1-r	0,00535	0.
Doble exp. $\tau = 1/2$	0,02705	0.

Cuadro 7.13 : Errores al integrar numéricamente el problema 4.9.

Problema 4.10, vamos a estudiar el PVI casi periódico que fue presentado por Stiefel y Bettis en [136] y que posteriormente ha aparecido en numerosas publicaciones ([30], [84], [88], [104], [112], [115], [142] ó [149], incluso en [116]

se muestran métodos que solucionan este problema de forma exacta):

$$y''(x) + y = 0,001e^{ix}, \quad y(0) = 1, \quad y'(0) = 0,9995i \quad (7.5.2)$$

y cuya solución exacta es

$$y(x) = (1 - 0,0005ix)e^{ix}.$$

Este problema no es *stiff*, pero nos permite ver de nuevo que este tipo de métodos funcionan mucho mejor que los métodos que tradicionalmente se utilizan para problemas *stiff* cuando el problema es oscilatorio. Representa el movimiento en una perturbación de una órbita circular en el plano complejo en el que la espiral lentamente se mueve hacia afuera.

El problema de valor inicial (7.5.2) se puede expresar como

$$\begin{cases} y_1'(x) = y_2(x), & y_1(0) = 1, \\ y_2'(x) = -y_1(x) + 0,001 \cos(x), & y_2(0) = 0, \\ y_3'(x) = y_4(x), & y_3(0) = 0, \\ y_4'(x) = -y_3(x) + 0,001 \sin(x), & y_4(0) = 0,9995. \end{cases}$$

(Aquí otra vez se pueden introducir dos nuevas variables para resolver de forma exacta dicho PVI).

Y la verdadera solución es

$$\begin{cases} y_1(x) = \cos(x) + 0,0005x \sin(x) \\ y_3(x) = \sin(x) - 0,0005x \cos(x). \end{cases}$$

Los valores propios de este problema son imaginarios puros, como ya hemos visto es una razón suficiente para provocar problemas en muchos integradores de problemas *stiff*.

Aquí hemos comparado el método explícito EF-1-k-2-r con el Radau5 y con el MEBDF. Y mostramos el error en el punto $x = 40\pi$ en el cuadro 7.14.

	Error	Número de e.f.
EF-1-k-2-r	$1,284 \times 10^{-5}$	480
MEBDF	$2,555 \times 10^{-5}$	708
Radau5	$6,953 \times 10^{-5}$	1689

Cuadro 7.14 : Errores al integrar numéricamente el problema 4.10.

Este problema nos sirve para mostrar como los métodos adaptados aquí presentados también pueden proporcionar muy buenos resultados frente a problemas que no son *stiff* y donde los autovalores del jacobiano son imaginarios puros, problemas donde se podría pensar que hay métodos que deberían funcionar mucho mejor.

Por eso hemos resuelto dicho problema con los métodos multipaso simétricos propuestos por Lambert and Watson ([88]), así como con las fórmulas de Stormer-Cowell de cinco pasos (ambos métodos tienen orden 6). Y hemos comparado los errores en el cuadro 7.15.

	EF-1-k-2-r explícito	Simétrico orden 6	Stormer-Cowell orden 6
$h = \frac{\pi}{4}$	$9,544 \times 10^{-5}$	0,031272	0,048014
$h = \frac{\pi}{12}$	$1,284 \times 10^{-5}$	$3,321 \times 10^{-5}$	$7,325 \times 10^{-5}$

Cuadro 7.15 : Errores al integrar numéricamente el problema 4.10.

Problema 4.11, consideramos el PVI no lineal (6.2.4), en el punto inicial el jacobiano de la función tiene como autovalores $\lambda_1 = -1004$, $\lambda_2 = -1,00199$ por lo que el sistema puede ser considerado *stiff*.

En la tabla 7.16 comparamos los resultados obtenidos en el punto $x = 10$ con el RadauIIA de dos pasos (paso constante) y el EF-1-k-3-r, con paso $h = 1/10$.

Método	error
EF-1-k-3-r	0
RadauIIA	$6,1476 \times 10^{-9}$

Cuadro 7.16 : Errores al integrar numéricamente el problema 4.11.

Problema 4.12, integramos el PVI (6.2.5) en el intervalo $[0, 10]$, se trata de un problema no lineal del tipo Prothero-Robinson de segundo orden, por tanto los dos valores propios del jacobiano son 0, por lo que si tomáramos el jacobiano de la función como parámetro tendríamos los BDF clásicos, sin embargo en el capítulo anterior ya se mostró una buena vía de hallar el parámetro del método.

En la tabla 7.17 comparamos los resultados obtenidos en el punto $x = 5$ con el RadauIIA de dos pasos (paso constante) y el EF-2-k-2-r, con paso $h = 1/20$.

Método	error
EF-2-k-2-r	0
RadauIIA	$1,7744 \times 10^{-6}$

Cuadro 7.17 : Errores al integrar numéricamente el problema 4.12.

Problema 4.13, hemos integrado el ya bien conocido problema E5 en el intervalo $[0, 10^7]$, pero reduciendo el sistema (6.2.6) a través de la igualdad $y_2(x) = y_3(x) + y_4(x)$ a un sistema de tres ecuaciones diferenciales. La solución exacta en el punto $x = 10^7$ la obtenemos de la página web de E. Hairer

<http://www.unige.ch/~hairer/testset/testset.html>

En este ejemplo hemos comparado el tiempo empleado por la CPU al ser

integrado por el radauIIA de dos pasos y el EF-1-k-3-r, pues los dos métodos han sido escritos en el mismo lenguaje, utilizando la misma precisión de la máquina y utilizando el mismo procesador un Intel Pentium M 715 a 1.50 GHz, como en todos los casos en los que se ha hecho la comparativa error-tiempo empleado. Dichos resultados se muestran en la figura 7.16.

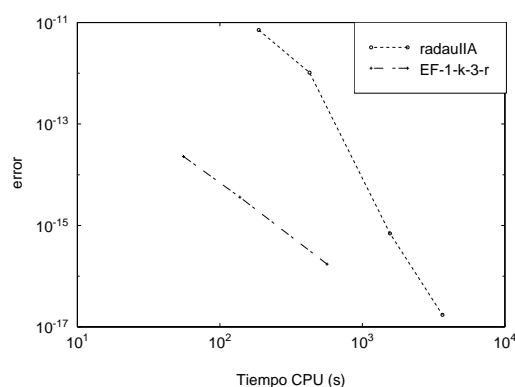


Figura 7.16: Error en la integración numérica del problema 4.13. Comparación del error cometido con el tiempo CPU empleado.

7.6. Ejemplos resueltos en paso variable y métodos BDF-Runge-Kutta.

Como ya se ha comentado, estudiar la estabilidad de los métodos en paso variable no resulta nada sencillo, por esa razón en esta sección únicamente vamos a tratar problemas donde el parámetro, $\lambda h_n \in R^-$, verifica que $\lambda h_n \ll -1$, por lo que podremos mantener durante varias iteraciones un crecimiento del paso relativamente alto, sin tener que preocuparnos en demasía por la estabilidad.

Los ejemplos numéricos que vamos a mostrar son ambos de la clase Prothero-Robinson: $y'(x) = \lambda(y(x) - g(x)) + g'(x)$, donde $\lambda \ll -1$ y con solución en los dos ejemplos $y(x) = e^{\lambda x} + g(x)$.

Para nuestros problemas hemos elegido que:

Problema 5.1: $\lambda = -10^8$ y $g(x) = \sin(x)$.

Problema 5.2: $\lambda = -10^6$ y $g(x) = e^{-x} + \cos(x) + 2x$.

Hemos utilizado el algoritmo en paso variable de 6 pasos A-I-k-6. No hemos permitido $h_n > 0'5$, $\frac{h_{n+1}}{h_n} < 0'2$, ni $\frac{h_{n+1}}{h_n} > 4'5$.

Para calcular la longitud del paso en la siguiente iteración vamos a utilizar la fórmula (similar a la que aparece en [16] ó [87])

$$h_{n+1} = \alpha h_n (1 + h_n + h_n^2) \left(\frac{tol}{estim} \right)^{1/7}, \quad (7.6.1)$$

donde *tol* es la tolerancia o máximo error permitido que introduce el programador y hemos considerado $\alpha = 0,95$. El valor *estim* que aparece en la fórmula, claro está, será el valor de truncado local que se ha calculado en el capítulo dedicado a los métodos de orden exponencial 1 (se utiliza la fórmula (2.5.3), considerando que $\nabla_P y(x) = f(x) = -\lambda g(x) + g'(x)$), pero en vez de calcular β_6^0 , lo aproximamos por $\theta_{0,6} = \frac{1}{6}$, para hacer más sencillos los cálculos.

Hemos presentado los resultados numéricos logrados con el MEBDF y el método adaptado en las siguientes figuras. En las dos primeras figuras comparamos el número total de pasos (figura 7.17) y el número de evaluaciones de la función (figura 7.18) en comparación con el error cometido al integrar el problema 5.1; en las otras dos figuras (7.19 y 7.20) hacemos lo propio con el problema 5.2.

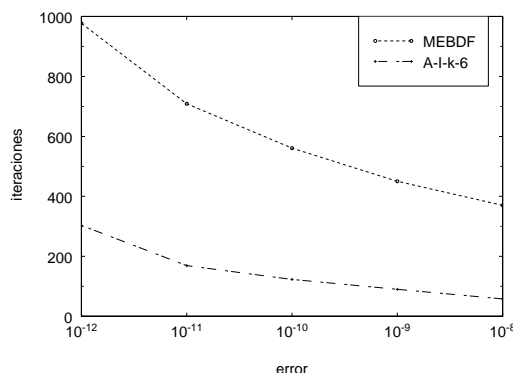


Figura 7.17 : Resultados al integrar numéricamente el problema 5.1.

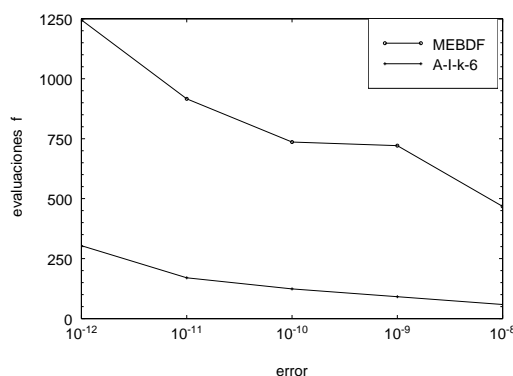


Figura 7.18 : Resultados al integrar numéricamente el problema 5.1.

En esta sección también vamos a incluir los métodos *exponential fitting* BDF-Runge-Kutta (que denotaremos a partir de ahora con EF-BDF-RK) . Estos métodos de un sólo paso no van a tener las dificultades en cuanto al estudio de la estabilidad de los métodos adaptados de tipo BDF de paso variable. Sin embargo aquí nos vamos a quedar con los algoritmos de paso fijo, ya que al variar el paso cambian los coeficientes y es preciso establecer una relación de cómo cambian estos al variar h a fin de que los algoritmos sean eficientes.

Problema 5.3, hemos integrado el conocido problema de Van der Pol que aparece en [64], pp. 403 y 404:

$$\begin{cases} y'(x) = z(x), & y(0) = 2, \\ z'(x) = 10^5((1 - y(x)^2)z(x) - y(x)), & z(0) = -0,6666654321121172. \end{cases}$$

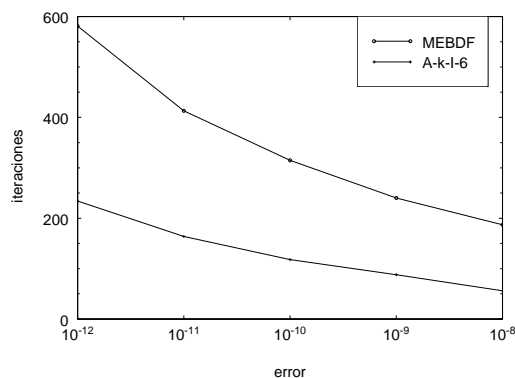


Figura 7.19 : Resultados al integrar numéricamente el problema 5.2.

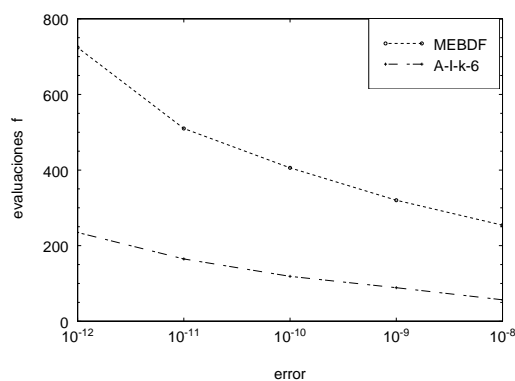


Figura 7.20 : Resultados al integrar numéricamente el problema 5.2.

En el cuadro 7.18 evaluamos el error cometido en el punto $x = 0.5$ (calculamos $\| \cdot \|_2$ del error en el punto final), que es el elegido por Hairer y Wanner en su libro, por diferentes métodos de orden 3 (el RadauIIA con $s = 2$) y 4 (Gauss con $s = 2$ y EF-BDF-RK con $s = 4$). Hemos tomado como parámetro del método el jacobiano de la función, sin embargo ya hemos visto que no suele ser la mejor forma de elegir el parámetro del método, pero esto nos sirve para verificar que la constante de error del método EF-BDF-RK-4 es muy pequeña y nos va a permitir obtener grandes resultados con este método.

Problema 5.4, de nuevo hemos integrado el PVI no lineal (6.2.4) en el intervalo $[0, 10]$ y que también se ha llamado Problema 4.13. Hemos elegido

Método	error
EF-BDF-RK-4 $h = 10^{-2}$	$8,19227 \times 10^{-11}$
RadauIIA $h = 5 \times 10^{-3}$	$2,90075 \times 10^{-9}$
RadauIIA $h = 10^{-3}$	$4,82439 \times 10^{-11}$
Gauss $h = 5 \times 10^{-3}$	$2,83671 \times 10^{-6}$
Gauss $h = 10^{-3}$	$1,55199 \times 10^{-8}$

Cuadro 7.18 : Errores al integrar numéricamente el problema 5.3.

el mismo parámetro que se tomó entonces y cuya explicación se encuentra en el capítulo 6. De nuevo el método es exacto para este problema y el error es 0, esto significa que las vías de obtención del parámetro Ah van a ser las mismas que las explicadas para los métodos adaptados de tipo BDF.

El hecho de que los métodos sean *exponential fitting* unido a que la constante del error sea muy pequeña hacen de este método EF-BDF-RK-4 muy prometedor al igual que los algoritmos de este tipo y orden superior.

Capítulo 8

Conclusiones.

Antes de este trabajo, el estudio de los métodos *exponential fitting* de tipo BDF era bastante reducido (ver [74]) y no había buenos análisis de las propiedades de estabilidad de los algoritmos que se habían propuesto, a pesar de que éstas son fundamentales para entender el comportamiento de los métodos al resolver PVI de tipo *stiff*.

Aquí se han generado diferentes familias de algoritmos *exponential fitting* y adaptados de tipo BDF y de tipo BDF-Runge-Kutta y se han mostrado las propiedades de consistencia y estabilidad de los mismos.

En este trabajo se comenzó generando los métodos BDF de orden exponencial 1 de paso fijo. Aunque los algoritmos explícitos tienen problemas de estabilidad, los métodos implícitos tienen buenas propiedades y es posible lograr resultados muy buenos enfrentándose incluso a métodos de paso variable y a veces a métodos de orden superior. También se han mostrado varias formas de construir los métodos de esta familia en paso variable, pero estos tienen varios inconvenientes: por una parte el coste computacional puede crecer de forma considerable si la dimensión del problema es grande y además el estudio de la estabilidad es complejo.

Posteriormente, se han mostrado los algoritmos de orden algebraico 1, que tienen mejores propiedades de estabilidad que los anteriores, destacando en este aspecto el explícito de orden 2, pero cuyos resultados numéricos dependen en mayor medida de una “adecuada” elección del parámetro del método. Aparte de estas dos familias de métodos se han construido varias más clases de *exponential fitting* de tipo BDF y se ha mostrado que estos los algoritmos implícitos tienen buenas propiedades al menos hasta orden 6. Pudiendo lograrse buenos resultados numéricos frente a problemas muy complejos.

Para ello resulta fundamental investigar como hallar un parámetro “adecuado” para el problema. En muchos casos se puede escoger el jacobiano de la función, y de hecho ésta es la forma que más se repite en la literatura científica para elegirlo. No obstante ya hemos explicado que no siempre da buenos resultados y hemos presentado ejemplos en los que se pueden tomar mejores parámetros. Con problemas *stiff* oscilatorios o altamente oscilatorios no lineales que tienen una ecuación lineal equivalente se han mostrado varios ejemplos en los que, calculando únicamente una vez el parámetro, se pueden conseguir resultados muy positivos, mejores y obtenidos con menor coste que si hubiéramos elegido el jacobiano.

Sin embargo, se han mostrado otro tipo de problemas donde el jacobiano de la función es más traicionera aún. Esos son los PVI que tienen un jacobiano de la función con valores propios positivos y sin embargo las soluciones del problema no tienen ninguna exponencial positiva. Estos son ejemplos numéricos que pueden resultar muy complejos para una gran cantidad de métodos clásicos, también para aquellos integradores que logran buenos resultados con problemas *stiff*, sin embargo, aquí se ha mostrado que algunos de estos ejemplos se pueden integrar perfectamente con los métodos nuevos, sólo hay

que elegir un buen parámetro y se han mostrado diferentes vías para hacer este cálculo: a partir de truncar el error y calculando aproximaciones a la solución mediante desarrollos de Taylor.

Otra familia de métodos que se ha mostrado como construir ha sido la de los *exponential fitting* con varios parámetros, estos algoritmos pueden favorecer la integración de problemas con alta variabilidad o con problemas que den complicaciones con la estabilidad, pero cuanto mayor sea el número de frecuencias más complicado será hacer un estudio detallado de las propiedades del método.

También se han desarrollado métodos *exponential fitting* de tipo BDF-Runge-Kutta hasta el orden 4, estos métodos se han mostrado muy prometedores ya que tienen unas constantes del error de truncado muy pequeñas y por esa razón son capaces de lograr grandes resultados.

Precisamente los resultados numéricos son la mayor garantía de que los métodos aquí presentados son eficaces, pues se ha procurado mostrar una relación amplia de diferente tipo de tests, la mayoría de ellos bastante habituales, y que los integradores con los que se comparaba tuvieran un amplio reconocimiento en la literatura científica, incluso se ha llegado a comparar con algoritmos de paso variable, a pesar de todo los resultados finales son muy interesantes.

Apéndice A

0-estabilidad de los métodos de orden exponencial 1.

Vamos a comenzar por estudiar la 0-estabilidad en los algoritmos adaptados y posteriormente daremos paso a los algoritmos *exponential fitting*.

Empezaremos por el método implícito de dos pasos, que escribiremos como

$$y_{n+1} = c_1 y_n + c_2 y_{n-1} + h_n c_0 f_{n+1},$$

su polinomio característico asociado es

$$x^2 - c_1 x - c_2 = 0$$

Dicho polinomio tiene dos raíces diferentes,

$$r_1 = e^u, \quad r_2 = \frac{1 - e^u + e^u u}{1 - e^u - u + 2e^u u}$$

siendo $u = \lambda h$. Obviamente si $\lambda \in C^-$ r_1 tiene módulo menor que 1. Sin embargo, la segunda raíz tiene regiones para $u \in C^-$ donde tiene módulo mayor que 1.

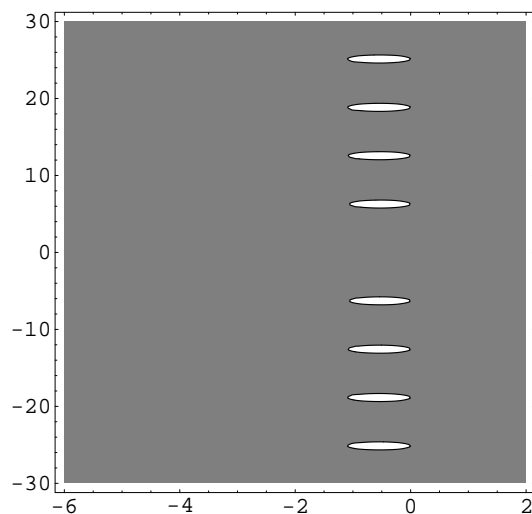


Figura A.1: Región de 0-estabilidad (en gris) de la segunda raíz del método implícito de dos pasos. El eje horizontal representa $Re(\lambda h)$, mientras que el vertical representa $Im(\lambda h)$.

Por supuesto, es necesario estudiar esas zonas y por eso en la figura A.1 mostramos dichas regiones.

Podemos observar que las regiones de 0-estabilidad incluyen todo el eje real negativo, sin embargo, para ciertos valores complejos hay longitudes de paso para las cuales el método será 0-inestable aun cuando $\lambda \in C^-$, si bien se puede observar que decreciendo el paso, será posible que el método sea 0-estable.

Muy diferente es el caso del correspondiente método explícito de dos pasos. Si consideramos la expresión

$$y_{n+1} = c_1^* y_n + c_2^* y_{n-1} + hc_0^* f_n,$$

para el método explícito, hemos de comprobar en que condiciones las raíces del polinomio

$$x^2 - c_1^* x - c_2^* = 0$$

tienen módulo 1, pero ahora una de las raíces resulta ser $-\frac{-1+e^u-u}{1-e^u+e^u u}$, y dicha raíz es mayor que 1 para toda $u = \lambda h \in C^-$. Llegando al siguiente resultado:

Teorema A.0.1. *El método adaptado explícito de dos pasos con parámetro u de forma que $Re(u) \neq 0$ es 0-inestable.*

Algo que resulta un tanto chocante, ya que es ampliamente conocido que cuando $u = 0$ (el caso del BDF clásico de dos pasos) el método sí es 0-estable.

Vamos a considerar el método de tres pasos como

$$y_{n+1} = \bar{c}_1 y_n + \bar{c}_2 y_{n-1} + \bar{c}_3 y_{n-2} + h_n \bar{c}_0 f_{n+1}.$$

Como antes debemos encontrar las regiones del plano, para las cuáles las raíces del polinomio

$$x^3 - \bar{c}_1 x^2 - \bar{c}_2 x - \bar{c}_3 = 0$$

tienen módulo menor que 1. Dicho polinomio tiene 3 raíces, la primera es $r_1 = e^u$ ($u = \lambda h$), mientras que las otras dos son complejas conjugadas. Aunque r_1 tiene módulo menor que 1 siempre que $\lambda \in C^-$, las otras dos presentan zonas del plano (en función del parámetro u) donde las fórmulas BDF adaptadas no son 0-estables. En la figura A.2 representamos la región de 0-estabilidad del algoritmo.

El mismo procedimiento se sigue con el correspondiente método explícito de tres pasos, llegando al siguiente resultado:

Teorema A.0.2. *Un método explícito adaptado de tres pasos es 0-inestable.*

A partir de ahora, seguiremos únicamente con los métodos implícitos, ya que los explícitos como ya se ha observado, muestran malas propiedades de estabilidad.

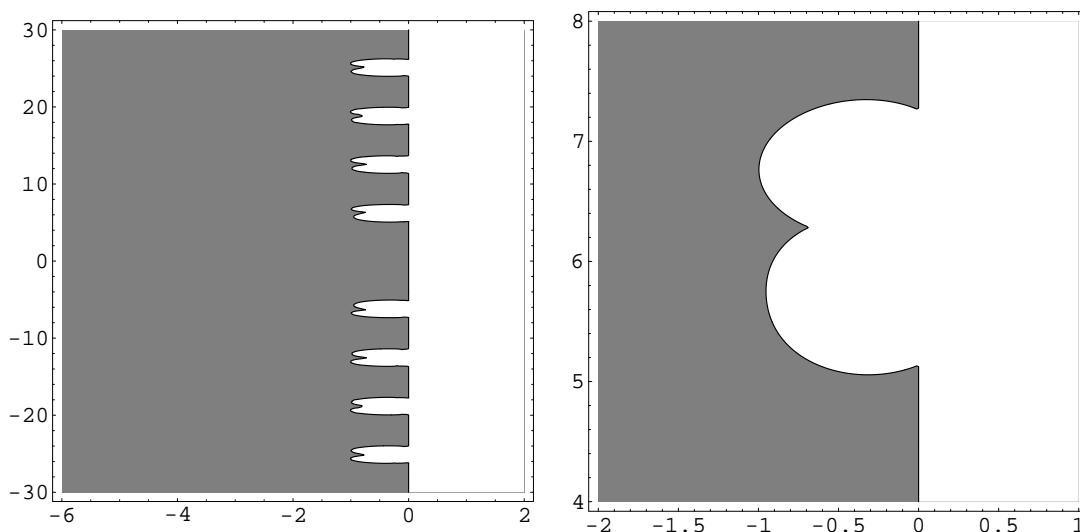
(a) *Region de 0-estabilidad (en gris).*(b) *Detalle de la zona.*

Figura A.2: Región de 0-estabilidad (en gris) del método implícito de tres pasos. El eje horizontal representa $Re(\lambda h)$ y el vertical representa $Im(\lambda h)$.

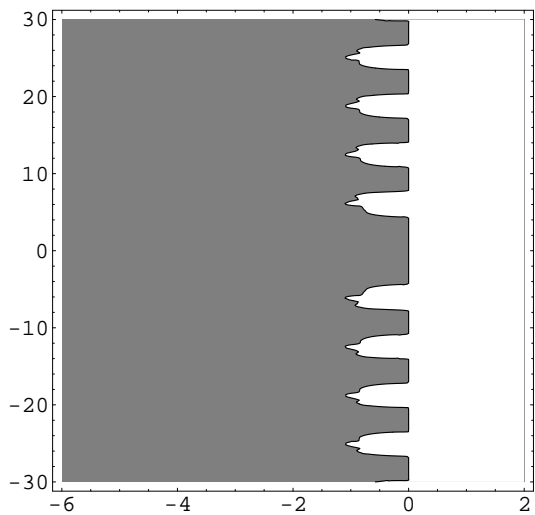
A continuación, en la figura A.3 presentamos las regiones donde los AM implícitos de 4 a 7 pasos (los denotaremos A-I-k-4, el método de 4 pasos, A-I-k-5 el de 5 pasos, ...) son 0-estables (la 0-estabilidad corresponde a la parte en gris).

Como sucede con los BDF clásicos el método implícito de 7 pasos presenta dificultades, como se puede observar en la figura A.4:

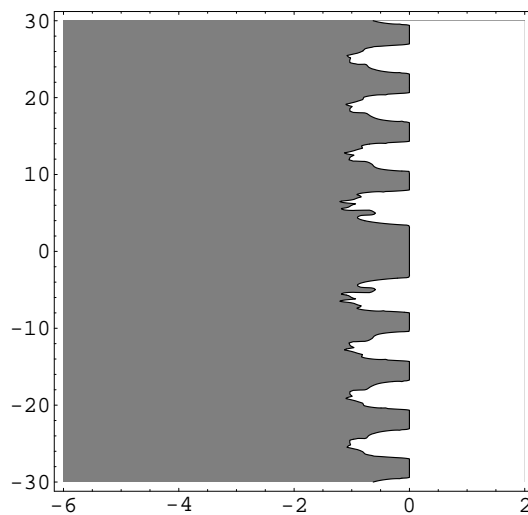
Teorema A.0.3. *Un método implícito adaptado de siete pasos (A-I-k-7) no es 0-estable en un entorno del origen.*

Por eso, aunque para ciertos valores de λh el método es 0-estable y los resultados con el método de 7 pasos son buenos, sin embargo, al decrecer el paso el método se vuelve inestable.

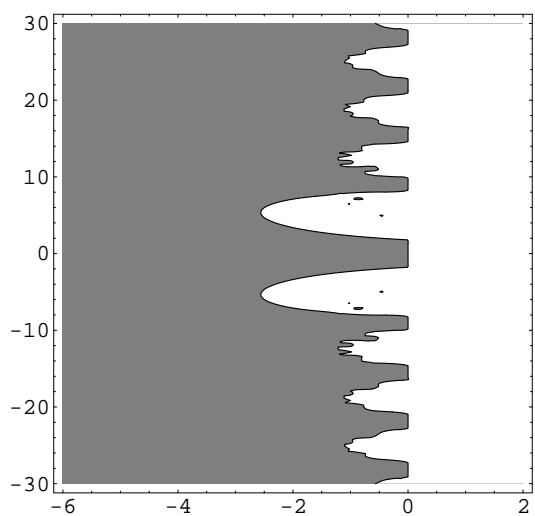
De esta forma, por ejemplo, el siguiente problema de valor inicial:



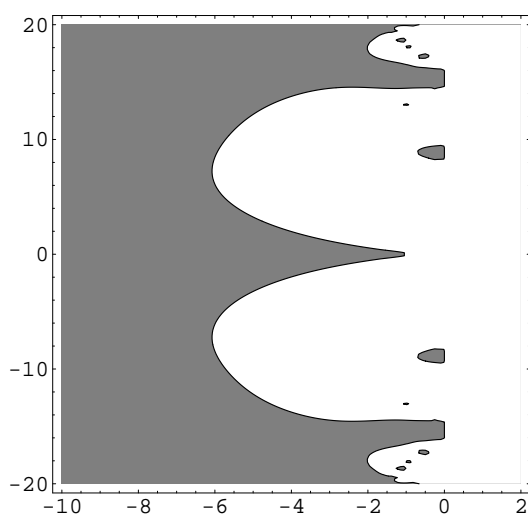
(a) $A-I-k-4$.



(b) $A-I-k-5$.



(c) $A-I-k-6$.



(d) $A-I-k-7$.

Figura A.3: Regiones de 0-estabilidad (en gris) de los métodos implícitos adaptados. Los ejes horizontal y vertical representan $Re(\lambda h)$ y $Im(\lambda h)$ respectivamente.

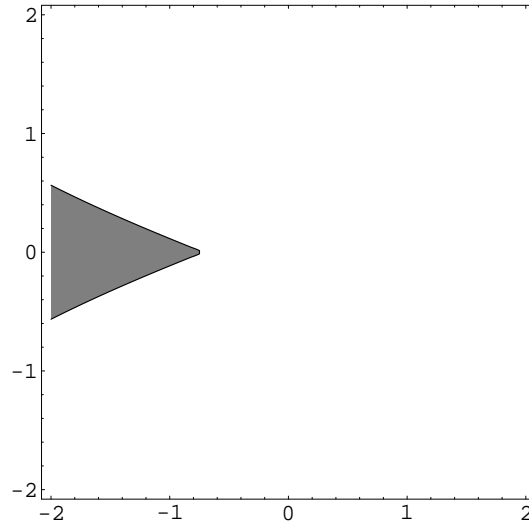


Figura A.4: Detalle de la 0-estabilidad (en gris) del método implícito adaptado de siete pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

$$y'(x) = -100(y(x) - \sin(x)) + \cos(x), \quad y(0) = 0, \quad (\text{A.0.1})$$

cuya solución es $y(x) = \sin(x)$, el método adaptado con $\lambda = -100$ y $h = 0,02$, al cabo de 5000 iteraciones, esto es, para el punto $x = 100$ da un error de $-6,28386 \times 10^{-14}$ ($\lambda h = -2$, el método es 0-estable), sin embargo, con la misma λ , pero con $h = 0,002$ al cabo de 50000 iteraciones, en el mismo punto $x = 100$ da un error aproximado de $2,9735 \times 10^{327}$, debido a que para esos parámetros el método no es 0-estable.

Similares resultados podemos encontrar con los métodos implícitos de 8 y 9 pasos, como podemos observar en la figura A.5.

Teorema A.0.4. *Los métodos implícitos adaptados de ocho y nueve pasos (A-I-k-8 y A-I-k-9) no son 0-estables en un entorno del origen.*

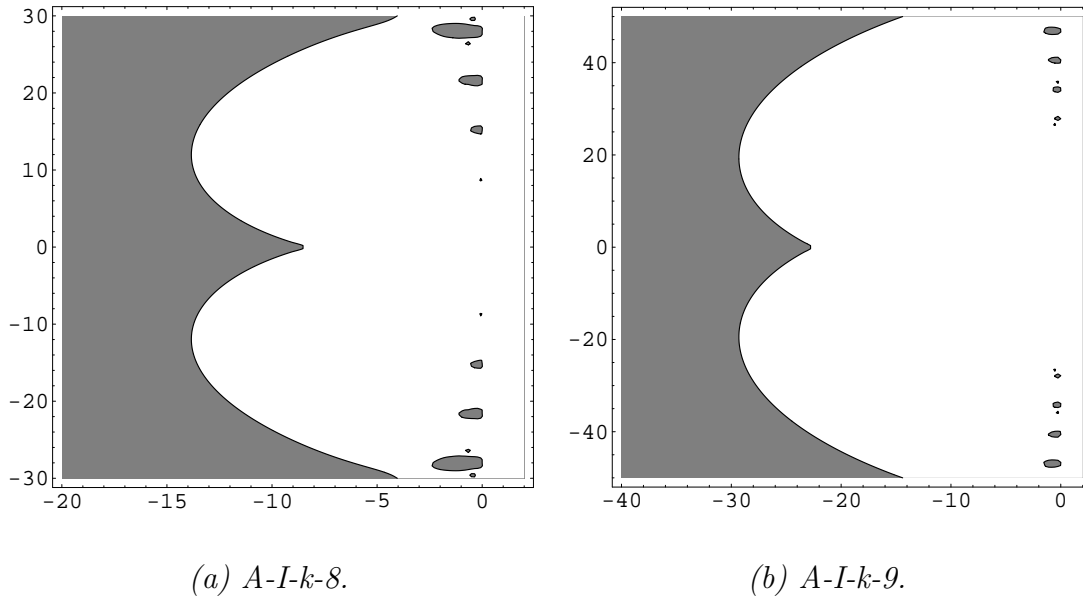


Figura A.5: Regiones de 0-estabilidad (en gris) de los métodos implícitos adaptados de ocho y nueve pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

De nuevo, es posible encontrar problemas para los cuales para determinada longitud de paso h los métodos de ocho y nueve pasos obtienen buenos resultados, sin embargo, al decrecer el paso los métodos se vuelven inestables.

Como ya se ha adelantado, al ser los coeficientes de los métodos EFM diferentes a los coeficientes de los métodos AM, las regiones de 0-estabilidad cambian, aún cuando el algoritmo en sí sea el mismo.

Mientras que el teorema A.0.1 nos decía que el método adaptado explícito de dos pasos con parámetro u de forma que $Re(u) \neq 0$ es 0-inestable, en cambio el método EFM explícito de dos pasos es 0-estable siempre que el parámetro $u = \lambda h \in C^-$. Aparentemente de las regiones de 0-estabilidad puede parecer que los algoritmos EFM y AM son totalmente diferentes, sin embargo son los mismos métodos. En el apéndice B dedicado a la estabilidad absoluta esta

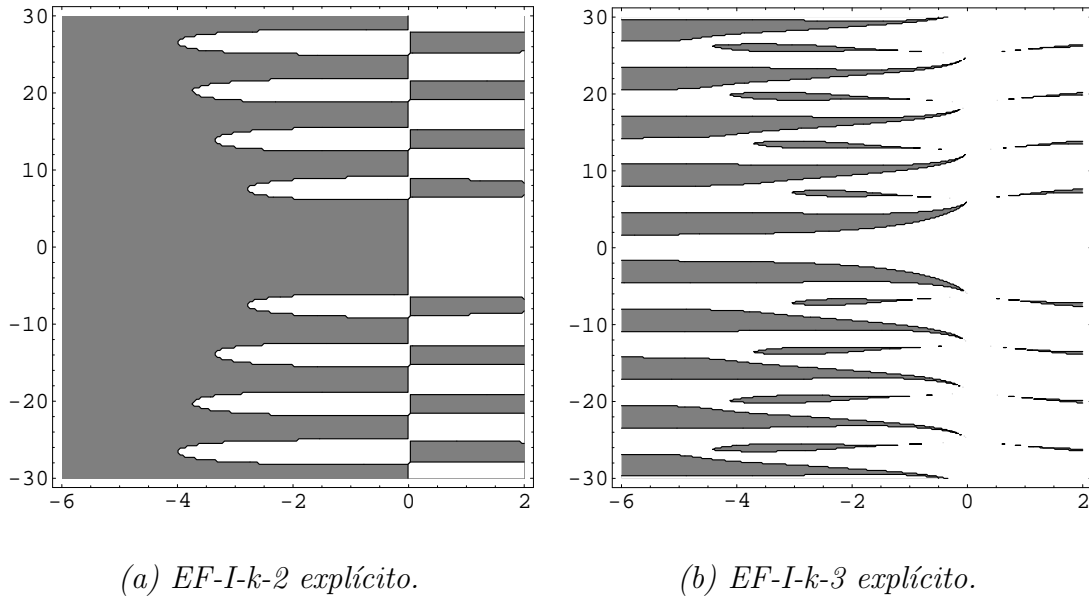


Figura A.6: Regiones de 0-estabilidad (en gris) de los métodos explícitos de tipo *exponential fitting* de dos y tres pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

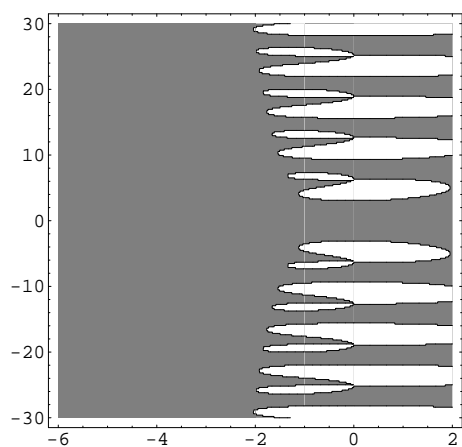
aparente contradicción quedará explicada.

A continuación mostramos las regiones de 0-estabilidad (en gris) de los algoritmos EFM explícitos de dos y tres pasos en la figura A.6.

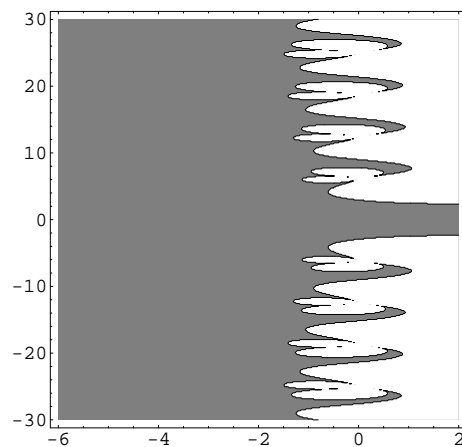
Aunque las diferencias son también notables entre los algoritmos EF-I-k implícitos y los A-I-k implícitos de mismo número de pasos, las diferencias no son tan flagrantes como en el caso del método explícito de dos pasos.

En la figura A.7 mostramos las regiones de 0-estabilidad de los algoritmos de tipo EF-I-k implícitos de dos a siete pasos.

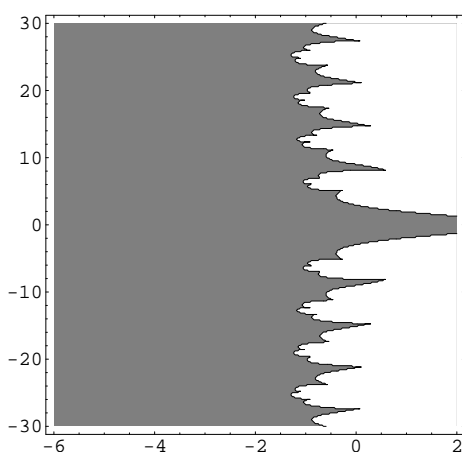
De nuevo, el algoritmo EF-I-k de siete pasos en un entorno del parámetro $\lambda h = 0$ no es 0-estable, aún cuando para gran cantidad de problemas, será posible encontrar valores de h para los cuales los resultados son buenos tal como hemos visto.



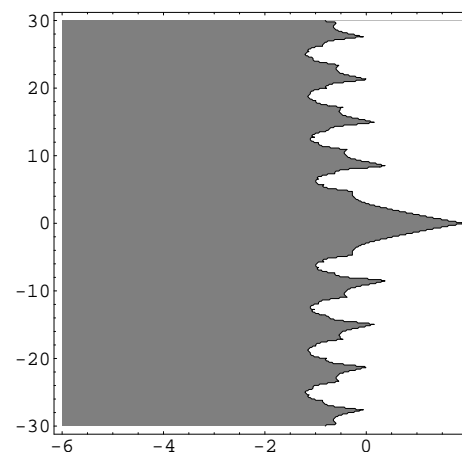
(a) Dos pasos.



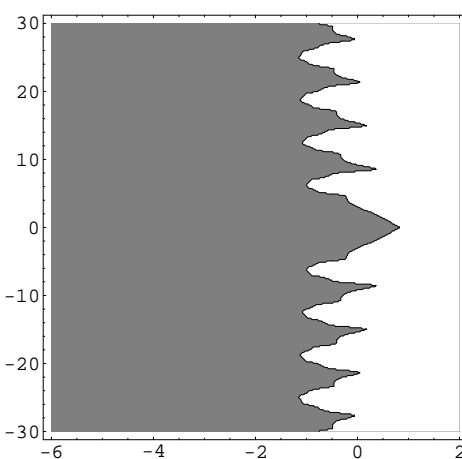
(b) Tres pasos.



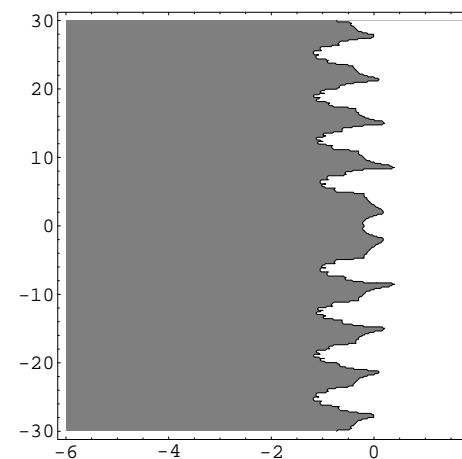
(c) Cuatro pasos.



(d) Cinco pasos.



(e) Seis pasos.



(f) Siete pasos.

Figura A.7: Regiones de 0-estabilidad (en gris) de los métodos EF-I-k implícitos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

En cambio para orden más alto a 7, las regiones de 0-estabilidad de los algoritmos EFM son prácticamente inexistentes.

Es posible deducir ya, cuando los métodos *exponential fitting* o adaptados (que integren de forma exacta los problemas de valor inicial cuando $y(x)$ pertenezca al subespacio $\langle e^{Ax}, 1, x, \dots, x^r \rangle$) tienen buenas regiones de estabilidad.

- i) Siempre que el parámetro introducido en el método sea el “adecuado” (en el capítulo dedicado a calcular el parámetro se explicó qué significaba que un parámetro sea el adecuado a un problema y vimos que no es cuestión sencilla) al problema y el parámetro $u = \lambda h$ pertenezca a las regiones de 0-estabilidad del correspondiente EF-I-k y del A-I-k del mismo número de pasos los resultados serán buenos.
- ii) Si por el contrario u se encuentra a la vez en las regiones de 0-inestabilidad del correspondiente EF-I-k y del A-I-k del mismo número de pasos, queda claro que el método será 0-inestable, como ya se ha comprobado en esta misma sección con un ejemplo.
- iii) Queda por definir el caso en el que u no pertenece a la región de 0-estabilidad del EF-I-k, pero sí a la del A-I-k del mismo número de pasos, o viceversa. Caso que se aclarará en la sección de estabilidad absoluta.

Apéndice B

Estabilidad absoluta de los métodos de orden exponencial

1.

La primera conclusión que se puede obtener es que debemos de ser muy cuidadosos con el valor u_1 que usamos en el método: si éste está cerca de la zona de 0-inestabilidad, entonces la región de inestabilidad puede llegar a ser enorme.

Para mostrar esta afirmación, vamos a poner un ejemplo: consideremos el algoritmo implícito adaptado de dos pasos. El valor $\lambda h = -0,1 + 100i$ está en la región de 0-estabilidad, sin embargo, por ejemplo, $\lambda h = -0,1 + 100,3i$ no está. Vemos en la figura B.1 la región de estabilidad absoluta, para $\lambda h = -0,1 + 100i$.

Observamos que la región de inestabilidad absoluta es muy grande, mucho más que si escogemos un parámetro lejos de la región de 0-inestabilidad, por

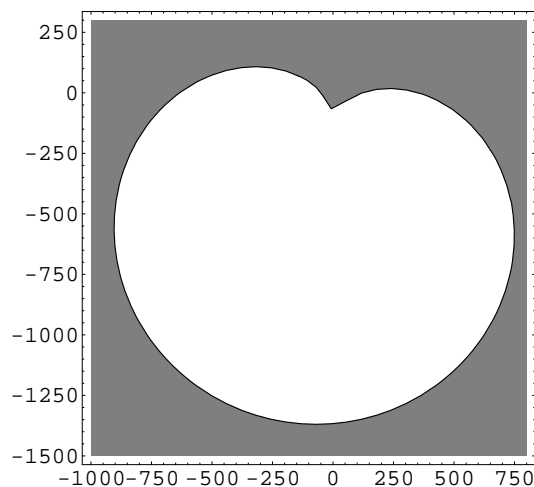


Figura B.1: Región de estabilidad absoluta (en gris) del algoritmo adaptado implícito de dos pasos. El parámetro del método está cerca de la zona de 0-inestabilidad. Los ejes horizontal y vertical representan $Re(\nu h)$ e $Im(\nu h)$ respectivamente.

ejemplo con $\lambda h = -3 + 100i$ (figura B.2).

De todas formas, podemos concluir que si introducimos en el método valores $\lambda h = a + bi$ donde $a < -3$, no tendremos problemas de 0-estabilidad en los métodos de número de pasos bajo (los métodos implícitos de dos, tres, incluso cuatro pasos).

Habíamos observado que el método adaptado explícito de dos pasos no era 0-estable para valores de $\lambda h \in R^-$, sin embargo, el mismo método, pero de tipo EFM sí que era 0-estable con la definición que habíamos dado de 0-estabilidad. En la figura B.3 mostramos unas regiones de estabilidad absoluta de los algoritmos explícitos EF-I-k-2 y A-I-k-2.

Se puede observar que en realidad los dibujos correspondientes al apartado (a) y el correspondiente al apartado (b) son el mismo, también sucede con

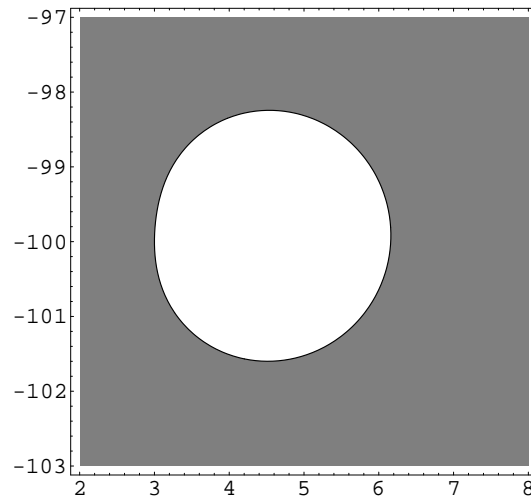


Figura B.2: Región de inestabilidad absoluta (en blanco) del algoritmo adaptado implícito de dos paso. Parámetro del método lejos de la zona de 0-inestabilidad. Los ejes horizontal y vertical representan $Re(\nu h)$ e $Im(\nu h)$ respectivamente.

los apartados (c) y (d) (tan sólo hay una traslación de λh). En los cuatro apartados el valor $(\lambda + \nu)h = 0$ pertenece a la región de estabilidad absoluta, este es un hecho que ya conocíamos a partir de la figura A.6 donde habíamos visto la región de 0-estabilidad del algoritmo EF-I-k-2 explícito, y habíamos visto que los parámetros $\lambda h = -0,01$ y $\lambda h = -100$ pertenecían a dicha región.

Sin embargo, si $\nu = 0$ y $\lambda h = -0,01$ ó $\lambda h = -100$, podemos comprobar que el método no es absolutamente estable (como ya sabíamos del teorema A.0.1).

Esta es la explicación por la que aunque los algoritmos EFM y AM corresponden a los mismos métodos, las regiones de 0-estabilidad de los algoritmos son diferentes.

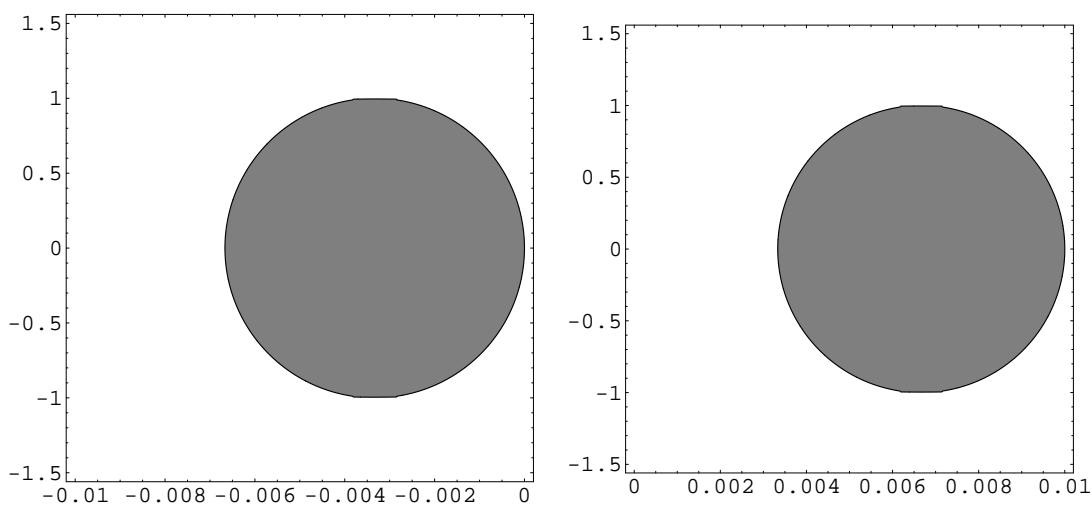
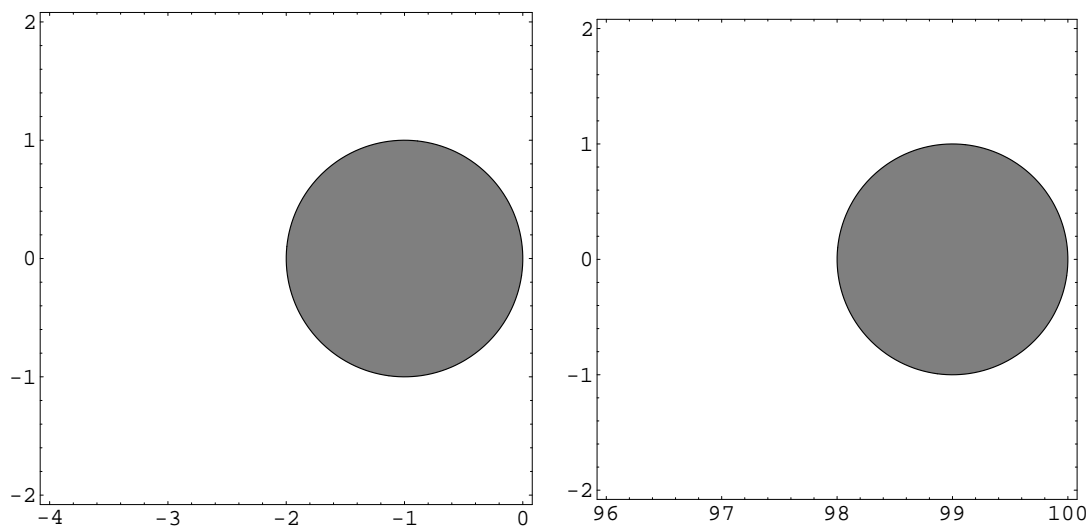
(a) $\lambda h = -0,01$. *EF-I-k-2*.(b) $\lambda h = -0,01$. *A-I-k-2*.(c) $\lambda h = -100$. *EF-I-k-2*.(d) $\lambda h = -100$. *A-I-k-2*.

Figura B.3: Regiones de estabilidad absoluta (en gris) de los métodos explícitos EF-I-k-2 y A-I-k-2. En los dibujos correspondientes al método EF-I-k-2 los ejes horizontal y vertical representan las partes $Re((\lambda + \nu)h)$ e $Im((\lambda + \nu)h)$, mientras que en los dibujos correspondientes al método A-I-k-2 los ejes horizontal y vertical representan las partes $Re(\nu h)$ e $Im(\nu h)$.

Para comprobar la relación entre la estabilidad de los métodos y las regiones de 0-estabilidad veamos cuál es el comportamiento que tiene dicho método con una serie de problemas.

Empecemos con un problema donde la solución $y(x)$ es una exponencial: $y(x) = e^{-x}$.

$$y'(x) + y(x) = 0, \quad y(0) = 1,$$

para resolver dicho problema utilizamos tanto el algoritmo EF-I-k-2 explícito como el A-I-k-2 explícito, aunque ya sabemos que ambos tienen el mismo el error. Utilizamos un paso $h = 0,01$, y tomamos $\lambda = -1$, y realizamos 7000 iteraciones, con un programa escrito en Mathematica.

El método comienza resolviendo de forma exacta el problema, pero debido a errores de truncamiento y a la inestabilidad del método al cabo de las 7000 iteraciones se ha dado un error aproximado de 0,0149565 con ambos algoritmos. El error debido a la inestabilidad sigue creciendo y al cabo de 20000 iteraciones (en el punto $x = 200$) el error es aproximadamente de $9,86834 \times 10^{16}$. Claramente el método es inestable con este problema.

Este método supuestamente integra de forma exacta el problema (2.1.1) cuando $y(x)$ pertenece al espacio generado por $\langle 1, x, e^{\lambda x} \rangle$. Y hemos visto que si el parámetro $\lambda + \nu = 0$, la región de estabilidad absoluta del método parece indicar que es estable. Por tanto vamos a probar con un problema donde la solución $y(x)$ sea un polinomio: $y(x) = 2 - x$.

$$y'(x) = -y(x) - x + 1, \quad y(0) = 2,$$

para resolver dicho problema utilizamos tanto el algoritmo EF-I-k-2 explícito como el A-I-k-2 explícito. Utilizamos un paso $h = 0,01$, tomamos $\lambda = -1$,

y realizamos 1000 iteraciones. El error aproximado es de $-1,4926 \times 10^{-9}$, y después de 20000 iteraciones es ya de $1,9891 \times 10^{17}$.

Aunque el método nuevamente comienza resolviendo de forma exacta el problema, debido a errores de truncamiento y al parámetro introducido en el método se produce la inestabilidad, ya que un error de truncamiento produce que el problema perturbado

$$y'(x) = -y(x) - x + 1, \quad y(0) = 2 + \varepsilon,$$

ya tenga una solución que depende de e^{-x} .

Por tanto el método es inestable si $Re(\lambda) < 0$ ó si $Re(\lambda) > 0$ (donde λ es el parámetro introducido en el método) y $\nu = -\lambda$ ó $\nu = 0$, pero para los problemas perturbados $\nu = -\lambda$.

Veamos que sucede ahora al integrar con el mismo método el problema

$$y'(x) + y(x) = \frac{y(x)}{2}, \quad y(0) = 1,$$

cuya solución es $y(x) = e^{-x/2}$, pero introduciendo el valor $\lambda = -1$, y con paso $h = 0,01$. Si observamos la figura B.3, el método es estable absolutamente con la definición que habíamos dado y efectivamente el método aunque en las primeras iteraciones llega a tener un error aproximado de 0,0049465, sin embargo, tras 1000 iteraciones (en el punto $x = 10$) el error es aproximadamente de $-1,42425 \times 10^{-7}$, y tras 20000 iteraciones es aproximadamente $-3,47244 \times 10^{-23}$.

Aunque el método no era 0-estable, sin embargo, para este problema concreto con estos parámetros concretos, sí que es estable. Por tanto, podemos llegar a la conclusión (daremos algún ejemplo más) de que la clave de la

estabilidad del método la tiene lo que hemos definido como estabilidad absoluta del método aplicada al problema. Es decir, la estabilidad va a depender del parámetro introducido en el método λh y del parámetro que sigan las “soluciones” del problema $((\lambda + \nu)h)$, y la estudiaremos con las regiones que hemos pintado en esta sección. Obsérvese que hemos dicho las soluciones del problema, sin embargo, si está bien formulado la solución del problema de valor inicial (2.1.1) debería ser única. Decimos “soluciones” porque hay que considerar también las soluciones de los problemas perturbados

$$y'(x) = \lambda y + f(x, y(x)), \quad y(x_0) = y_0 + \varepsilon.$$

Obviamente, la 0-estabilidad no es en absoluto despreciable. En primer lugar, ya hemos explicado que en zonas cercanas a donde el método no es 0-estable la región donde el método no es estable absolutamente crece, y por tanto hemos de ser más cuidadosos con el parámetro introducido.

En segundo lugar, si para un parámetro λh el método no es 0-estable para el algoritmo adaptado, ya hemos comprobado que eso se traduce en que no podemos elegir el λh más apropiado a nuestro problema, con lo cual el error puede ser bastante considerable dependiendo del problema que se esté estudiando.

Lo explicaremos con un ejemplo, consideremos el problema

$$y'(x) = -50y(x) + 50 \sin(x) + \cos(x), \quad y(0) = 1, \quad (\text{B.0.1})$$

cuya solución es $y(x) = e^{-50x} + \sin(x)$. Si tomamos un paso de $h = 0,1$, nos damos cuenta que $(\lambda + \nu)h = -5$, es decir, para que el algoritmo sea estable tendríamos que buscar un λ que nos permitiera que para el par $u =$

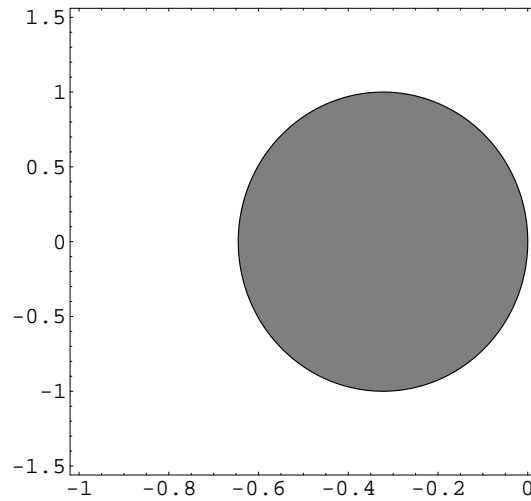


Figura B.4: Región de estabilidad absoluta (en gris) del método explícito de dos pasos. El parámetro del método es $\lambda h = -1$. Los ejes horizontal y vertical representan $Re((\lambda + \nu)h)$ e $Im((\lambda + \nu)h)$ respectivamente.

$(\lambda/10, -5)$ el algoritmo *exponential fitting* fuera absolutamente estable con la definición que hemos dado.

Sin embargo, ya hemos visto a partir de la figura B.3 que ni siquiera, con $\lambda h = -100$, es decir, con $\lambda = -1000$ (muy distante del -50 que es el parámetro que nos interesaría) el método sería estable.

A pesar de todo, sí podemos elegir unos datos con los que podemos integrar el problema (B.0.1) con el algoritmo A-I-k-2 explícito. Basta tomar un paso más pequeño, por ejemplo, $h = 0,01$, ahora con $\lambda = -100$ y $\nu = -50$ el método sí es estable como se puede observar en la figura B.4.

De hecho con esos parámetros el error tras 100 iteraciones es aproximadamente de $2,69702 \times 10^{-5}$, que es similar al obtenido tras 100000 iteraciones: $2,64958 \times 10^{-5}$.

Obviamente, este método no resulta muy operativo si el problema es muy

stiff, ya que necesitaríamos un paso pequeño, para que el método pudiera ser estable y poder utilizar un parámetro λ adecuado. Como ya hemos explicado de este tipo de métodos *exponential fitting* o adaptados sólo el método de un paso que es ya ampliamente conocido tiene buenas propiedades de estabilidad.

Una vez ya ha sido estudiado el comportamiento de los métodos referente a la 0-estabilidad de los mismos tanto en su versión *exponential fitting* como en la versión adaptada, ahora vamos a enunciar una serie de propiedades que observamos en las regiones de estabilidad absoluta de los métodos. Es obvio que, dado el número de métodos de distinto orden ya descritos, y que para cada uno de ellos se podrían tomar infinitos λh (tanto parámetros reales negativos, como valores complejos cuya parte real sea negativa), el número resultante de figuras que podríamos hacer para realizar un estudio detallado pudiera ser tan elevado como quisiéramos. Aquí vamos a procurar mostrar un número comedido de figuras para cada una de las propiedades que enumeremos. Además como ya hemos observado que las regiones de estabilidad absoluta de los algoritmos adaptados y *exponential fitting* son iguales salvo una traslación, de ahora en adelante, nos reduciremos a mostrar las figuras de estabilidad absoluta de los métodos *exponential fitting*.

Empecemos tomando valores $\lambda h \in R^-$, en ese caso observamos que si $\lambda h \rightarrow 0^-$, la región de estabilidad absoluta del método tiende a ser la misma que la del BDF clásico del mismo número de pasos.

Teorema B.0.5. *Si $\lambda h \rightarrow 0^-$, los coeficientes del método nuevo convergen a los coeficientes del BDF tradicional, por ello si $\lambda h \rightarrow 0^-$, los métodos nuevos convergen a los métodos tradicionales.*

Pero si $\lambda h \rightarrow -\infty$, estas regiones de estabilidad crecen. Este hecho se muestra en las figuras B.5, B.6 y B.7, donde hemos dibujado la región de estabilidad absoluta en gris.

Incluso el método de 7 pasos presenta unas regiones parecidas (salvo para parámetros pequeños en valor absoluto, donde alguno de los algoritmos, ó los dos, no es 0-estable), como se puede observar en la figura B.8.

Estas afirmaciones que hemos hecho para parámetros reales negativos se pueden extrapolar en otras direcciones del plano complejo. Si estas direcciones no cortan zonas de 0-inestabilidad, las regiones de estabilidad absoluta crecen a la vez que $\|\lambda h\| \rightarrow \infty$. Este hecho se puede observar en las figuras B.9, B.10 y B.11.

También se puede observar que las figuras de estabilidad absoluta para los parámetros $\lambda h = a + bi$ y $\lambda h = a - bi$ son simétricas entre sí respecto del eje real.

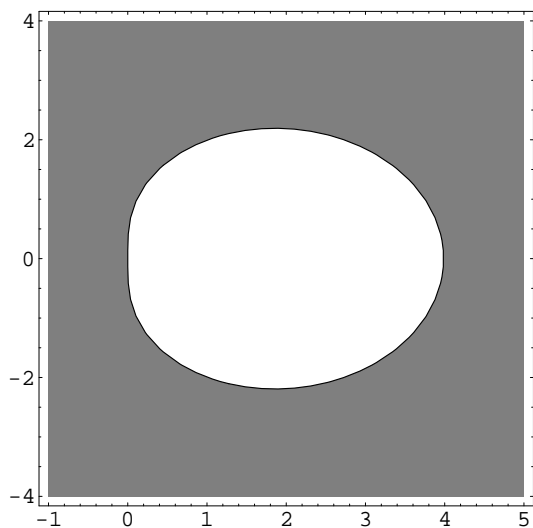
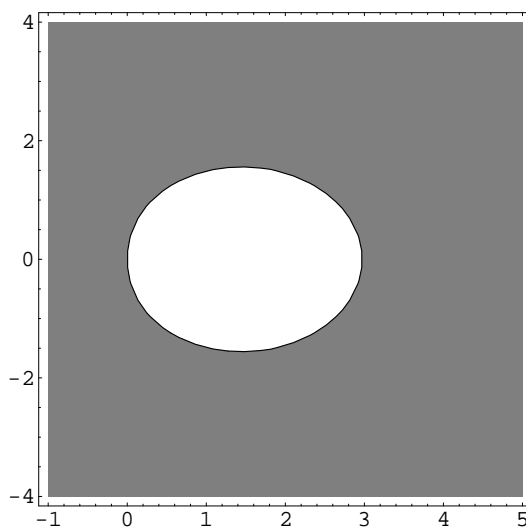
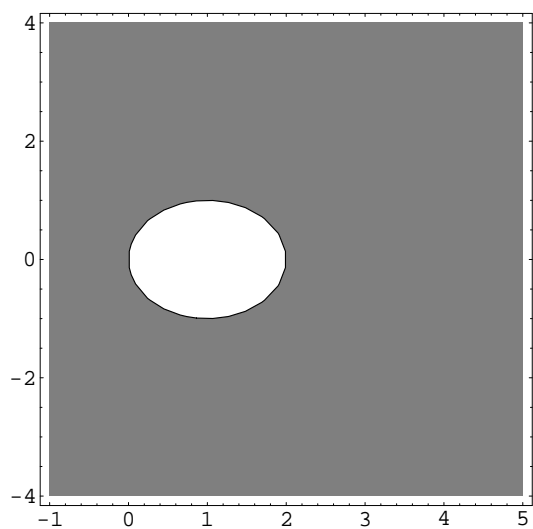
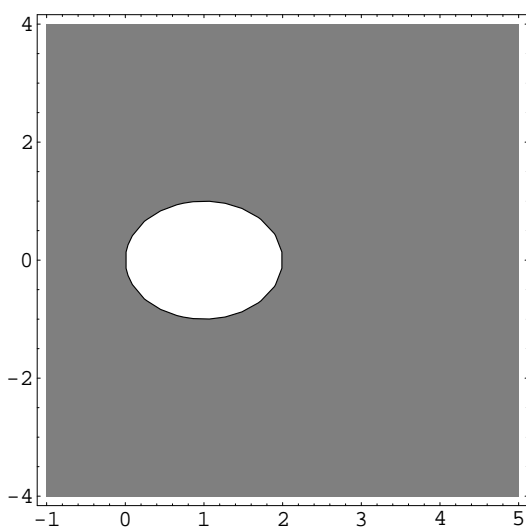
(a) $\lambda h = -0,01$.(b) $\lambda h = -1$.(c) $\lambda h = -100$.(d) $\lambda h = -10000$.

Figura B.5: Regiones de estabilidad absoluta (en gris) del método implícito de dos pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

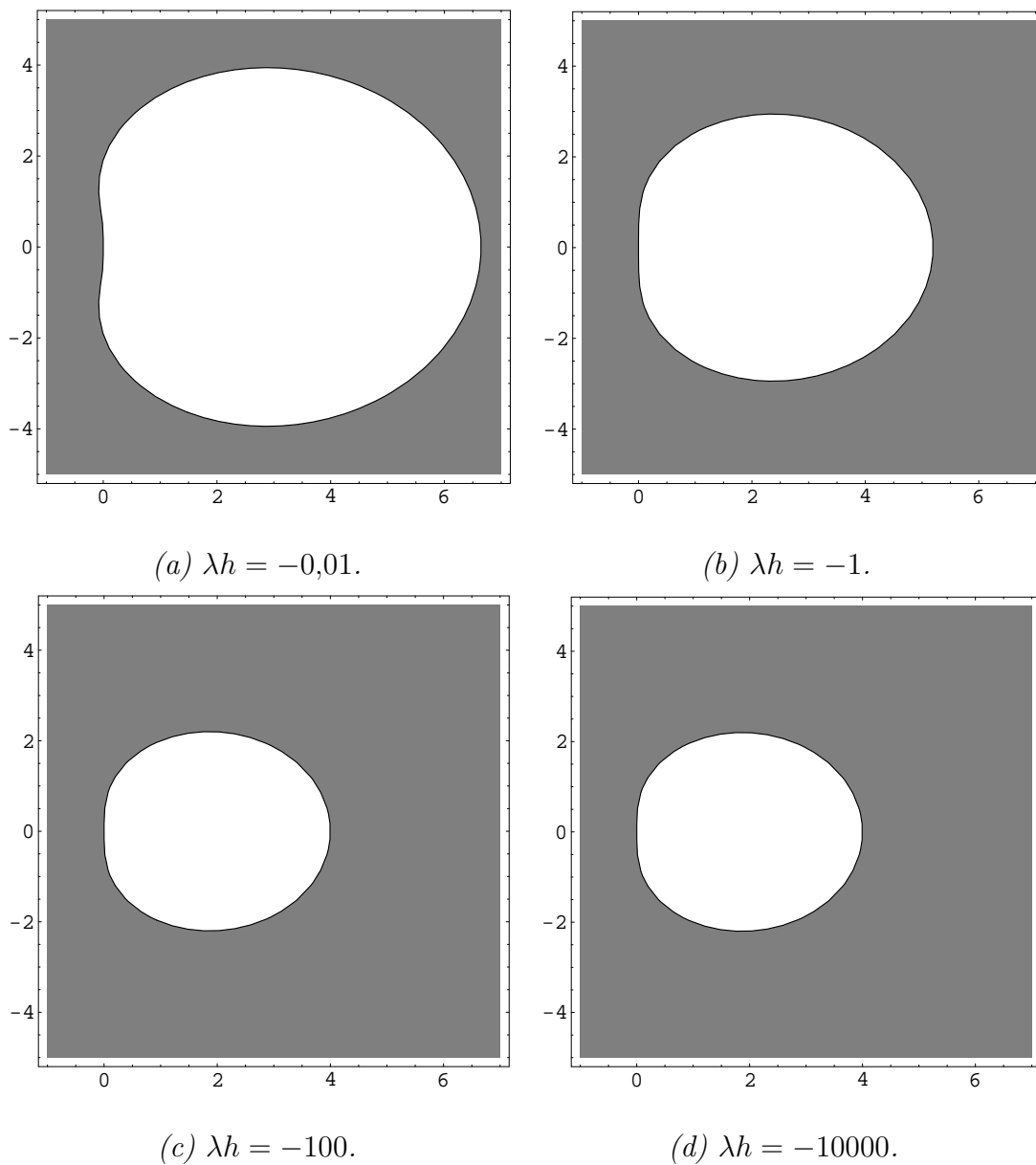


Figura B.6: Regiones de estabilidad absoluta (en gris) del método implícito de tres pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

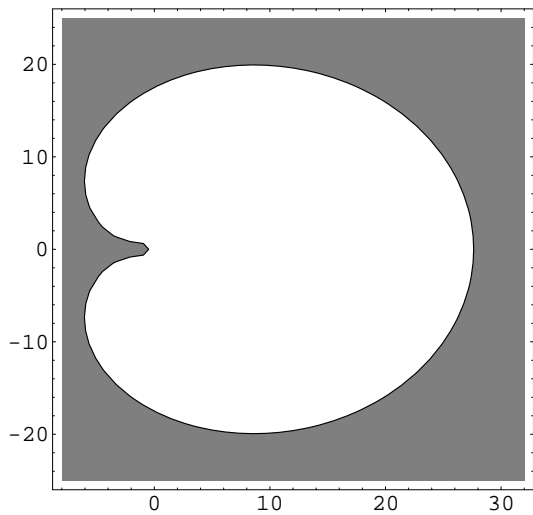
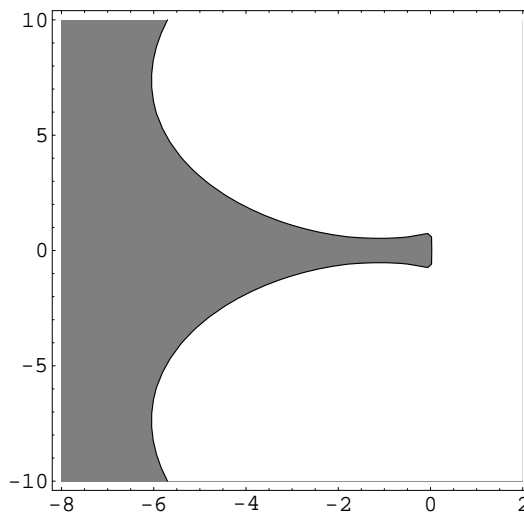
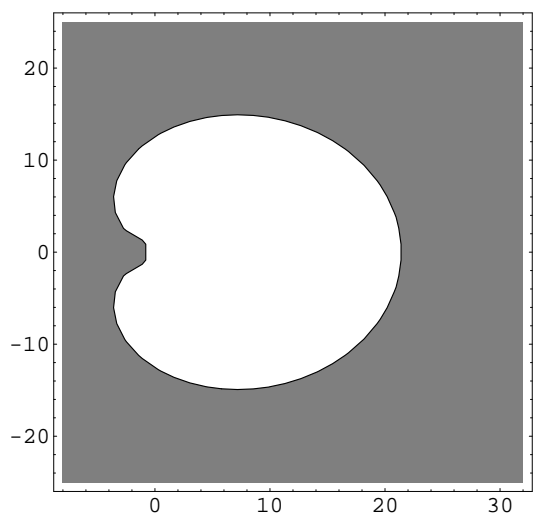
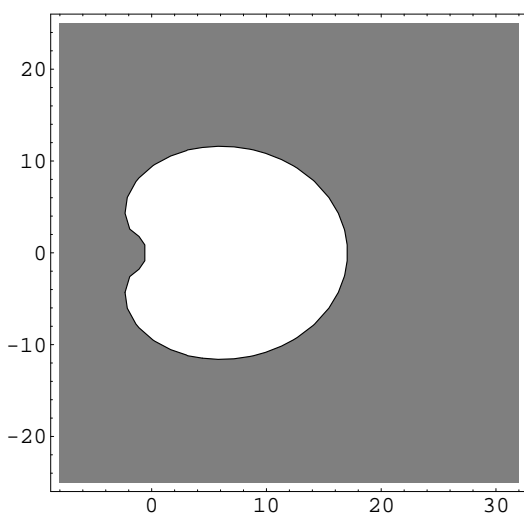
(a) $\lambda h = -0,01$.(b) $\lambda h = -0,01$. *Detalle de la región.*(c) $\lambda h = -1$.(d) $\lambda h = -100$.

Figura B.7: Regiones de estabilidad absoluta (en gris) del método implícito de seis pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

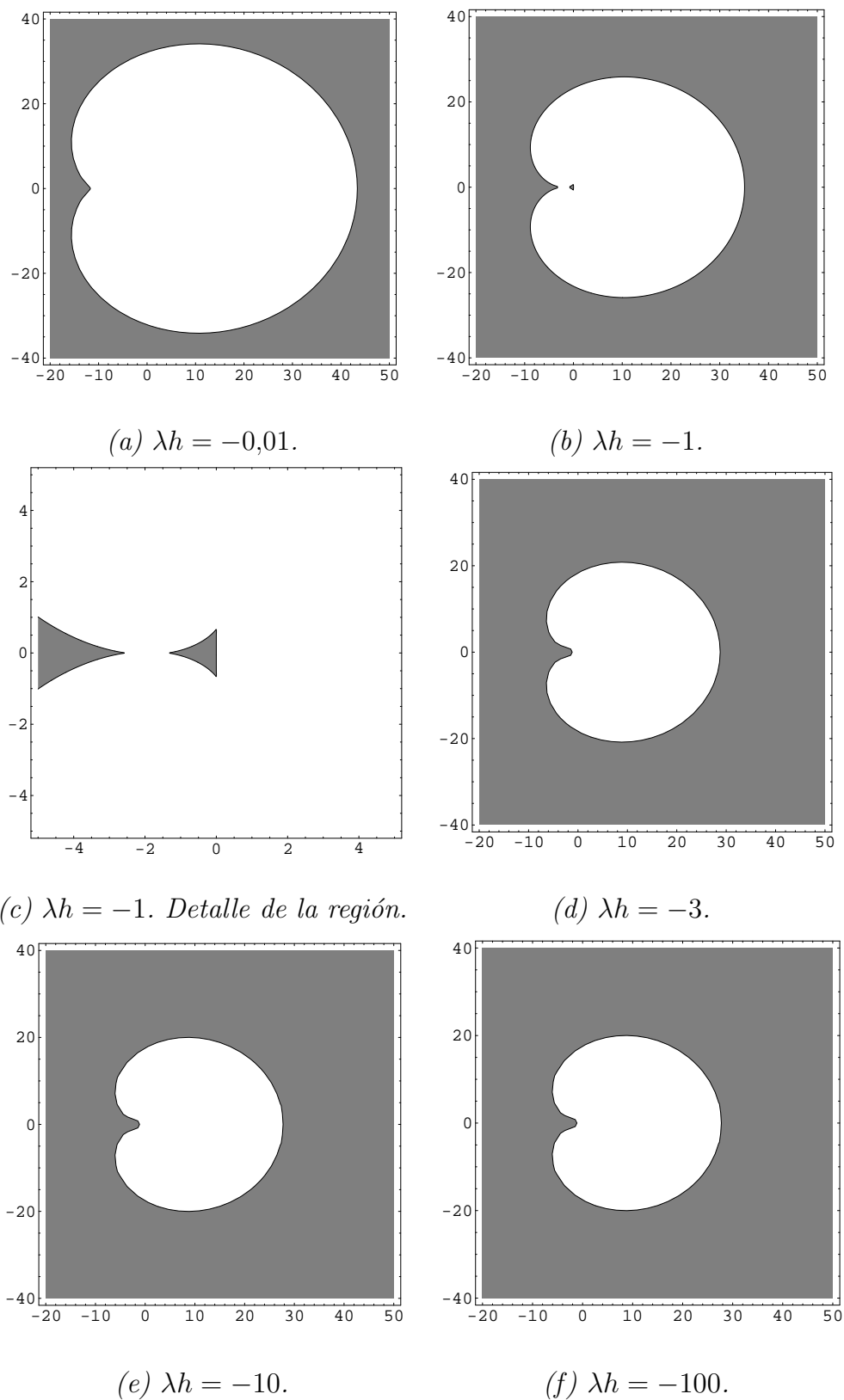


Figura B.8: Regiones de estabilidad absoluta (en gris) del método implícito de siete pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

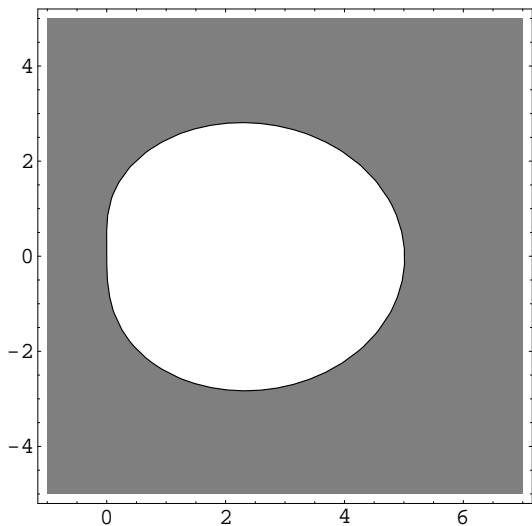
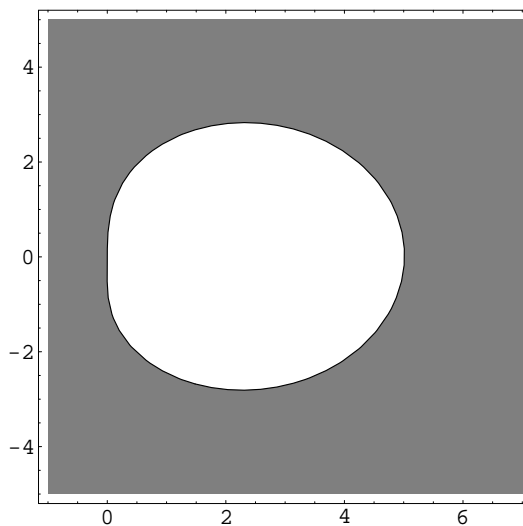
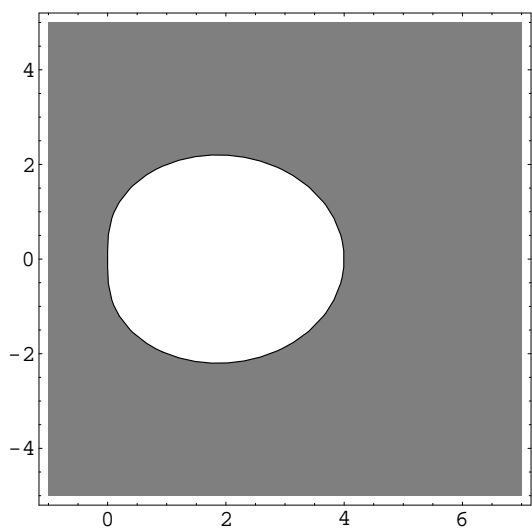
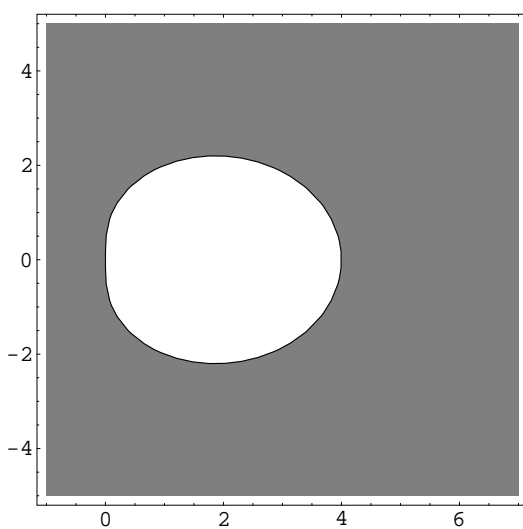
(a) $\lambda h = -1,5 + 5i$.(b) $\lambda h = -1,5 - 5i$.(c) $\lambda h = -30 + 100i$.(d) $\lambda h = -30 - 100i$.

Figura B.9: Regiones de estabilidad absoluta (en gris) del método implícito de tres pasos. Parámetro imaginario del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

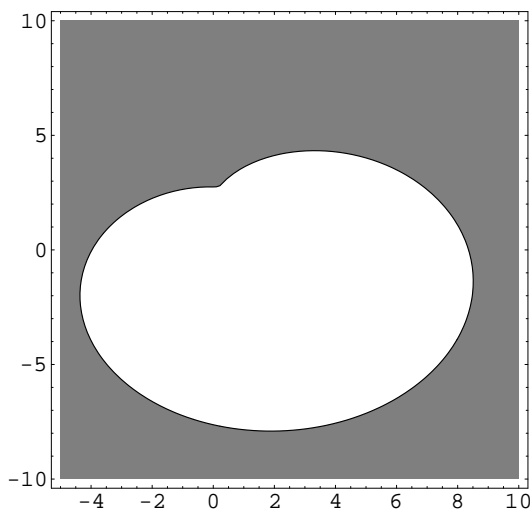
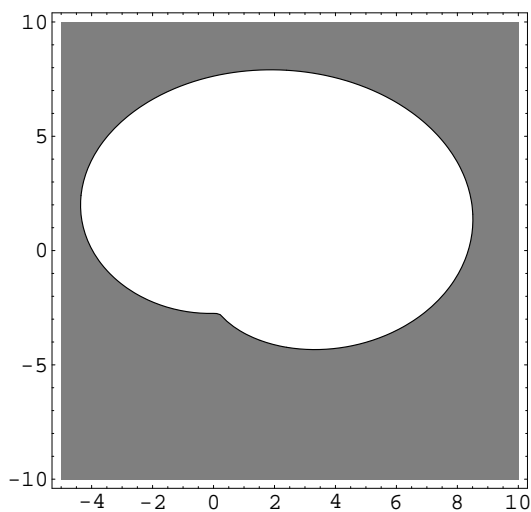
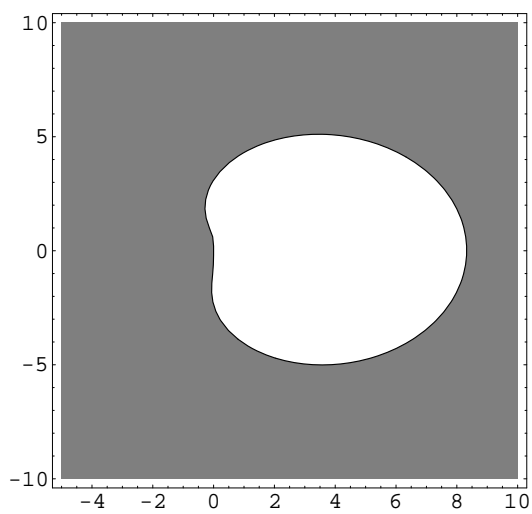
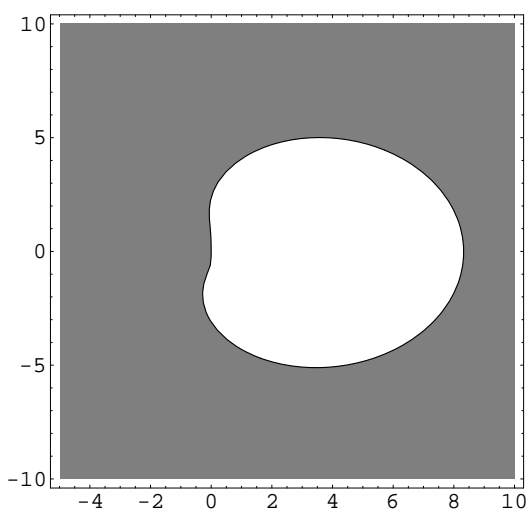
(a) $\lambda h = -1,5 + 5i$.(b) $\lambda h = -1,5 - 5i$.(c) $\lambda h = -30 + 100i$.(d) $\lambda h = -30 - 100i$.

Figura B.10: Regiones de estabilidad absoluta (en gris) del método implícito de cuatro pasos. Parámetro imaginario del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

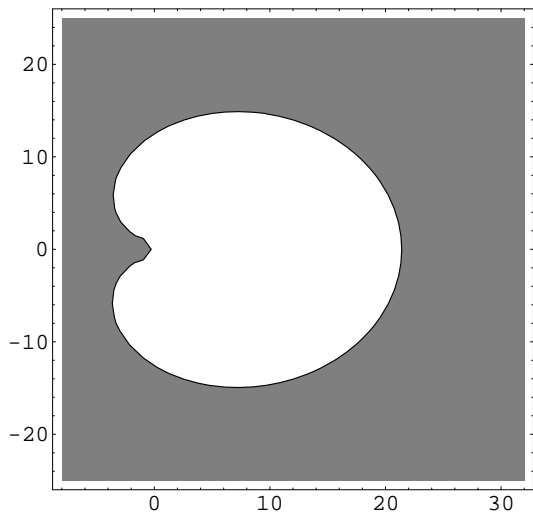
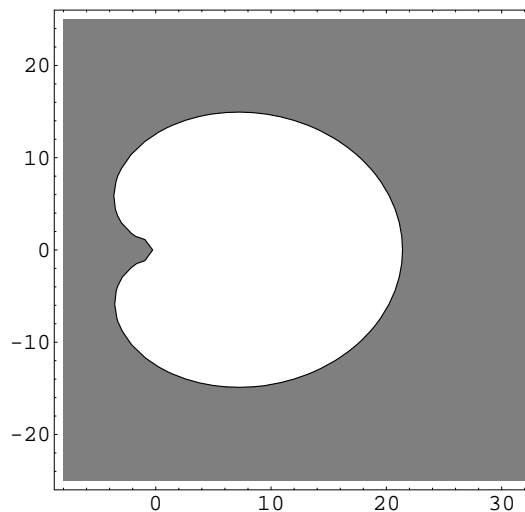
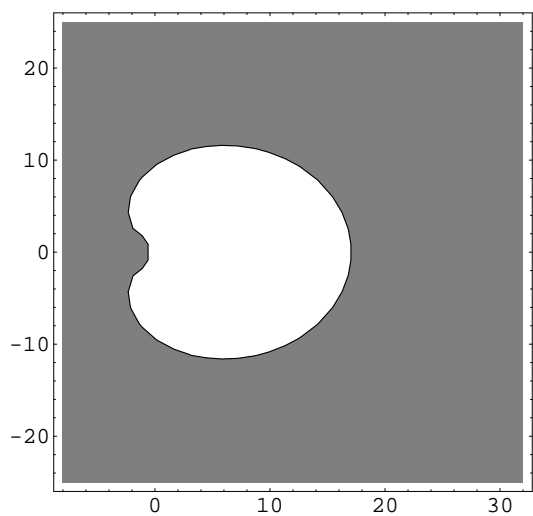
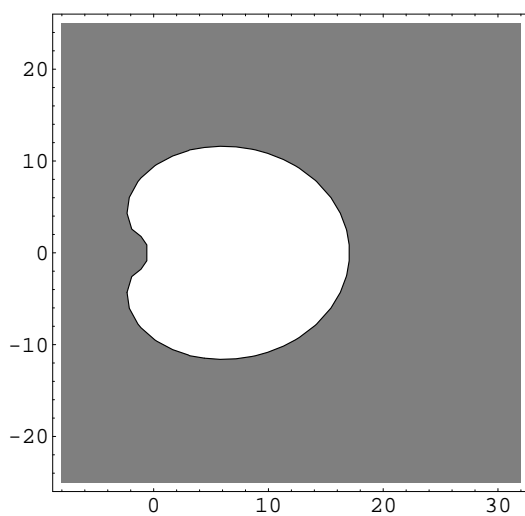
(a) $\lambda h = -1 + 0,1i$.(b) $\lambda h = -1 - 0,1i$.(c) $\lambda h = -10 + 1i$.(d) $\lambda h = -10 - 1i$.

Figura B.11: Regiones de estabilidad absoluta (en gris) del método implícito de seis pasos. El parámetro λh del método es imaginario. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

Apéndice C

0-estabilidad de los métodos de orden algebraico 1.

El mismo razonamiento que hicimos en el capítulo dedicado a los métodos de orden exponencial 1 es perfectamente válido en este, por tanto, comprobaremos si para cada valor propio λ todas las raíces r_k del polinomio característico

$$p(r) = \sum_{i=0}^m \alpha_i r^{m-i}, \quad (\text{C.0.1})$$

verifican que $|r_k| < 1$, y en aquellas raíces que tengan módulo 1 ver cuándo son simples.

A continuación mostramos las figuras de 0-estabilidad de los algoritmos adaptados y *exponential fitting* de los métodos explícitos de tipo BDF de orden algebraico 1 y bajo número de pasos en la figura C.1.

A partir de los apartados (a) y (b) de la figura C.1 podemos sacar la siguiente conclusión:

Teorema C.0.6. *El método exponential fitting o adaptado explícito de dos pasos con parámetro $u = \lambda h$ de forma que $\text{Re}(u) \leq 0$ es 0-estable.*

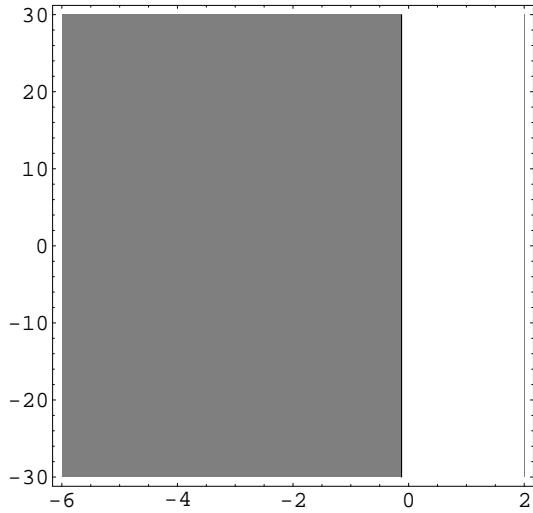
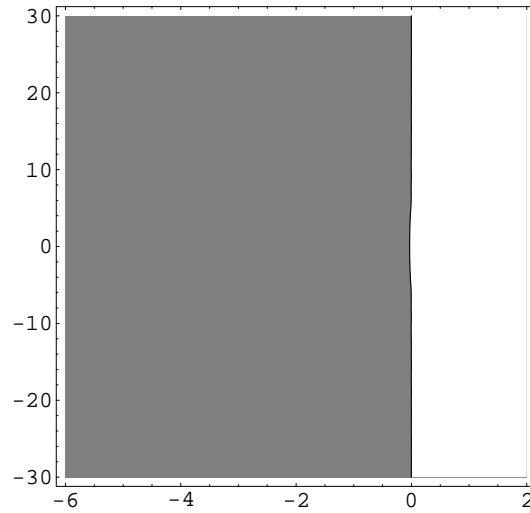
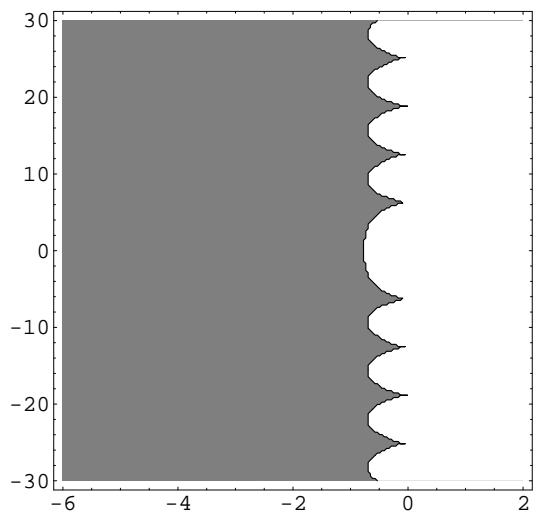
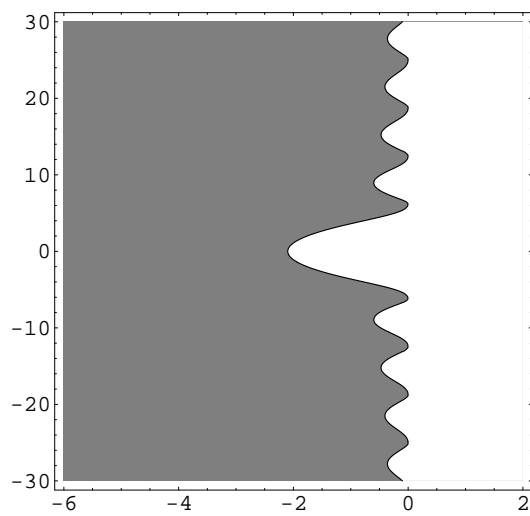
(a) Método *EF-I-r-2*.(b) Método *A-I-r-2*.(c) Método *EF-I-r-3*.(d) Método *A-I-r-3*.

Figura C.1: Regiones de 0-estabilidad (en gris) de los métodos EF-I-r y A-I-r explícitos de dos y tres pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

De hecho en [95] y en el capítulo referente a los ejemplos numéricos se puede ver sobre diferentes problemas, que dicho método es capaz de lograr buenos resultados frente a métodos implícitos de paso variable y orden mayor.

Los métodos explícitos de tres, cuatro (y en adelante) pasos no verifican la misma propiedad que el homólogo de dos pasos, sin embargo, para un buen número de problemas *stiff* o de problemas del tipo Prothero-Robinson, ..., es posible encontrar longitudes de paso h para las cuales dichos métodos dan buenos resultados. En este aspecto su comportamiento es similar al del algoritmo A-I-k-7 que se vió en el capítulo anterior. En la figura C.2 mostramos las regiones de 0-estabilidad de los algoritmos explícitos de 4 y 5 pasos.

Incluso estas propiedades se mantienen con métodos de un número muy elevado de pasos, como ejemplo en la figura C.3 mostramos las regiones de 0-estabilidad de los algoritmos explícitos de 10 y 15 pasos.

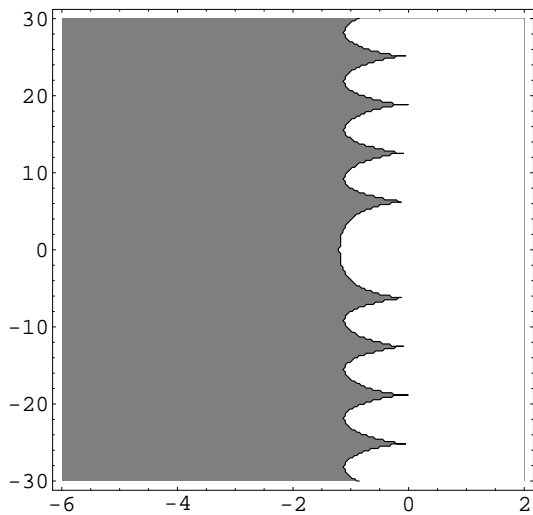
Pasemos ahora a estudiar la 0-estabilidad de los métodos implícitos de este tipo.

A continuación, en las figuras C.4, C.5, C.6 y C.7 presentamos las regiones donde los AM implícitos de 2 a 9 pasos (los denotaremos A-I-r-2, el método de 2 pasos, A-I-r-3 el de 3 pasos, ...) son 0-estables, la 0-estabilidad corresponde a la parte en gris.

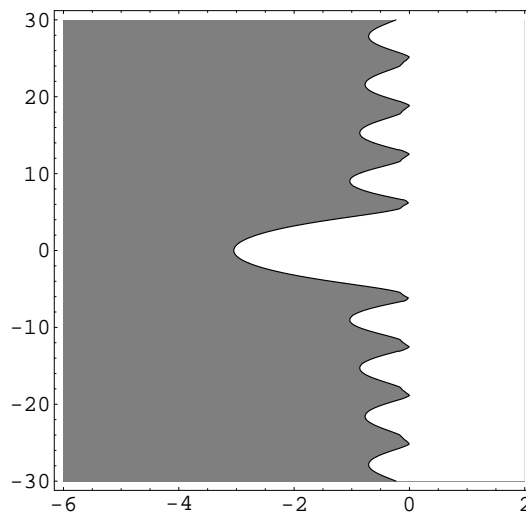
Como sucede con los BDF clásicos o con los métodos de orden exponencial 1, el método implícito de 7 pasos presenta ciertas dificultades. De hecho como se observa en la figura C.8:

Teorema C.0.7. *Los algoritmos implícitos adaptados y exponential fitting de siete pasos (A-I-r-7 y EF-I-r-7) no son 0-estables en un entorno del origen.*

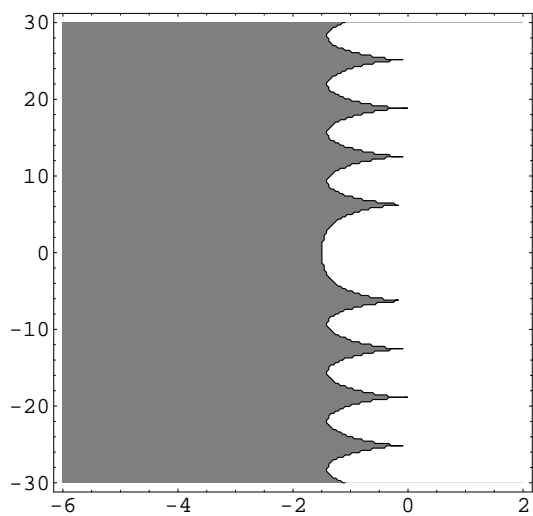
Lo mismo sucede con los algoritmos de ocho pasos (ver figura C.9).



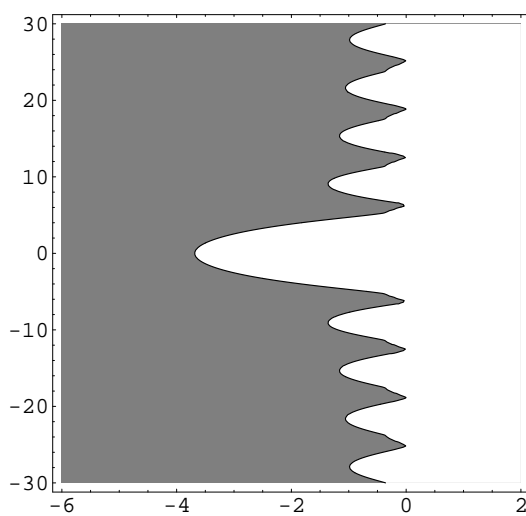
(e) Método EF-I-r-4.



(f) Método A-I-r-4.

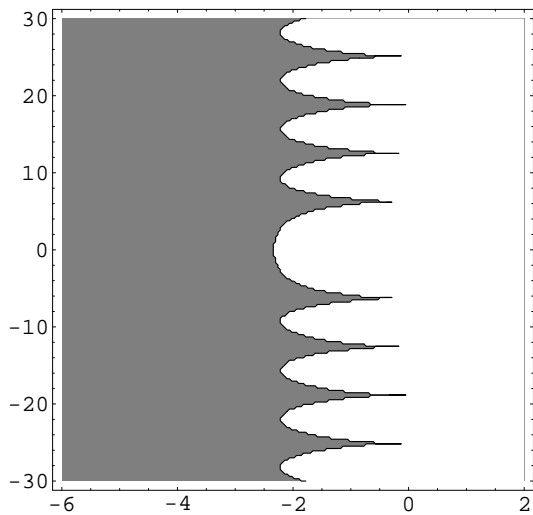


(g) Método EF-I-r-5.

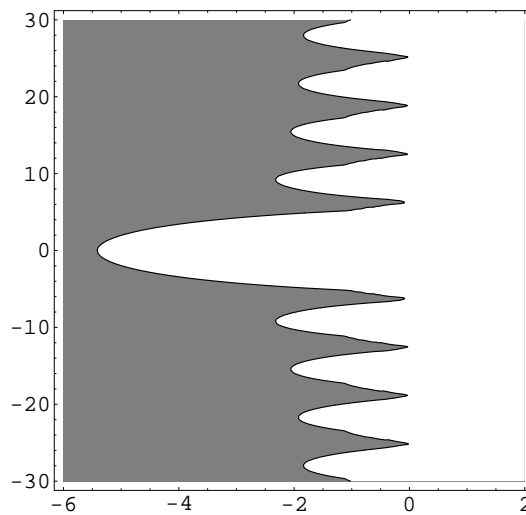


(h) Método A-I-r-5.

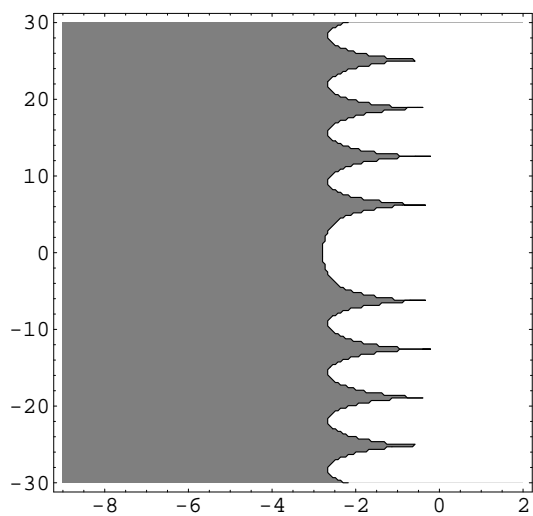
Figura C.2: Regiones de 0-estabilidad (en gris) de los métodos EF-I-r y A-I-r explícitos de cuatro y cinco pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.



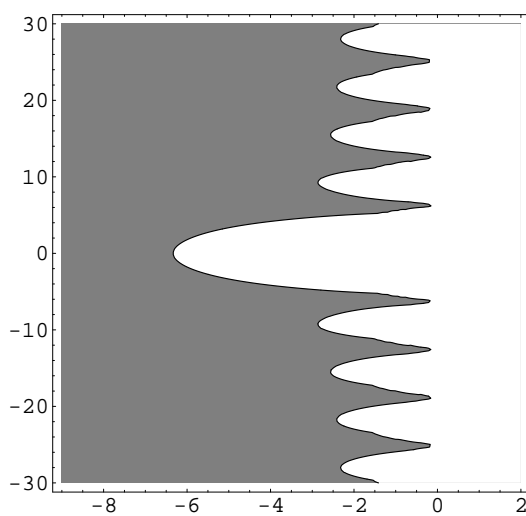
(i) Método EF-I-r-10.



(j) Método A-I-r-10.



(k) Método EF-I-r-15.



(l) Método A-I-r-15.

Figura C.3: Regiones de 0-estabilidad (en gris) de los métodos EF-I-r y A-I-r explícitos de número elevado de pasos (diez y quince pasos). Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

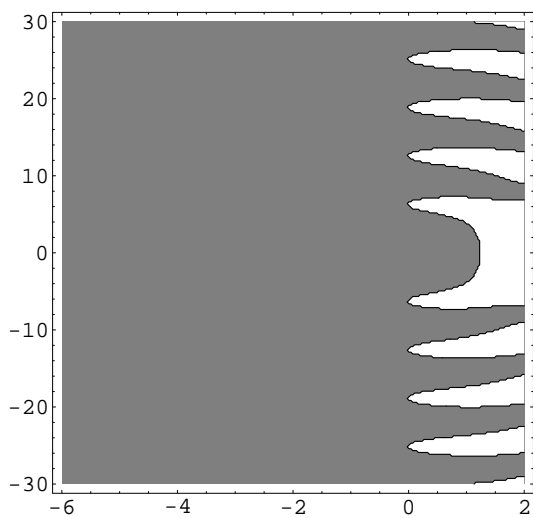
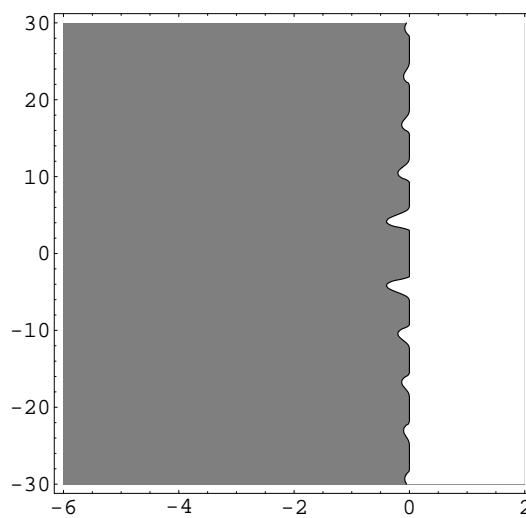
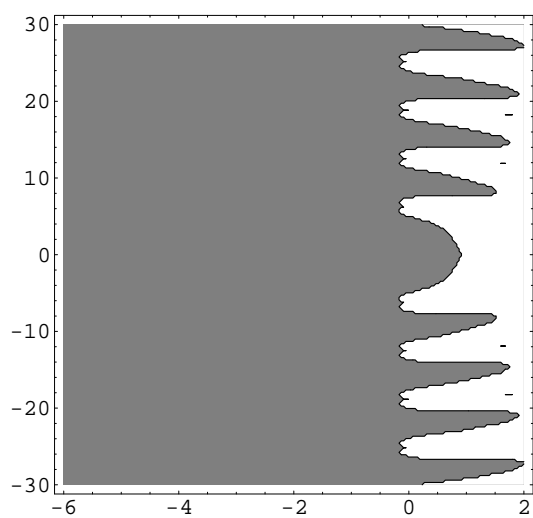
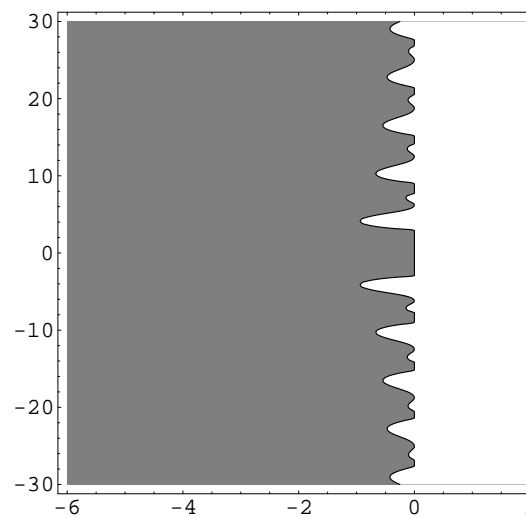
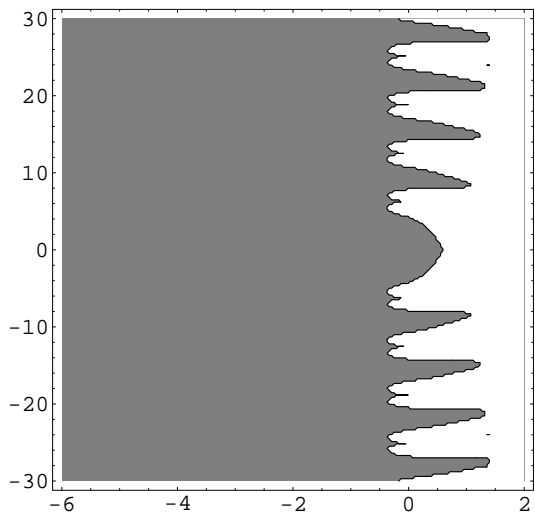
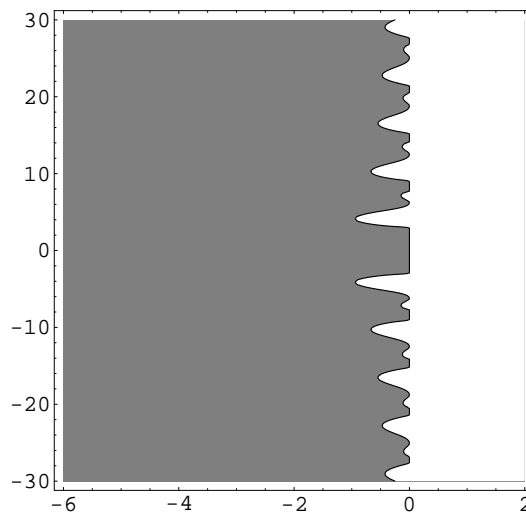
(a) Método *EF-I-r-2*.(b) Método *A-I-r-2*.(c) Método *EF-I-r-3*.(d) Método *A-I-r-3*.

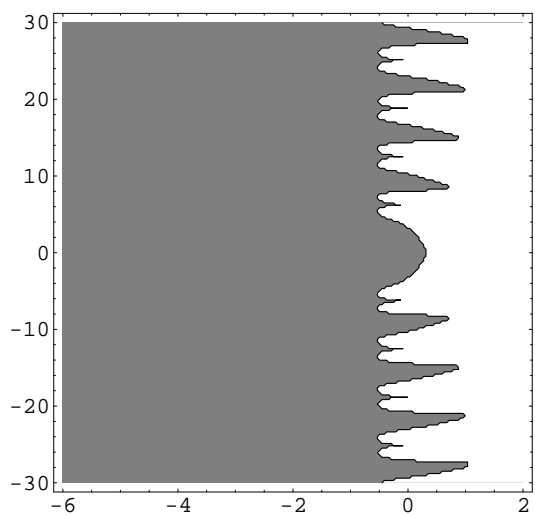
Figura C.4: Regiones de 0-estabilidad (en gris) de los métodos EF-I-r y A-I-r implícitos de orden bajo (de dos y tres pasos). Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.



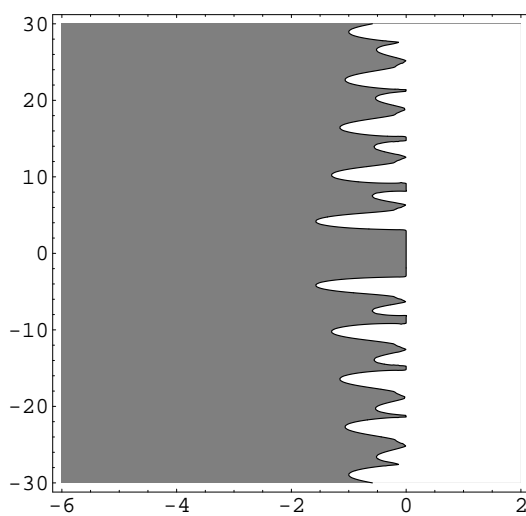
(e) Método *EF-I-r-4*.



(f) Método *A-I-r-4*.

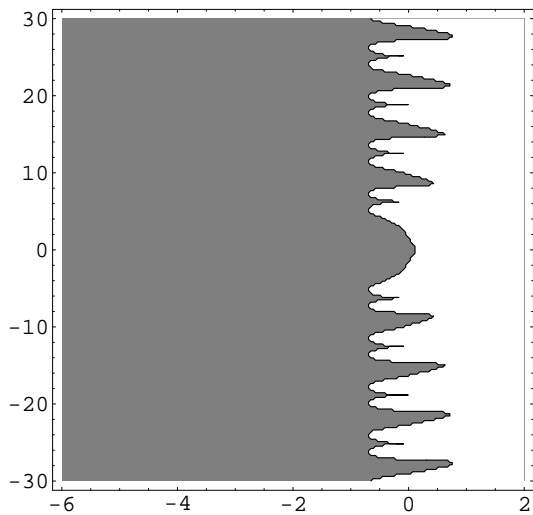


(g) Método *EF-I-r-5*.

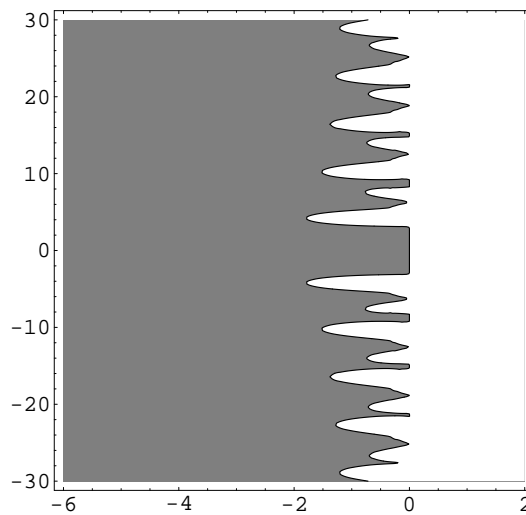


(h) Método *A-I-r-5*.

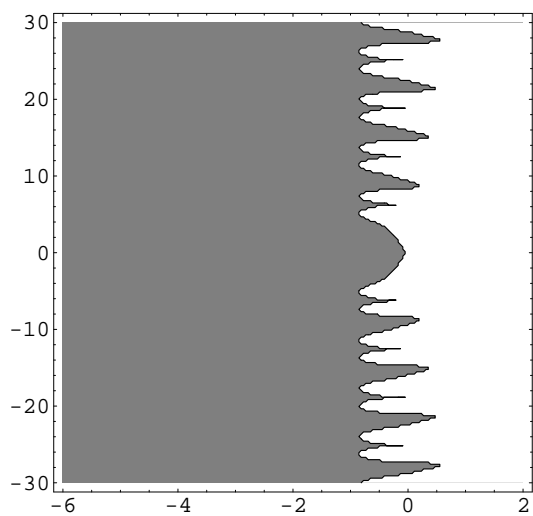
Figura C.5: Regiones de 0-estabilidad (en gris) de los métodos EF-I-r y A-I-r implícitos de orden medio (de cuatro y cinco pasos). Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.



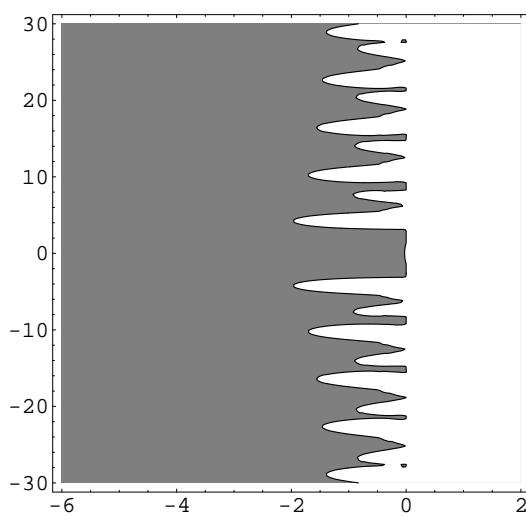
(i) Método EF-I-r-6.



(j) Método A-I-r-6.

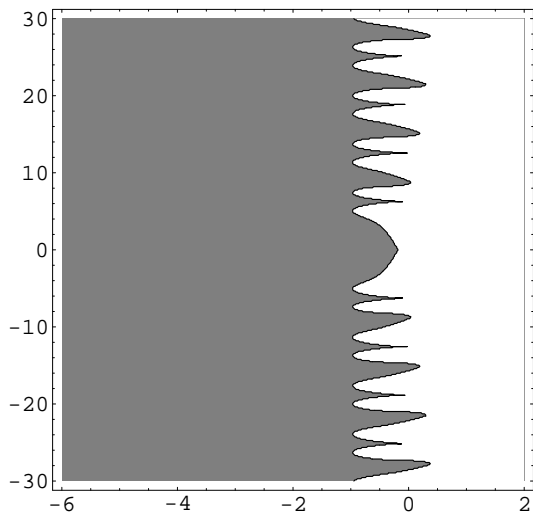


(k) Método EF-I-r-7.

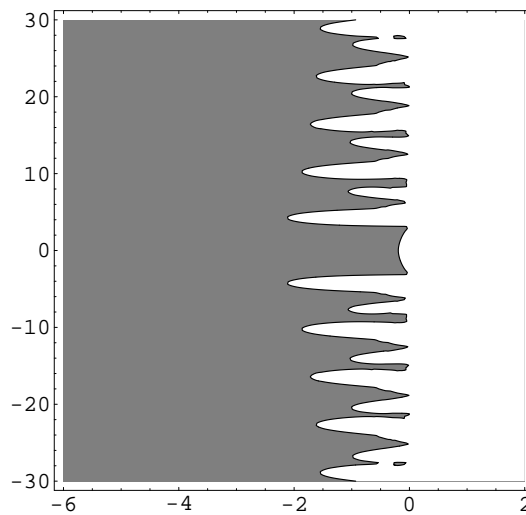


(l) Método A-I-r-7.

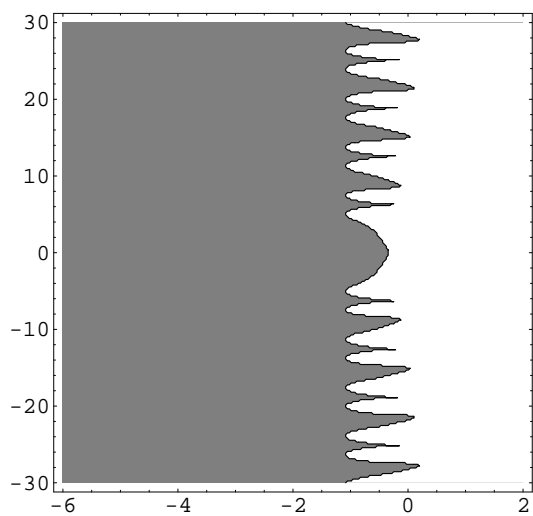
Figura C.6: Regiones de 0-estabilidad (en gris) de los métodos EF-I-r y A-I-r implícitos de orden alto (de seis y siete pasos). Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.



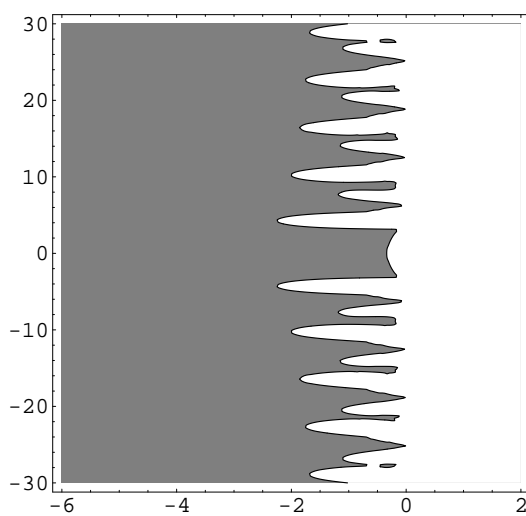
(m) Método EF-I-r-8.



(n) Método A-I-r-8.



(p) Método EF-I-r-9.



(q) Método A-I-r-9.

Figura C.7: Regiones de 0-estabilidad (en gris) de los métodos EF-I-r y A-I-r implícitos de orden muy alto (de ocho y nueve pasos). Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

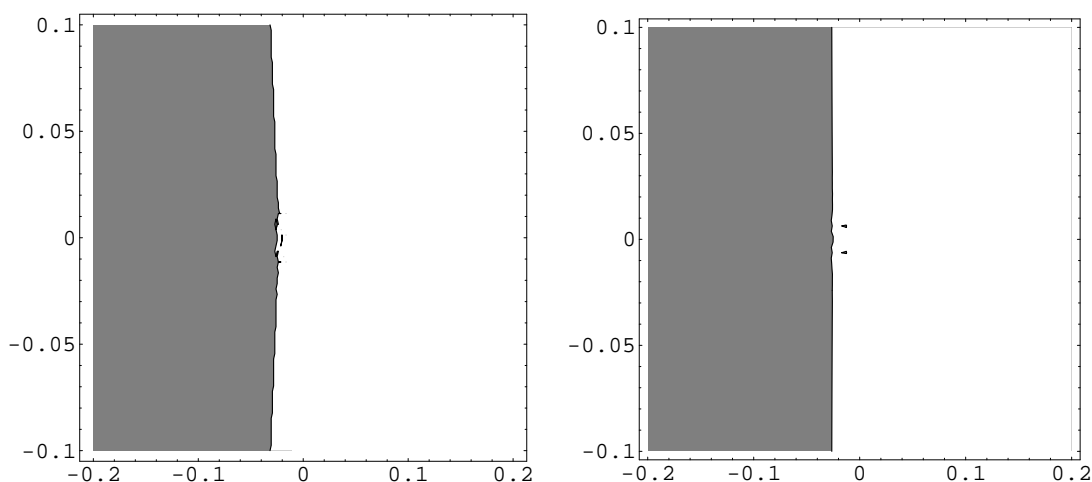
(a) *EF-I-r-7*.(b) *A-I-r-7*.

Figura C.8: Detalle de la 0-estabilidad (en gris) del método implícito de siete pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

Teorema C.0.8. *Los algoritmos implícitos adaptados y exponential fitting de ocho pasos (*A-I-r-8* y *EF-I-r-8*) no son 0-estables en un entorno del origen.*

De nuevo, es posible encontrar problemas para los cuales para determinada longitud de paso h los métodos obtienen buenos resultados, sin embargo, al decrecer el paso los métodos se vuelven inestables.

Es posible alcanzar similares resultados para algoritmos implícitos de un número más alto de pasos, por ejemplo el caso de los algoritmos de 9 pasos se puede observar en la figura C.7, o en la figura C.10 la de los algoritmos de 10 y 15 pasos.

De las figuras de 0-estabilidad que se han mostrado en este apéndice y en el apéndice A es fácil llegar a la conclusión de que los métodos de orden algebraico 1 van a tener propiedades de estabilidad mejores que los métodos

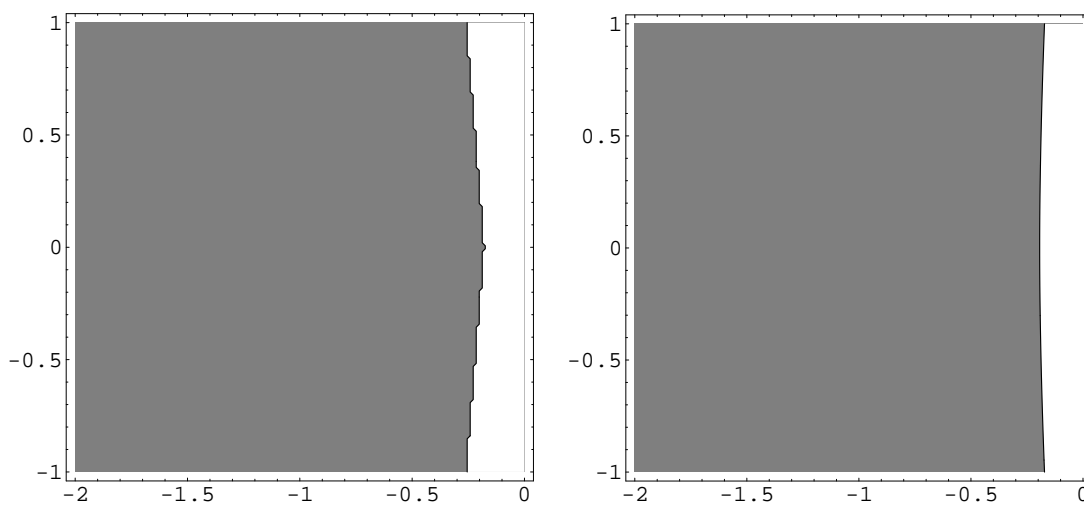
(a) *EF-I-r-8*.(b) *A-I-r-8*.

Figura C.9: Detalle de las regiones de 0-estabilidad (en gris) de los algoritmos implícitos de ocho pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

de orden exponencial 1. Esta conclusión se verá más detenidamente en el apéndice D.

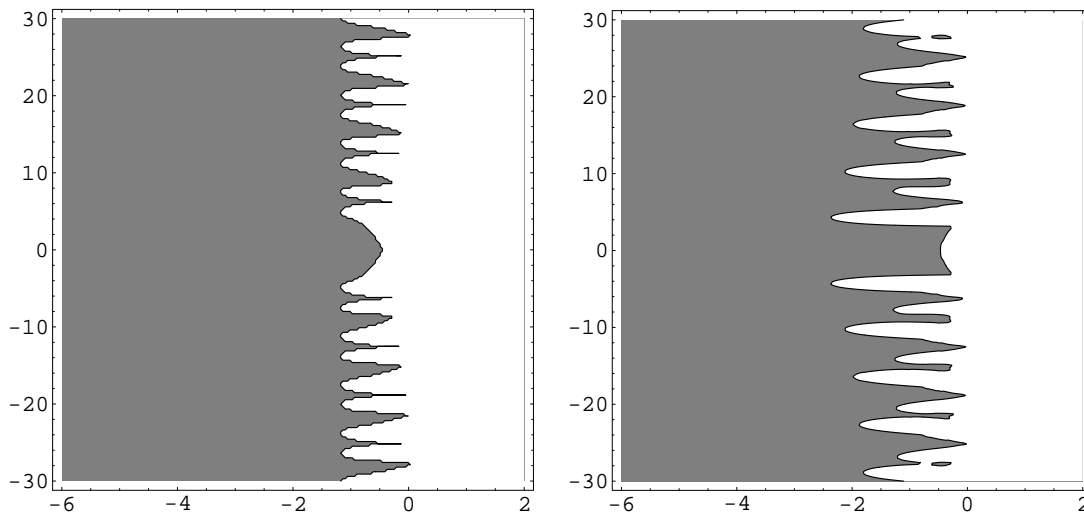
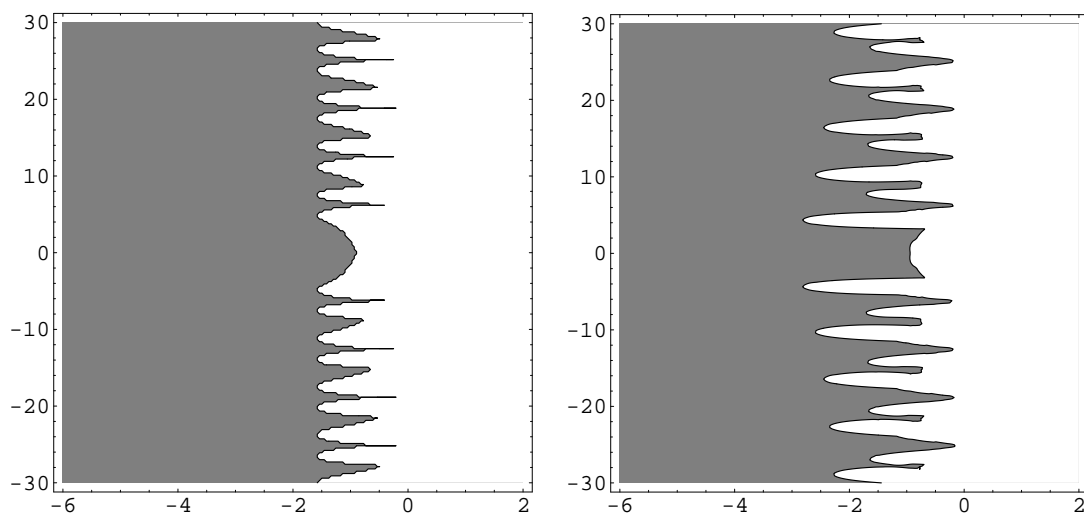
*(r) EF-I-r-10.**(s) A-I-r-10.**(t) EF-I-r-15.**(u) A-I-r-15.*

Figura C.10: Detalle de las regiones de 0-estabilidad (en gris) de los algoritmos implícitos de diez y quince pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

Apéndice D

Estabilidad absoluta de los métodos de orden algebraico 1.

Las propiedades que se observan en los métodos de orden algebraico 1, no distan demasiado (en general) de las de los métodos de orden exponencial 1.

La primera característica que se repite es que debemos ser muy cuidadosos con el valor u_1 que usamos en el método: si éste está cerca de la zona de 0-inestabilidad, entonces la región donde el método no es estable puede llegar a ser muy grande.

Para mostrar esta afirmación, vamos a poner un ejemplo: consideremos el método implícito de tres pasos. El valor $\lambda h = -1 + 5i$ está en la región de 0-estabilidad tanto del algoritmo adaptado como del *exponential fitting*, sin embargo, como se puede observar en la figura D.1, hay parámetros cerca para los cuales los algoritmos no son 0-estables.

Eso se traduce en que la región donde el método es inestable absolutamente es más grande que si escogemos un parámetro más alejado de las regiones

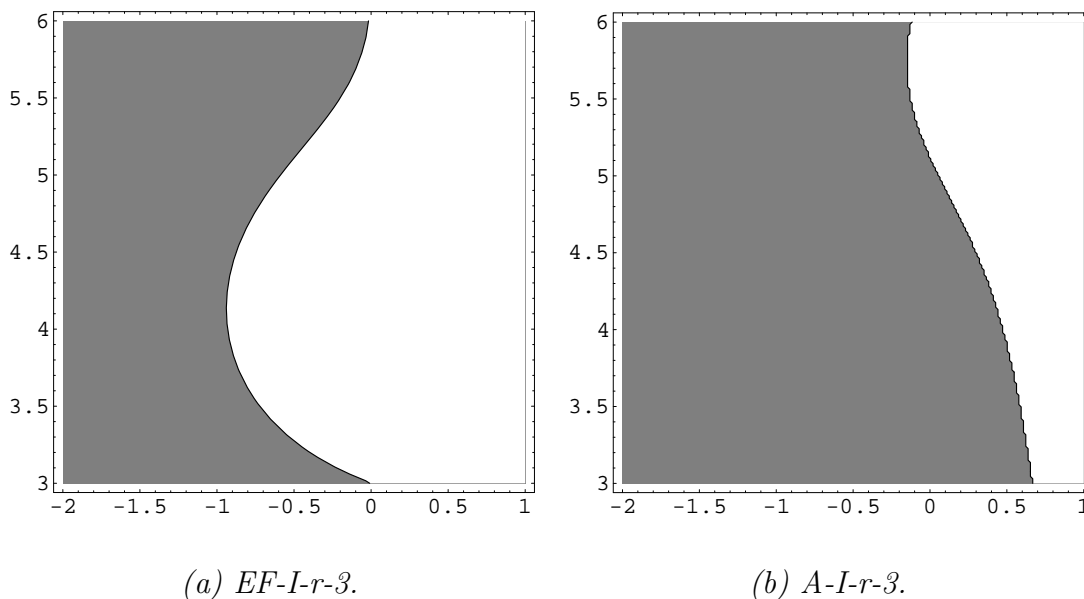


Figura D.1: Detalle de las regiones de 0-estabilidad de los algoritmos EF-I-r-3 y A-I-r-3. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

de 0-estabilidad, como es $\lambda h = -3 + 5i$. En la figura D.2 se muestran las regiones de estabilidad absoluta para ambos parámetros.

Sin embargo, podemos concluir que si introducimos en el método valores $\lambda h = a + bi$ donde $a < -3$, no tendremos problemas de 0-estabilidad en los métodos de número de 2 a 6 pasos.

También podemos concluir que al igual que en los métodos del anterior capítulo, en los nuevos también hay una relación entre las regiones de estabilidad absoluta de los algoritmos adaptados y las de los algoritmos *exponential fitting*. Es obvio que las regiones de estabilidad en sí van a ser iguales, la única diferencia es que hay una traslación (de λh). Esto es debido a que:

Teorema D.0.9. *Dados los algoritmos EF-I-r y A-I-r con parámetro λh , los polinomios característicos de estabilidad absoluta para los pares $u = (u_1 =$*

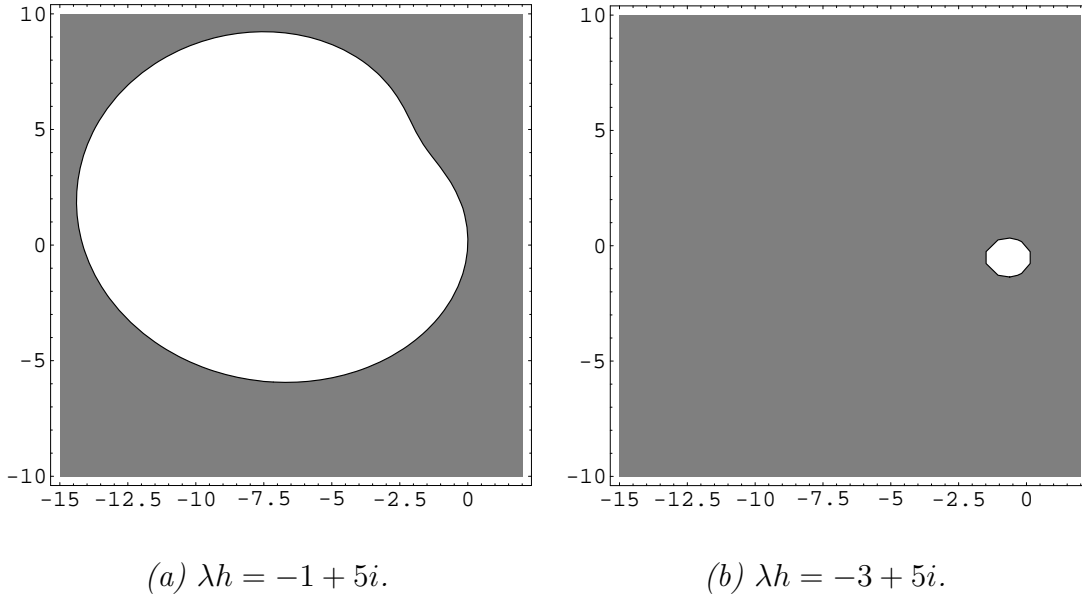


Figura D.2: Evolución de la estabilidad absoluta en función de la cercanía del parámetro λh a las regiones de 0-inestabilidad.

$\lambda h, u_2 = (\lambda + \nu)h$ y $\bar{u} = (\bar{u}_1 = \lambda h, \bar{u}_2 = \nu h)$ (respectivamente), son el mismo. Por tanto tienen las mismas raíces y regiones de estabilidad absoluta, salvo una traslación del parámetro λh introducido en el método.

Por eso, de ahora en adelante nos vamos a limitar a mostrar las regiones de estabilidad absoluta de los algoritmos *exponential fitting* (que nos indicarán en cuanto nos podemos equivocar al elegir el parámetro del problema).

Habíamos observado que los algoritmos explícitos de dos pasos, tanto el *exponential fitting* como el adaptado esta vez sí eran 0-estable para valores de $\lambda h \in R^-$. En las figuras D.3 y D.4 mostramos las regiones de estabilidad absoluta del algoritmo explícito EF-I-r-2.

Como podemos observar el método explícito de orden 2 presenta buenas propiedades de estabilidad, de hecho en [95] ó en el capítulo de ejemplos numéricos se vió que este método es capaz de lograr buenos resultados en

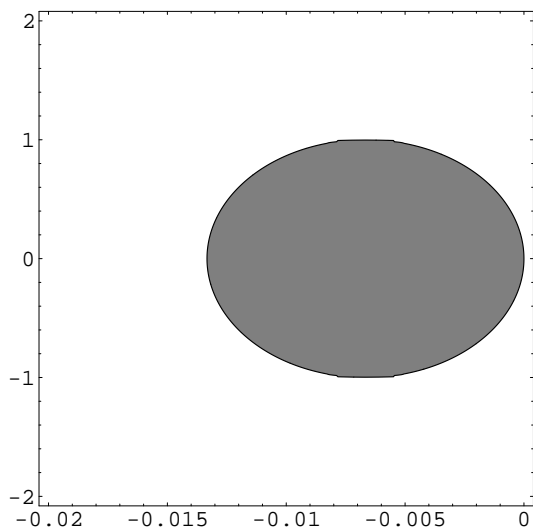
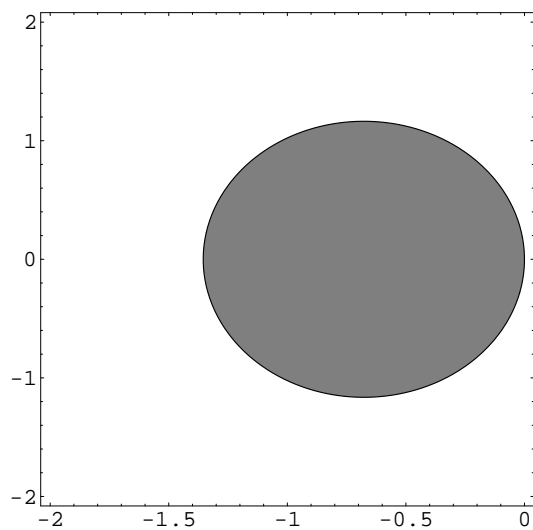
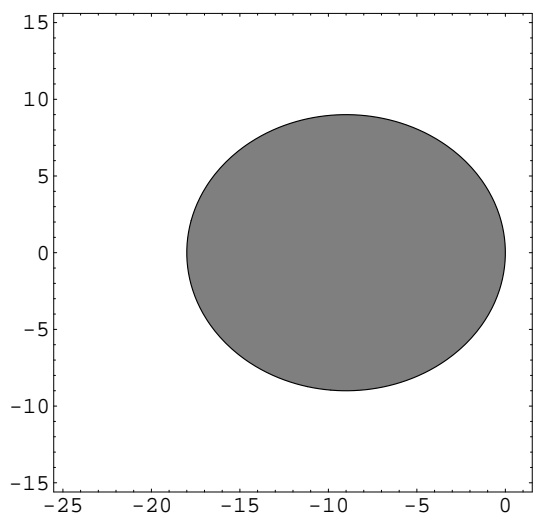
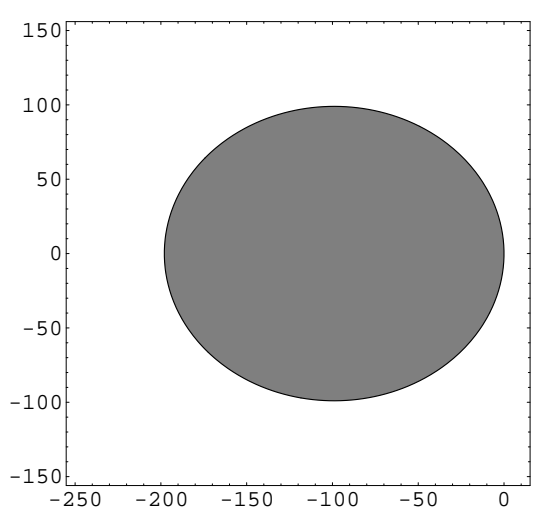
(a) $\lambda h = -0,01$.(b) $\lambda h = -1$.(c) $\lambda h = -10$.(d) $\lambda h = -100$.

Figura D.3: Regiones de estabilidad absoluta (en gris) del método explícito de dos pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

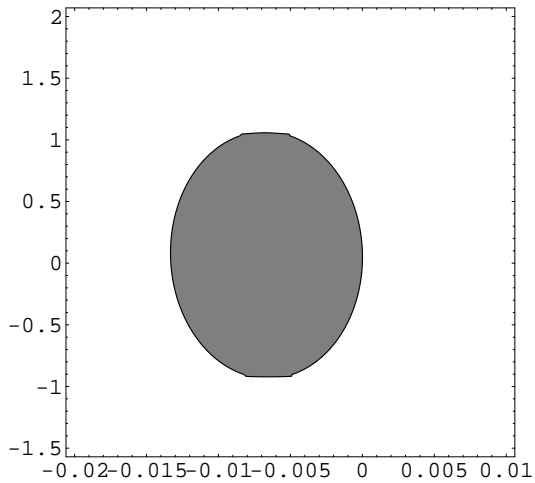
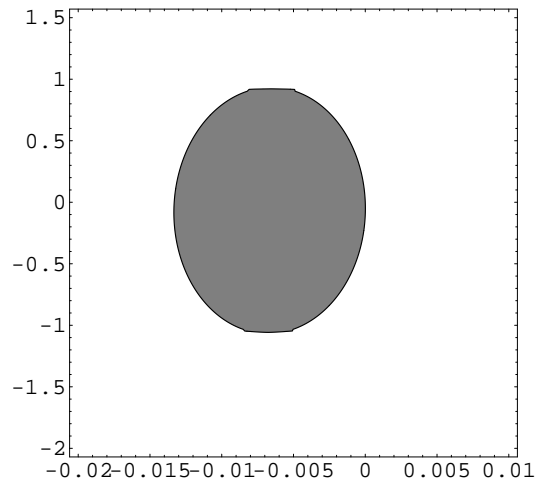
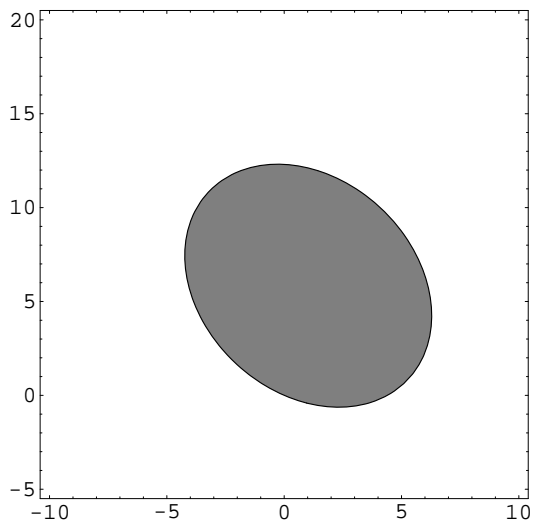
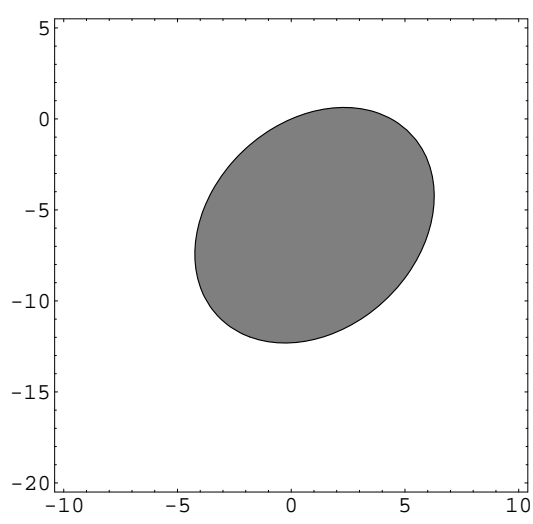
(a) $\lambda h = -0,01 + 0,1i$.(b) $\lambda h = -0,01 - 0,1i$.(c) $\lambda h = -1 + 10i$.(d) $\lambda h = -1 - 10i$.

Figura D.4: Regiones de estabilidad absoluta (en gris) del método explícito de dos pasos. El parámetro imaginario del método es λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

diferente tipo de problemas. Se observa que cuando $\|\lambda h\|$ crece la región donde el método es estable absolutamente también lo hace, y además lo hace de una forma similar al crecimiento de $\|\lambda h\|$.

En los métodos explícitos de orden exponencial 1 (como pudimos observar en el capítulo anterior), los métodos de varios pasos tenían una estabilidad muy compleja y débil. Las propiedades de estabilidad de los nuevos métodos explícitos son mucho mejores.

De hecho pueden ser bastante eficientes con un número bastante elevado de problemas donde los anteriores métodos tenían numerosas dificultades.

De alguna forma el comportamiento de la estabilidad de los nuevos métodos explícitos de orden superior a 2 es parecida a la del EF-I-k-7.

En los apéndices A y B vimos que para el método implícito de orden 7, había gran cantidad de problemas para los cuales el método alcanzaba buenos resultados cuando la longitud de paso h era grande, sin embargo, si decrecíamos h el método se volvía inestable. Con estos métodos sucede algo similar. De momento mostramos las regiones de estabilidad absoluta de los algoritmos explícitos EF-I-r-3 (figura D.5) y EF-I-r-5 (figura D.6), para distintos valores de $\lambda h \in R^-$.

Ya se puede observar que entre métodos con el mismo número de pasos, los métodos de orden algebraico 1, tienen mejores propiedades de estabilidad que los métodos de orden exponencial 1. De hecho en el próximo capítulo intentaremos mostrar que, en general, entre métodos con el mismo número de pasos, a mayor orden exponencial, en general habrá una mejor estabilidad.

Pasemos a comprobar si en los métodos implícitos se verifican las mismas propiedades que con los explícitos.

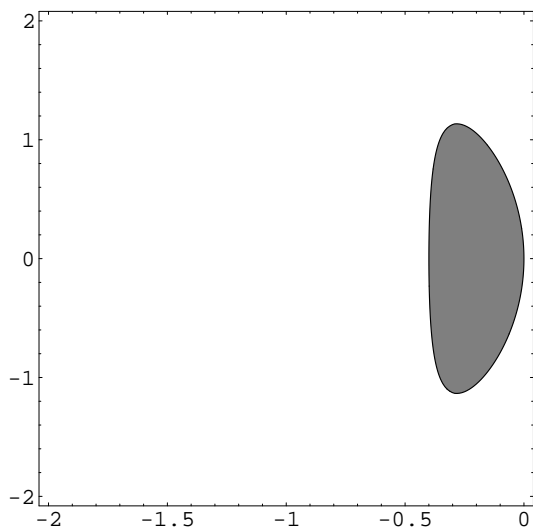
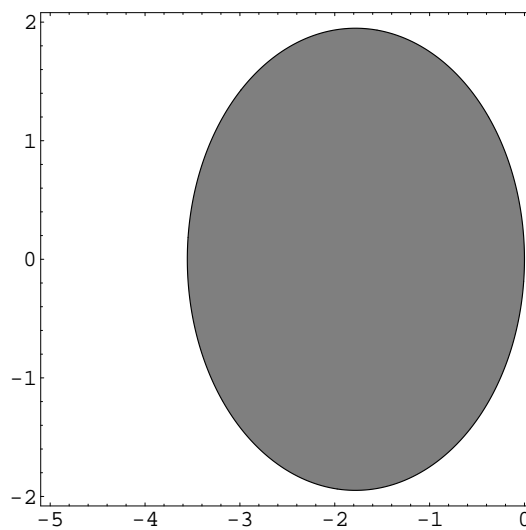
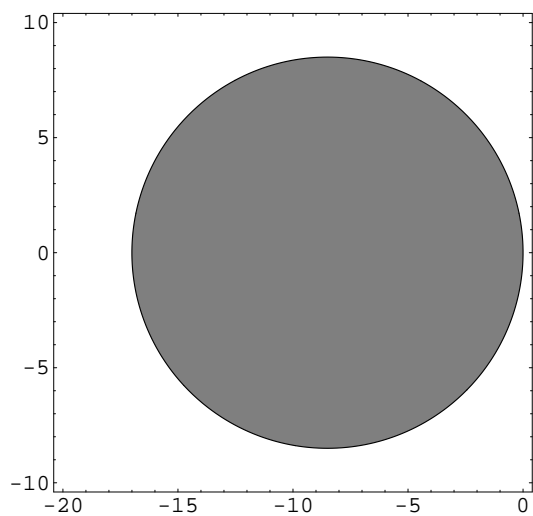
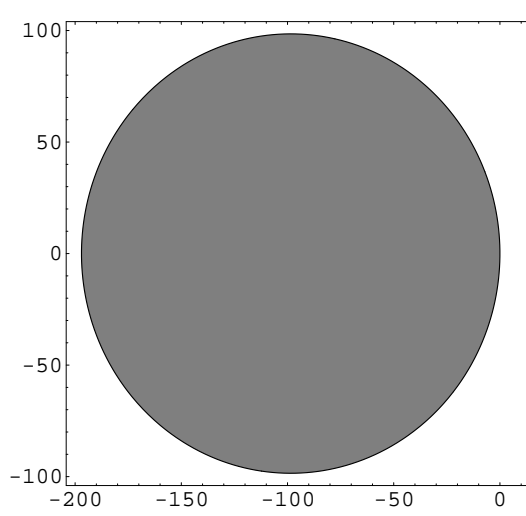
(a) $\lambda h = -1$.(b) $\lambda h = -3$.(c) $\lambda h = -10$.(d) $\lambda h = -100$.

Figura D.5: Regiones de estabilidad absoluta (en gris) del método explícito de tres pasos. El parámetro real del método es λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

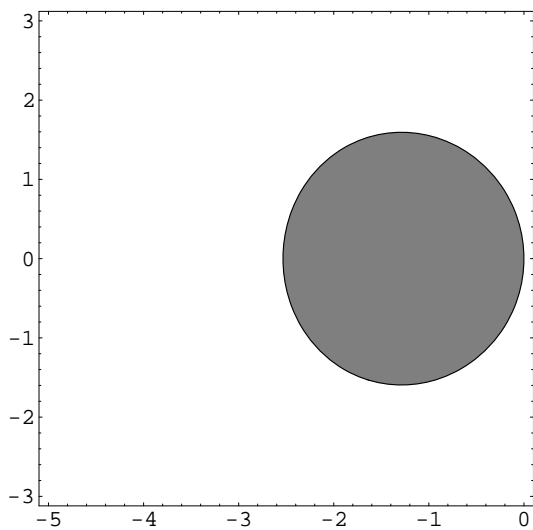
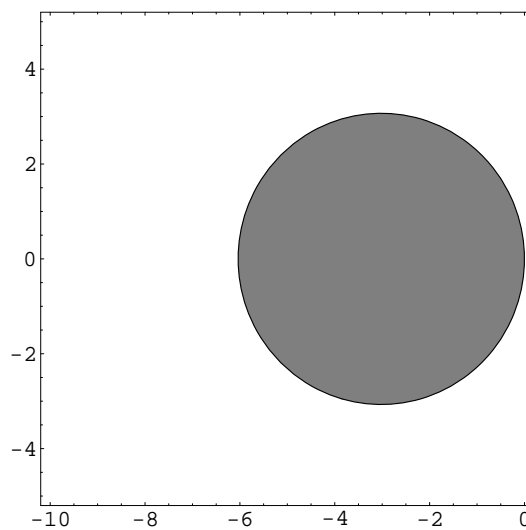
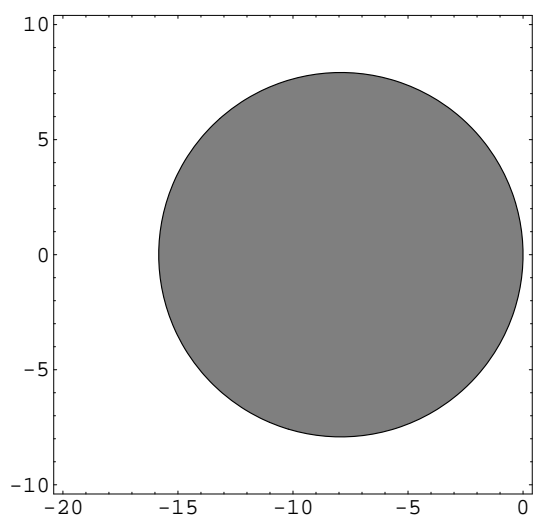
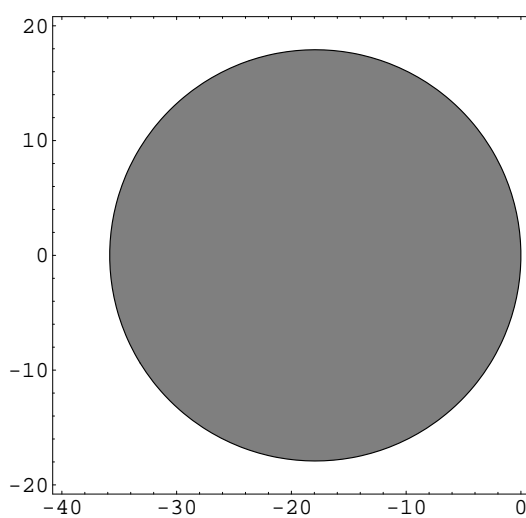
(a) $\lambda h = -3$.(b) $\lambda h = -5$.(c) $\lambda h = -10$.(d) $\lambda h = -20$.

Figura D.6: Regiones de estabilidad absoluta (en gris) del método explícito de cinco pasos. El parámetro real del método es λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

Para empezar, podemos mostrar una pequeña representación de las regiones de estabilidad absoluta de los métodos implícitos de 2 (figura D.7), 4 (figura D.8), 6 (figura D.9) y 8 pasos (figura D.10), con parámetro $\lambda h \in R$.

De nuevo vemos que las regiones de estabilidad absoluta de los métodos de orden algebraico 1 son más grandes que las regiones de estabilidad absoluta de los métodos de orden exponencial 1, sobre todo si $\lambda h \in R$ y $\lambda h < -1$.

También sucede que si $\lambda h \rightarrow -\infty$, las regiones de estabilidad crecen. Aunque no es cierto que las regiones de estabilidad crezcan (o las regiones de inestabilidad decrezcan) de forma proporcional al crecimiento del parámetro en valor absoluto (como más o menos pasaba con los métodos explícitos).

Estas afirmaciones que hemos hecho para parámetros reales negativo se pueden extrapolar en otras direcciones del plano complejo. Si estas direcciones no cortan zonas de 0-inestabilidad, las regiones de estabilidad absoluta crecen rápidamente cuando el parámetro crece en valor absoluto. Este hecho se puede observar en las figuras D.11 y D.12.

También se puede observar, que las regiones de estabilidad absoluta para los parámetros $\lambda h = a + bi$ y $\lambda h = a - bi$ son simétricas entre sí respecto del eje real.

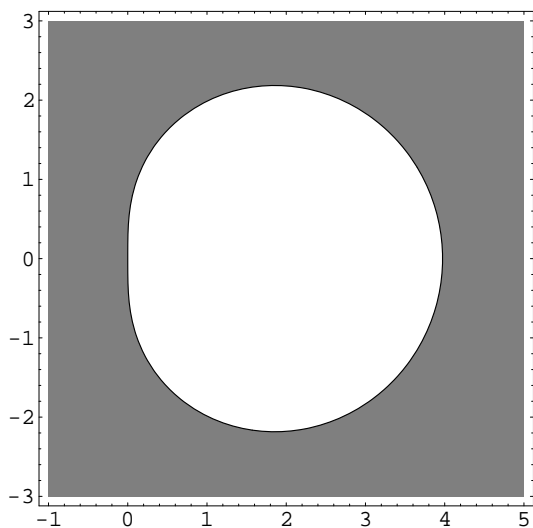
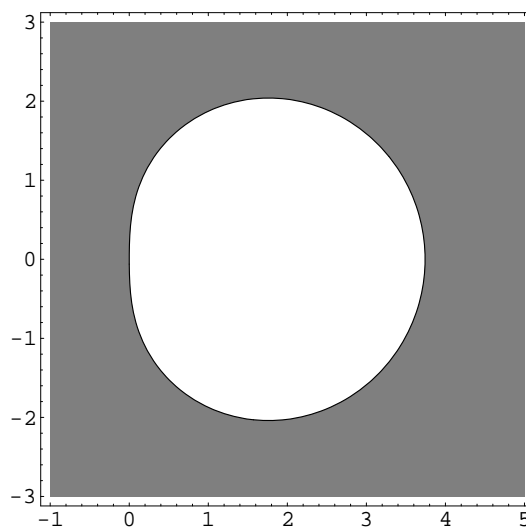
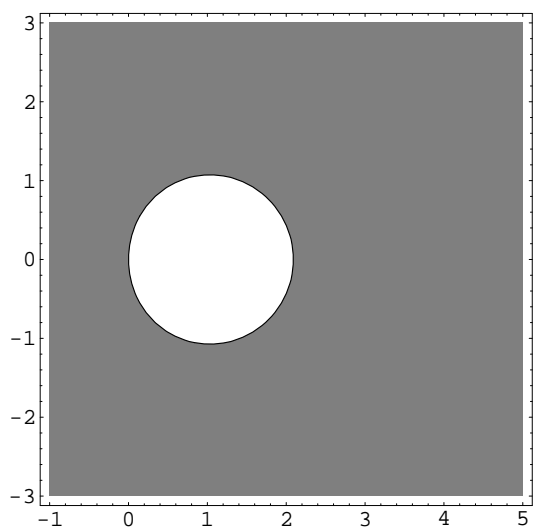
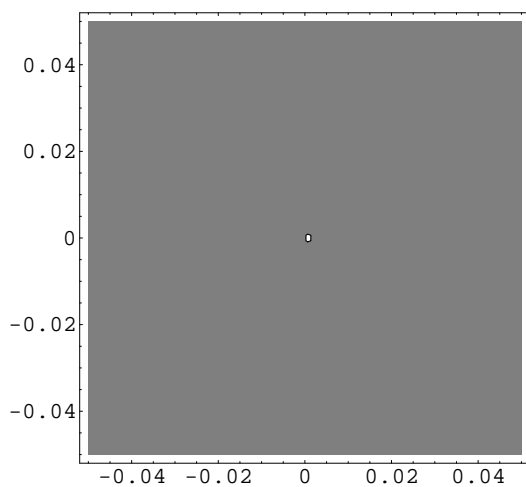
(a) $\lambda h = -0,01$.(b) $\lambda h = -0,1$.(c) $\lambda h = -1$.(d) $\lambda h = -10$.

Figura D.7: Regiones de estabilidad absoluta (en gris) del método implícito de dos pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

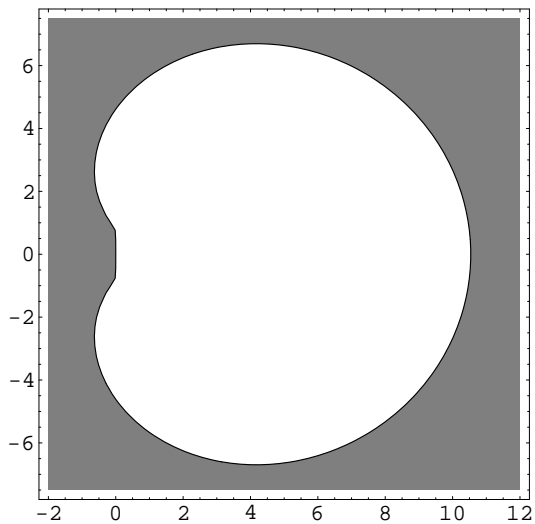
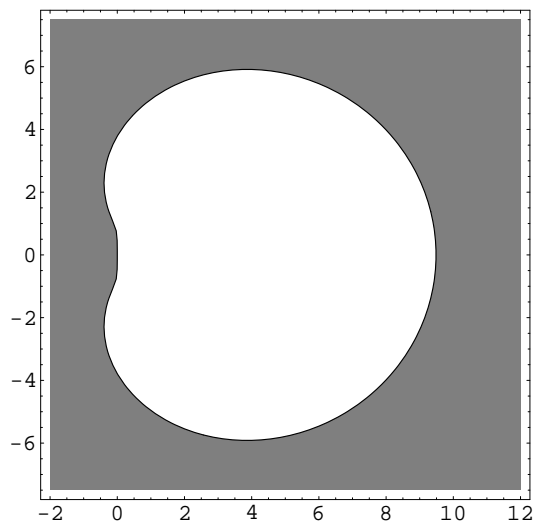
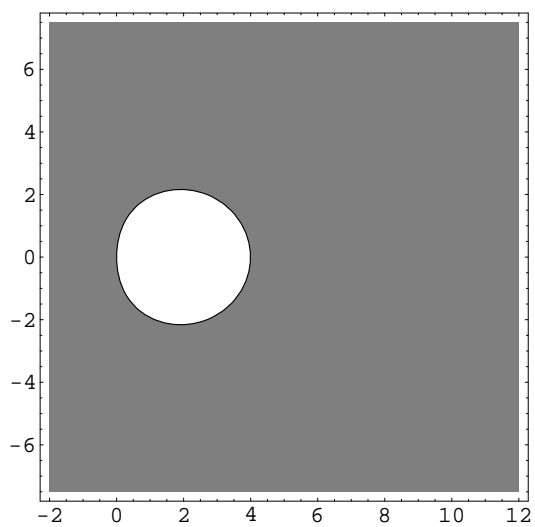
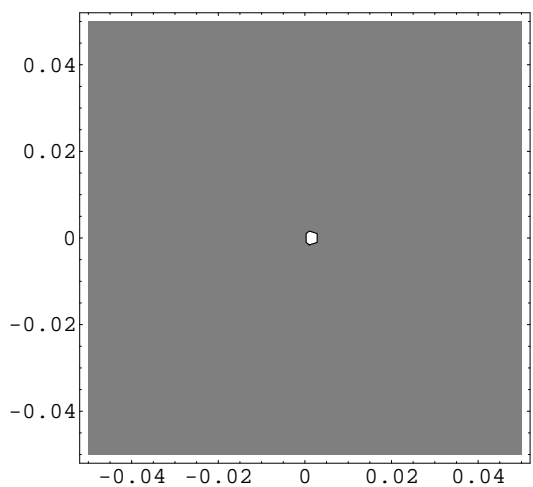
(a) $\lambda h = -0,01$.(b) $\lambda h = -0,1$.(c) $\lambda h = -1$.(d) $\lambda h = -10$.

Figura D.8: Regiones de estabilidad absoluta (en gris) del método implícito de cuatro pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

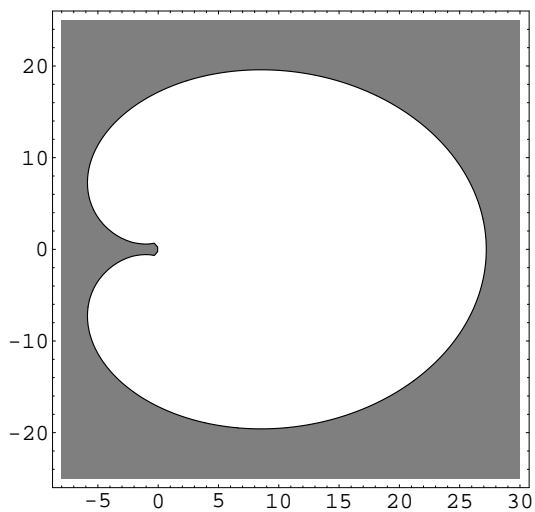
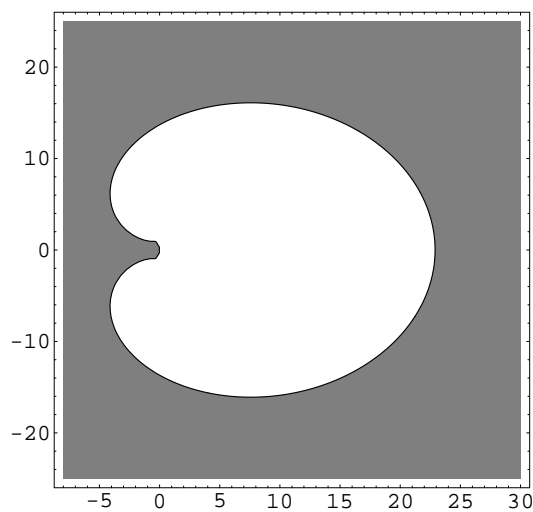
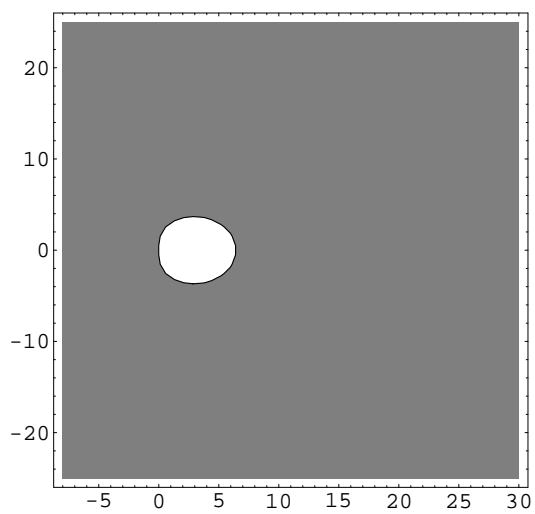
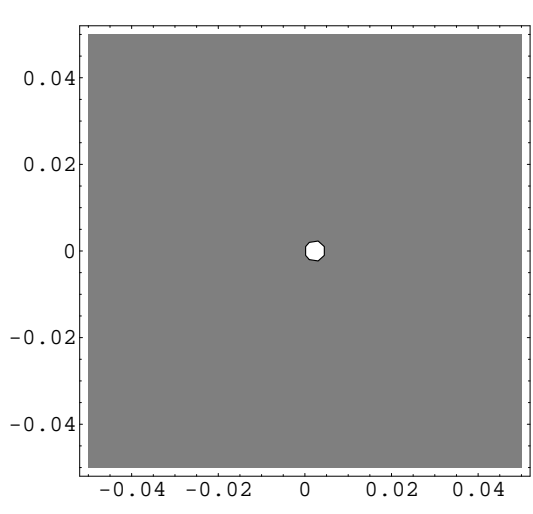
(a) $\lambda h = -0,01$.(b) $\lambda h = -0,1$.(c) $\lambda h = -3$.(d) $\lambda h = -10$.

Figura D.9: Regiones de estabilidad absoluta (en gris) del método implícito de seis pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

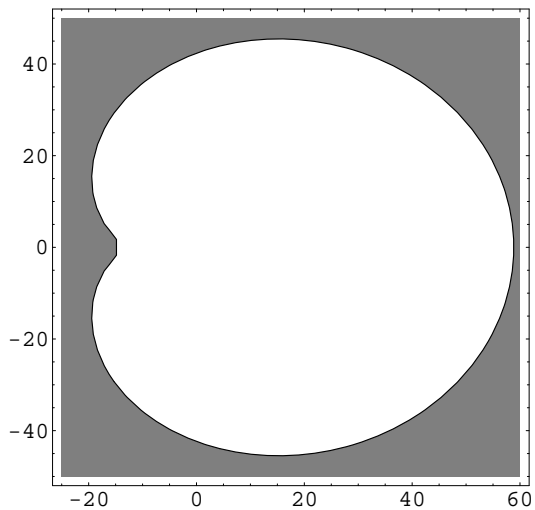
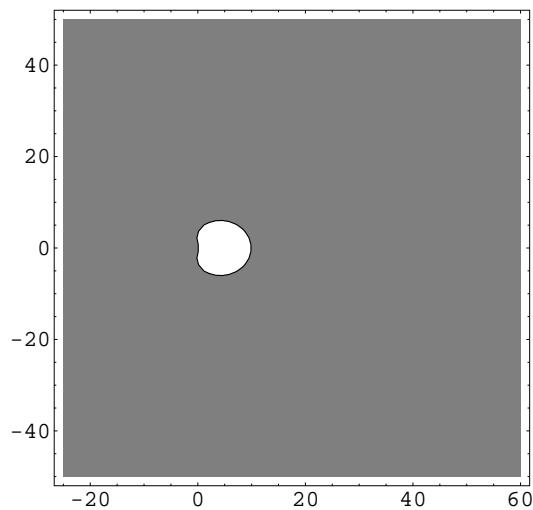
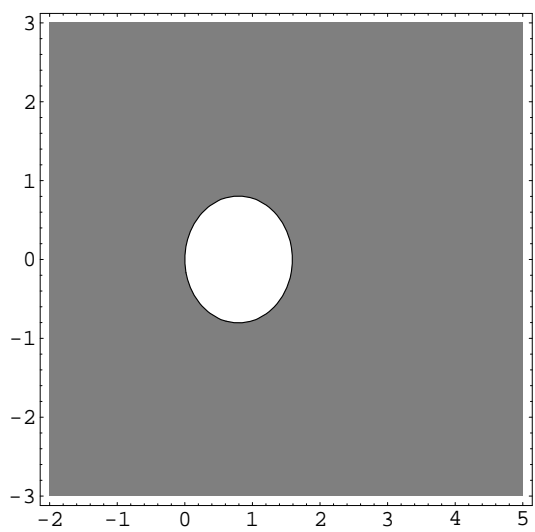
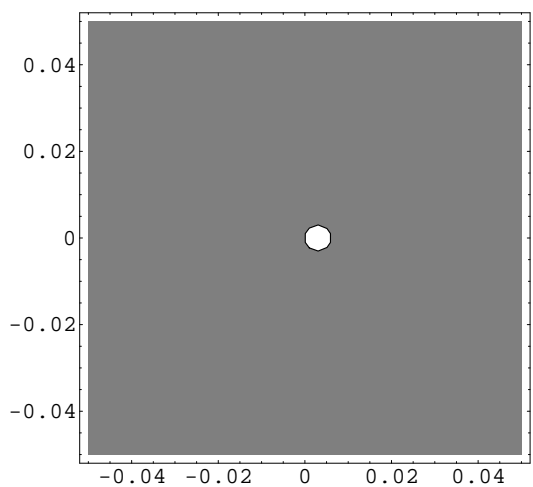
(a) $\lambda h = -0,1$.(b) $\lambda h = -1$.(c) $\lambda h = -3$.(d) $\lambda h = -10$.

Figura D.10: Regiones de estabilidad absoluta (en gris) del método implícito de ocho pasos. Parámetro real del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

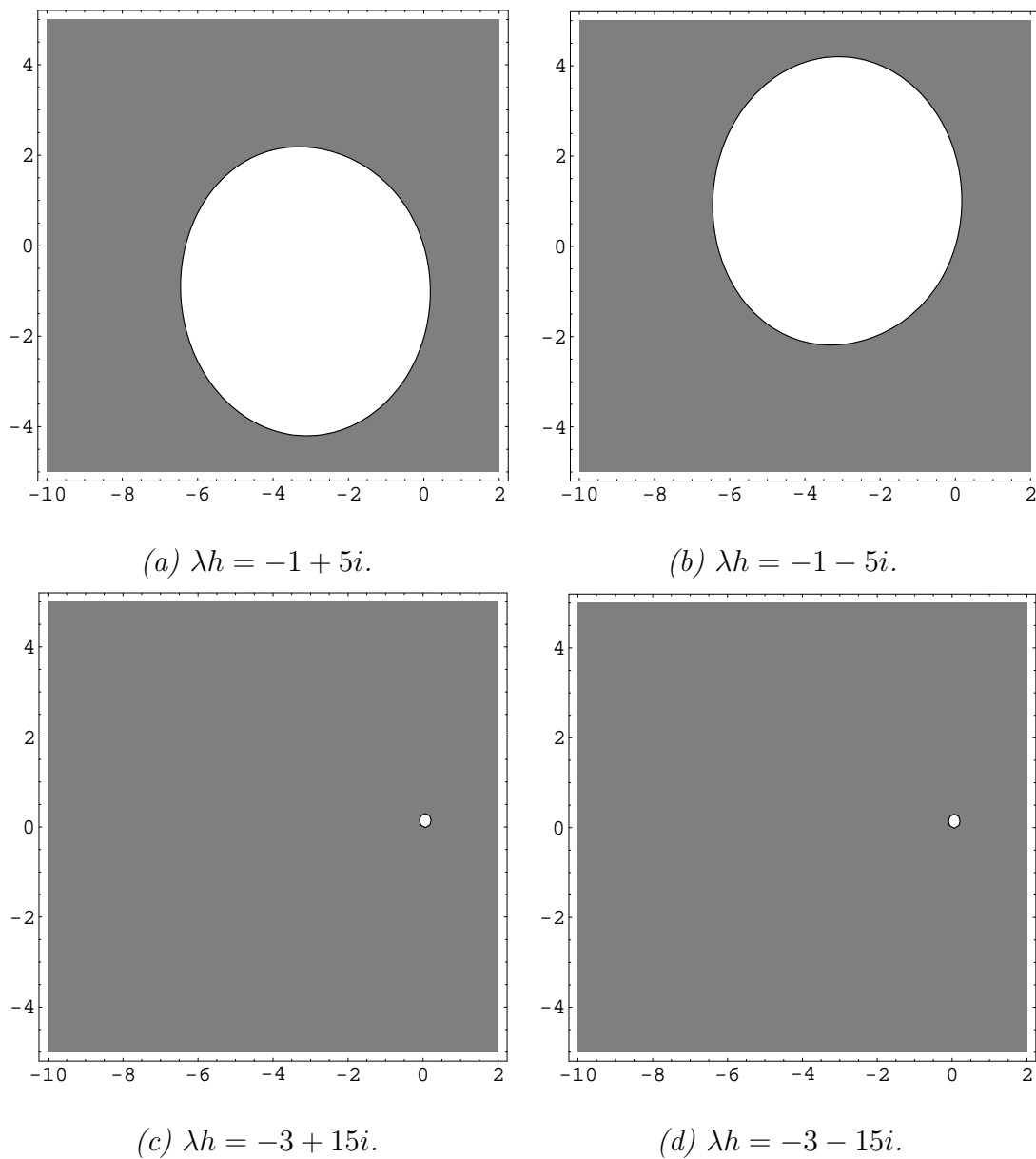


Figura D.11: Regiones de estabilidad absoluta (en gris) del método implícito de cuatro pasos. Parámetro imaginario del método λh . Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

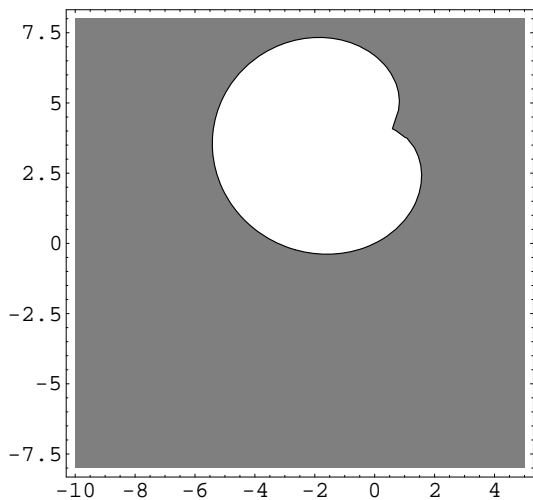
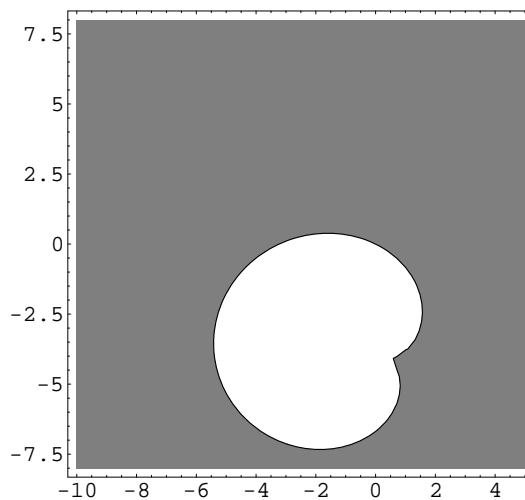
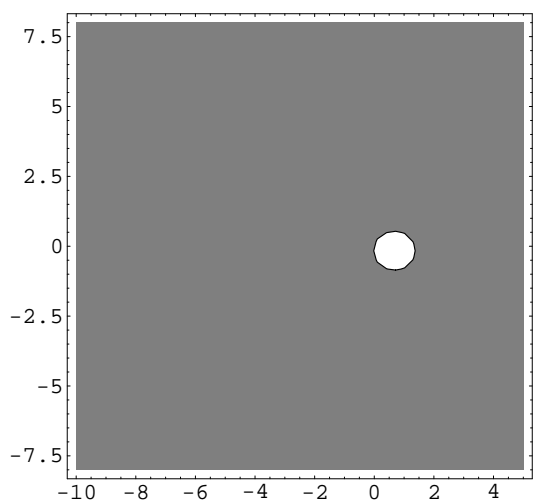
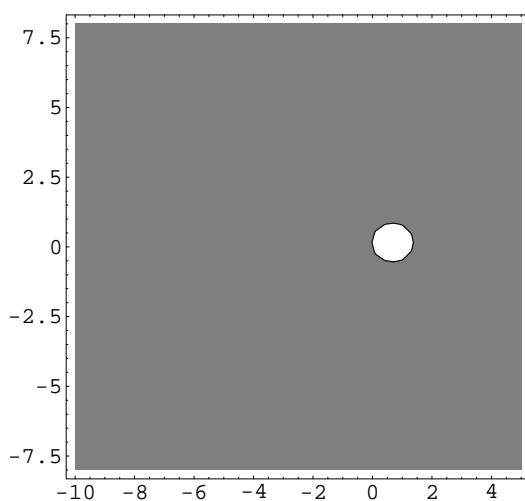
(a) $\lambda h = -1 + 4i$.(b) $\lambda h = -1 - 4i$.(c) $\lambda h = -5 + 20i$.(d) $\lambda h = -5 - 20i$.

Figura D.12: Regiones de estabilidad absoluta (en gris) del método implícito de cinco pasos. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

Apéndice E

0-estabilidad de otros métodos

EF- γ -k- δ -r o A- γ -k- δ -r.

El mismo razonamiento que hicimos en los apéndices y capítulos anteriores es perfectamente válido en este, por tanto, comprobaremos si para cada valor propio λ las raíces r_k del polinomio característico (C.0.1) se verifica que $|r_k| < 1$, y en aquellas raíces que tengan módulo 1 ver cuándo son simples.

Ya hemos comentado que las propiedades de estabilidad de los métodos de orden algebraico 1 son mejores que las de los métodos de orden exponencial 1. De hecho entre métodos (tanto los métodos explícitos por una parte, como los métodos implícitos, por otra) del mismo orden de convergencia, cuanto mayor es el orden exponencial mejores son las propiedades de estabilidad (en general).

Esto es algo que se observa con las regiones de estabilidad absoluta. Por el contrario en las figuras E.1 (algoritmo adaptado) y E.2 (algoritmo *exponential fitting*), hemos comparado las figuras de 0-estabilidad de los métodos de

4 pasos implícitos, donde no parece que se llegue a esta misma conclusión.

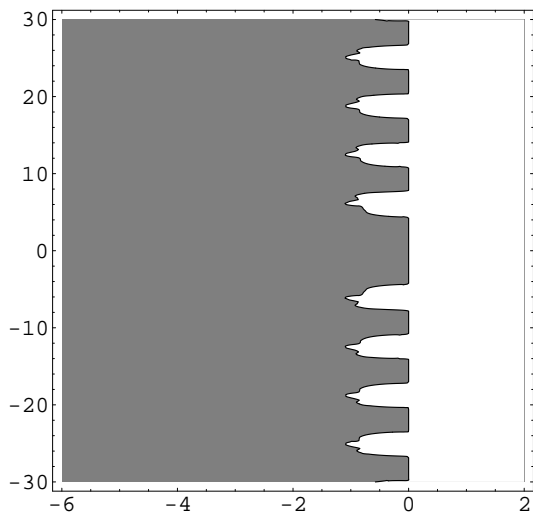
A continuación, en la figura E.3 vamos a mostrar una serie de algoritmos de tipo *exponential fitting* y adaptados explícitos y que no han sido estudiados hasta ahora, con regiones relativamente amplias.

Aunque algunos algoritmos *exponential fitting* explícitos muestran regiones de 0-estabilidad interesantes, las regiones de estabilidad absoluta son muy pobres. Además sólo los métodos explícitos de orden algebraico 1 (y bastante menos los de orden algebraico 2) tienen buenas regiones de 0-estabilidad.

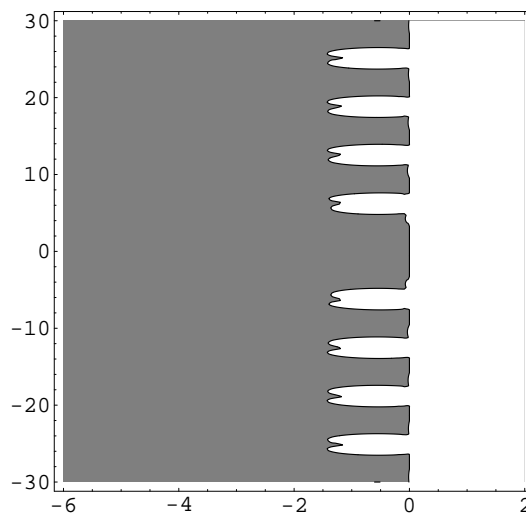
Los implícitos, claro, tienen mejores propiedades de estabilidad. Comenzaremos mostrando las regiones de 0-estabilidad del único método de 3 pasos, que todavía no ha sido mostrado. Podemos verlo en la figura E.4.

En la figura E.5 vamos a mostrar las regiones de 0-estabilidad de los métodos implícitos de orden 5 (que no han sido estudiados hasta ahora), mientras que en las figuras E.6 y E.7 mostramos las regiones de 0-estabilidad de los métodos implícitos de orden 6 (que no han sido estudiados hasta ahora).

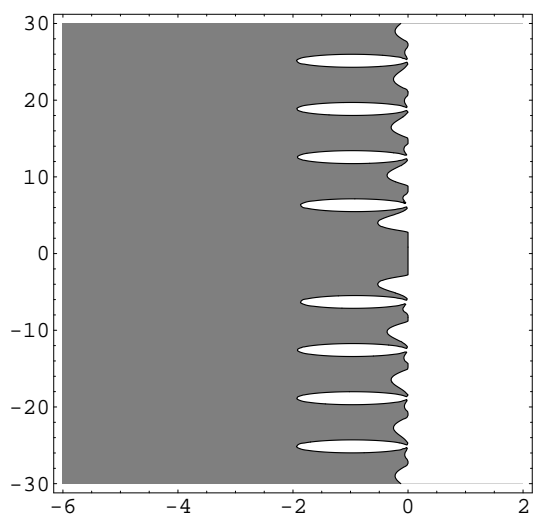
Por último, aunque hay métodos de más de 7 pasos cuyas regiones de 0-estabilidad pueden ser interesantes, no vamos a mostrar las regiones de estos métodos. Tan sólo lo haremos con las regiones de 0-estabilidad de los de 7 pasos, para mostrar que estos ya tienen un entorno del origen donde los algoritmos son 0-inestables. Estas regiones se pueden observar en las figuras E.8 y E.9.



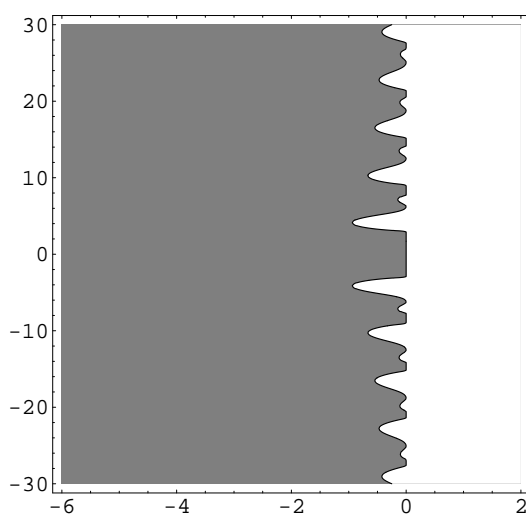
(a) $A-1-k-4-r$.



(b) $A-2-k-3-r$.



(c) $A-3-k-2-r$.



(d) $A-4-k-1-r$.

Figura E.1: Regiones de 0-estabilidad (en gris) de los algoritmos adaptados de 4 pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

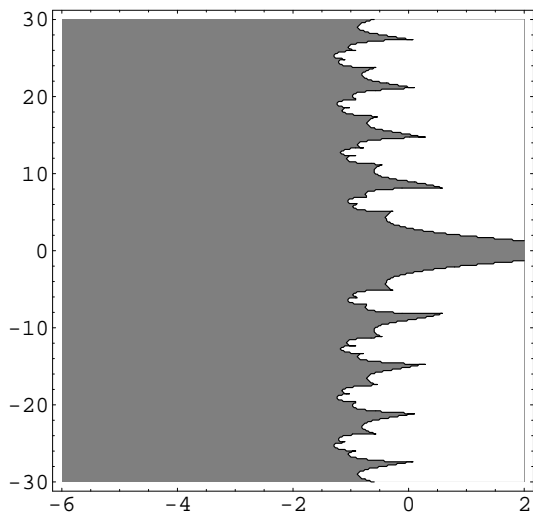
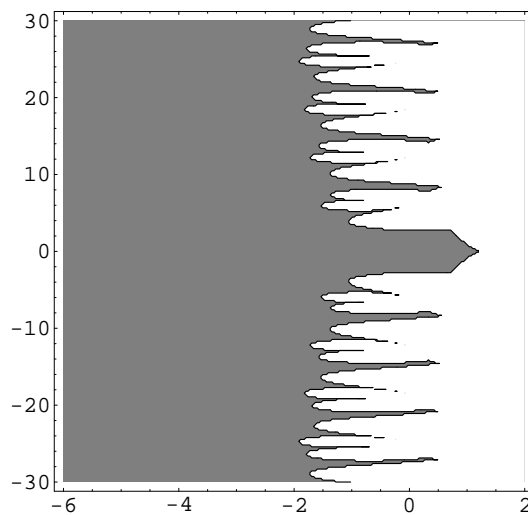
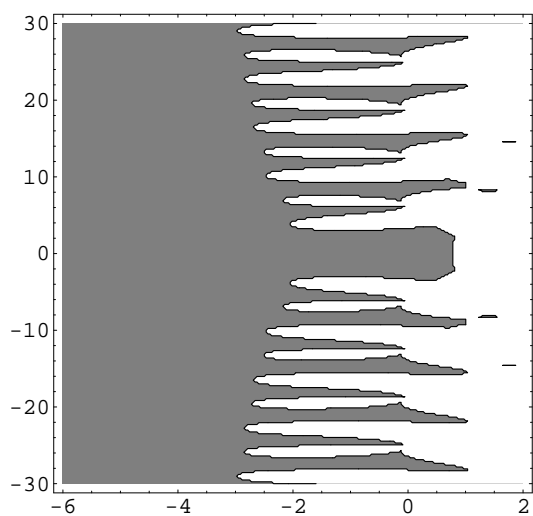
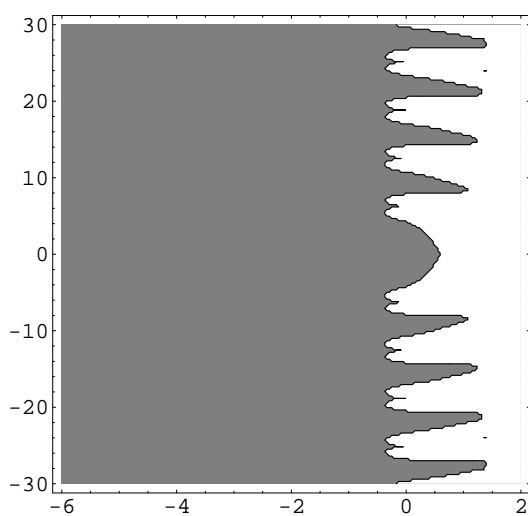
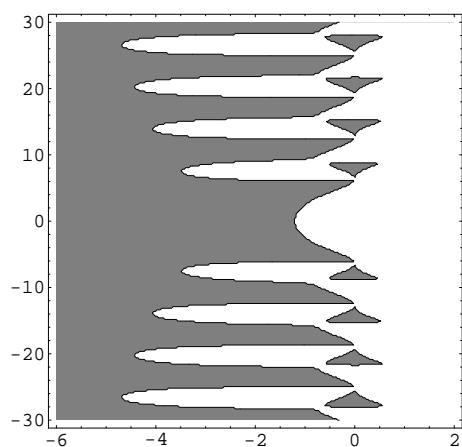
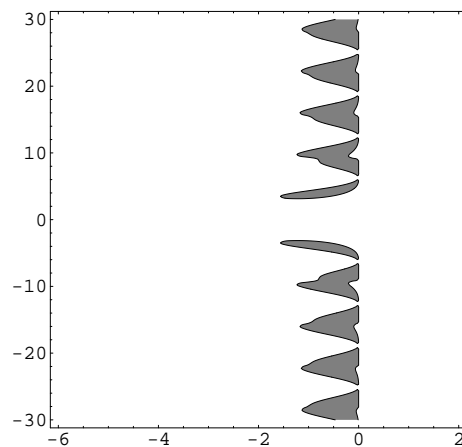
(a) $EF-1-k-4-r$.(b) $EF-2-k-3-r$.(c) $EF-3-k-2-r$.(d) $EF-4-k-1-r$.

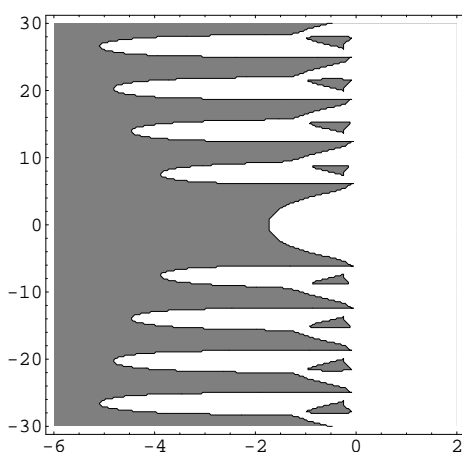
Figura E.2: Regiones de 0-estabilidad (en gris) de los algoritmos *exponential fitting* de 4 pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.



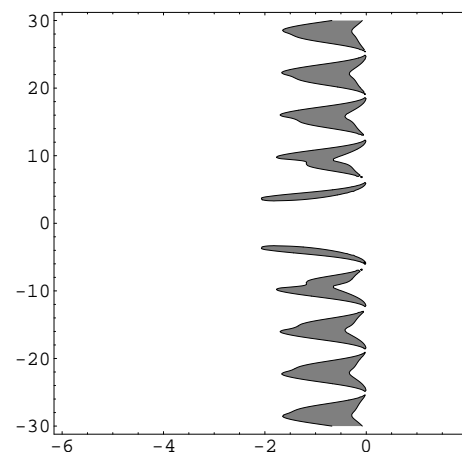
(a) $EF-2-k-2-r$.



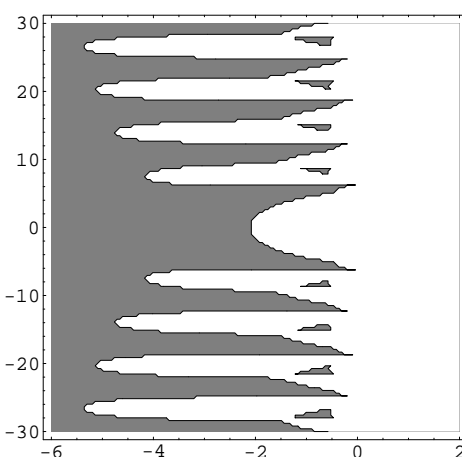
(b) $A-2-k-2-r$.



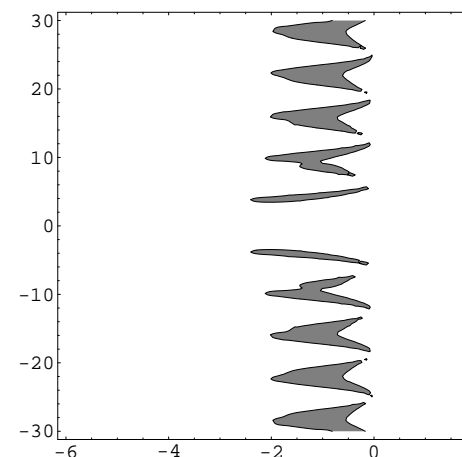
(c) $EF-3-k-2-r$.



(d) $A-3-k-2-r$.

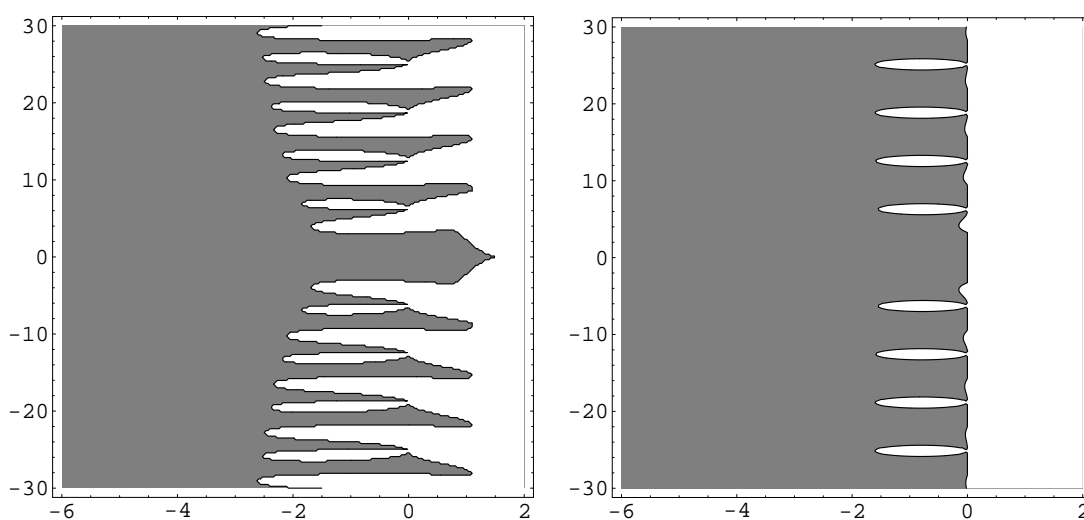


(e) $EF-4-k-2-r$.



(f) $A-4-k-2-r$.

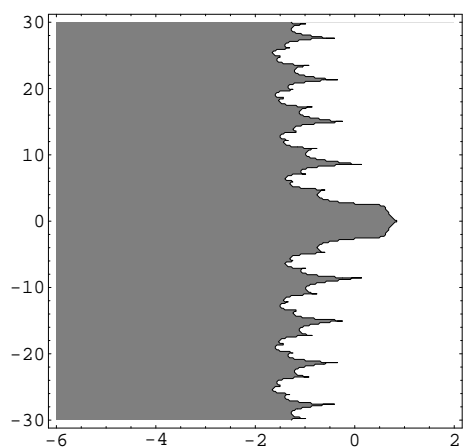
Figura E.3: Regiones de 0-estabilidad (en gris) de algoritmos explícitos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.



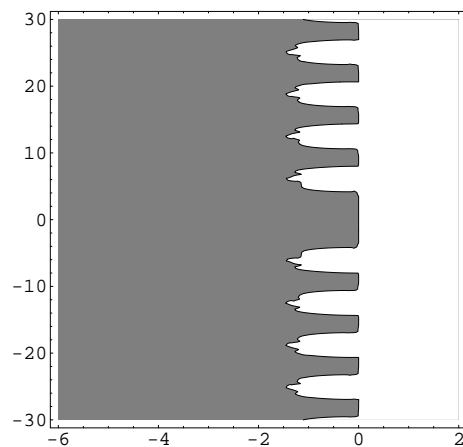
(a) $EF-2-k-2-r$.

(b) $A-2-k-2-r$.

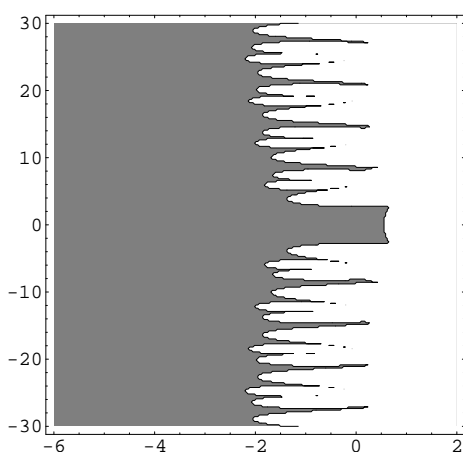
Figura E.4: Regiones de 0 -estabilidad (en gris) de los algoritmos $EF-2-k-2-r$ y $A-2-k-2-r$. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.



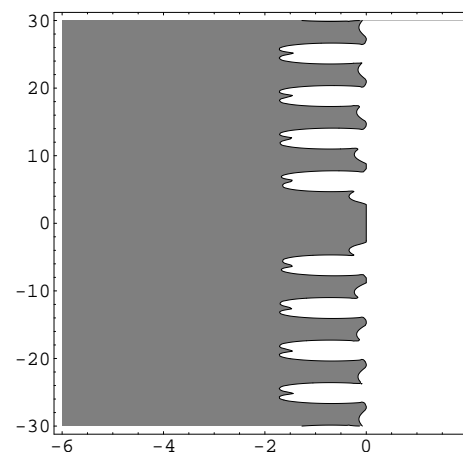
(a) $EF-2-k-4-r$.



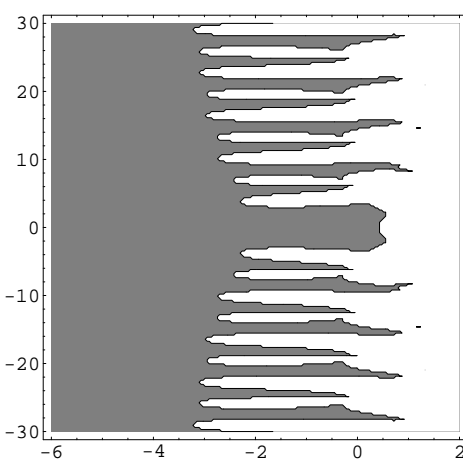
(b) $A-2-k-4-r$.



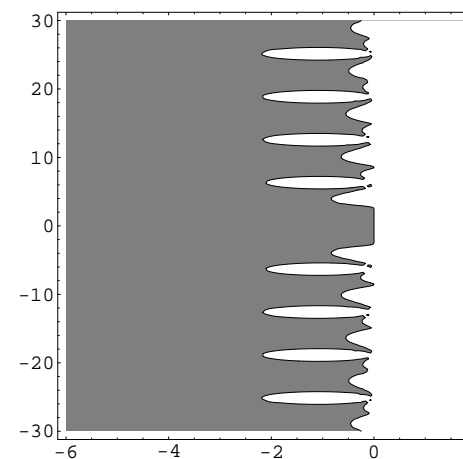
(c) $EF-3-k-3-r$.



(d) $A-3-k-3-r$.



(e) $EF-4-k-2-r$.



(f) $A-4-k-2-r$.

Figura E.5: Regiones de 0-estabilidad (en gris) de los algoritmos de cinco pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

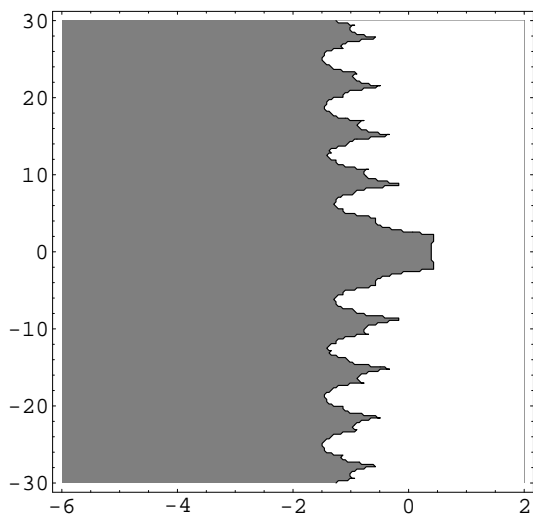
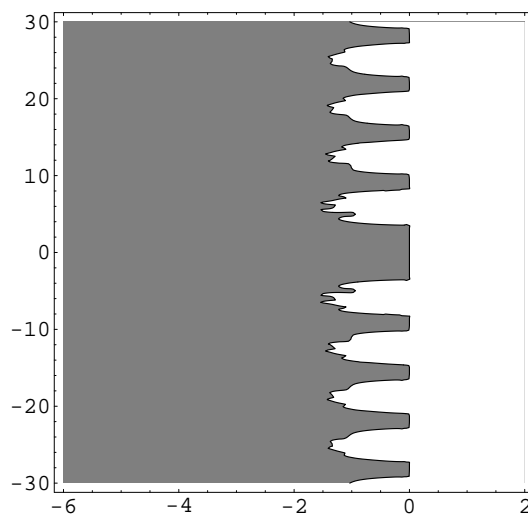
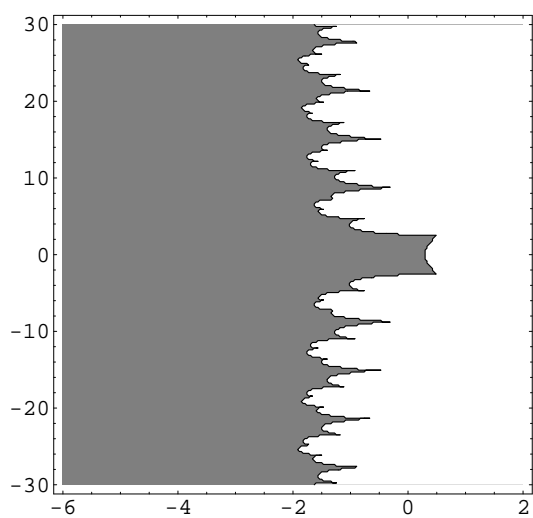
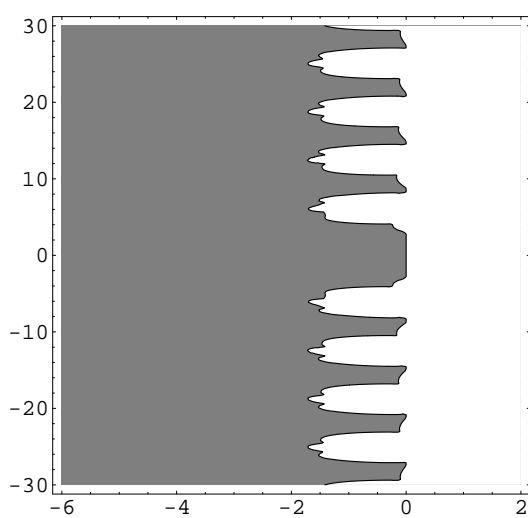
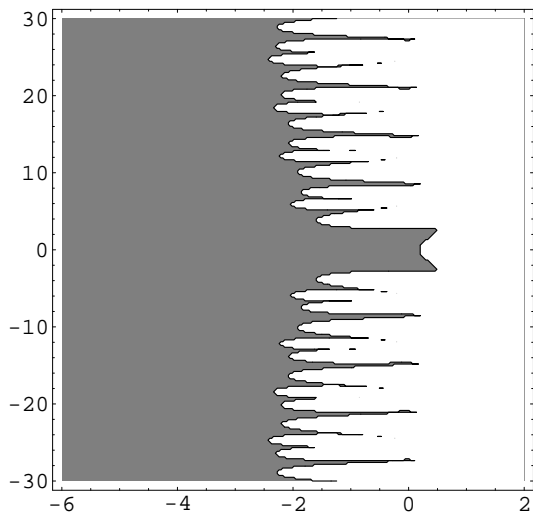
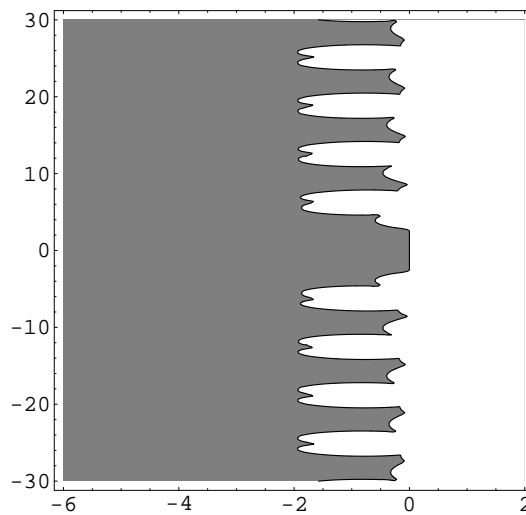
(a) $EF-2-k-5-r$.(b) $A-2-k-5-r$.(c) $EF-3-k-4-r$.(d) $A-3-k-4-r$.

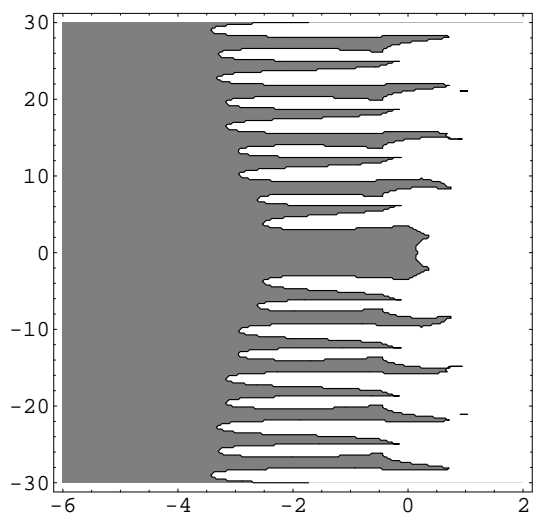
Figura E.6: Regiones de 0-estabilidad (en gris) de los algoritmos de seis pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.



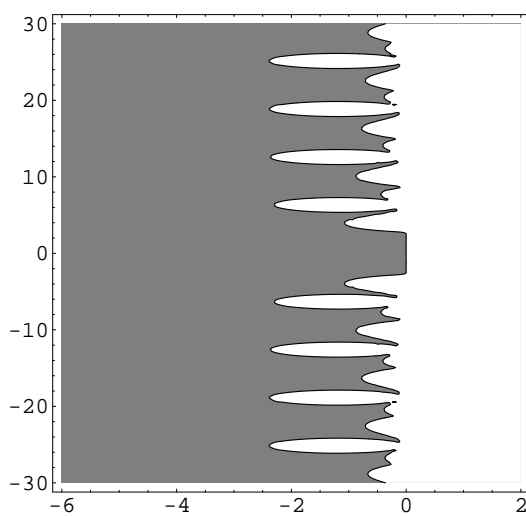
(e) $EF-4-k-3-r$.



(f) $A-4-k-3-r$.



(g) $EF-5-k-2-r$.



(h) $A-5-k-2-r$.

Figura E.7: Regiones de 0-estabilidad (en gris) de los algoritmos de seis pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

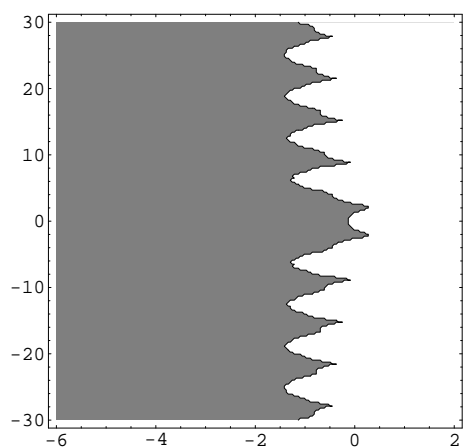
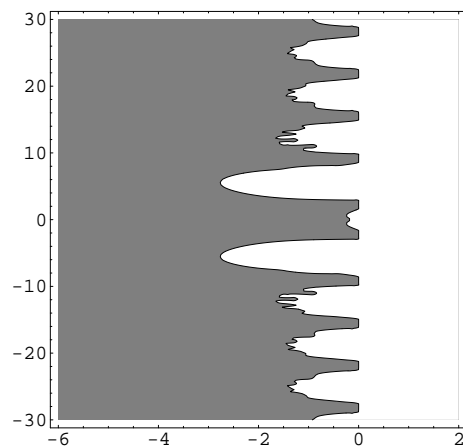
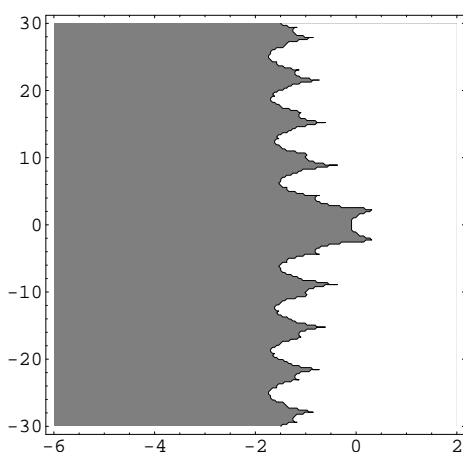
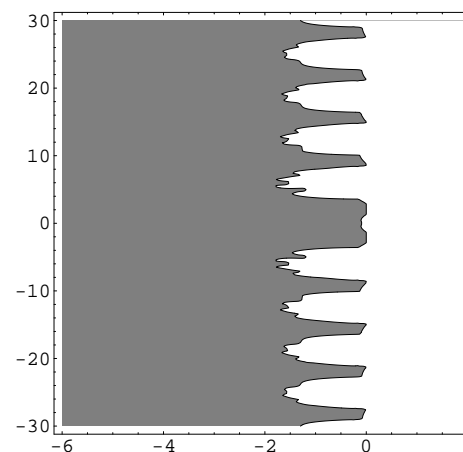
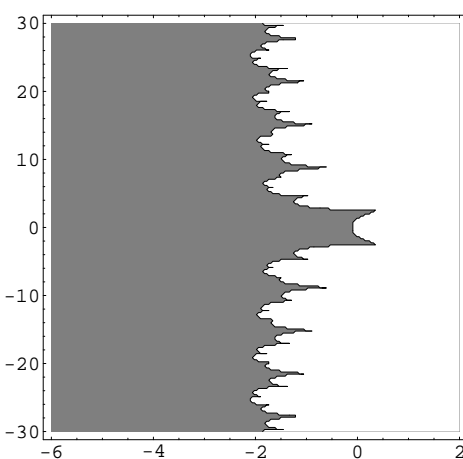
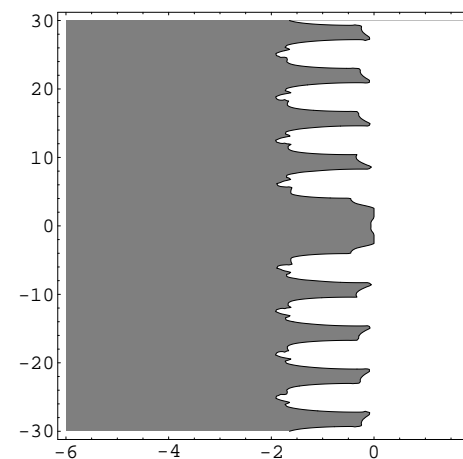
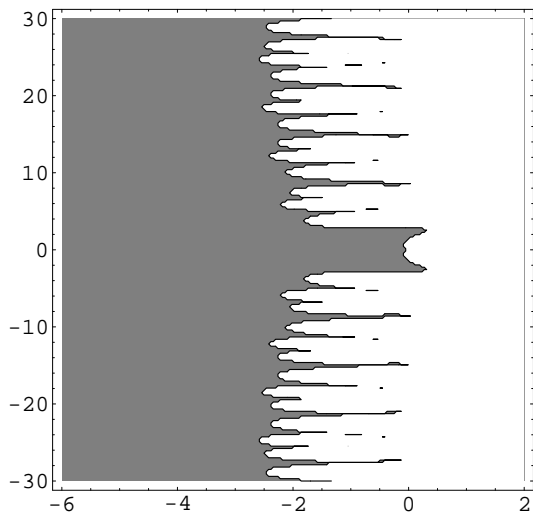
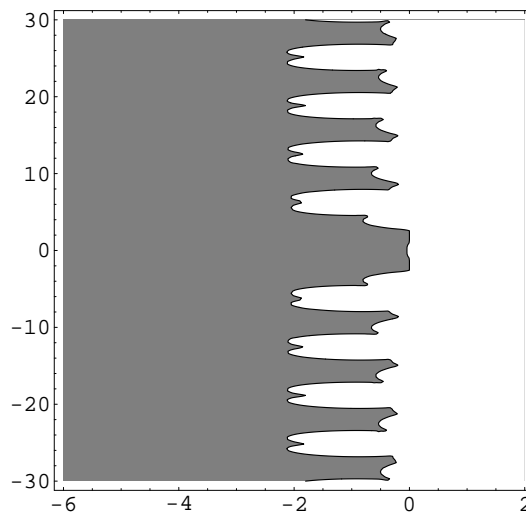
(a) $EF-2-k-6-r$.(b) $A-2-k-6-r$.(c) $EF-3-k-5-r$.(d) $A-3-k-5-r$.(e) $EF-4-k-4-r$.(f) $A-4-k-4-r$.

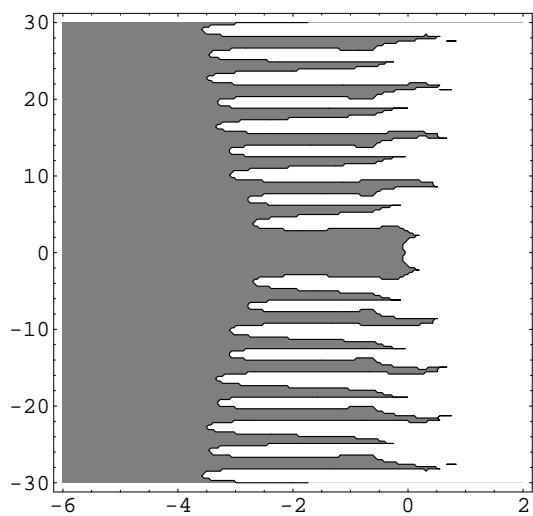
Figura E.8: Regiones de 0 -estabilidad (en gris) de los algoritmos de siete pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.



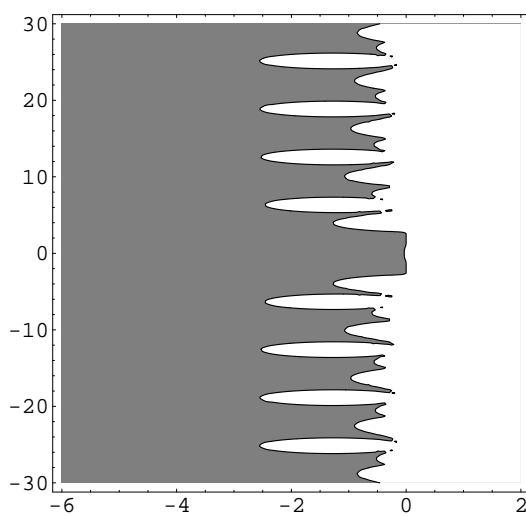
(g) $EF-5-k-3-r$.



(h) $A-5-k-3-r$.



(i) $EF-6-k-2-r$.



(j) $A-6-k-2-r$.

Figura E.9: Regiones de 0-estabilidad (en gris) de los algoritmos de siete pasos. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

Apéndice F

Estabilidad absoluta de otros métodos EF- γ -k- δ -r o A- γ -k- δ -r.

Como se ha explicado en el capítulo referente a los métodos *exponential fitting* o adaptados de orden exponencial y algebraico mayor que 1, vamos a pretender llegar a las siguientes conclusiones:

- 1.- Observamos que las regiones de inestabilidad absoluta de un método dado pueden serán más grandes cuanto λh esté más cerca de las regiones de 0-inestabilidad de los algoritmos correspondientes a dicho método.
- 2.- Observamos que las regiones de estabilidad absoluta crecen si $\lambda h \rightarrow -\infty$ ($\lambda h \in R^-$); si el parámetro $|\lambda h|$ es muy grande, el error al elegir dicho parámetro puede ser más grande y sin embargo, el parámetro continúa siendo estable.
- 3.- Si $\lambda h \in C^-$, las regiones de estabilidad absoluta crecen para λht cuando $t \rightarrow \infty$ (considerando que los λht pertenecen a las regiones de 0-estabilidad).
- 4.- Si $\lambda_1 h = a + ib \in C^-$, $a, b \in R$ y $\lambda_2 h = a - ib$, tomamos un único método y

fijamos por una parte $\lambda_1 h$ y por otra $\lambda_2 h$, las regiones de estabilidad absoluta de las dos figuras son simétricas entre sí.

Además, ya hemos mencionado que entre dos métodos del mismo número de pasos, el de mayor orden exponencial tiene mejores propiedades de estabilidad, pero es precisamente en este apéndice donde mejor vamos a poder comprobar esta afirmación.

Sin duda, para mostrar estas propiedades, podríamos hacer uso de un número muy elevado de figuras, aquí hemos elegido una pequeña representación, como podríamos haber elegido cualquier otra. Eso sí, una vez ya han sido comprobadas las pobres regiones de 0-estabilidad de los métodos explícitos que no tienen orden algebraico 1, las figuras aquí presentadas pertenecen todas a métodos implícitos.

Para mostrar la propiedad 1, nos puede valer cualquier método, y eligiendo un parámetro cerca de las regiones de 0-inestabilidad y un parámetro bastante similar pero más alejado de la zona de 0-inestabilidad se observa este comportamiento. Puesto que es una propiedad que hemos observado con un método de orden exponencial 1 y con otro de orden algebraico 1, ahora vamos a elegir otros dos que no pertenezcan a ninguna de estas familias de métodos.

Para empezar elegimos el método EF-2-k-2-r, hemos dibujado la región de 0-estabilidad del algoritmo adaptado en torno al valor $\lambda h = -1 + 7i$, que está bastante cerca de una zona de 0-inestabilidad. Como se puede comprobar con la figura (F.1).

Por esa razón la región de inestabilidad absoluta para el parámetro $\lambda h = -1 + 7i$ es mucho mayor que la correspondiente al parámetro $\lambda h = -1 + 8i$

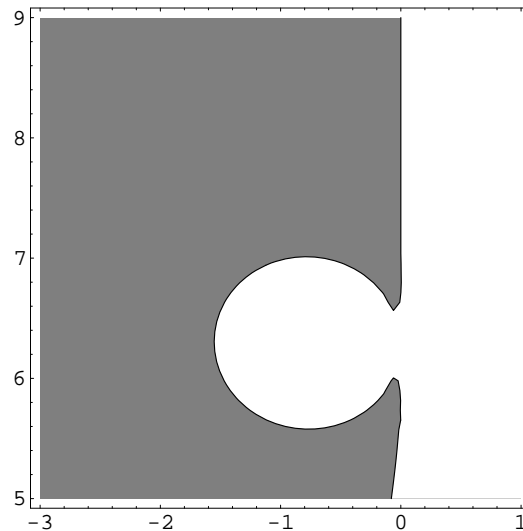


Figura F.1: Detalle de la región de 0-estabilidad del método implícito A-2-k-2-r. El eje horizontal representa $Re(\lambda h)$, mientras que el vertical representa $Im(\lambda h)$.

(más alejado de la zona de 0-inestabilidad), como se puede observar en la figura (F.2).

También hemos elegido para observar esta afirmación el método EF-2-k-3-r. Como se ve en la figura (F.3) el parámetro $\lambda h = -1 + 4,9i$ queda muy cerca de la zona de 0-inestabilidad del algoritmo adaptado, mucho más que el parámetro $\lambda h = -1 + 3i$, las consecuencias en las propiedades de estabilidad absoluta se pueden observar en la figura (F.4).

Una vez que hemos visto como se repite la propiedad 1, vamos a estudiar la 2. Para ello hemos representado dibujos de estabilidad absoluta de todos los métodos de 4 pasos con varios parámetros ($\lambda h = -0,01, -1, -10$) en las figuras F.5, F.6 y F.7, respectivamente, y de los métodos de 6 pasos con los parámetros $\lambda h = -0,1, -1, -10$ (salvo para los métodos EF-1-k-6-r y EF-6-k-1-r, que son ya bastante más conocidos) en las figuras F.8, F.9 y F.10.

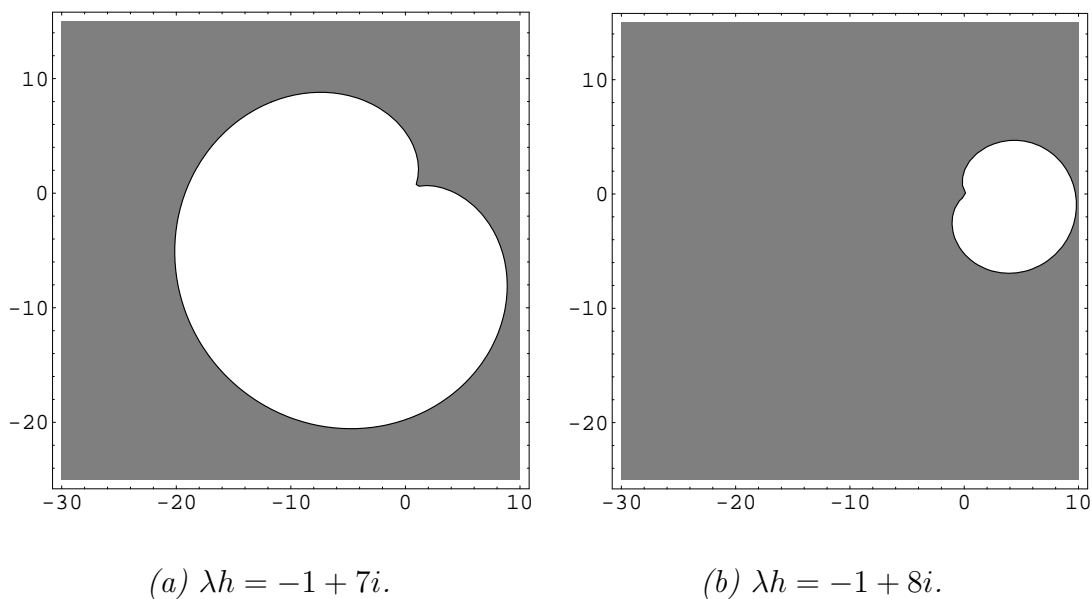


Figura F.2: Regiones de estabilidad absoluta (en gris) del método EF-2-k-2-r, con un parámetro más cercano a la zona de 0-inestabilidad ($\lambda h = -1 + 7i$) y el otro más alejado ($\lambda h = -1 + 8i$). Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

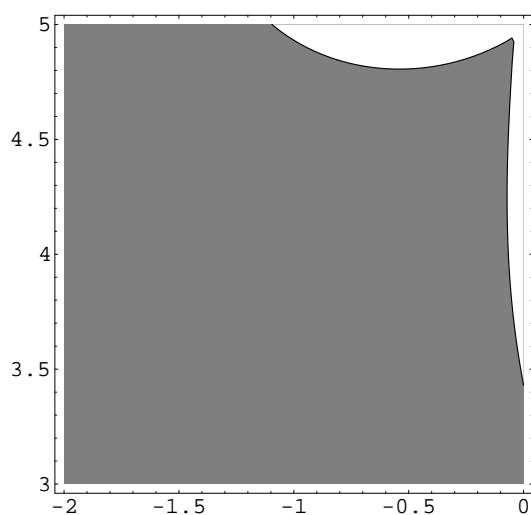


Figura F.3: Detalle de la región de 0-estabilidad del método implícito A-2-k-3-r. El eje horizontal representa $Re(\lambda h)$, mientras que el vertical representa $Im(\lambda h)$.

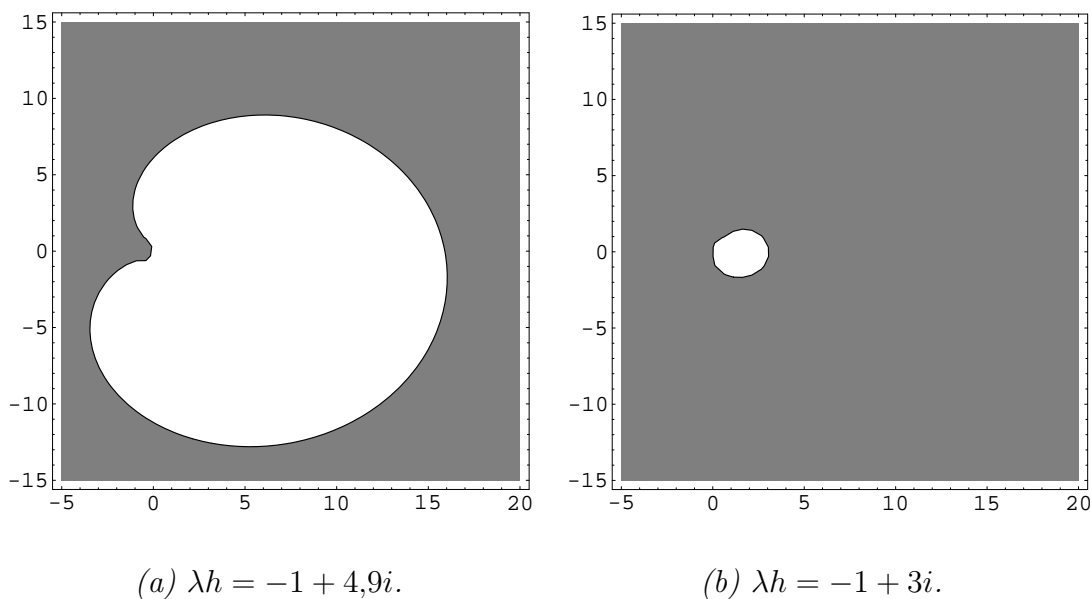


Figura F.4: Regiones de estabilidad absoluta (en gris) del método EF-2-k-3-r, con un parámetro más cercano a la zona de 0-inestabilidad ($\lambda h = -1 + 4,9i$) y el otro más alejado ($\lambda h = -1 + 3i$). Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

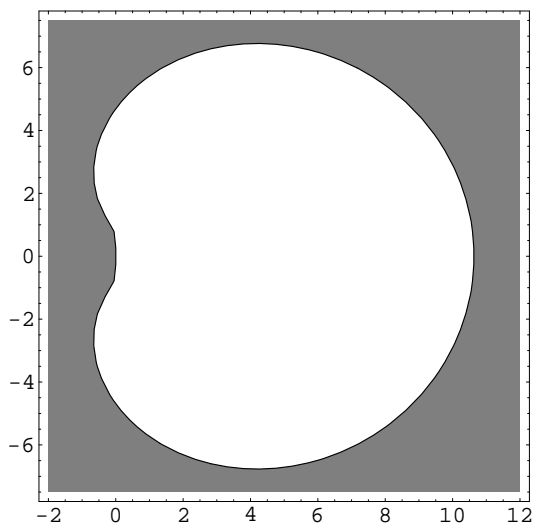
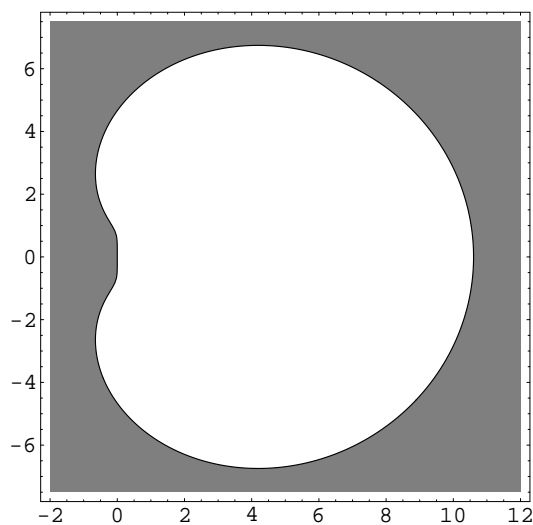
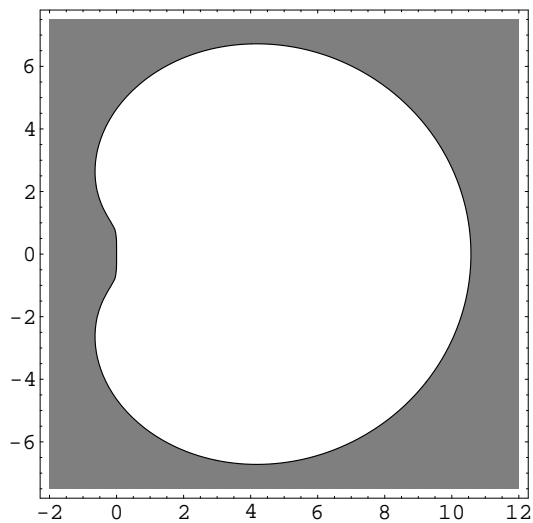
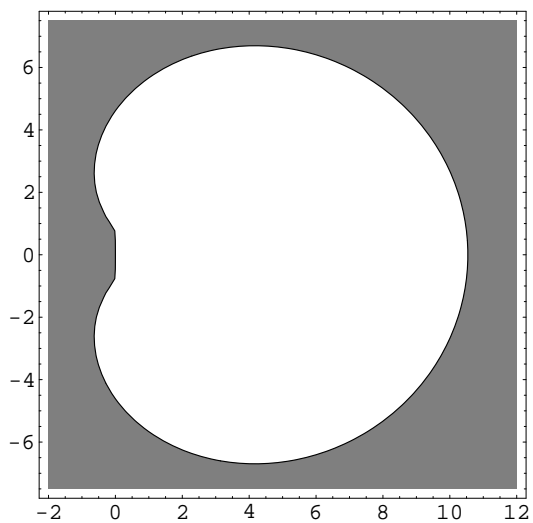
(a) $EF-1-k-4-r$.(b) $EF-2-k-3-r$.(c) $EF-3-k-2-r$.(d) $EF-4-k-1-r$.

Figura F.5: Regiones de estabilidad absoluta (en gris) de los métodos implícitos de cuatro pasos. El parámetro real del método es $\lambda h = -0,01$. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

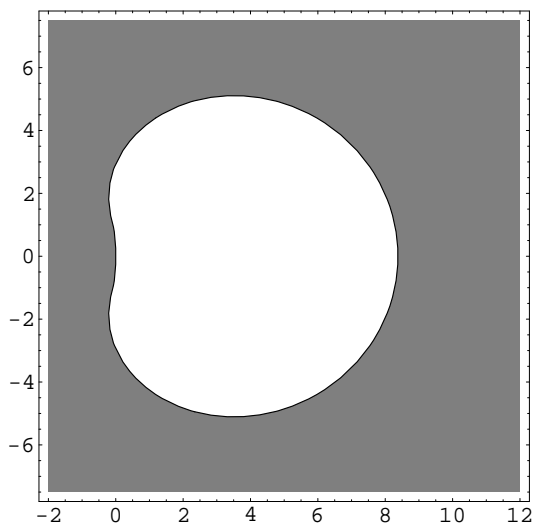
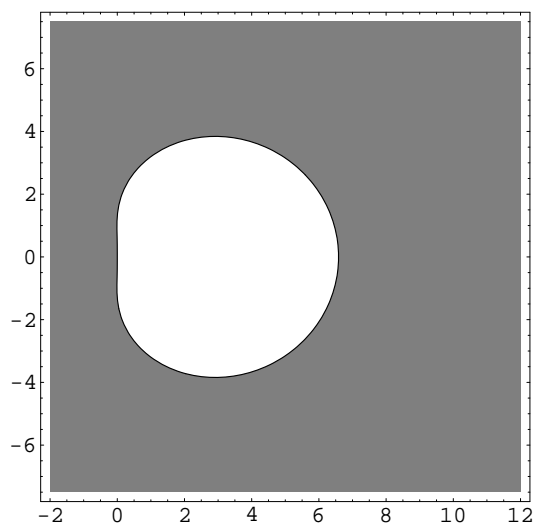
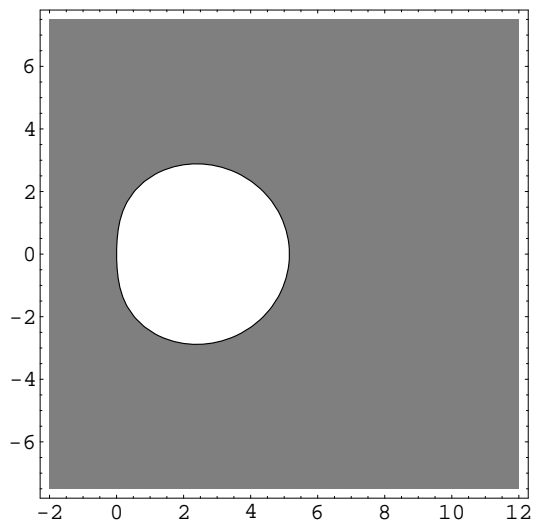
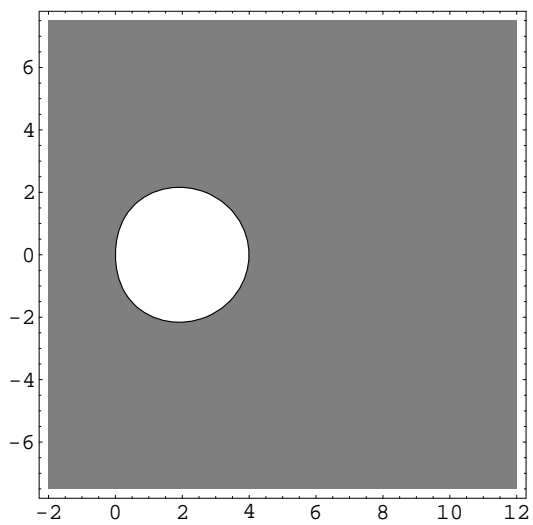
(a) $EF-1-k-4-r$.(b) $EF-2-k-3-r$.(c) $EF-3-k-2-r$.(d) $EF-4-k-1-r$.

Figura F.6: Regiones de estabilidad absoluta (en gris) de los métodos implícitos de cuatro pasos. El parámetro real del método es $\lambda h = -1$. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

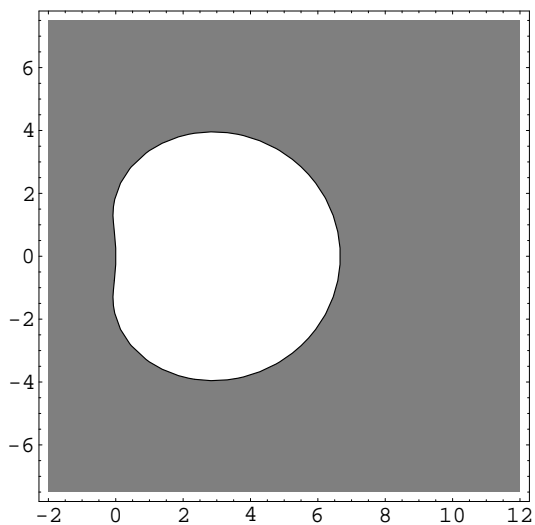
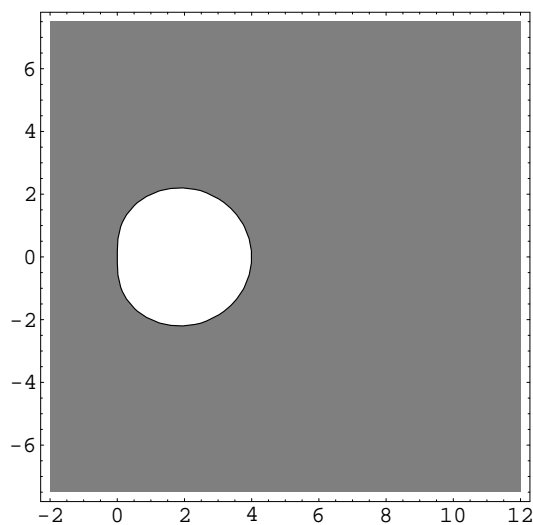
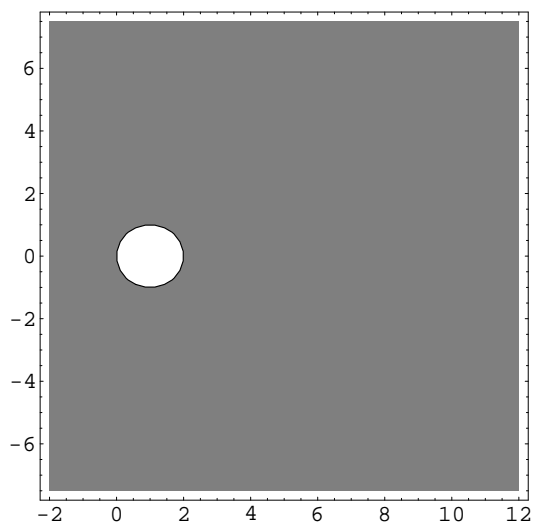
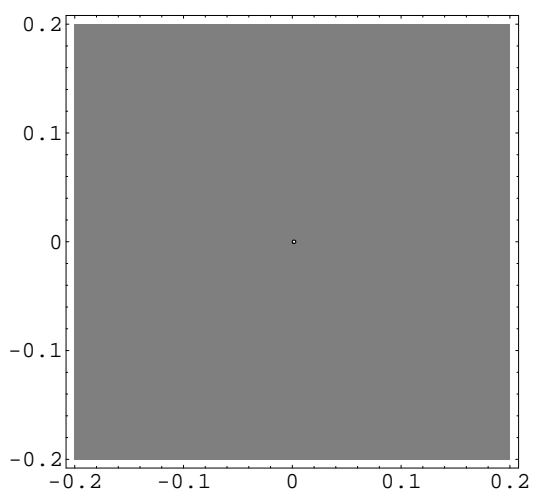
(a) $EF-1-k-4-r$.(b) $EF-2-k-3-r$.(c) $EF-3-k-2-r$.(d) $EF-4-k-1-r$.

Figura F.7: Regiones de estabilidad absoluta (en gris) de los métodos implícitos de cuatro pasos. El parámetro real del método es $\lambda h = -10$. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

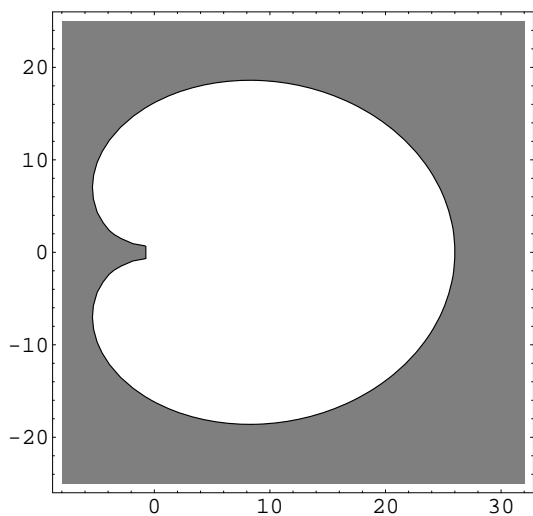
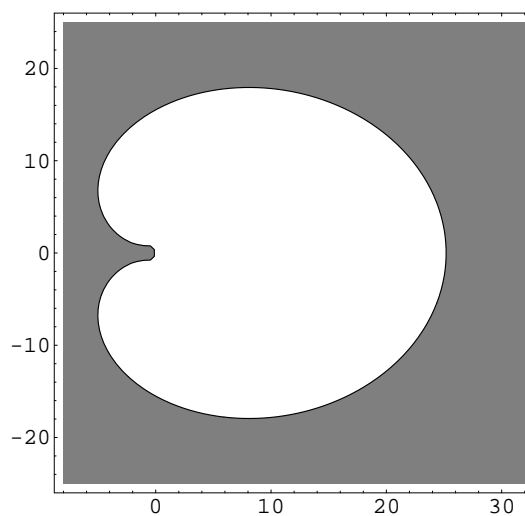
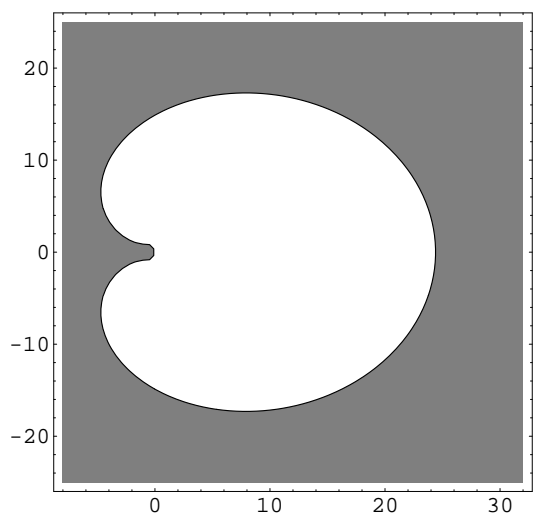
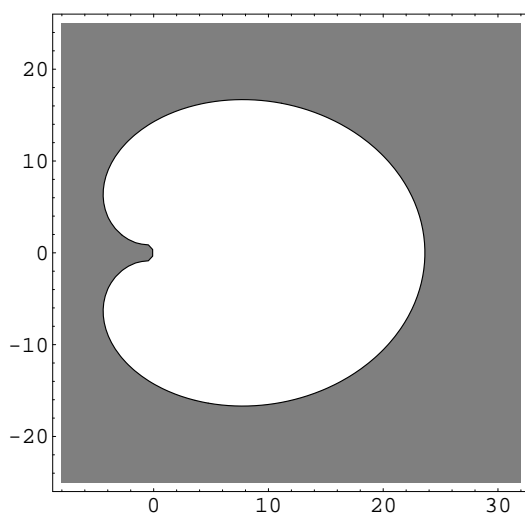
(a) $EF-2-k-5-r$.(b) $EF-3-k-4-r$.(c) $EF-4-k-3-r$.(d) $EF-5-k-2-r$.

Figura F.8: Regiones de estabilidad absoluta (en gris) de los métodos implícitos de seis pasos. El parámetro real del método es $\lambda h = -0,1$. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

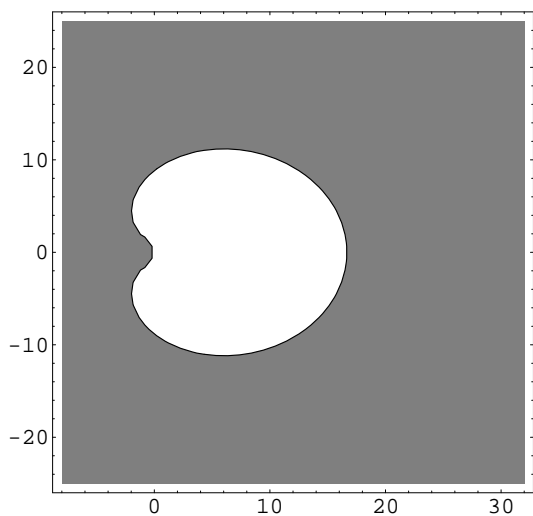
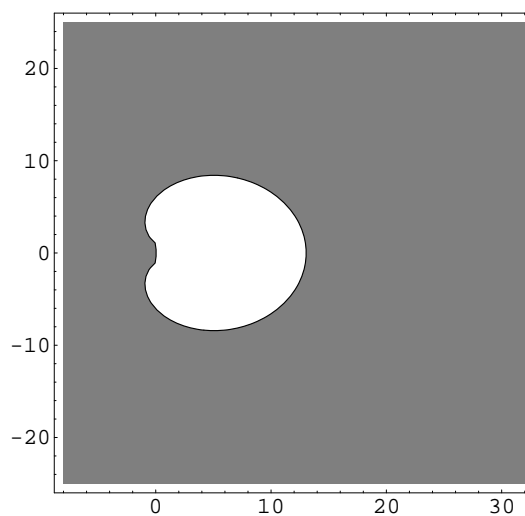
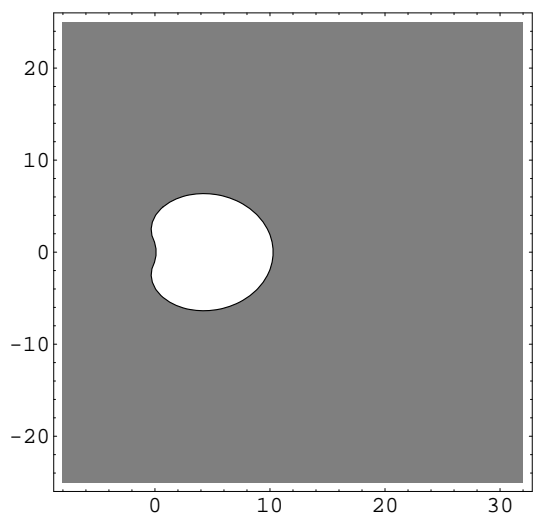
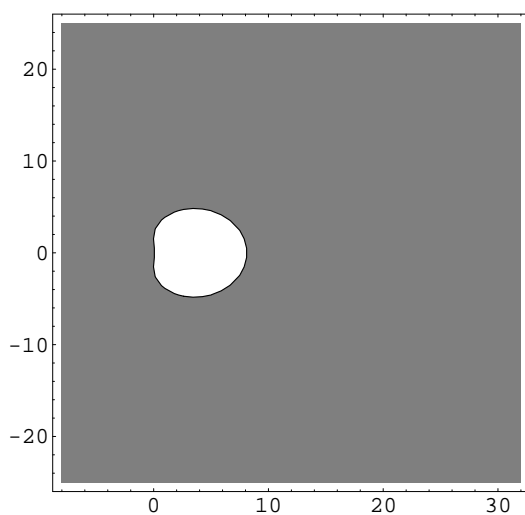
(a) $EF-2-k-5-r$.(b) $EF-3-k-4-r$.(c) $EF-4-k-3-r$.(d) $EF-5-k-2-r$.

Figura F.9: Regiones de estabilidad absoluta (en gris) de los métodos implícitos de seis pasos. El parámetro real del método es $\lambda h = -1$. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

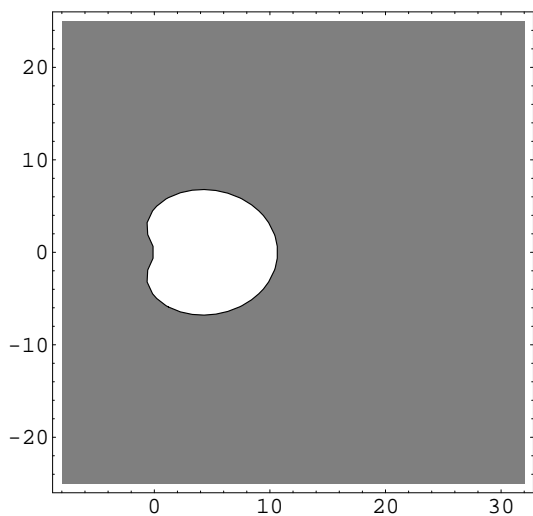
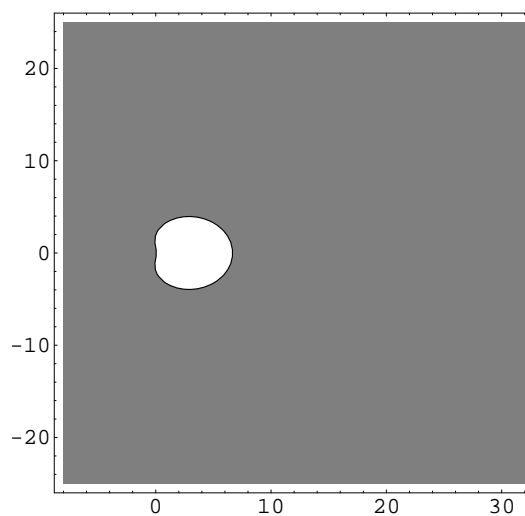
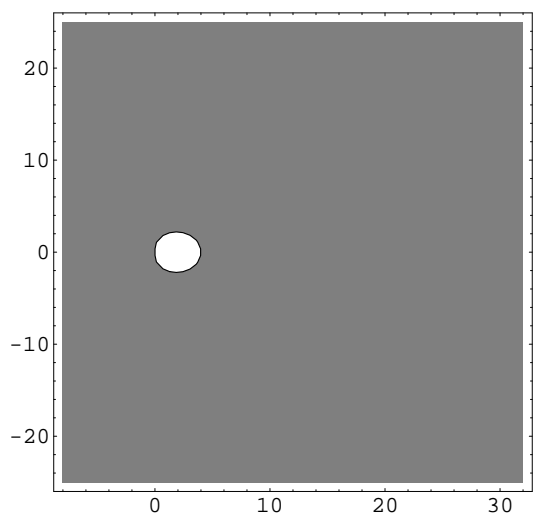
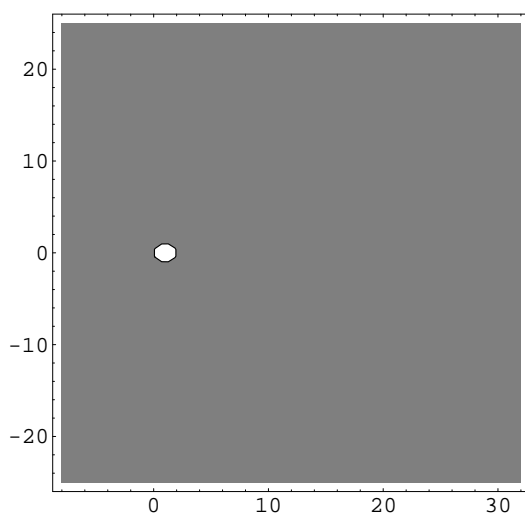
(a) $EF-2-k-5-r$.(b) $EF-3-k-4-r$.(c) $EF-4-k-3-r$.(d) $EF-5-k-2-r$.

Figura F.10: Regiones de estabilidad absoluta (en gris) de los métodos implícitos de seis pasos. El parámetro real del método es $\lambda h = -10$. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

De paso con estas figuras empezamos a observar que efectivamente entre métodos del mismo número de pasos, cuanto mayor es el orden exponencial, el método tiene mejores propiedades de estabilidad. De hecho en la figura F.5 se observa que para un parámetro cercano a 0 ($\lambda h = -0,01$) las regiones de estabilidad son muy parecidas para los 4 métodos a la región del BDF tradicional de 4 pasos (la razón como ya se ha dicho es que los coeficientes de los métodos tradicionales convergen a los del tradicional cuando $\lambda h \rightarrow 0$), sin embargo, según crece $\|\lambda h\|$ (con $\lambda h \in R^-$), las regiones de inestabilidad (la zona en blanco), cambian de forma muy diferente según el método. Mientras para el método EF-1-k-4-r la zona de inestabilidad decrece lentamente, para el método EF-4-k-1-r lo hace mucho más rápido, los otros dos métodos quedan claramente en el medio de ambos. Un comportamiento muy similar tienen los métodos de 6 pasos.

Por último hemos elegido los métodos de cinco pasos EF-2-k-4-r (figura F.11), EF-3-k-3-r (figura F.12) y EF-4-k-2-r (figura F.13) para observar como se repiten las propiedades 3 y 4.

Aquí volvemos a notar que entre dos métodos implícitos del mismo número de pasos, el de mayor orden exponencial tiene mejores propiedades de estabilidad (sobre todo se nota si $\lambda h = a + bi$, con $a \ll -1$) se verifica que para el mismo parámetro del método el de mayor orden exponencial tiene una región de estabilidad absoluta mayor y además crece más rápidamente cuando $\|\lambda h\| \rightarrow \infty$ ($\lambda h \in C^-$).

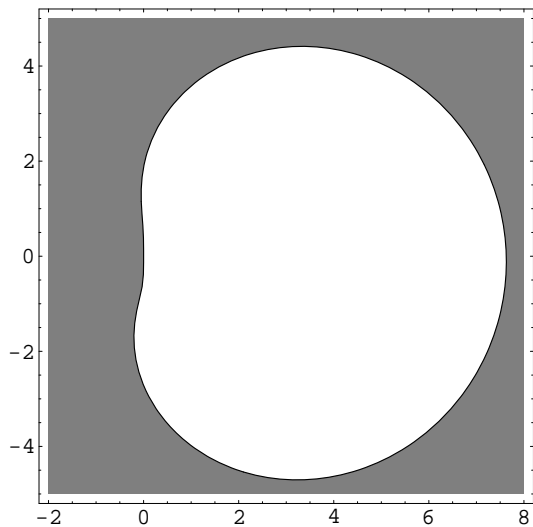
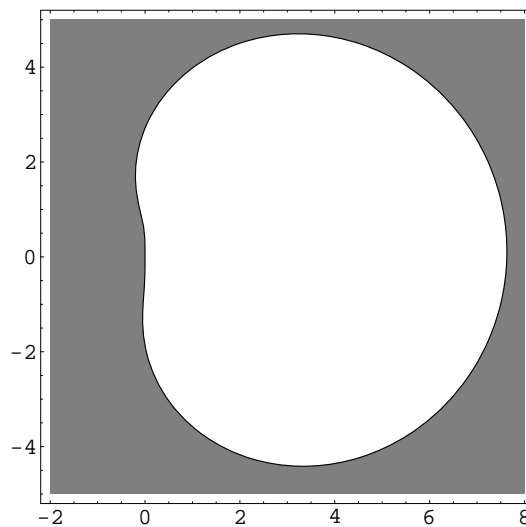
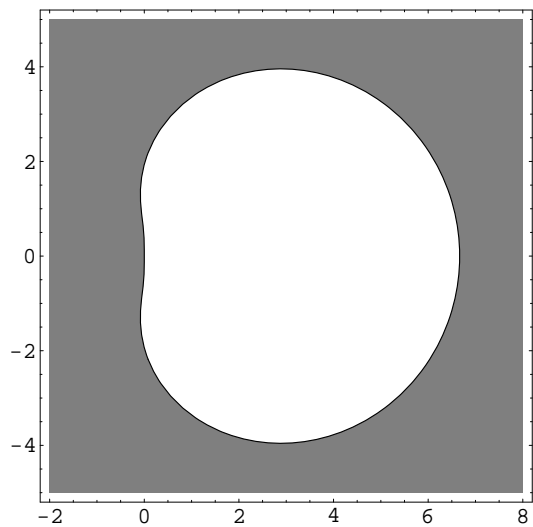
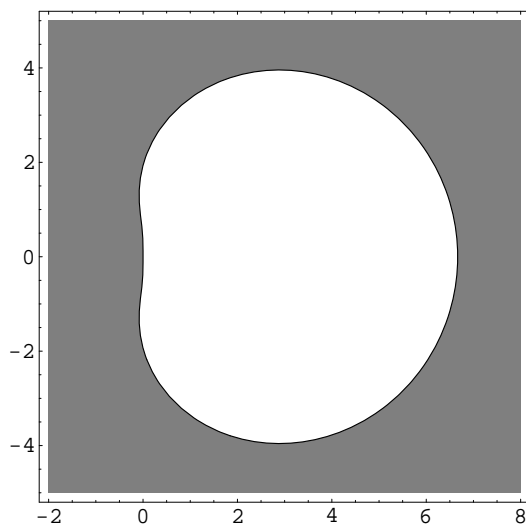
(a) $\lambda h = -2 + 1i$.(b) $\lambda h = -2 - 1i$.(c) $\lambda h = -10 + 5i$.(d) $\lambda h = -10 - 5i$.

Figura F.11: Regiones de estabilidad absoluta (en gris) del método implícito de cinco pasos EF-2-k-4-r. El parámetro del método es imaginario. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

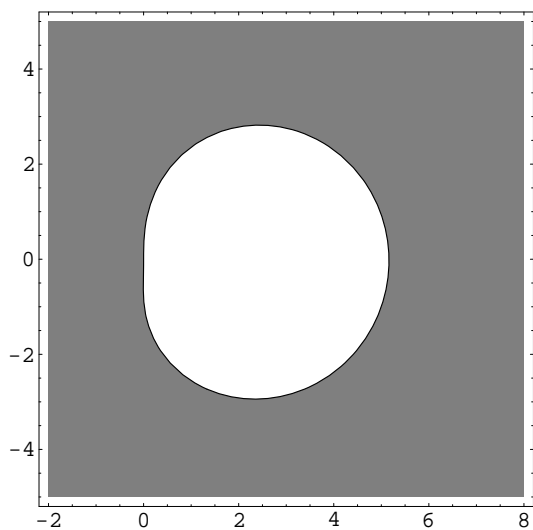
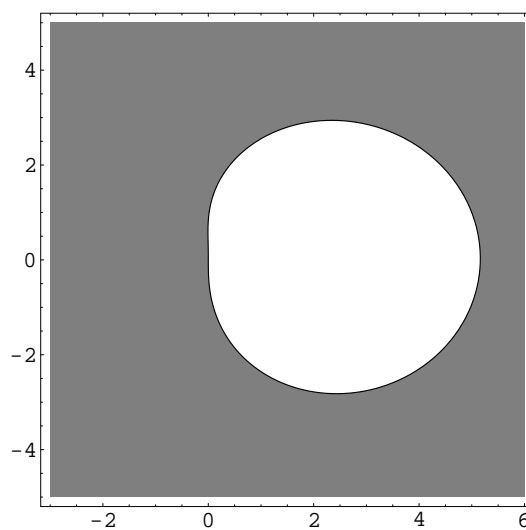
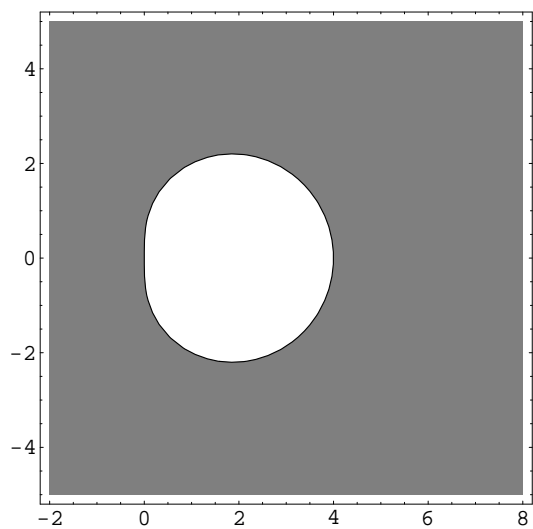
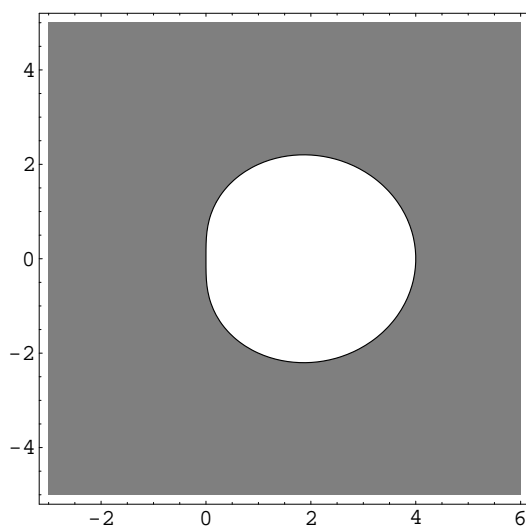
(a) $\lambda h = -2 + 1i$.(b) $\lambda h = -2 - 1i$.(c) $\lambda h = -10 + 5i$.(d) $\lambda h = -10 - 5i$.

Figura F.12: Regiones de estabilidad absoluta (en gris) del método implícito de cinco pasos $EF-3-k-3-r$. El parámetro del método es imaginario. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

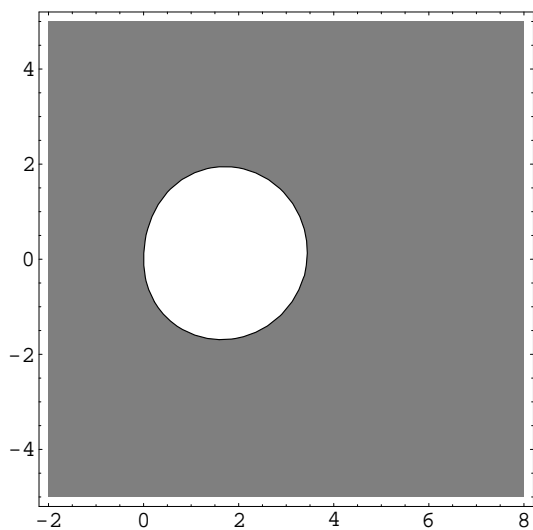
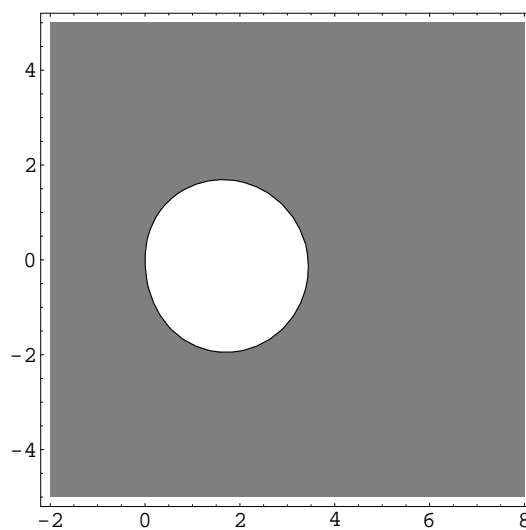
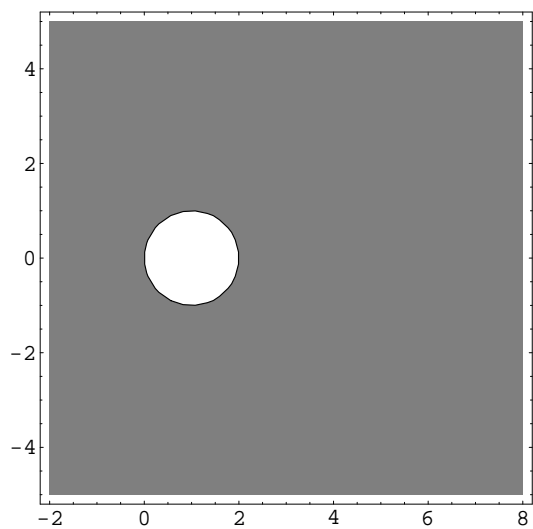
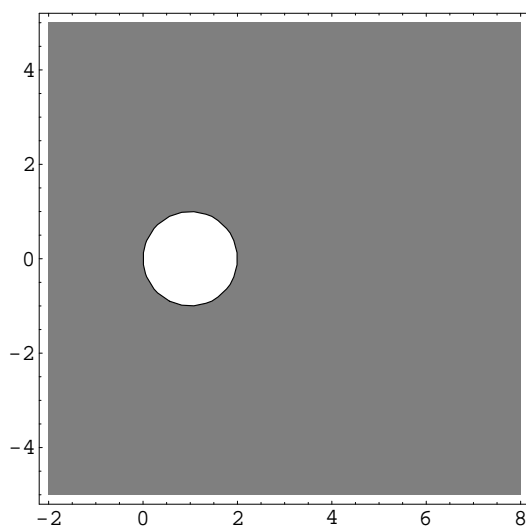
(a) $\lambda h = -2 + 1i$.(b) $\lambda h = -2 - 1i$.(c) $\lambda h = -10 + 5i$.(d) $\lambda h = -10 - 5i$.

Figura F.13: Regiones de estabilidad absoluta (en gris) del método implícito de cinco pasos EF-4-k-2-r. El parámetro del método es imaginario. Los ejes horizontal y vertical representan respectivamente las partes $Re(h(\lambda + \nu))$ e $Im(h(\lambda + \nu))$.

Apéndice G

0-estabilidad de los métodos exponenciales con dos parámetros.

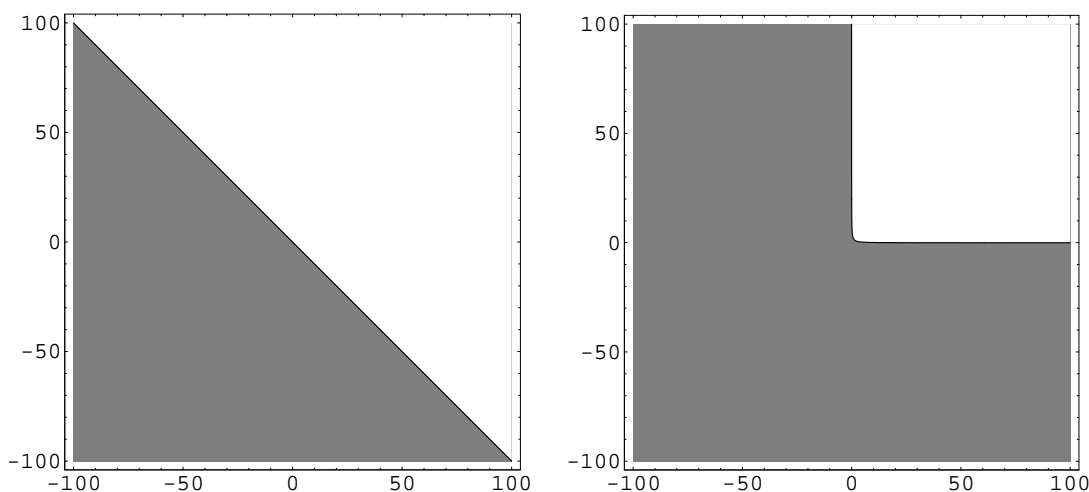
Como ya hemos comentado antes, la estabilidad la estudiaremos del problema lineal (2.5.4), pero como ya se ha comentado nos podemos reducir a estudiar el caso escalar, con $\lambda \in C^-$. De esta forma pedimos que

$$\|y(x)\| \rightarrow 0 \text{ cuando } x \rightarrow \infty,$$

y por tanto que para $\lambda \in C^-$, $\tau \in C$, las raíces r_k del polinomio característico (C.0.1) verifiquen que $|r_k| < 1$, y en aquellas raíces que tengan módulo 1 que sean simples.

Una de las raíces del polinomio característico asociado al método explícito es $r_1 = 1$, mientras que la otra es

$$r_2 = \frac{e^{(\lambda+\tau)h}(\tau(1 - e^{\lambda h}) + \lambda(-1 + e^{\tau h}))}{-\tau e^{\tau h}(1 - e^{\lambda h}) + \lambda e^{\lambda h}(1 - e^{\tau h})}.$$



(a) Método explícito.

(b) Método implícito.

Figura G.1: Regiones de 0-estabilidad (en gris) de los métodos exponenciales con dos parámetros (parámetros reales). Los ejes horizontal y vertical representan λh y τh respectivamente.

Por contra las del método implícito son $r_1^* = 1$ y

$$r_2^* = \frac{e^{(\lambda+\tau)h}(\tau(1 - e^{\lambda h}) + \lambda(-1 + e^{\tau h}))}{-\tau e^{\tau h}(1 - e^{\lambda h}) + \lambda e^{\lambda h}(1 - e^{\tau h})}.$$

Por supuesto tanto la raíz r_2 del método explícito como la raíz r_2^* del implícito dependen de ambos parámetros, λh y τh que son números complejos, por tanto resulta complicado llegar a conclusiones generales para cualquier valor de los parámetros.

En primer lugar y a modo de orientación en la figura G.1 hemos dibujado las regiones de 0-estabilidad de ambos métodos en el caso escalar, es decir si $\lambda h, \tau h \in \mathbb{R}$.

Ya podemos observar que el método explícito será 0-estable, por ejemplo, si $\lambda h \in \mathbb{R}^-$ y $\tau \leq \|\lambda\|$ (si $\tau = -\lambda$, entonces $r_2 = -1$). Por contra el

método implícito es 0-estable siempre que uno de los parámetros λh ó τh sean negativos (y reales).

En el caso de que queramos considerar que al menos uno de los dos es complejo el estudio se complica claramente, en este caso podemos elegir por ejemplo entre fijar el valor τh o fijar la relación entre τ y λ .

Empecemos mostrando unas regiones de 0-estabilidad del método explícito: en primer lugar en la figura G.2 hemos comparado las regiones de 0-estabilidad para parámetros $\tau \in C^-$ y los mismos valores τ pero con signo contrario. Podemos observar que hay una relación entre ambas figuras, dada la que tiene como parámetro $\tau \in C^-$, la correspondiente figura con parámetro $-\tau \in C^+$ es simétrica respecto al eje que representa $Im(\lambda h)$, pero cambiando la zona estable por la inestable. Por eso en la figura G.3 nos restringimos a fijar diferentes valores $\tau h \in C^-$ y pintar las regiones de 0-estabilidad en función del valor de λh .

Por el contrario en la figura G.4 hemos fijado diferentes relaciones $\tau = t\lambda$ (con $t \in [-1, 1]$, ya que por ejemplo la región de 0-estabilidad de la relación $t = 2$, es idéntica a la de la relación $\frac{1}{2}$, salvo que las dimensiones en el dibujo son la mitad) y pintamos las regiones de 0-estabilidad en función del valor de λh .

La primera conclusión que se puede obtener es que elegir valores $\tau \in R^+$ va a ser peligroso y habrá que ser muy cuidadoso en esos casos. Es más en el próximo apéndice de estabilidad absoluta veremos que no aporta beneficios en cuanto a la estabilidad y sí perjuicios.

También se puede observar ya que cuánto más negativa es la parte real del parámetro τh , mejor parece que va a ser la estabilidad, esto es algo que se

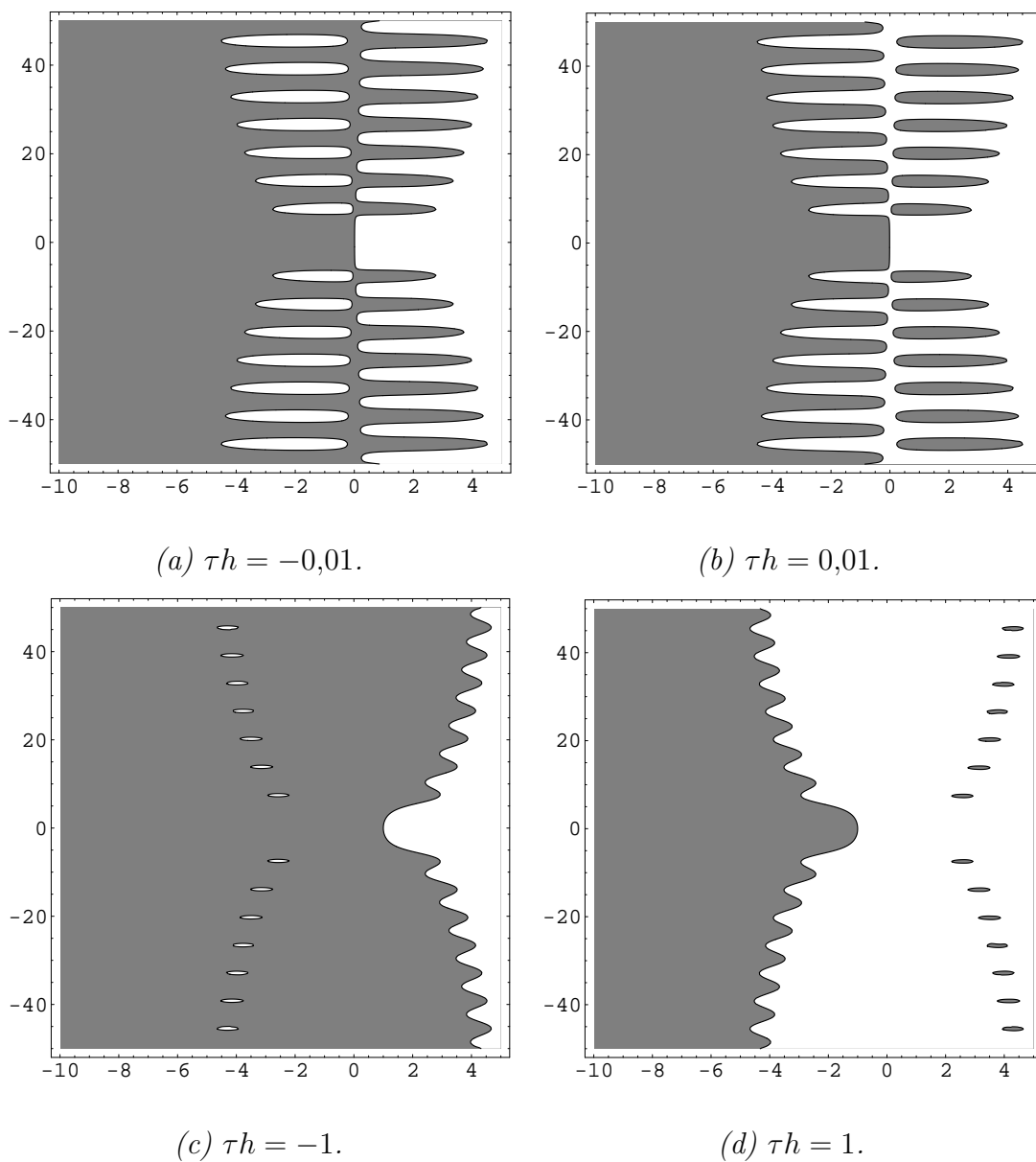


Figura G.2: Regiones de 0-estabilidad (en gris) del método explícito exponencial con dos parámetros, para ciertos valores de τh . Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

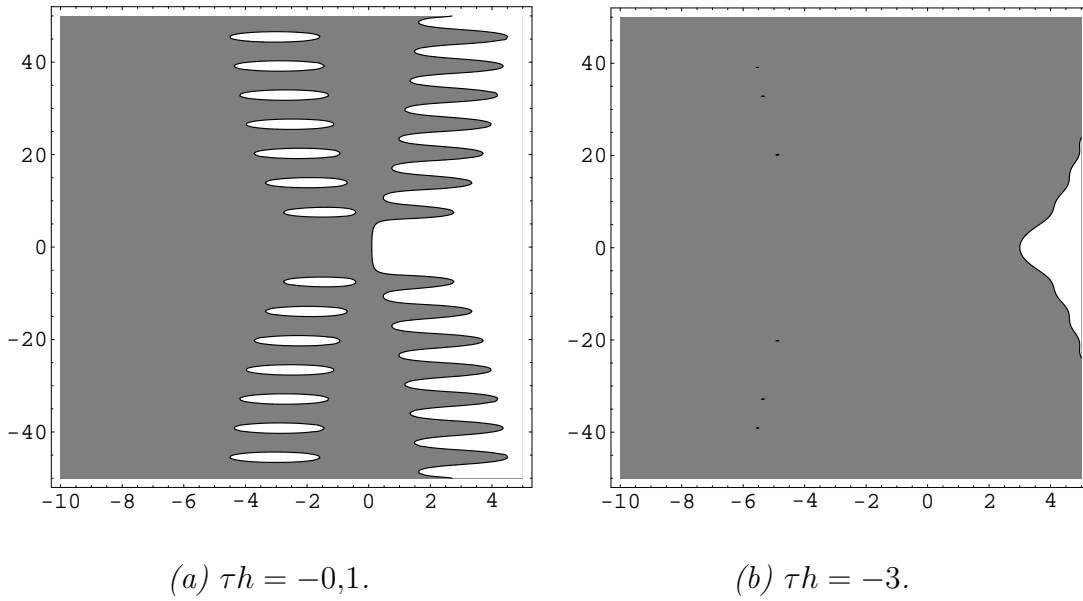


Figura G.3: Regiones de 0-estabilidad (en gris) del método explícito exponencial con dos parámetros, para ciertos valores de τh . Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

observa mejor en el apéndice de estabilidad absoluta.

Para el estudio de la 0-estabilidad del método implícito hemos actuado de forma similar a la del caso explícito: en primer lugar en la figura G.5 hemos fijado diferentes valores de $\tau h \in R$ y pintamos las regiones de 0-estabilidad en función del valor de λh , mientras que en la figura G.6 hemos fijado diferentes relaciones $\tau = t\lambda$ (con $t \in [-1, 1]$) y pintamos las regiones de 0-estabilidad en función del valor de λh .

Aquí se observa que las regiones de 0-estabilidad no verifican la misma simetría que en el caso explícito.

Pero si se vuelve a observar que cuanto más pequeña es la parte real del parámetro τh mejores son las propiedades de estabilidad.

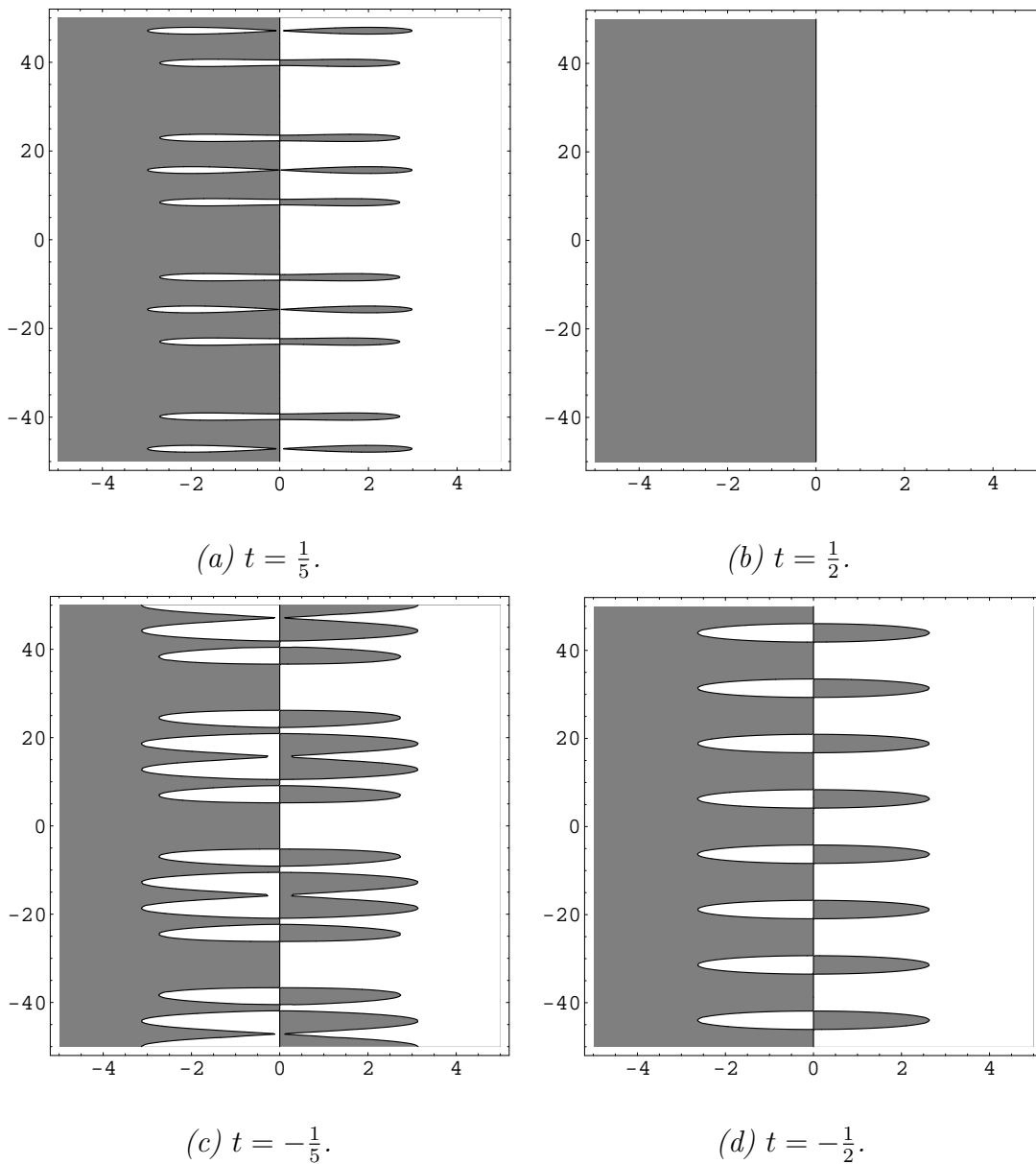


Figura G.4: Regiones de 0-estabilidad (en gris) del método explícito exponencial con dos parámetros, para ciertos valores de $t = \frac{\tau}{\lambda}$. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

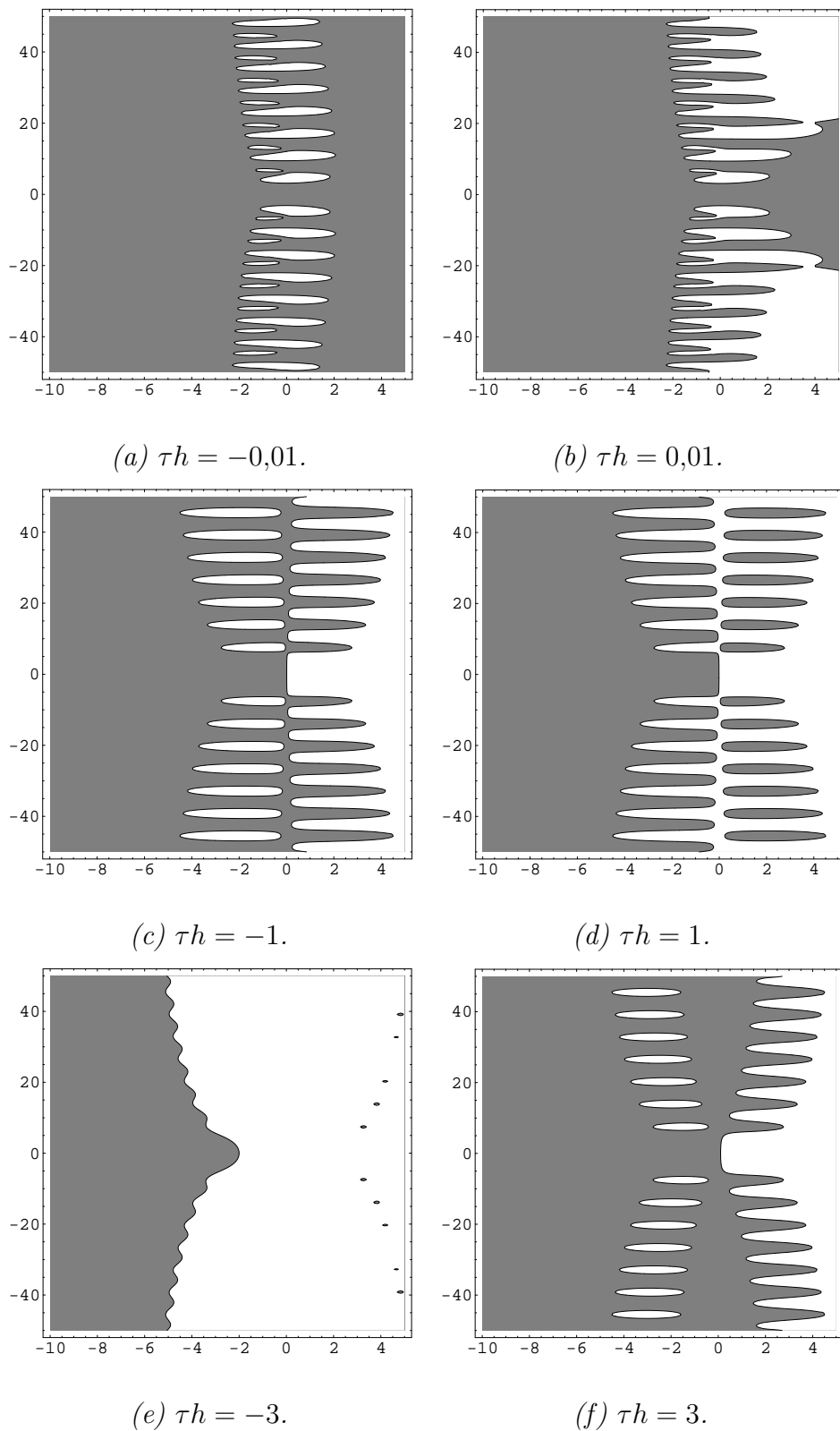


Figura G.5: Regiones de 0-estabilidad (en gris) del método implícito exponencial con dos parámetros, para ciertos valores de τh . Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

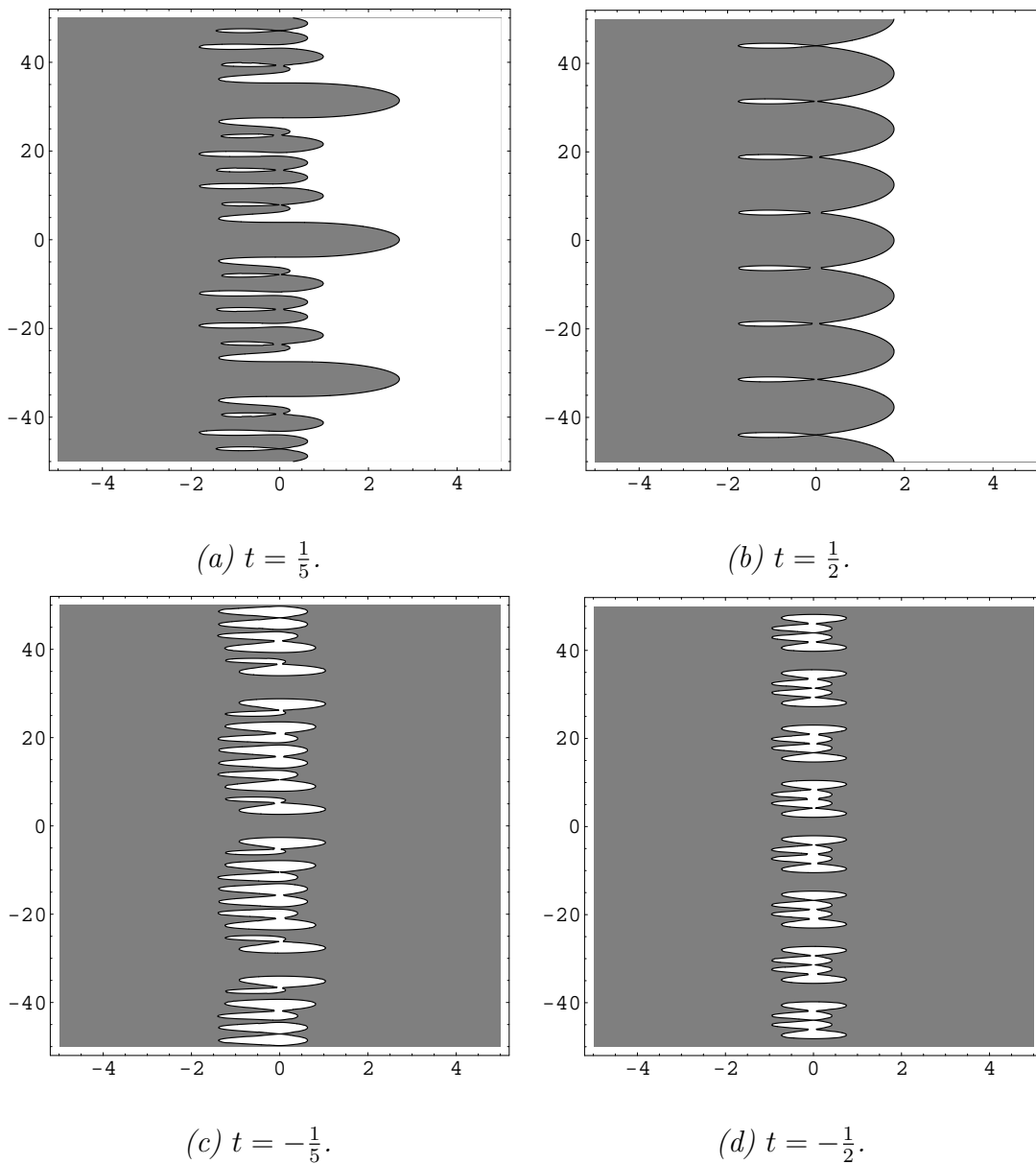


Figura G.6: Regiones de 0-estabilidad (en gris) del método implícito exponencial con dos parámetros, para ciertos valores de $t = \frac{\tau}{\lambda}$. Los ejes horizontal y vertical representan $Re(\lambda h)$ e $Im(\lambda h)$ respectivamente.

Apéndice H

Estabilidad absoluta de los métodos exponenciales con dos parámetros.

Como ya se ha comentado para estudiar la estabilidad absoluta vamos a estudiar el problema (6.4.2), por lo que las regiones de estabilidad van a depender de tres parámetros, $\lambda h \in C$, $\tau h \in C$ y $\mu h \in C$, por lo que habrá que fijar dos de estos parámetros (λh y τh) y las regiones de estabilidad absoluta dependerán del valor de μh . Por supuesto no vamos a poder dibujar más que un número finito de regiones de estabilidad y las observaciones no serán concluyentes.

Para tener una idea previa nos pueden ayudar los siguientes dos teoremas:

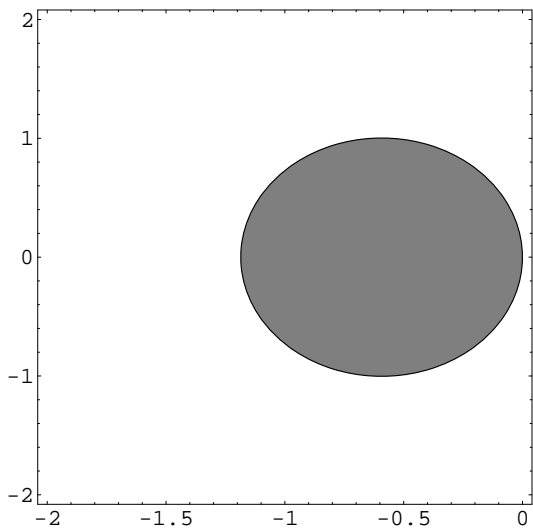
Teorema H.0.10. *Si $\tau h \rightarrow 0$, los coeficientes de los métodos exponenciales con dos parámetros tienden a los coeficientes de los métodos EF-I-k-2.*

Teorema H.0.11. *Si $\tau h \rightarrow \lambda h$, los coeficientes de los métodos exponenciales con dos parámetros tienden a los coeficientes de los métodos EF-I-r-2.*

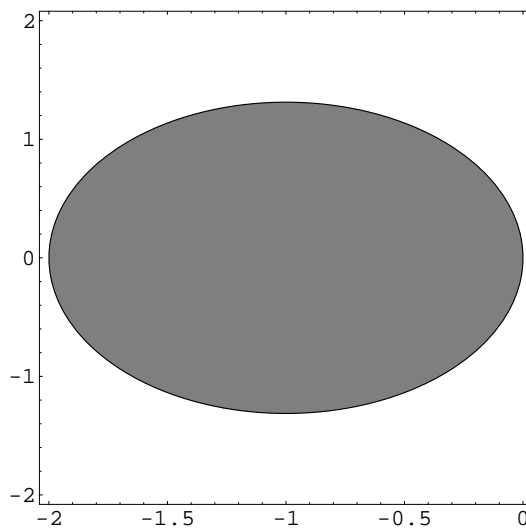
Esto ya nos induce a pensar que los métodos exponenciales con dos parámetros van a cumplir las propiedades 1, 2, 3 y 4 enunciadas en el apéndice dedicado a los métodos *exponential fitting* ó adaptados de orden exponencial y algebraico mayor que uno, y que además las regiones de estabilidad de los métodos serán mayores cuando $\tau h \rightarrow \lambda h \in C^-$, que cuando $\tau h \rightarrow 0$. Hecho que se puede comprobar empíricamente, además ya sabemos (sobre todo en el caso explícito) que las regiones de estabilidad de los métodos serán mayores cuando $\tau h \rightarrow 0$ que cuando $\tau h \rightarrow -\lambda h \in C^+$, y en general se observa que si $\tau h = t\lambda h$ (donde $\lambda h \in C^-$), cuanto mayor sea t , más grande es la región de estabilidad absoluta.

También se verifica que si $\lambda h = v_1$ y $\tau h = v_2$ los coeficientes de los métodos son los mismos que si $\lambda h = v_2$ y $\tau h = v_1$, y por tanto las regiones de estabilidad son las mismas.

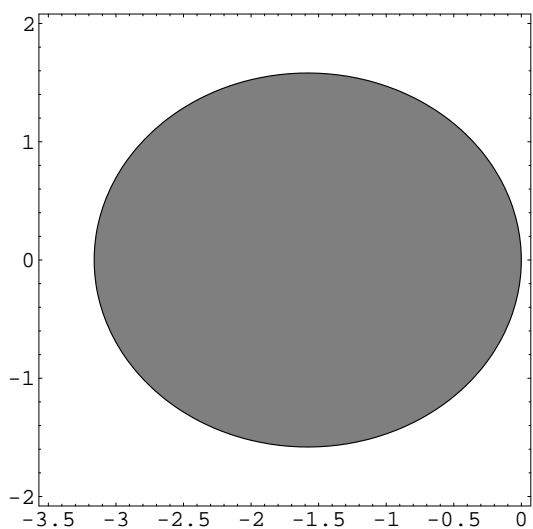
Mostramos una serie de regiones de estabilidad absoluta a modo de ejemplos en las figuras H.1 y H.2 para el método explícito exponencial con dos parámetros y en las figuras H.3 y H.4 para el implícito.



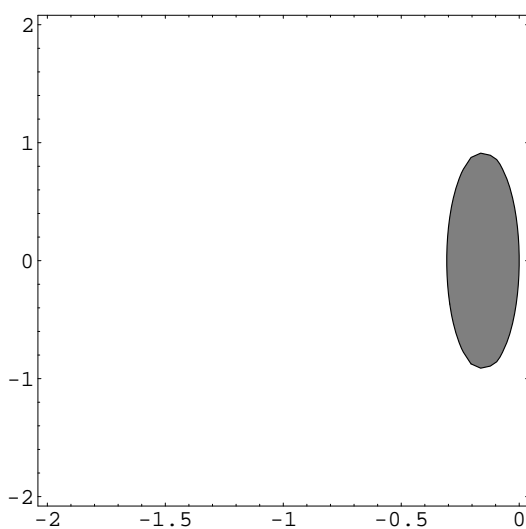
(a) $\lambda = -0,01, \tau = -2.$



(b) $\lambda = -1, \tau = -2.$

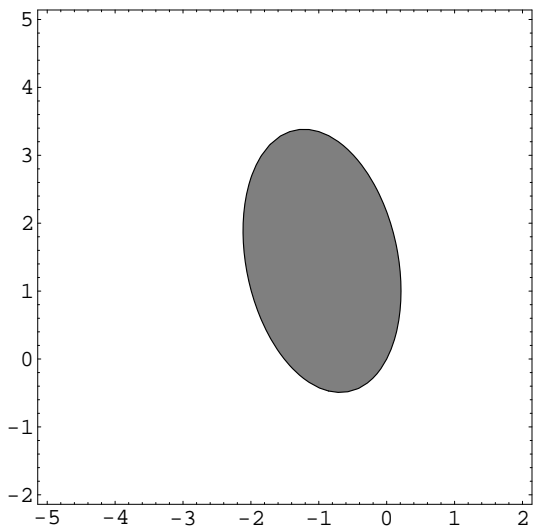


(c) $\lambda = -1, \tau = -10.$

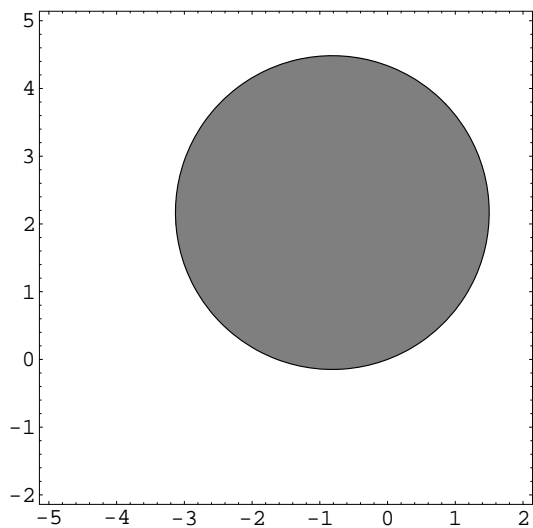


(d) $\lambda = -1, \tau = 0,5$ (valor positivo).

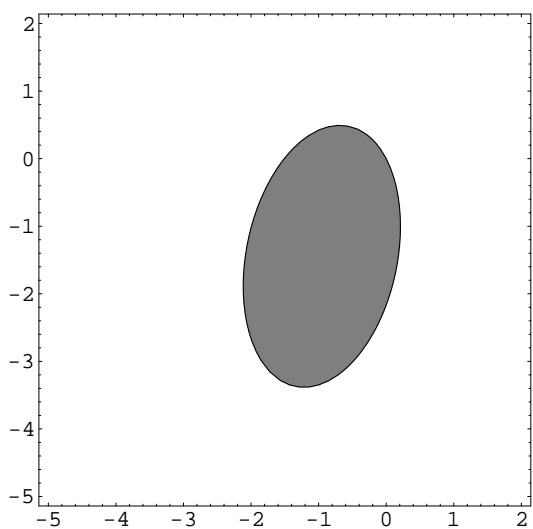
Figura H.1: Regiones de estabilidad absoluta (en gris) del método explícito exponencial con dos parámetros, para ciertos valores de λ y τ reales. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente.



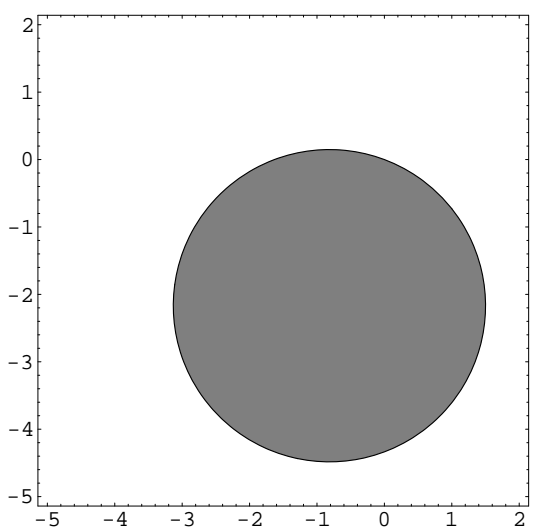
(a) $\lambda = -1 + 3i, \tau = -1.$



(b) $\lambda = -1 + 3i, \tau = -50.$

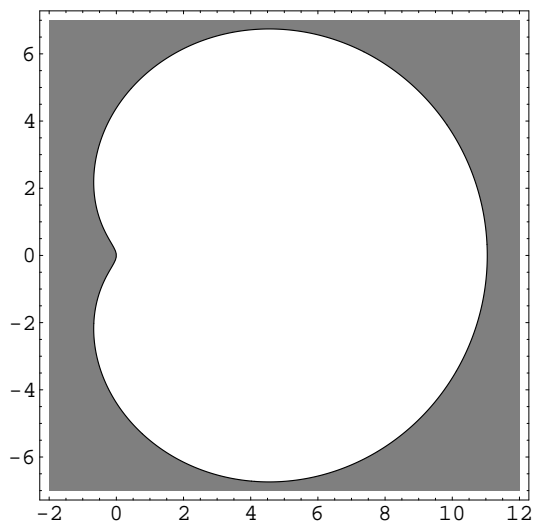
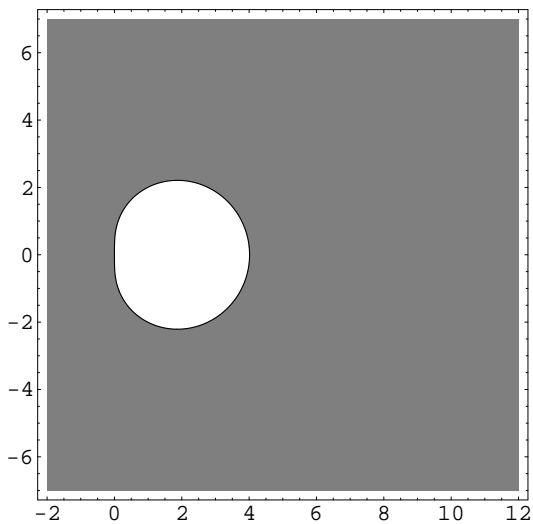


(c) $\lambda = -1 - 3i, \tau = -1.$

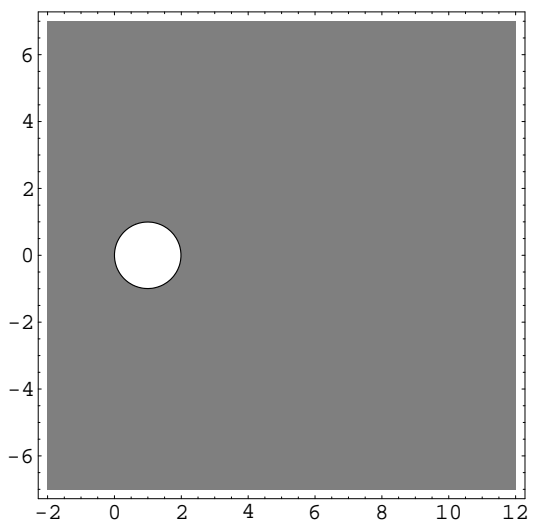
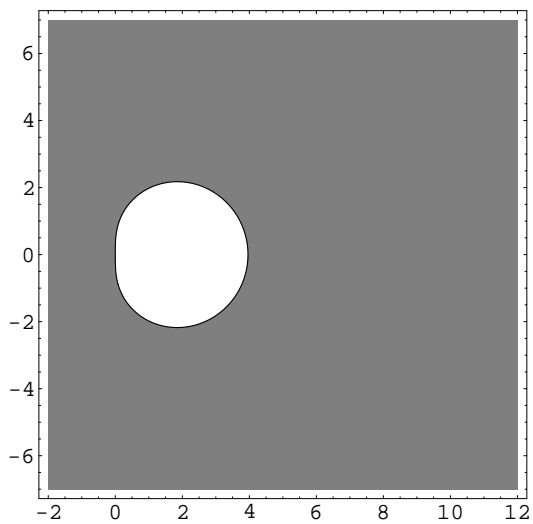


(d) $\lambda = -1 - 3i, \tau = -50.$

Figura H.2: Regiones de estabilidad absoluta (en gris) del método explícito exponencial con dos parámetros, para ciertos valores de λ y τ complejos. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente.



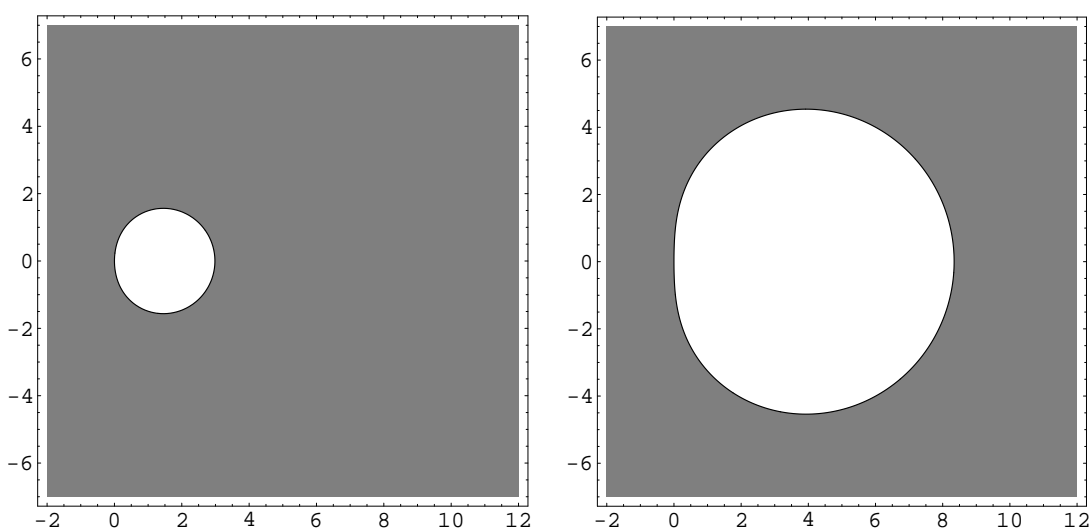
(a) $\lambda = -0,01$, $\tau = 0,02$ (valor positivo). (b) $\lambda = -0,01$, $\tau = 3$ (valor positivo).



(c) $\lambda = -0,01$, $\tau = -0,02$.

(d) $\lambda = -0,01$, $\tau = -10$.

Figura H.3: Regiones de estabilidad absoluta (en gris) del método implícito exponencial con dos parámetros, para ciertos valores de λ y τ reales. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente.



(a) $\lambda = -1$, $\tau = 0,01$ (valor positivo). (b) $\lambda = -1$, $\tau = 3$ (valor positivo).

Figura H.4: Regiones de estabilidad absoluta (en gris) del método implícito exponencial con dos parámetros, para ciertos valores de λ y τ reales. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente.

Apéndice I

Estabilidad absoluta de los métodos BDF-Runge-Kutta.

Una vez calculadas las funciones de estabilidad de los métodos basta con ver para que valores de $u = \lambda h$ y $z = h\mu$ se verifica que $|R(z)| \leq 1$.

Comenzamos dando las regiones de estabilidad absoluta del método BDF-Runge-Kutta cuando $s = 2$. En primer lugar vamos a mostrar las regiones de estabilidad absoluta del método BDF-Runge-Kutta cuando $\lambda h \in R^-$.

La primera conclusión que observamos en el caso del BDF-Runge-Kutta es que la región de inestabilidad disminuye si $\lambda h \rightarrow -\infty$, sin embargo si cuando $\lambda h \rightarrow 0$ el método es A-estable, para valores más pequeños el método deja de ser A-estable como se observa en la figura I.1.

Las regiones de estabilidad absoluta del método tienen un comportamiento similar si $\lambda h \in C^-$ como se puede observar en la figura I.2.

De todas formas tenemos que tener en cuenta que si introducimos el parámetro $\lambda h = -3$ no es muy preocupante que haya problemas de estabilidad si la solu-

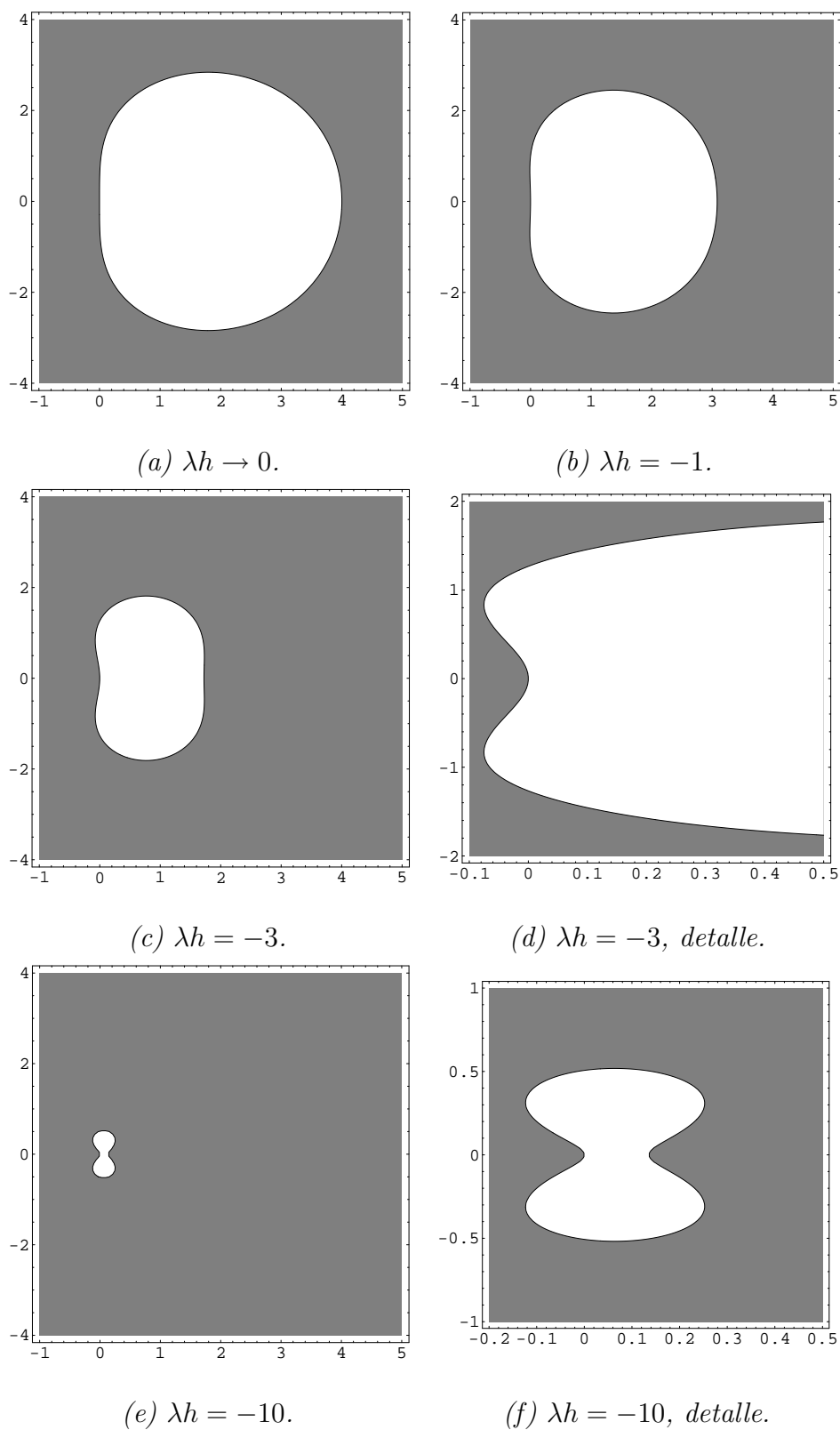


Figura I.1: Regiones de estabilidad absoluta (en gris) del método BDF-RK con $s = 2$. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente.

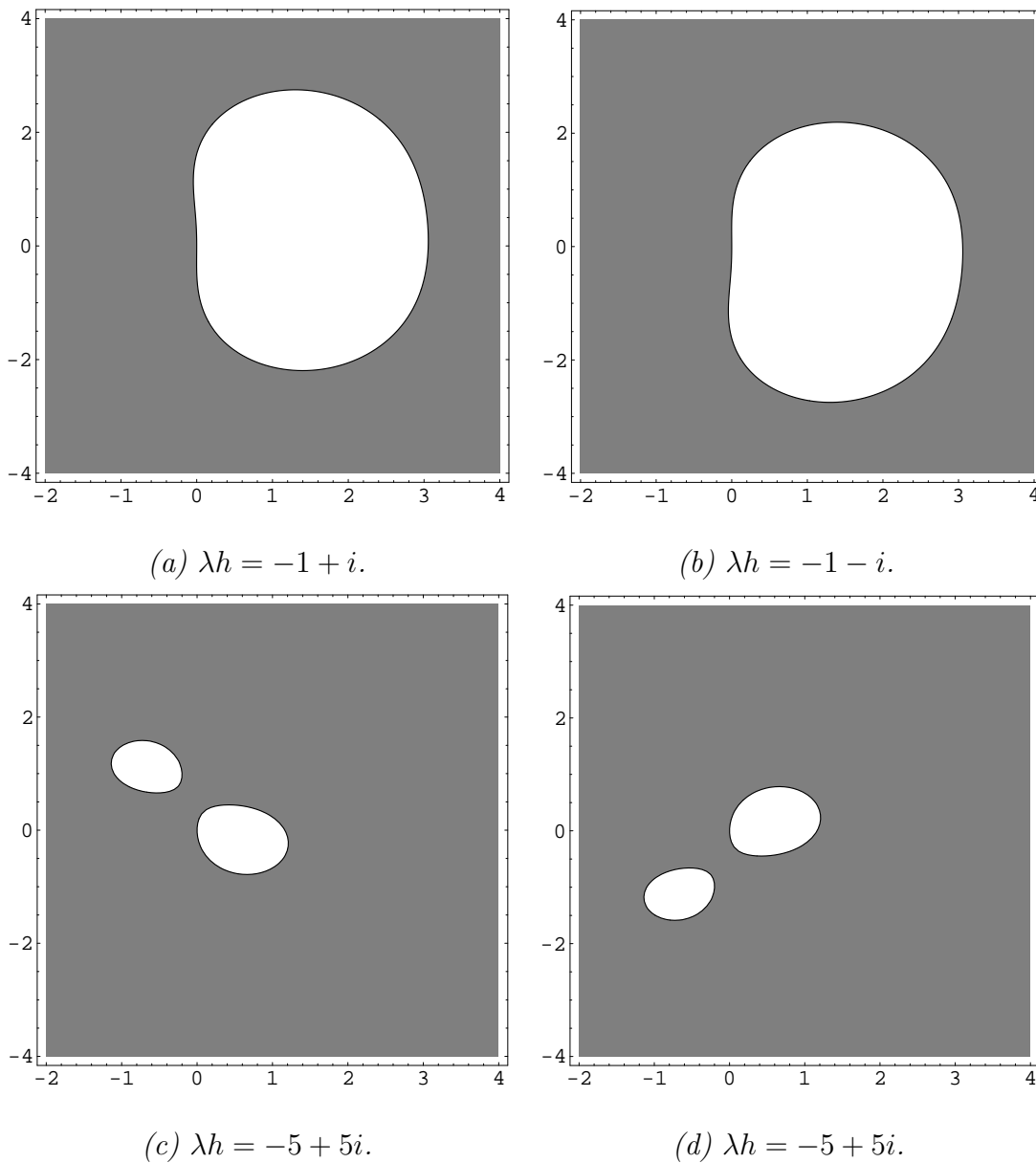


Figura I.2: Regiones de estabilidad absoluta (en gris) del método BDF-RK con $s = 2$. El parámetro del método es imaginario. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente.

ción depende de la exponencial de $\mu h = -0,05 + i$. El error al introducir el parámetro λh ha de ser considerable para que el método tenga un comportamiento de inestabilidad con el problema elegido.

Podemos observar un comportamiento muy similar con el método BDF-RK con $s = 3$ como se observa en la figura I.3 (para valores $\lambda h \in R^-$) y en la figura I.4 (para $\lambda h \in C^-$).

Por último mostramos las regiones de estabilidad absoluta del método BDF-RK con $s = 4$ en las figuras I.5 (para valores $\lambda h \in R^-$) e I.6 (para $\lambda h \in C^-$).

Aquí se puede observar una situación nueva, y es que si los métodos BDF-RK para $s = 2$ y $s = 3$ eran A-estables cuando $\lambda h \rightarrow 0$, en el caso $s = 4$ no sucede lo mismo. Sin embargo, sí que es $A(\alpha)$ -estable (cuando $\lambda h \rightarrow 0$) para $\alpha = 89,7473^\circ$, es decir el conjunto

$$\{z \in C : | \arg(-z) | < \alpha, z \neq 0\}$$

está contenido en le dominio de estabilidad.

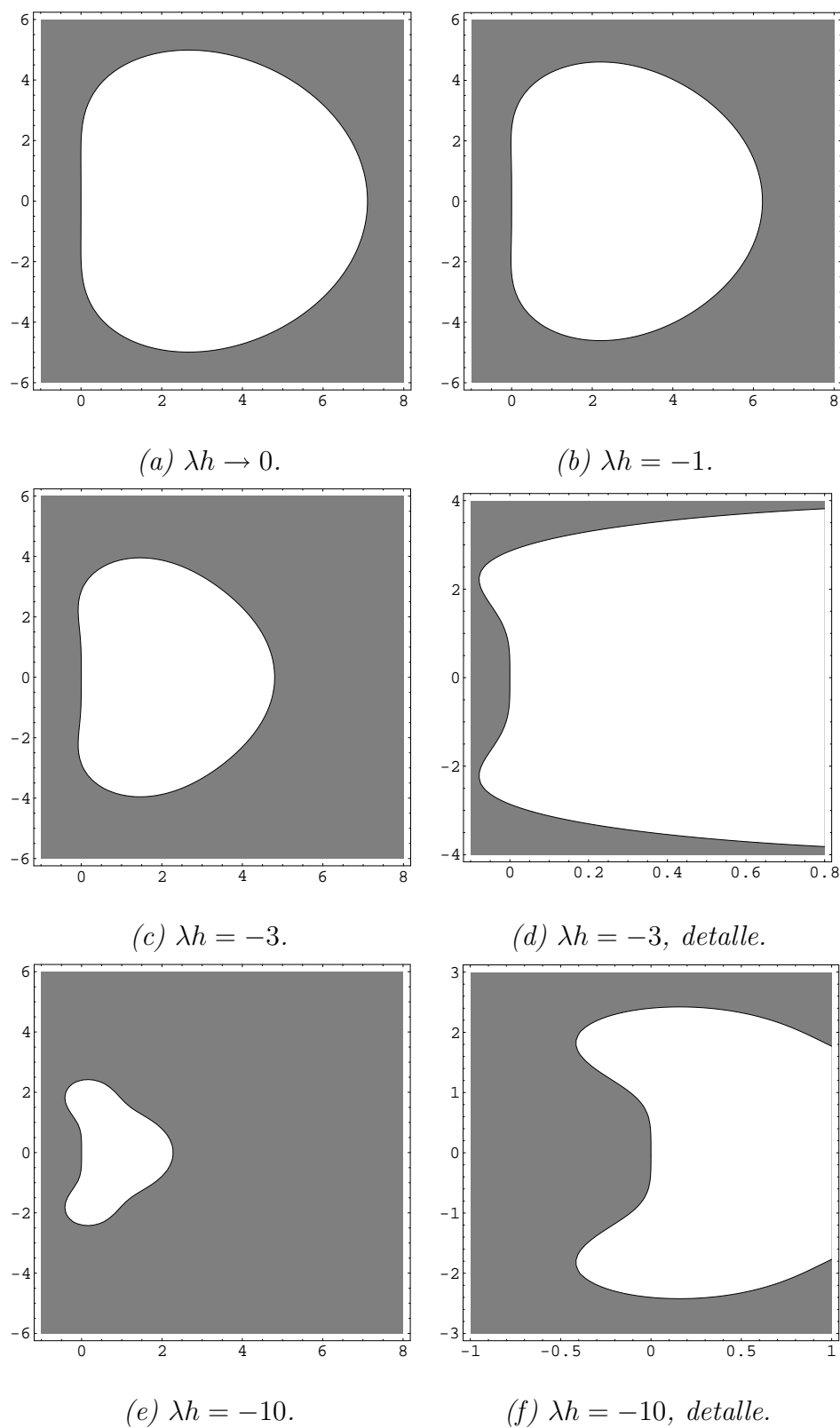


Figura I.3: Regiones de estabilidad absoluta (en gris) del método BDF-RK con $s = 3$. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente.

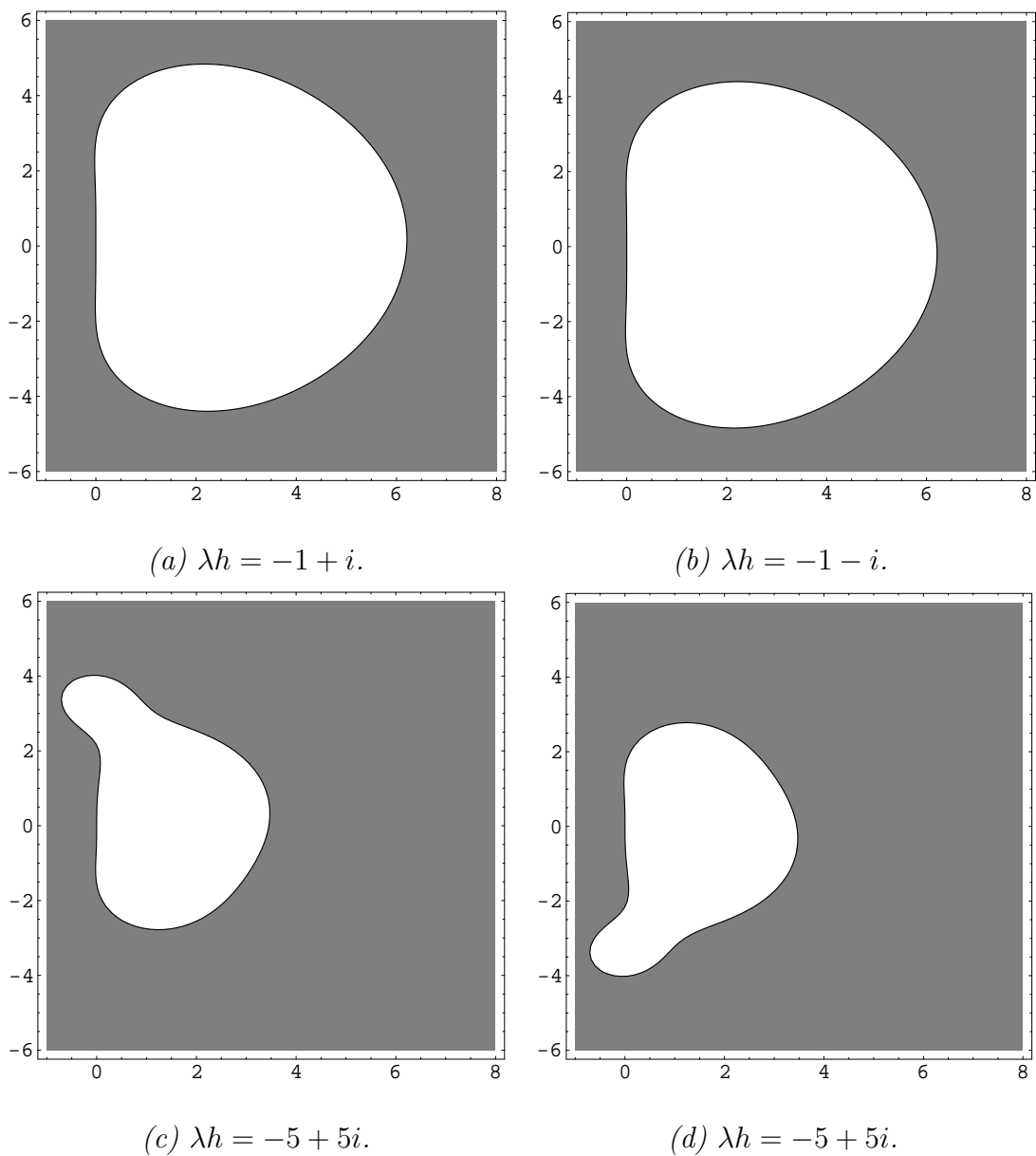
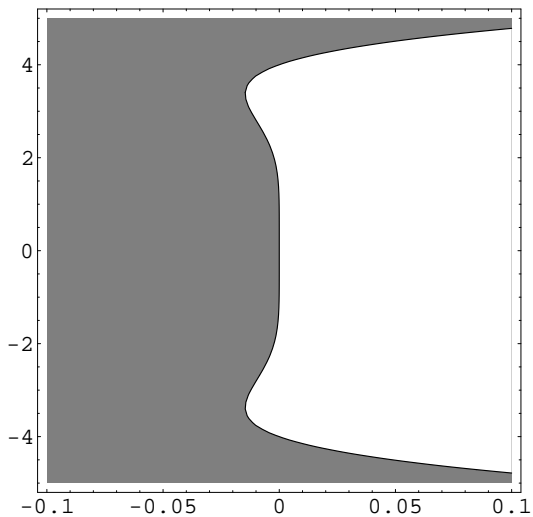
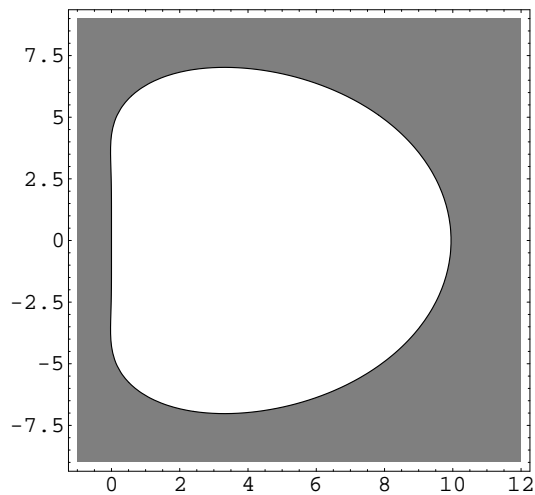


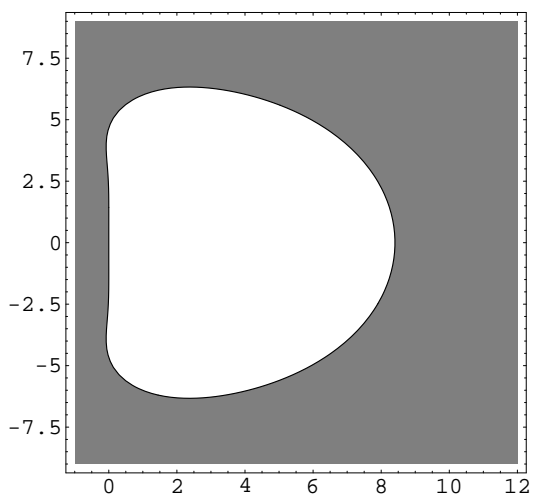
Figura I.4: Regiones de estabilidad absoluta (en gris) del método BDF-RK con $s = 3$. El parámetro del método es imaginario. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente.



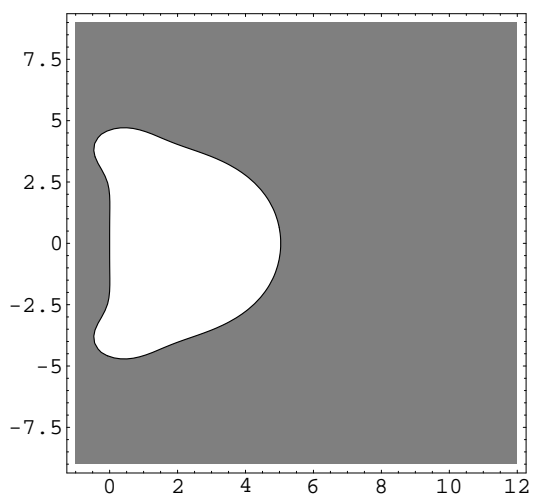
(a) $\lambda h \rightarrow 0$, detalle.



(b) $\lambda h = -1$.



(c) $\lambda h = -3$.



(d) $\lambda h = -10$.

Figura I.5: Regiones de estabilidad absoluta (en gris) del método BDF-RK con $s = 4$. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente.

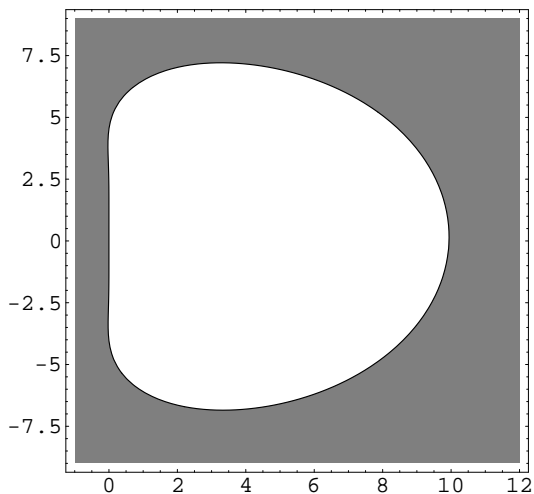
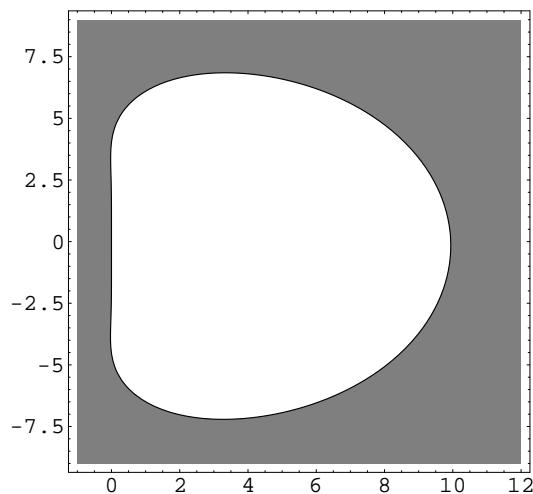
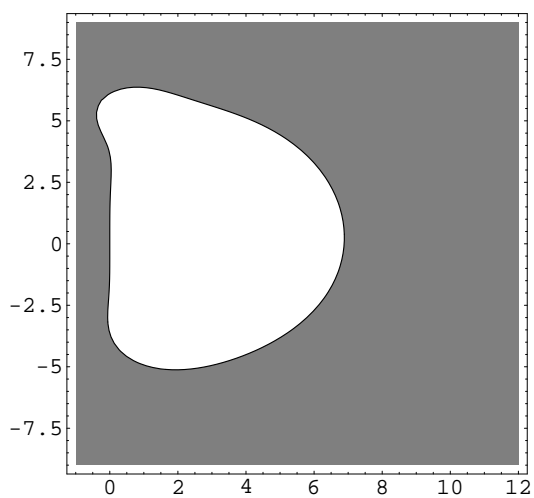
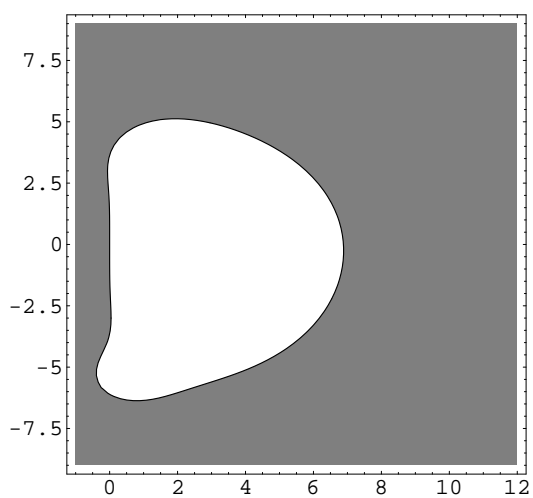
(a) $\lambda h = -1 + i$.(b) $\lambda h = -1 - i$.(c) $\lambda h = -5 + 5i$.(d) $\lambda h = -5 - 5i$.

Figura I.6: Regiones de estabilidad absoluta (en gris) del método BDF-RK con $s = 4$. El parámetro del método es imaginario. Los ejes horizontal y vertical representan $Re(\mu h)$ e $Im(\mu h)$ respectivamente.

Bibliografía

- [1] Aiken, R.C. (1985), *Stiff computation*, Oxford University Press, New York.
- [2] Alexander, R. (1977), *Diagonally implicit Runge-Kutta methods for stiff ODEs*, SIAM J. Numer. Anal. 14, pp. 1006-1021.
- [3] Álvarez, J. (2002), *Métodos GRK para ecuaciones diferenciales ordinarias*, Tesis doctoral, Universidad de Valladolid.
- [4] Álvarez, J. y Rojo, J. (2002), *An improved class of generalized Runge-Kutta methods for stiff problems. Part I: the scalar case*, Appl. Math. and Comp., v. 130, i. 2-3, pp. 537-560.
- [5] Ang, W. (2003), *The two-dimensional reaction-diffusion Brusselator system: a dual-reciprocity boundary element solution*, Engineering Analysis with Boundary Elements 27, pp. 897-903.
- [6] Axelsson, O. y Golobov, S.V. (2001), *Stability and error estimates for the θ -method for strongly monotone and infinitely stiff evolution equations*, Numer. Math. 89, pp. 31-48.
- [7] Barker, T., Bowles, R. y Williams, W. (1973), *Development and application of a local linearization algorithm for integration of quaternion rate equations in real-time flight simulation problems*, NASA TN-D-7347.

- [8] Barton, D. (1980), *On Taylor series and stiff equations*, ACM Trans. Math. Software 6, pp. 280-294.
- [9] Bogoliubov, N.N. y Mitropolsky, Y.A. (1985), *Asymptotic methods in the theory of non-linear oscillations*, Gordon and Breach, Science Publishers, Inc. Glasgow.
- [10] Brock, P. y Murray, F.J. (1952), *The use of exponential sums in step by step integration*, M. T. A. C. 6, pp. 63-78 y 138-150.
- [11] Brydon, D. y Hindmarsh, A. (1987), *Stiff ODE solvers: A review of current and coming attractions*, J. of Comput. Physics 70, pp. 1-62.
- [12] Brydon, D., Pearson, J. y Marder, M. (1998), *Solving stiff differential equations with the method of patches*, J. of Comput. Physics 144, pp. 280-298.
- [13] Bujosa, A., Criado, R. y Vega, C. (1998), *Jordan normal form via elementary transformations*, SIAM Rev., v. 40, n. 4, pp. 947-956.
- [14] Butcher, J.C. y Chen, D.J.L. (2001), *On the implementation of ESIRK methods for stiff IVPs*, Numerical Algorithms 26, pp. 201-218.
- [15] Butcher, J.C. y Rattenbury, N. (2005), *ARK methods for stiff problems*, Appl. Num. Math. 53, pp. 165-181.
- [16] Calvo, M., Montijano, J.I. y Rández, L. (198-?), *Curso de análisis numérico. Métodos lineales multipaso para la resolución de ecuaciones diferenciales ordinarias*, Universidad de Zaragoza.
- [17] Cash, J.R. (1979), *Diagonally implicit Runge-Kutta formulae with error estimates*, J. IMA 24, pp. 293-301.
- [18] Cash, J.R. (1979), *Stable Recursions: with applications to the numerical solution of Stiff Systems*, Academic Press, New York.

- [19] Cash, J.R. (1981), *On the design of high order exponentially fitted formulae for the numerical integration of stiff systems*, Numerische Mathematik 36, pp. 253-266.
- [20] Cash, J.R. (1981), *On the exponential fitting of composite multiderivative linear multistep methods*, SIAM J. Numer. Anal., v. 18, 5, pp. 808-821.
- [21] Cash, J.R. (1983), *The integration of stiff initial value problems in ODEs using modified extended backward differentiation formulae*, Computers and Mathematics with Applications, v. 9, i. 5, pp. 645-657.
- [22] Cash, J.R. y Lambrou, N. (1985), *Blended extended linear multistep methods for the accurate numerical integration of stiff initial value problems*, Applied Numerical Mathematics, v. 1, i. 3, pp. 195-216.
- [23] Cash, J.R. y Considine, S. (1992) *An MEBDF code for stiff initial value problems*, ACM Trans. Math. Soft. 18, pp. 142-158.
- [24] Cash, J.R. (1998) *A comparison of some codes for the stiff oscillatory problem*, Computers Math. Applic., v. 36, n. 1, pp. 51-57.
- [25] Cash, J.R. (2000) *Modified extended backward differentiation formulae for the numerical solution of stiff initial problems in ODEs and DAEs*, J. Comput. Appl. Math. v. 125, pp. 117-130.
- [26] Castillo, P. y Saad Y. (1998), *Preconditioning the matrix exponential operator with applications*, J. Sci. Comput., v. 13, n. 3, pp. 275-302.
- [27] Celledoni, E. y Iserles A. (2000), *Approximating the matrix exponential from a Lie algebra to a Lie group*, Math. Comp. 69, pp. 1457-1480.
- [28] Celledoni, E. y Iserles A. (2001), *Methods for the approximations of the matrix exponential in a Lie-algebraic setting*, IMA J. Numer. Anal. 21, pp. 463-488.

- [29] Coleman, J.P. y Ixaru, L.G. (1996), *P-stability and exponential fitting methods for $y'' = f(x, y)$* , IMA J. Num. Anal., v. 16, i. 2, pp. 179-199.
- [30] Coleman, J.P. y Duxbury, S.C. (2000), *Mixed collocation methods for $y'' = f(x, y)$* , J. Computational and Applied Mathematics 126, pp. 47-75.
- [31] Collings, A.G. y Tee, G.J. (1979), *Stability and accuracy of the generalized Euler method for ordinary differential equations, with reference to structural dynamics problem*, Engineering Structures, v. 1, i. 2, pp. 99-108.
- [32] Collings, A.G. y Tee, G.J. (1979), *The solution of structural dynamics problems by the generalized Euler method*, Computers and Structures, v. 10, i. 3, pp. 505-515.
- [33] Cox, S.M. y Mathews, P.C. (2002), *Exponential time differencing for stiff systems*, J. Comp. Phys. 176, pp. 430-455.
- [34] Corliss, G.F., Griewank, A., Henneberger, P., Kirlinger, G. Potra, F.A. y Stetter, H.J. (1997), *High-order ODE solvers via automatic differentiation and rational prediction*, Numerical analysis and its applications (Rousse, 1996), pp. 114-125. Springer, Berlin.
- [35] Crouzeix M. (1975), *Sur l'approximation des equations differentiales operationelles lineaires par des methodes Runge-Kutta*, Tesis doctoral, Universidad de París VI, París.
- [36] Curtiss, C.F. y Hirschfelder, J.O. (1952), *Integration of stiff equations*, Proc. Nat. Acad. Sci. 38, pp. 235-243.
- [37] Dahlquist (1963), *A special stability problem for linear multistep methods*, BIT 3, pp. 27-43.

- [38] Dekker, K. and Verwer, J.G. (1984), *Stability of Runge-Kutta methods for stiff nonlinear differential equations*, Elsevier Science Publishers B.V., Amsterdam.
- [39] Dennis, S.C.R. (1960), *The numerical integration of ordinary differential equations possessing exponential type solutions*, Proc. Cambridge Philos. Soc. 56, pp. 240-246.
- [40] Dilão, R. (2005), *Turing instabilities and patterns near a Hopf bifurcation*, Applied Mathematics and Computation 164, pp. 391-414.
- [41] Dogan, A. (2004), *A Galerkin finite element approach to Burgers' equation*, Applied Mathematics and Computation 157, pp. 331-346.
- [42] Dongarra, J.J., Bunch, J.R., Moler, C.B. y Stewart, G.W. (1979), *LINPACK User's guide*, SIAM, Philadelphia, Pennsylvania.
- [43] Edwards, W.S., Tuckerman, L.S., Friesner, R.A. y Sorensen, D.C. (1994), *Krylov methods for the incompressible Navier-Stokes equations*, J. Comput. Physics, v. 110, pp. 82-102.
- [44] Ehle, B.L. (1969), *On Pade approximations to the exponential function and A-stable methods for the numerical solution of initial value problems*, University of Waterloo, Dept. Applied Analysis and computer Sci., Research Rpt. N. CSRR 2010.
- [45] Emeux, T. y Reiss, E. (1983), *Brusselator isolas*, SIAM J. Appl. Math. 43 pp. 1240-1246.
- [46] Enright, W.H. (1974), *Optimal second derivative methods for stiff systems*, In stiff differential systems, R.A. Willoughby, Ed., Plenum Press, New York, pp. 95-111.
- [47] Enright, W. H., Hull, T.E. y Lindberg, B. (1975), *Comparing numerical methods for stiff systems of ODE's*, BIT 15, pp. 10-48.

- [48] Enright, W. H. y Hull, T.E. (1976), *Comparing numerical methods for the solution of stiff systems of ODEs arising in chemistry*, Numerical Methods for Differential Systems, Lapidus, L. and Schiesser, W.E., Eds., Academic Press, pp. 45-66.
- [49] Enright, W. H. y Pryce, J.D. (1987), *Two FORTRAN packages for assessing initial value methods*, ACM Trans. Math. Software 13, pp. 1-27.
- [50] Eriksson, K; Johnson, C. and Logg, A. (2003), *Explicit time-stepping for stiff ODE's*, SIAM J. Scientific Computing, vol 25, 4, pp. 1142-1157.
- [51] Fjeld, S.A. (1980), *An implementation of estimation techniques to a hydrological model for prediction of runoff to a hydroelectric power station*, IEEE Trans. Automat. Control AC-25, pp. 151-163.
- [52] Fox, L. y Parker, I.B. (1968), *Chebyshev Polynomials in Numerical Analysis*, Oxford U. Press, London.
- [53] Franco, J.M., Gómez, I. y Rández, L. (1997), *SDIRK methods for stiff ODEs with oscillating solutions*, J. Comput. Appl. Math., v. 81, pp. 197-209.
- [54] Franco, J.M. (2004), *Exponentially fitted explicit Runge-Kutta-Nyström methods*, J. Comput. Appl. Math., v. 167, pp. 1-19.
- [55] Friesner, R.A. Tuckerman, L.S., Dornblaser, B.C. y Russo, T.C. (1989), *A method for exponential propagation of large systems of stiff nonlinear differential equations*, J. Sci. Comput., v. 4, pp. 327-354.
- [56] Gallopoulos, E. y Saad Y. (1992), *Efficient solution of parabolic equations by Krylov approximation methods*, SIAM J. Sci. Stat. Comput., v. 13, i. 5, pp. 1236-1264.

- [57] Gautschi, W. (1961), *Numerical integration of ordinary differential equations based on trigonometric polynomials*, Numer. Math. 3, pp. 381-397.
- [58] Gear, C.W. (1982), *Numerical initial value problems in ordinary differential equations*, Prentice-Hall, Englewood Cliffs, NJ.
- [59] Gottwald, B.A. (1977), *MISS-ein einfaches Simulations-System für biologische und chemische Prozesse*, EDV in Medizin und Biologie 3, pp. 85-90.
- [60] Greenwood, R.E. (1949), *Numerical integration of linear sums of exponential functions*, Ann. Math. Statist. 20, pp. 608-611.
- [61] Guang-Wu, Yan y Li, Y. (2001), *A lattice Boltzmann method for the chemical clock in the Belousov-Zhabotinskii reaction*, Chinese Physics Letters 18, pp. 918-920.
- [62] Guzel, N. y Bayram, M. (en prensa), *On the numerical solution of stiff systems*, Appl. Math. and Comput., prueba corregida, en internet desde 29 Enero 2005.
- [63] Hairer, E. y Wanner, G. (1991), *Solving Ordinary Differential Equations I. Nonstiff Problems*, Springer-Verlag, Berlin.
- [64] Hairer, E. y Wanner, G. (1991), *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems*, Springer-Verlag, Berlin.
- [65] Henrici, P. (1962), *Discrete variable Methods in Ordinary Differential Equations*, John Wiley, New York.
- [66] Higuera, I. y Roldán, T. (2000), *Starting algorithms for some DIRK methods*, Num. Algorith. 23, pp. 357-369.

- [67] Hindmarsh A.C. (1977), *GEARB: Solution of ordinary differential equations having banded jacobian*, Lawrence Livermore Report UCID-30059, Rev. 2, June, Livermore, California.
- [68] Hindmarsh A.C. (1980), *LSODE and LSODI, two new initial value ordinary differential equation solvers*, ACM-Signum Newsletter 15, 10-11.
- [69] Hochbruck, M. y Lubich, Ch. (1997), *On Krylov subspace approximations to the matrix exponential operator*, SIAM J. Numer. Anal. 34, pp. 1911-1925.
- [70] Hochbruck, M., Lubich, Ch. y Selhofer, H. (1998), *Exponential integrators for large systems of differential equations*, SIAM J. Sci. Comput. 19, pp. 1552-1574.
- [71] Iavernaro, F. y Mazzia, F. (1998), *Solving ordinary differential equations by generalized Adams methods*, Appl. Numer. Math. 28, pp. 107-126.
- [72] Ixaru, L.Gr., Rizea, M., De Meyer, H. y Vanden Berghe, G. (2001), *Weights of the exponential fitting multistep algorithms for ODEs*, J. of Comput. and Appl. Math., vol. 132, pp. 83-93.
- [73] Ixaru, L.Gr., Vanden Berghe, G. y De Meyer, H. (2002), *Frequency evaluation in exponential fitting multistep algorithms for ODEs*, J. of Comput. and Appl. Math., vol. 140, pp. 423-434.
- [74] Ixaru, L.Gr., Vanden Berghe, G. y De Meyer, H. (2003), *Exponentially fitted variable two-step BDF algorithms for first order ODEs*, Comput. Phys. Comm. 100, pp. 56-70.
- [75] Jalbert, F. y Zahar, R.V.M. (1985), *A highly precise Taylor series method for stiff ODEs*, Proceedings of the fourteenth Manitoba con-

- ference on numerical mathematics and computing (Winnipeg, Man., 1994), v. 46, pp. 347-358.
- [76] Janssen, M., Van Hentenryck, P. y Deville Y. (2001), *Optimal Pruning in Parametric Differential Equations*, Practice of Constraint Programming, Springer.
- [77] Jackson, L.W. y Kenue, S.K. (1974), *A fourth order exponentially fitted method*, SIAM J. Numer. Anal., v. 11, 5, pp. 965-978.
- [78] Jiménez, J.C., Biscay, R., Mora, C. y Rodríguez, L.M. (2002), *Dynamic properties of the local linearization method for initial-value problems*, Appl. Math. Comput. 126, pp. 63-81.
- [79] Jorba, A. y Zou, M. (2004), *A software package for the numerical integration of ODEs by means of high-order Taylor methods*, Experimental Math. v. 14.
- [80] Kassam, A. y Trefethen, L.N. (2005), *Fourth-order time stepping for stiff PDEs*, SIAM J. Scientific Computing, v. 26, n. 4, pp. 1214-1233.
- [81] Kirlinger, G. y Corliss, G.F. (1992), *On implicit Taylor series methods for stiff ODEs*, Computer arithmetics and enclosure methods (Oldenburg, 1991), pp. 371-379. North-Holland, Amsterdam.
- [82] Kiss, I., Merkin, J.H., Scott, S.K. y Simon, P.L. (2004), *Electric field effects on travelling waves in the Oregonator model for the Belousov-Zhabotinsky reaction*, Quarterly Journal of Mechanics and Applied Mathematics 57, n. 4, pp. 467-494.
- [83] Knizhnerman, L.A. (1991), *Computations of functions of unsymmetric matrices by means of Arnoldi's method*, Comput. Math. Math. Phys., v. 31, i. 1, pp. 5-16.

- [84] Konguetsof, A. y Simos, T.E. (2003), *An exponentially-fitted and trigonometrically-fitted method for the numerical solution of periodic initial-value problems*, Computers and Mathematics with Applications 45, pp. 547-554.
- [85] Kosloff, R. (1994), *Propagation methods for quantum molecular dynamics*, Annu. Rev. Phys. Chm 45, pp. 145-178.
- [86] Lambert, J.D. and Sigurdsson, S.T. (1972), *Multistep methods with variable matrix coefficients*, SIAM J. Numer. Anal., vol 9, pp. 715-733.
- [87] Lambert J.D. (1991), *Numerical Methods for Ordinary Differential Systems. The initial Value Problem*, Wiley, Chichester.
- [88] Lambert, J.D. and Watson, I.A. (1976), *Symmetric multistep methods for periodic initial value problems*, J. Inst. Math. Appl., vol. 18, pp. 189-202.
- [89] Liniger, W. y Willoughby, R.A. (1967), *Efficient numerical integration of ordinary differential equations*, Res. Rep. RC 1970, IBM, Yorktown Heights, N.Y.
- [90] Liniger, W. y Willoughby, R.A. (1970), *Efficient integration methods for stiff systems of ordinary differential equations*, Res. Rep. RC 1970, 7, pp. 47-60.
- [91] Lioen, W.M., Swart, J.J.B. de y Van der Veen, W.A. (1996), *Test set for IVP solvers*, Rep. NM-R 9615, CWI, Amsterdam.
- [92] Lioen, W.M. y Swart, J.J.B. de (1998), *Test set for initial value problem solvers*, Rep. MAS-R 9832, CWI, Amsterdam.
- [93] Magherini, C. (2004), *Numerical solution of stiff ODE-IVPs via Blended Implicit methods*, Tesis Doctoral, Universita degli studi di Firenze.

- [94] Mazzia, F. y Iavernò, F. (2003), *Test set for initial value problem solvers*, Report 40/2003, Bari.
- [95] Martín-Vaquero J. y Vigo-Aguiar, J. (en prensa), *Exponential fitting BDF algorithms: explicit and implicit 0-stable methods*, J. Comp. Appl. Math., prueba corregida, en internet desde 24 Junio 2005.
- [96] Martín-Vaquero, J. y Vigo-Aguiar, J. (sometido), *Adapted BDF Algorithms: higher-order methods and their stability*, J. Comp. Appl. Math.
- [97] Meerbergen, K. y Sadkane, M. (1999), *Using Krylov approximation to the matrix exponential operator in Davidson's method*, Appl. Numer. Math., v. 32, i. 3, pp. 331-351.
- [98] Miranker, W.L. (1981), *Numerical methods for stiff equations and singular perturbation problems*, Reidel Publ. Comp., Dordrecht.
- [99] Moler, C.B. y Van Loan, C.F. (1978), *Nineteen dubious ways to compute the exponential of a matrix*, SIAM Rev., v. 29, 4, pp. 801-836.
- [100] Morini, B. y Macconi, M. (1999), *Inexact methods in the numerical solution of stiff initial value problems*, Computing 63 , pp. 265-281.
- [101] Nauts, A. y Wyatt, R.E. (1983), *New approach to many-state quantum dynamics: The recursively residue-generation method*, Phys. Rev. Lett. 51, pp. 2238-2241.
- [102] Norsett, S.P. (1974), *Semi explicit Runge-Kutta methods*, Mathematics and Computing Rpt. N. 6/74, University of Trondheim.
- [103] Nour-Omid, B. (1989), *Applications of the Lanczos algorithm*, Comput. Phys. Commun. 53, pp. 153-168.
- [104] Papageorgiou, G., Famelis, I.Th. y Tsitouras, Ch. (1998), *A P-stable singly diagonally implicit Runge-Kutta-Nyström method*, Numerical Algorithms 17, pp. 345-353.

- [105] Park, H.M., Lee, M.W. y Jang, Y.D. (1998), *An efficient computational method of boundary optimal control problems for the Burgers equation*, Computer Methods in Applied Mechanics and Engineering 166, pp. 298-308.
- [106] Park, T.J. y Light, J.C. (1986), *Unitary quantum time evolution by iterative Lanczos reduction*, J. Chem. Phys. 85, pp. 5870-5876.
- [107] Peng, R. y Wang, M. (2005), *Pattern formation in the Brusselator system*, J. Math. Anal. and Appl. 309, i. 1, pp. 151-166.
- [108] Petzold, L.R. (1981), *An efficient numerical method for highly oscillatory ordinary differential equations*, SIAM J. Numer. Anal. 18, pp. 455-479.
- [109] Petzold, L.R. (1993), *A description of DASSL: a differential-algebraic system solver*, IMACS Trans. Scientific Computing, eds. R.S. Stepleman et al., North-Holland, Amsterdam, pp. 65-68.
- [110] Prothero, A. y Robinson, A. (1974), *On the stability and accuracy of one-step methods for solving stiff systems of ordinary differential equations*, Math. of Comput. 28, pp. 145-162.
- [111] Psihoyios, G. (2004), *A class of implicit advanced step-point methods with a parallel feature for the solution of stiff initial value problems*, Mathematical and Computer Modelling, v. 40, i. 11-12, pp. 1199-1224.
- [112] Psihoyios, G. y Simos, T.E. (2003), *Trigonometrically fitted predictor-corrector methods for IVPs with oscillating solutions*, Journal of Computational and Applied Mathematics 158, pp. 135-144.
- [113] Prigogine, I. y Lefever, R. (1968), *Symmetry breaking instabilities in dissipative systems II*, J. Chem. Phys. 48, pp. 1695-1700

- [114] Putzer, E.J. (1996), *Avoiding the Jordan canonical form in the discussion of linear systems with constant coefficients*, Amer. Math. Monthly 73, pp. 2-7.
- [115] Qinghong, L. y Xinyuan, W. (2003), *A two step explicit P-stable method for solving second order initial value problems*, Applied Mathematics and Computation 138, pp. 435-442.
- [116] Ramos, H. (2004), *Problemas de valores iniciales de segundo orden de tipo oscilatorio*, Tesis Doctoral, Universidad de Salamanca.
- [117] Ramos, H. y Vigo-Aguiar, J. (sometido), *A new BDF-type method of fourth order for stiff problems*, J. Comput. Appl. Math.
- [118] Ramos, J.I. y García-López, C.M. (1997), *Piecewise-linearized methods for initial-value problems*, Appl. Math. and Comput. 82, pp. 273-302.
- [119] Ramos, J.I. (2005), *A piecewise-analytical method for singularly perturbed parabolic problems*, Appl. Math. and Comput. 161, pp. 501-512.
- [120] Ramos, J.I. (2005), *An exponentially-fitted method for singularly perturbed, one-dimensional, parabolic problems*, Appl. Math. and Comput. 161, pp. 513-523.
- [121] Ramos, J.I. (2005), *An exponentially-fitted method for singularly-perturbed ordinary differential equations with turning points and parabolic problems*, Appl. Math. and Comput. 165, pp. 549-564.
- [122] Ratowsky, R.P. y Fleck Jr., J.A. (1991), *Accurate numerical solution of the Helmholtz equation by iterative Lanczos reduction*, Optic Letters 73, pp. 787-789.
- [123] Robertson, H.H. (1966), *The solution of a set of reaction rate equations*, Academic Press, New York pp. 178-182.

- [124] Ropp, D.L., Shadid, J.N. y Ober, C.C. (2004), *Studies of the accuracy of time integration methods for reaction-diffusion equations*, J. Comput. Physics 194, pp. 544-574.
- [125] Rosebrock, H.H. (1963), *Some general implicit processes for the numerical solution of differential equations*, Comp. J. 5, pp. 329-330.
- [126] Saad, Y. (1992), *Analysis of some Krylov subspace approximations to the matrix exponential operator*, SIAM J. Numer. Anal., v. 29, 1, pp. 209-228.
- [127] Schäfer, E. (1975), *A new approach to explain the high irradiance responses of photomorphogenesis on the basis of phytochrome*, J. of Math. Biology 2, pp. 41-56.
- [128] Scherer, R. (1979), *A necessary condition for B-stability*, BIT 19, pp. 111-115.
- [129] Shampine, L.F. y Gordon, M.K., (1975), *Computer Solution of ordinary differential equations: the initial value problem*, W.H. Freeman, S. Francisco.
- [130] Shampine, L.F. (1994), *Numerical solution of ordinary differential equations*, Chapman y Hall, New York.
- [131] Shampine, L.F. and Reichelt, M. (1997), *The MATLAB ODE Suite*, SIAM J. Sci. Comput., vol. 18, pp. 1-22.
- [132] Sidge, R.B. (1998), *Expokit: software package for computing matrix exponentials*, ACM Trans. Math. Software, v. 24, 1, pp. 130-156.
- [133] Smith, J. (1997), *Mathematical Modelling and Digital Simulations for Engineerings and Scientists*, Wiley, Chichester.
- [134] Spijker, M.N. (1996), *Stiffness in the numerical initial-value problems*, J. of Comput. and Appl. Math., vol. 72, pp. 393-406.

- [135] Stetter, H.J. (1973), *Analysis of discretization methods for ordinary differential equations*, Springer, Berlin.
- [136] Stiefel, E. y Bettis, D.G. (1969), *Stabilization of Cowell's method*, Numer. Math., vol. 13, pp. 154-175.
- [137] Strizhak, P.E., Didenko, O.Z. y Ivashchenko, T.S. (2001), *Determination of traces of thallium using the transient chaotic regime in the Belousov-Zhabotinskii oscillating chemical reaction*, Analytica Chimica Acta, vol. 428, pp. 15-21.
- [138] Swart, J.J.B. de y Lioen, W.M. (1998), *Collecting real-life problems to test solvers for implicit differential equations*, CWI Quaterly 11(1), pp. 83-100.
- [139] Tang, Ch., Yan, H., Zhang, H. and Li, W. (2004), *The various order explicit multistep exponential fitting for systems of ordinary differential equations*, J. of Comput. and Appl. Math., v. 169, i. 1, pp. 171-182.
- [140] Tang, Ch., Yan, H., Zhang, H. and Chen, Z. (2005), *The arbitrary order implicit multistep schemes of exponential fitting and their applications*, J. of Comput. and Appl. Math., v. 173, i. 1, pp. 155-168.
- [141] Thomée, V. (1991), *Galerkin finite element methods for parabolic problems*, Springer, Berlin.
- [142] Tsitouras, Ch. y Simos, T.E. (2002), *High algebraic, high phase-lag order embedded Numerov-type methods for oscillatory problems*, Applied Mathematics and Computation 131, pp. 201-211.
- [143] Van de Vyver, H. (2005), *Frequency evaluation for exponentially fitted Runge-Kutta methods*, J. Comput. Appl. Math. 184, i. 2, pp. 442-463.

- [144] Van der Hoewen, P.J., Sommeijer, B.P. y Van der Veen, W.A. (1993), *Parallelism across the steps in iterated Runge-Kutta methods for stiff initial value problems*, Report NM-R9322, CWI, Amsterdam.
- [145] Van der Pol, B. (1926), *On relaxation oscillations*, Phil. Mag. 2, pp. 978-992, reproducido en Selected Scientific Papers 1, North-Holland Publ. Comp. Amsterdam, 1960.
- [146] Van der Veen, W.A. (1995), *Step-parallel algorithm for stiff initial value problems*, Report NM-R9507, CWI, Amsterdam.
- [147] Vanden Berghe, G., Ixaru, L.Gr. y Van Daele, M. (2001), *Optimal implicit exponentially fitted Runge-Kutta methods*, Comput. Phys. Comm 140, pp. 346-357.
- [148] Van Hecke, T., Vanden Berghe, G., Van Daele, M. y De Meyer, H. (1996), *A variable-step variable-order algorithm for systems of stiff ODE's*, Intern. J. Computer. Math. 63, pp. 149-157.
- [149] Vigo-Aguiar, J. y Ferrández, J. M. (1998), *A general procedure for the adaptation of multistep algorithms to the integration of oscillatory problems*, SIAM Journal Numerical Analysis, v. 35, 4, pp. 1684-1708.
- [150] Vigo-Aguiar, J. (1999), *An approach to variable coefficients multistep methods for special differential equations*, Int. J. Appl. Math., v. 1, n. 8, pp. 911-921.
- [151] Vigo-Aguiar, J., Martín-Vaquero, J. y Criado, R. (2005), *On the stability of exponential Fitting BDF Algorithms*, J. Comput. Appl. Math., v. 175, i. 1, pp. 183-194.
- [152] Vigo-Aguiar, J. y Ramos, H. (sometido), *A Chebyshev backward differentiation formulae for the solution of stiff initial-value problems*, SIAM J. Scientific Computing.

- [153] Vigo-Aguiar, J., Martín-Vaquero, J. y Wade, B. (sometido), *Adapted BDF Algorithms Applied to Parabolic Problems*, Numer. Meth. Part. Differ. Equ.
- [154] Willoughby, R.A. (1974), *Stiff differential systems*, New York, Plenum Press.
- [155] Winfree, A.T. (1984), *The prehistory of the Belousov-Zhabotinsky oscillator*, J. of Chemical Education, v. 61, pp. 661-663.
- [156] Yu, D.W. (2000), *Generalized Euler process for systems of nonlinear differential equations*, Korean J. Comput. and Appl. Math. (Series A), v. 7, 3, pp. 941-958.
- [157] Yu, D.W. (2004), *The generalized Euler process for exponentially dominant systems*, Computers and Math. with Applications 47, pp. 161-181.
- [158] Zaikin, A.N. y Zhabotinsky, A.M. (1970) *Concentration wave propagation in two-dimensional liquid-phase self-oscillating system*, Nature 225, pp. 535-537.
- [159] Zhu, R. y Shu Li, Q. (2003), *Eliminating chaos in the Belousov-Zhabotinsky reaction by no-delay feedback and delayed feedback*, Theoretica Chimica Acta 110, pp. 85-91.