
**stichting
mathematisch
centrum**



AFDELING NUMERIEKE WISKUNDE

NN 2/74

JUNI

S.P.N. VAN KAMPEN
EEN 4-de ORDE GEGENERALISEERDE RUNGE-KUTTA METHODE

2e boerhaavestraat 49 amsterdam

Printed at the Mathematical Centre, 49, 2e Boerhaavestraat, Amsterdam.

The Mathematical Centre, founded the 11-th of February 1946, is a non-profit institution aiming at the promotion of pure mathematics and its applications. It is sponsored by the Netherlands Government through the Netherlands Organization for the Advancement of Pure Research (Z.W.O), by the Municipality of Amsterdam, by the University of Amsterdam, by the Free University at Amsterdam, and by industries.

Een 4^{de} orde gegeneraliseerde Runge-Kutta methode

S.P.N. van Kampen

VOORWOORD

Dit rapport beschrijft een afleiding van een gegeneraliseerde Runge-Kutta methode voor het oplossen van beginwaardeproblemen.

Het ALGOL 60 programma en de testvoorbeelden zijn getest op de Electrologica X8 computer van het Mathematisch Centrum en gedeeltelijk op de Control Data Cyber 73-28 van de Stichting Academisch Rekencentrum Amsterdam. De schrijver betuigt zijn dank aan Dr. P.J. van der Houwen voor diens waardevolle hulp.

INHOUD

	blz.
1. Inleiding	1
2. Semi-impliciete Runge-Kutta methoden	2
2.1. Structuur van het gegeneraliseerde Runge-Kutta schema	2
2.2. Consistentievoorwaarden	3
2.3. Stabiliteit	9
2.4. Stapkeuzestrategie	14
3. De procedure effosirk	16
3.1. Heading en parameters	16
3.2. De procedure body	18
3.3. De procedure stepsize	21
3.4. De procedure coefficient	21
3.5. De procedure difference scheme	21
3.6. Bibliotheekprocedures	22
4. Testresultaten	22
4.1. Een lineair stelsel differentiaalvergelijkingen	22
4.2. Een niet-lineair stelsel differentiaalvergelijkingen	24
4.3. Een stelsel stijve differentiaalvergelijkingen met een positieve eigenwaarde	27
4.4. Een eenvoudige, niet stijve, 2 ^e orde vergelijking	32
5. Conclusie	35
Referenties	36

1. INLEIDING

We beschouwen het beginwaardeprobleem

$$(1.1) \quad \frac{dy}{dx} = f(y), \quad y_0 = y(x_0).$$

Hierin is $x \in \mathbb{R}$ de onafhankelijk veranderlijke en $f \in \mathbb{R}^n \rightarrow \mathbb{R}^n$ een vector van functies die voldoende vaak continu differentieerbaar zijn naar y . De oplossing y is een vector van functies $\in \mathbb{R}^n \rightarrow \mathbb{R}$.

Voor het oplossen van (1.1) is een groot aantal methoden ontwikkeld. In het geval dat de afgeleiden van het rechterlid moeilijk te berekenen zijn kunnen Runge-Kutta methoden gebruikt worden. In de conventionele Runge-Kutta methoden (zie bijv. Lapidus & Seinfeld [1971]) zijn de parameters constanten. Als deze methoden expliciet zijn vraagt verhoging van de orde van consistentie en vergroting van het stabiliteitsgebied een groot aantal rechterlid evaluaties. Bij impliciete methoden kan een hoge orde van consistentie bereikt worden (zie Butcher [1964]) terwijl het stabiliteitsgebied het gehele linker halfvlak kan bestrijken (A-stabiliteit in de zin van Dahlquist [1963]). Deze laatste methoden hebben echter het nadeel dat een transcendent stelsel vergelijkingen met een iteratief proces opgelost moet worden.

Een generalisatie van de gewone Runge-Kutta methoden krijgen we als een of meer parameters vervangen worden door een functie van de Jacobiaan van het stelsel. Dergelijke gegeneraliseerde Runge-Kutta methoden zijn voor de orden 2 en 3 gegeven door Rosenbrock [1963] en Calahan [1968] en voor orden kleiner of gelijk 4 door Van der Houwen [1973].

In de volgende sectie zal bewezen worden dat bij deze gegeneraliseerde Runge-Kutta methoden dezelfde orde van consistentie mogelijk is als bij conventionele impliciete methoden. Vergeleken bij de gewone impliciete schema's hebben deze gegeneraliseerde methoden echter het voordeel dat, alleen indien minstens één van de parameters een rationale functie van de Jacobiaan is, een lineair stelsel vergelijkingen opgelost moet worden.

Voorts beschouwen we een methode uit een klasse van 2-punts 4^e orde consistente schema's, waarin de stabiliteitsfunctie nog vrij gekozen kan

worden (Van der Houwen [1973]). Door een geschikte keuze van deze stabiliteitsfunctie is bijvoorbeeld A-stabiliteit en exponentiële aanpassing in één of meer eigenwaarden van de Jacobiaan mogelijk.

In de derde sectie wordt een ALGOL 60 procedure van deze methode besproken.

In de laatste sectie wordt de methode getest op een aantal beginwaardeproblemen.

2. SEMI-IMPLICIETE RUNGE-KUTTA METHODEN

In deze sectie geven we een beknopt overzicht van de theorie van gegeneraliseerde Runge-Kutta methoden. We geven een afleiding van de consistentievoorwaarden waaraan het schema moet voldoen en leiden een uitdrukking af voor de maximale orde van het schema. Tevens worden de stabiliteitseigenschappen en de stapkeuzestrategie van een 2-punts 4^e orde semi-impliciete methode besproken.

2.1. Structuur van het gegeneraliseerde Runge-Kutta schema

We beschouwen het m-punts Runge-Kutta schema

$$(2.1) \quad \left\{ \begin{array}{l} y_{n+1} = y_n + \sum_{j=0}^{m-1} \theta_j K_n^{(j)}, \\ K_n^{(j)} = h_n f(y_n + \sum_{\ell=0}^{j-1} \Lambda_{j,\ell} K_n^{(\ell)}), \quad j=0,1,\dots,m-1, \\ \Lambda_{0,0} = \Lambda_{0,-1} = 0. \end{array} \right.$$

Hierin is y_n de numerieke oplossing in het punt $x = x_n$ en h_n de lengte van de $(n+1)^e$ -integratiestap. De parameters θ_j en $\Lambda_{j,\ell}$ zijn functies van $h_n J_n^e$, waarbij

$$J_n = \frac{\partial}{\partial y} f(y) \Big|_{y=y_n}$$

de Jacobiaan is van het stelsel in $y = y_n$.

Indien alle θ_j en $\Lambda_{j,l}$ polynomen zijn in $h_n J_n$ noemen we de methode *explicit*, terwijl we de methode *semi-implicit* noemen als minstens één van de parameters een rationale functie is in $h_n J_n$.

2.2. Consistentievoorwaarden

Stellen we schema (2.1) compact voor door $y_{n+1} = E(y_n)$ en is $y(x_{n+1})$ een lokaal-analytische oplossing van (1.1) door (x_n, y_n) dan noemen we schema (2.1) p^e orde consistent indien

$$(2.2) \quad y(x_{n+1}) - E(y(x_n)) = O(h_n^{p+1}), \quad h_n \rightarrow 0.$$

We beschouwen nu het 2-punts gegeneraliseerde Runge-Kutta schema

$$(2.3) \quad y_{n+1} = y_n + \theta_0(z) h_n f(y_n) + \theta_1(z) h_n f(y_n + \Lambda(z) h_n f(y_n)),$$

met $z = h_n J_n$. Voor de afgeleiden van θ_j en Λ naar z in $z = 0$ gebruiken we de volgende notaties.

$$\theta_j = \theta_j(0), \quad \theta'_j = \frac{d}{dz} \theta_j(z) \Big|_{z=0}, \quad \text{enz.},$$

$$\lambda = \Lambda(0), \quad \lambda' = \frac{d}{dz} \Lambda(z) \Big|_{z=0}, \quad \text{enz.}$$

Voor de consistentievoorwaarden van formule (2.3) geldt nu de volgende stelling (zie Van der Houwen [1973]).

Stelling 2.1. Formule (2.3) is consistent van de orde 1 als

$$(2.4) \quad \theta_0 + \theta_1 = 1,$$

en van de orde 2 als ook

$$(2.5) \quad \theta_0' + \theta_1' + \theta_1 \lambda = \frac{1}{2},$$

en van de orde 3 als ook

$$(2.6) \quad \begin{cases} \frac{1}{2} \theta_0'' + \frac{1}{2} \theta_1'' + \theta_1 \lambda' + \theta_1' \lambda = \frac{1}{6}, \\ \frac{1}{2} \theta_1 \lambda^2 = \frac{1}{6}, \end{cases}$$

en van de orde 4 als ook

$$(2.7) \quad \begin{cases} \frac{1}{6} \theta_0''' + \frac{1}{6} \theta_1''' + \frac{1}{2} \theta_1 \lambda'' + \theta_1' \lambda' + \frac{1}{2} \theta_1'' \lambda = \frac{1}{24}, \\ \frac{1}{2} \theta_1' \lambda^2 = \frac{1}{24}, \\ \theta_1 \lambda \lambda' = \frac{3}{24}, \\ \frac{1}{6} \theta_1 \lambda^3 = \frac{1}{24}. \end{cases}$$

Bewijs. De Taylorontwikkeling van de lokaal-analytische oplossing $y(x_{n+1})$ in $x = x_n$ luidt:

$$(2.8) \quad y(x_{n+1}) = y_n + h_n f(y_n) + \frac{1}{2} \frac{d}{dx} f(y) \Big|_{y=y_n} + \frac{1}{6} \frac{d^2}{dx^2} f(y) \Big|_{y=y_n} + \dots$$

Stel dat de vectoren f en y bestaan uit de componenten f^k en y^k , $k=1,2,\dots,s$. We duiden nu de partiële afgeleide van f^j naar y^k in $x = x_n$ aan met f_k^j . Voor boven- en onderindices die in een uitdrukking tweemaal voorkomen gebruiken we de sommatieconventie van Einstein. Met behulp van de kettingregel vinden we nu voor de 2^e , 3^e en 4^e afgeleide van y naar x in $x = x_n$:

$$(2.9) \quad \begin{cases} \frac{d}{dx} f(y) = f_j f^j, \text{ (dit is dus een notatie voor } \sum_{j=1}^s \left(\frac{\partial}{\partial y^j} f \right) f^j), \\ \frac{d^2}{dx^2} f(y) = \frac{d}{dx} (f_j f^j) = f_j f_k^j f^k + f_{jk} f^j f^k, \\ \frac{d^3}{dx^3} f(y) = f_j f_k^j f_\ell^k f^\ell + f_j f_{k\ell}^j f^k f^\ell + 3 f_{jk} f_\ell^j f^\ell f^k + f_{j k \ell} f^j f^k f^\ell. \end{cases}$$

We ontwikkelen θ_0 in machten van $h_n J_n$:

$$(2.10) \quad \theta_0(h_n J_n) = \theta_0 + \theta'_0 h_n J_n + \frac{1}{2} \theta''_0 h_n^2 J_n^2 + \dots$$

met analoge reeksen voor θ_1 en Λ . De Taylorontwikkeling van $f(y_n + \Lambda h_n f(y_n))$ in $y = y_n$ luidt:

$$(2.11) \quad f(y_n + \Lambda h_n f(y_n)) = f + h_n f_j (\Lambda f)^j + \frac{1}{2} h_n^2 f_{jk} (\Lambda f)^j (\Lambda f)^k + \dots$$

We merken op dat $J_n f = f_j f^j$, $J_n^2 f = f_j f_k^j f^k$, enz., en vinden dan uit (2.10) en (2.11) de volgende ontwikkeling:

$$(2.12) \quad \begin{aligned} y_{n+1} = y_n &+ h_n (\theta_0 + \theta_1) f + h_n^2 (\theta'_0 + \theta'_1 + \theta_1 \lambda) f_j f^j + \\ &+ h_n^3 \left[\left(\frac{1}{2} \theta''_0 + \frac{1}{2} \theta''_1 + \theta_1 \lambda' + \theta'_1 \lambda \right) f_j f_k^j f^k + \frac{1}{2} \theta_1 \lambda^2 f_{jk} f^j f^k \right] + \\ &+ h_n^4 \left[\left(\frac{1}{6} \theta'''_0 + \frac{1}{6} \theta'''_1 + \frac{1}{2} \theta_1 \lambda'' + \theta'_1 \lambda' + \frac{1}{2} \theta''_1 \lambda \right) f_j f_k^j f_\ell^k f^\ell + \right. \\ &+ \frac{1}{2} \theta'_1 \lambda^2 f_j f_{k\ell}^j f^k f^\ell + \theta_1 \lambda \lambda' f_{jk} f_\ell^j f^k f^\ell + \\ &\left. + \frac{1}{6} \theta_1 \lambda^3 f_{jkl} f^j f^k f^\ell \right] + O(h_n^5), \quad h_n \rightarrow 0. \end{aligned}$$

Substitueren we nu (2.9) in formule (2.8) en vergelijken we de overeenkomstige termen in (2.8) en (2.12), dan vinden we de voorwaarden (2.4) tot (2.7) uit de stelling. $\square$

Uit stelling 2.1 blijkt dat een 1-punts gegeneraliseerde Runge-Kutta methode wel 2^e orde exact maar niet 3^e orde exact kan zijn. Een 2-punts methode kan 4^e orde nauwkeurig zijn. In het algemeen geldt hiervoor de volgende stelling:

Stelling 2.2. De consistentie-orde van het m-punts gegeneraliseerde Runge-Kutta schema (2.1) is hoogstens 2m.

Bewijs. In het bewijs van deze stelling gebruiken we dezelfde conventies

betreffende de boven- en onder indices van de vectorfunctie $f = f(y)$ en zijn afgeleiden als in het bewijs van stelling 2.1.

In navolging van Butcher [1963] definiëren we nu het begrip elementaire differentiaal:

1. $f = f(y_n)$ is de enige elementaire differentiaal van de orde 1;
2. alle elementaire differentiaal van de orde $r + 1$ verkrijgen we door differentiatie naar x van alle elementaire differentiaal van de orde r , d.w.z. alle elementaire differentiaal van de orde $r + 1$ zijn termen van de vorm

$$f_{j_1 j_2 \dots j_s} F^{j_1}_{j_1} F^{j_2}_{j_2} \dots F^{j_s}_{j_s}, \quad s=1,2,\dots,r.$$

Hierin is $F^{j_i}_{j_i}$ de j_i de component van een elementaire differentiaal van de orde $r_i \leq r - s + 1$, zodanig dat

$$r + 1 = 1 + r_1 + r_2 + \dots + r_s.$$

De enige elementaire differentiaal van de orde 2 is dus $f_{j_1} f^{j_1}_{j_1}$ en de elementaire differentiaal van de orde 3 zijn $f_{j_1 j_2} f^{j_1 j_2}_{j_1 j_2}$ en $f_{j_1 j_2} f^{j_1}_{j_1} f^{j_2}_{j_2}$, enz. Het zal duidelijk zijn dat

$$\frac{d^n}{dx^n} f(y) \Big|_{y=y_n}$$

is samengesteld uit elementaire differentiaal van de orde $n + 1$.

We zullen nu bewijzen dat de consistentie-orde van het m -punts gegeneraliseerde Runge-Kutta schema kleiner of gelijk $2m$ is met behulp van het volgende lemma:

Lemma 1. Alle elementaire differentiaal van de orde $\leq 2m$, die voorkomen in de Taylorontwikkeling van de lokaal-analytische oplossing (2.8), komen ook voor in de ontwikkeling van het m -punts gegeneraliseerde Runge-Kutta schema (2.1).

Bewijs lemma 1. Het lemma wordt bewezen met volledige inductie. Lemma 1 is kennelijk juist voor $m = 1$. We veronderstellen nu dat het lemma juist is voor een k -punts schema met $k=1,2,\dots,m-1$. In het m -punts schema is de laatste term

$$\Theta_{m-1} K_n^{(m-1)} = \Theta_{m-1} h_n f(y_n + \Lambda_{m-1,0} K_n^{(0)} + \Lambda_{m-1,1} K_n^{(1)} + \dots + \Lambda_{m-1,m-2} K_n^{(m-2)}).$$

We stellen nu

$$(2.13) \quad h_n B_{m-1} = \Lambda_{m-1,0} K_n^{(0)} + \Lambda_{m-1,1} K_n^{(1)} + \dots + \Lambda_{m-1,m-2} K_n^{(m-2)}.$$

Met behulp van $\Theta_{m-1} = \theta_{m-1} + h_n \theta'_{m-1} h_n J_n + O(h_n^2)$, $h_n \rightarrow 0$, en $J_n F = f_{j_0} F^{j_0}$, krijgen we nu voor de Taylorontwikkeling van $K_n^{(m-1)}$ in $y = y_n$:

$$(2.14) \quad \begin{aligned} \Theta_{m-1} K_n^{(m-1)} &= h_n \theta_{m-1} f + h_n^2 \theta_{m-1} f_{j_1} B_{m-1}^{j_1} + \frac{1}{2} h_n^3 \theta_{m-1} f_{j_1 j_2} B_{m-1}^{j_1 j_2} + \dots \\ &\dots + \frac{1}{(2m-1)!} h_n^{2m} \theta_{m-1} f_{j_1 j_2 \dots j_{2m-1}} B_{m-1}^{j_1 j_2 \dots j_{2m-1}} + \\ &+ h_n^2 \theta'_{m-1} f_{j_0} f^{j_0} + h_n^3 \theta'_{m-1} f_{j_0 j_1} f^{j_0 j_1} B_{m-1}^{j_1} + \frac{1}{2} h_n^4 \theta'_{m-1} f_{j_0 j_1 j_2} f^{j_0 j_1 j_2} B_{m-1}^{j_1 j_2} + \dots \\ &\dots + \frac{1}{(2m-2)!} h_n^{2m} \theta'_{m-1} f_{j_0 j_1 j_2 \dots j_{2m-2}} f^{j_0} B_{m-1}^{j_1 j_2 \dots j_{2m-2}} + O(h_n^{2m+1}), \quad h_n \rightarrow 0. \end{aligned}$$

In (2.14) zijn de termen met θ_{m-1}'' enz. weggelaten.

De uitdrukking B_{m-1} heeft dezelfde structuur als een $(m-1)$ -punts Runge-Kutta formule en brengt volgens de inductie veronderstelling alle elementaire differentiaal van de orde $\leq 2m - 2$ voort. We beschouwen nu in (2.14) alle termen die θ_{m-1} en B_{m-1} bevatten. Bij ontwikkeling brengt de term

$$(2.15) \quad \frac{1}{2} h_n^3 \theta_{m-1} f_{j_1 j_2} B_{m-1}^{j_1 j_2}$$

alle elementaire differentiaal, beginnend met $f_{j_1 j_2}$, van de orde $2m - 1$ en $2m$ voort. Immers $B_{m-1}^{j_1}$ produceert alle elementaire differentiaal van de orde r_i met $1 \leq r_i \leq 2m - 2$ en de orde r van de elementaire differentiaal (2.15) wordt bepaald door $r = 1 + r_1 + r_2$. Deze redenering gaat op voor alle termen met θ_{m-1} en B_{m-1} , behalve voor de term

$$h_n^2 \theta_{m-1} f_{j_1} B_{m-1}^{j_1}$$

die slechts de elementaire differentiaal, beginnend met f_{j_1} , tot en met orde $2m - 1$ voortbrengt. Het is nu echter gemakkelijk na te gaan dat al deze elementaire differentiaal van de orde $2m$, wel voorkomen in de ontwikkeling van de termen met θ_{m-1}' .

We hebben nu aangetoond dat alle elementaire differentiaal tot en met de orde $2m$ die voorkomen in de Taylorontwikkeling van (2.8) ook voorkomen in de ontwikkeling van het m -punts gegeneraliseerde Runge-Kutta schema. Hiermee is lemma 1 bewezen. $\square$

We bewijzen nu met lemma 2 dat schema (2.1) niet de consistentie-orde $2m + 1$ kan hebben.

Lemma 2. In de ontwikkeling van het gegeneraliseerde Runge-Kutta schema (2.1) ontbreekt een elementaire differentiaal $f_{j_1 j_2} F_{j_1}^{j_1} f^{j_2}$ van de orde $2m + 1$.

Bewijs lemma 2. Het lemma is juist voor $m = 1$. Immers, als $\theta_1 = 0$, dan komt in formule (2.12) de term $f_{j_1 j_2} F_{j_1}^{j_1} f^{j_2}$ niet voor. We veronderstellen nu dat het lemma juist is voor het $(m-1)$ -punts schema, dus de term $f_{k\ell} F_{k\ell}^{k\ell} f^{j_2}$ van de orde $2m - 1$ ontbreekt.

In het m -punts schema bekijken we nu de elementaire differentiaal

$$f_{j_1 j_2} f_{k\ell}^{j_1} F_{k\ell}^{k\ell} f^{j_2}$$

van de orde $2m + 1$. Uit (2.14) blijkt dat deze alleen uit

$$f_{j_1 j_2} B_{m-1}^{j_1} B_{m-1}^{j_2}$$

voortgebracht kan worden. B_{m-1} heeft de structuur van een $(m-1)$ -punts schema en kan volgens de inductie veronderstelling niet $f_{k\ell}^k f^{\ell}$ voortbrengen. Dit betekent dat

$$f_{j_1 j_2}^{j_1} f_{k\ell}^{k\ell} f^{j_2}$$

niet voorkomt in de ontwikkeling van het m -punts schema.

Hiermee zijn lemma 2 en stelling 2.2 bewezen. $\square$

Gevolg. Uit het bewijs van stelling 2.2 volgt nu dat het m -punts schema (2.1) slechts een orde van consistentie $2m$ kan hebben als θ_{m-1} niet een constante is.

2.3. Stabiliteit

Bij het onderzoek naar de stabiliteitseigenschappen van methode (2.3) is het voldoende als we ons beperken tot de stabiliteit van de modelvergelijking $y' = \delta y$, $\delta \in \mathbb{C}$. Schema (2.3) wordt in dit geval herleid tot

$$(2.16) \quad y_{n+1} = R(h_n \delta) y_n,$$

met

$$(2.17) \quad R(z) = 1 + z[\theta_0(z) + \theta_1(z)] + z^2 \theta_1(z) \Lambda(z).$$

$R(z)$ heet de stabiliteitsfunctie van (2.3). Voor expliciete gegeneraliseerde Runge-Kutta methoden zijn θ_0 , θ_1 en Λ polynomen in z en is $R(z)$ diengevolge ook een polynoom. Voor semi-impliciete methoden is $R(z)$ een rationale functie. Stellen we daarom

$$R(z) = \frac{\sum_{j=0}^{m_1} \beta_j z^j}{\sum_{j=0}^{m_2} \alpha_j z^j}, \quad \alpha_0 = 1,$$

dan is een oplossing van de niet-lineaire vergelijkingen (2.4) - (2.7)

(zie Van der Houwen [1973]):

$$\begin{aligned} \Theta_0(z) &= \frac{R(z)-1}{z} - \frac{(1 + \frac{3}{4}z + \frac{9}{32}z^2)(\frac{16}{27} + \frac{4}{27}(4\alpha_1+1)z)}{1 + \alpha_1 z + \alpha_2 z^2 + \dots + \alpha_{m_2} z^{m_2}}, \\ (2.18) \quad \Theta_1(z) &= \frac{\frac{16}{27} + \frac{4}{27}(4\alpha_1+1)z}{1 + \alpha_1 z + \alpha_2 z^2 + \dots + \alpha_{m_2} z^{m_2}}, \\ \Lambda(z) &= \frac{3}{4} + \frac{9}{32} z, \end{aligned}$$

waarin $z = h_n J_n$.

De stabiliteitsfunctie $R(z)$ kan in (2.18) nog vrij gekozen worden afhankelijk van de gewenste eigenschappen van de methode. Om de methode geschikt te maken voor stijve differentiaalvergelijkingen, kiezen we in dit verslag een stabiliteitsfunctie die:

- a) consistent is van de orde 4,
- b) exponentiële aanpassing mogelijk maakt in één reëel punt,
- c) A-stabiël is.

We zullen nu achtereenvolgens deze eisen behandelen.

a) Consistentie

De stabiliteitsfunctie $R(z)$ noemen we consistent van de orde p als

$$\frac{d^j}{dz^j} R(z) \Big|_{z=0} = \frac{d^j}{dz^j} e^z \Big|_{z=0} = 1, \quad j=0,1,2,\dots,p.$$

Uit (2.16) blijkt dat de consistentie-orde van de integratiemethode niet hoger kan zijn dan de consistentie-orde van de stabiliteitsfunctie. In het geval de orden van consistentie gelijk zijn, noemen we de stabiliteitsfunctie compatibel met de integratiemethode.

In schema (2.18) kiezen we nu voor $R(z)$

$$(2.19) \quad R(z) = \frac{1 + \frac{1}{2}(12\alpha_3+1)z + \frac{1}{12}(24\alpha_3+1)z^2}{1 + \frac{1}{2}(12\alpha_3-1)z + \frac{1}{12}(1-48\alpha_3)z^2 + \alpha_3 z^3}.$$

Deze functie is consistent van de orde 4 en is dus compatibel met (2.18). Uit (2.18) krijgen we nu de 2-punts 4^e orde semi-impliciete Runge-Kutta methode:

$$(2.20) \quad \begin{aligned} \theta_0(z) &= \frac{\frac{11}{27} + \frac{2}{27}(33\alpha_3-4)z - \frac{1}{18}(1+66\alpha_3)z^2 + \frac{1}{24}(1-24\alpha_3)z^3}{1 + \frac{1}{2}(12\alpha_3-1)z + \frac{1}{12}(1-48\alpha_3)z^2 + \alpha_3 z^3}, \\ \theta_1(z) &= \frac{\frac{16}{27} + \frac{4}{27}(24\alpha_3-1)z}{1 + \frac{1}{2}(12\alpha_3-1)z + \frac{1}{12}(1-48\alpha_3)z^2 + \alpha_3 z^3}, \end{aligned}$$

$$\Lambda(z) = \frac{3}{4} + \frac{9}{32} z.$$

b) Exponentiële aanpassing

De vrije parameter α_3 in (2.19) kan gebruikt worden om de stabiliteitsfunctie exponentieel aan te passen. Als δ_0 het reële gedeelte van de eigenwaarde van de Jacobiaan is met de grootste modulus, dan passen we $R(z)$ exponentieel aan in $z_0 = h_n \delta_0$ door de stabiliteitsfunctie in z_0 gelijk te maken aan de analytische stabiliteitsfunctie:

$$R(z_0) = e^{z_0}.$$

Hieruit volgt voor α_3

$$(2.21) \quad \alpha_3 = \frac{1}{12z_0} \frac{e^{z_0}(z_0^2-6z_0+12) - (z_0^2+6z_0+12)}{-e^{z_0}(z_0^2-4z_0+6) + 2z_0 + 6}.$$

c) A-stabiliteit

We definiëren het stabiliteitsgebied S van $R(z)$ als

$$S = \{z \mid |R(z)| < 1, z \in \mathbb{C}\}.$$

Methode (2.3) heet stabiel voor een stelsel differentiaalvergelijkingen als voor alle eigenwaarden δ van de Jacobiaan met negatief reëel deel $\operatorname{Re} \delta$ behoort tot gebied S . Indien S het hele linker halfvlak omvat dan noemen we de methode *A-stabiel* (Dahlquist [1963]). Uit het maximummodulus theorema volgt dat de stabiliteitsfunctie (2.19) A-stabiel is, als voldaan is aan:

- (1) $|R(z)| \leq 1$ voor $\operatorname{Re}(z) = 0$,
- (2) $\lim_{|z| \rightarrow \infty} |R(z)| \leq 1$ voor $\operatorname{Re}(z) \leq 0$,
- (3) $R(z)$ heeft geen polen met negatief reëel deel.

Het is eenvoudig na te gaan dat voor alle α_3 voldaan is aan (1) terwijl uit (2.19) blijkt dat $R(z) \rightarrow 0$ voor $|z| \rightarrow \infty$ (asymptotische stabiliteit). Om voorwaarde (3) te verifiëren berekenen we numeriek de wortels van de noemer van $R(z)$:

$$1 + \frac{1}{2}(12\alpha_3 - 1)z + \frac{1}{12}(1 - 48\alpha_3)z^2 + \alpha_3 z^3 = 0.$$

Voor $-\frac{1}{2} < \alpha_3 < \frac{1}{2}$ zijn de reële wortel en het reële en imaginaire deel van de complexe wortels getekend in figuur 2.1.

Uit de figuur blijkt dat voor $\alpha_3 < 0$ alle wortels in het rechter halfvlak liggen. Met behulp van (2.21) kunnen we afleiden dat bij exponentiële aanpassing in $z_0 \in (-\infty, 0]$, α_3 ligt in het interval $(-\frac{1}{24}, -\frac{1}{60}]$. Hiermee is voorwaarde (3) bewezen.

We hebben nu aangetoond dat methode (2.20) A-stabiel is, mits de stabiliteitsfunctie exponentieel aangepast wordt in een reëel punt z_0 , $z_0 \leq 0$. In figuur 2.2 zijn de krommen $R(z) = 1$ getekend voor $\alpha_3 = -\frac{1}{24}$ (exponentiële aanpassing in $-\infty$) en $\alpha_3 = -\frac{1}{60}$ (aangepassing in 0).

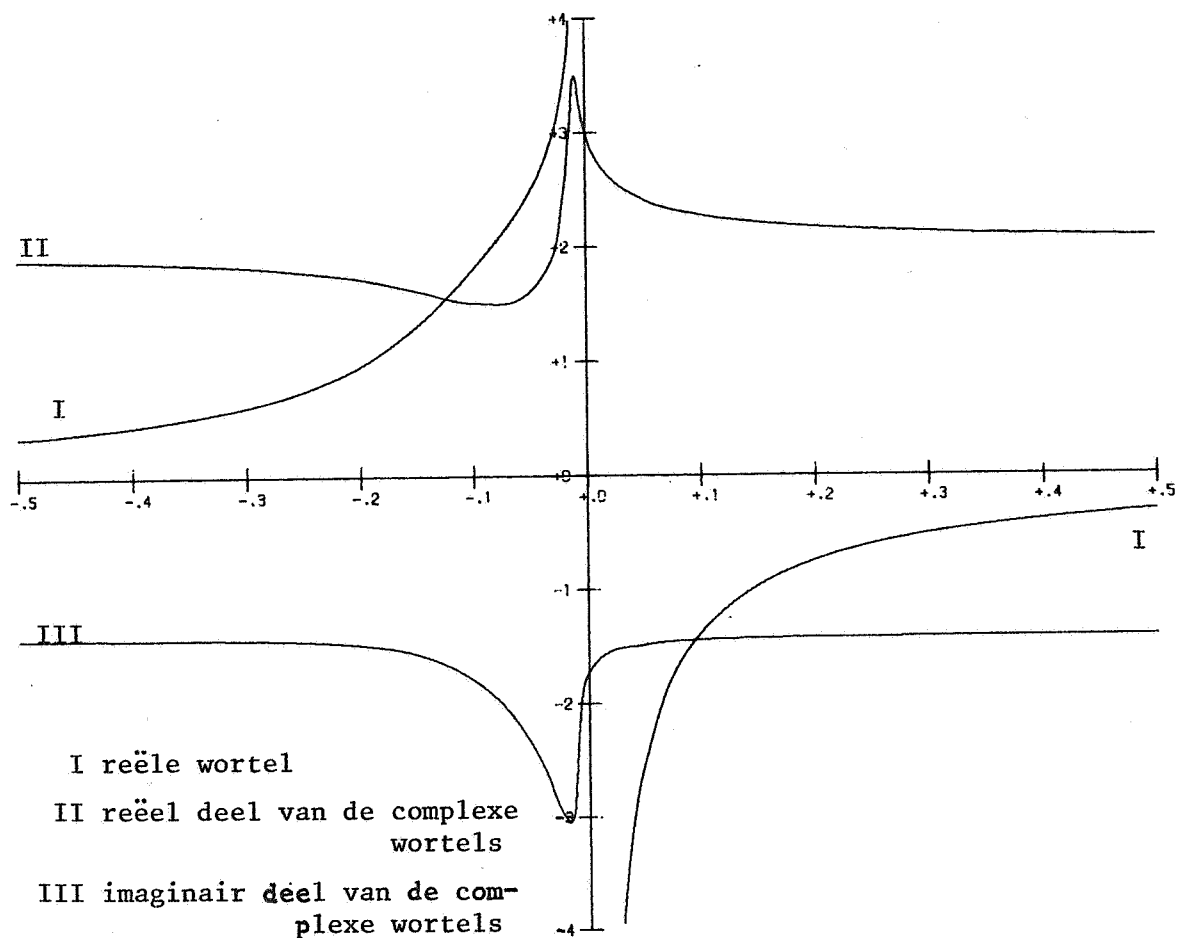


Fig. 2.1

Reële en complexe wortels van $1 + \frac{1}{2}(12\alpha_3 - 1)z + \frac{1}{12}(1 - 48\alpha_3)z^2 + \alpha_3 z^3 = 0$.

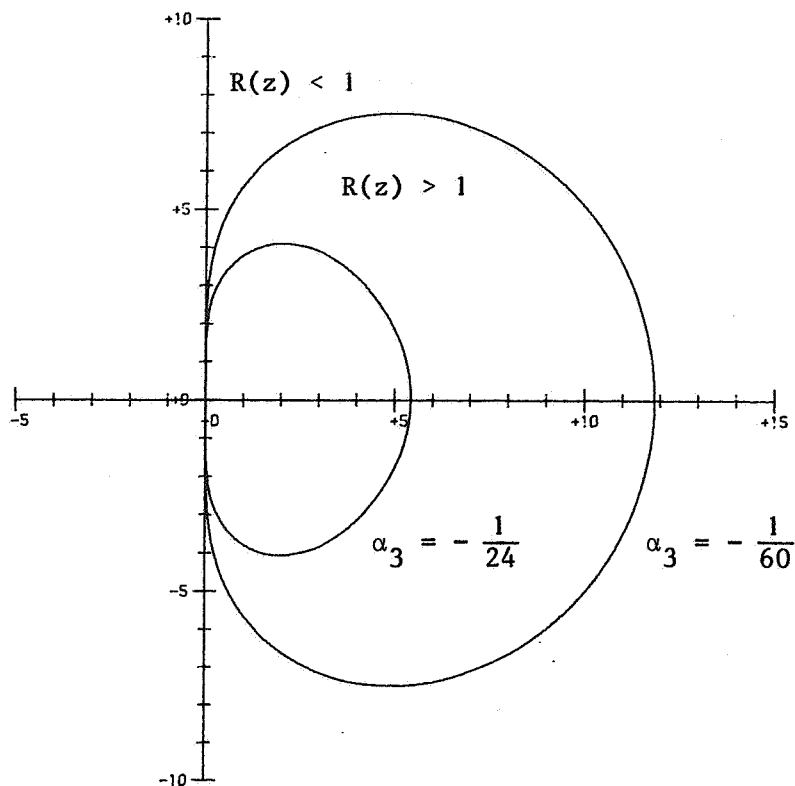


Fig. 2.2

Gebied met $R(z) < 1$ en $R(z) > 1$ voor $\alpha_3 = -\frac{1}{24}$ en $\alpha_3 = -\frac{1}{60}$.

2.4. Stapkeuzestrategie

De grootte van de integratiestap bij de A-stabiele methode uit de vorige paragraaf zal niet beperkt worden door stabiliteitseisen. De stabiliteitseigenschappen hebben we echter afgeleid voor lineaire of lokaal ge-lineariseerde differentiaalvergelijkingen en de methode zal daarom gevoelig zijn voor sterk niet-lineaire differentiaalvergelijkingen. De strategie die de grootte van de integratiestap bepaalt willen we om deze reden niet baseren op de nauwkeurigheid van de oplossing maar op de niet-lineariteit van het stelsel. Daartoe berekenen we een referentie-oplossing $\tilde{y}_{n+1}$ van de vorm

$$(2.22) \quad \tilde{y}_{n+1} = y_n + v_0 \frac{1}{N(z)} h_n f(y_n) + v_1 \frac{\Lambda(z)}{N(z)} h_n f(y_n) + \\ + v_2 \frac{1}{N(z)} h_n f(y_n + \Lambda(z) h_n f(y_n)) + v_3 h_n f(y_{n+1}),$$

met stabiliteitsfunctie

$$\tilde{R}(z) = 1 + v_0 \frac{z}{N(z)} + v_1 \frac{\Lambda(z)z}{N(z)} + v_2 \frac{z[1+\Lambda(z)z]}{N(z)} + v_3 z R(z).$$

$N(z)$ is de noemer van de stabiliteitsfunctie (2.19). Eisen we nu dat

$$(2.23) \quad \tilde{R}(z) = R(z),$$

dan zal voor lineaire differentiaalvergelijkingen de referentie-oplossing $\tilde{y}_{n+1}$ gelijk zijn aan y_{n+1} . Voor niet-lineaire differentiaalvergelijkingen is $\|\tilde{y}_{n+1} - y_{n+1}\|$ een maat voor de niet-lineariteit van het stelsel. De eis (2.23) leidt tot het stelsel lineaire vergelijkingen:

$$v_0 + \frac{3}{4} v_1 + v_2 + v_3 = 1, \\ \frac{9}{32} v_1 + \frac{3}{4} v_2 + \frac{1}{2} (12\alpha_3 + 1) v_3 = 6\alpha_3, \\ \frac{9}{32} v_2 + \frac{1}{12} (24\alpha_3 + 1) v_3 = -\alpha_3.$$

Stellen we nu $v_2 = 0$ dan is dit stelsel equivalent met

$$v_3 = \frac{-12\alpha_3}{24\alpha_3 + 1}, \\ v_1 = \frac{64\alpha_3(12\alpha_3 + \frac{2}{3})}{24\alpha_3 + 1}, \\ v_0 = 1 - \frac{3}{4} v_1 - v_3.$$

De referentie-oplossing (2.22) ligt nu vast. Omdat de stabiliteitsfunctie van $\tilde{y}_{n+1}$ voldoet aan (2.23) is $\tilde{y}_{n+1}$ minstens 2^e orde exact en er geldt

$$\|\tilde{y}_{n+1} - y_{n+1}\| = c_n h_n^3.$$

Veronderstellen we nu dat $c_n \approx c_{n-1}$ dan wordt de nieuwe integratiestap h_n bepaald door

$$(2.24) \quad h_n = \sqrt[3]{\frac{\text{tol}}{c_n}} \approx h_{n-1} \sqrt[3]{\frac{\text{tol}}{\|\tilde{y}_n - y_n\|}},$$

met tol een vereiste lokale tolerantie.

3. DE PROCEDURE EFFOSIRK

In dit hoofdstuk bespreken we de ALGOL 60 procedure effosirk (exponentially fitted, fourth order, semi-implicit Runge-Kutta) geprogrammeerd volgens het schema uit het vorige hoofdstuk.

3.1. Heading en parameters

De heading van de procedure is:

```
procedure effosirk (x, xe, m, y, delta, derivative, jacobian, j, n, aeta,
                    reta, hmin, hmax, linear, output);
```

```
value m; integer m,n;
```

```
real x, xe, delta, aeta, reta, hmin, hmax;
```

```
procedure derivative, jacobian, output;
```

```
Boolean linear;
```

```
array y,j;
```

De betekenis van de formele parameters is:

x: <variable>;

de onafhankelijke variabele x;

bij aanroep van de procedure moet x de beginwaarde x_0 van de

onafhankelijke variabele hebben; na uitvoering van de procedure heeft x de waarde van xe;

xe: <arithmetic expression>;
de eindwaarde van x;

m : <arithmetic expression>;
het aantal differentiaalvergelijkingen van het op te lossen stelsel;

y : <array identifier>;
een één-dimensionaal array y[1:m];
bij aanroep van effosirk moet y de startwaarde $y(x_0)$ hebben;
gedurende het integratieproces heeft y de waarden van de berekende oplossing in het punt x;

delta: <arithmetic expression>;
het reële gedeelte van het punt waarin exponentiële aanpassing is gewenst;
alternatieven:
delta = (een schatting van) het reële gedeelte van de in absolute waarde grootste eigenwaarde van de Jacobiaan van het stelsel;
delta = 0, teneinde 5^e orde consistentie te verkrijgen bij lineaire of bijna lineaire vergelijkingen;
delta = $-\infty$, teneinde asymptotische stabiliteit te verkrijgen;

derivative:<procedure identifier>;
procedure derivative (a); array (a);
als in effosirk derivative wordt aangeroepen bevat a de waarden van array y; na uitvoering van derivative moet a de waarden van f(y) bevatten; omdat de differentiaalvergelijking in autonome vorm moet zijn, mag de variabele x niet gebruikt worden in derivative;

jacobian : <procedure identifier>;
procedure jacobian (j,y); array j,y;
na uitvoering van jacobian moet array j de waarden van de Jacobiaan van f(y) bevatten: $j[k,\ell] = \frac{\partial}{\partial y^\ell} f^k$;

j : <array identifier>;

```

    een twee-dimensionaal array j [1:m,1:m];
    j is een hulparray dat gebruikt wordt in de procedure
    jacobian;
n :    <variable>;
       n telt het aantal integratiestappen;
aeta, reta:<arithmetic expression>;
       de vereiste absolute en relatieve lokale tolerantie;
hmin, hmax:<arithmetic expression>;
       minimale en maximale waarde van de integratiestap;
linear: <Boolean expression>;
       als linear = true wordt de procedure jacobian alleen in de
       eerste stap geëvalueerd en de integratie wordt uitgevoerd met
       een stapgrootte hmax; de overeenkomstige vermindering in reken-
       tijd kan gebruikt worden in geval van lineaire of bijna li-
       neaire differentiaalvergelijkingen;
output: <procedure identifier>;
       procedure output;
       in output kan men uitvoer vragen van bijvoorbeeld x, y, j en n;
       bovendien kan men in output de waarden van xe, aeta, reta, hmin,
       hmax en linear veranderen tijdens het integratieproces, zonder
       dat effosirk opnieuw gestart behoeft te worden.

```

3.2. De procedure body

De complete ALGOL 60 tekst van de procedure is:

```

procedure effosirk(x, xe, m, y, delta, derivative, jacobian, j,
                   n, aeta, reta, hmin, hmax, linear, output);
value m; integer m, n;
procedure derivative, jacobian, output;
real x, xe, delta, aeta, reta, hmin, hmax;
Boolean linear;
array y, j;

```

```

begin real step, h, mu0, mu1, mu2, mu3, mu10, mu11,
  labda1, labda2, labda3, nu1, nu2, nu3;
array j1[1 : m, 1 : m], f, f0, f1, f2[1 : m];
integer array p[1 : m];
integer k, l;
Boolean lin;

real procedure stepsize;
begin real s, discr, eta, c;
  if linear then s:= h:= hmax else
  if hmin = hmax v n = 1 then s:= h:= hmin else
  begin eta:= aeta + reta × sqrt(vecvec(1, m, 0, y, y));
    c:= nu3 × step; for k:= 1 step 1 until m do
      f2[k]:= f2[k] + c × f[k] - y[k];
    discr:= sqrt(vecvec(1, m, 0, f2, f2));
    s:= h:= (eta / (0.75 × (eta + discr)) + 0.33) × h;
    if h < hmin then s:= h:= hmin else
    if h > hmax then s:= h:= hmax
  end;
  if x + s > xe then s:= xe - x;
  lin:= step = s ^ linear; stepsize:= s
end stepsize;

procedure coefficient;
begin real z1, ex, alpha3, a, b, c, d;
  own real z2;
  z1:= step × delta; if n = 1 then z2:= z1 + z1;
  if abs(z2 - z1) > 10-3 × abs(z1) v z1 > - 1 then
  begin if abs(z1) < 0.075 then
    begin c:= z1 × (z1 × 71 / 350 - .1) + 1; d:= - 60
    end else
    if z1 < - 1010 then
    begin c:= 1; d:= - 24 end else
    if z1 < - 30 then
    begin c:= - z1 × (z1 + 6) - 12;
      d:= (z1 + z1 + 6) × z1 × 12
    end else
    begin ex:= exp(z1); a:= z1 × z1; b:= 6 × z1;
      c:= (a - b + 12) × ex - (a + b + 12);
      d:= -(a - 4 × z1 + 6) × ex + z1 + z1 + 6) × z1 × 12
    end;
    alpha3:= c / d; a:= 66 × alpha3; mu0:= 11 / 27;
    mu1:= (a - 8) / 27; mu2:= - (a + 1) / 18;
    mu3:= - alpha3 + 1 / 24; mu10:= 16 / 27;
    mu11:= (96 × alpha3 - 4) / 27;
    labda1:= 6 × alpha3 - 0.5;
    labda2:= (1 - 48 × alpha3) / 12; labda3:= alpha3;
    a:= 24 × alpha3 + 1; b:= 12 × alpha3; nu3:= - b / a;
    nu2:= (b + 2 / 3) × 64 × alpha3 / a;
    nu1:= 1 - 0.75 × nu2 - nu3;
    z2:= z1
  end
end coefficient;

```

```

procedure difference scheme;
begin real h2, h3, h4, a0, a1, a2, a3, b0, b1, b2, b3,
    fk, f0k;
    derivative(f); step:= stepsize;
    if  $\neg$  linear v n = 1 then jacobian(j, y);
    h2:= step  $\times$  step; h3:= h2  $\times$  step; h4:= h3  $\times$  step;
    if  $\neg$  lin then
    begin coefficient;
        a1:= labda1  $\times$  step; a2:= labda2  $\times$  h2; a3:= labda3  $\times$  h3;
        for k:= 1 step 1 until m do
            begin for l:= 1 step 1 until m do
                j1[k, l]:= matmat(1, m, k, l, j, j);
                for l:= 1 step 1 until m do
                    f0[l]:= matmat(1, m, k, l, j1, j);
                    for l:= 1 step 1 until m do
                        j1[k, l]:= a1  $\times$  j[k, l] + a2  $\times$  j1[k, l] + a3  $\times$  f0[l];
                        j1[k, k]:= j1[k, k] + 1
                    end;
                det(j1, m, p)
            end;
            for k:= 1 step 1 until m do
                f0[k]:= matvec(1, m, k, j, f);
                for k:= 1 step 1 until m do
                    f1[k]:= matvec(1, m, k, j, f0);
                a0:= step  $\times$  0.75; a1:= h2  $\times$  9 / 32; a3:= step  $\times$  nu1;
                b0:= step  $\times$  mu0; b1:= h2  $\times$  mu1; b2:= h3  $\times$  mu2;
                b3:= h4  $\times$  mu3;
                for k:= 1 step 1 until m do
                    begin fk:= f[k]; f0k:= f0[k]; a2:= a0  $\times$  fk + a1  $\times$  f0k;
                        f[k]:= y[k] + a2;
                        f0[k]:= b0  $\times$  fk + b1  $\times$  f0k + b2  $\times$  f1[k] +
                            b3  $\times$  matvec(1, m, k, j, f1);
                        f2[k]:= a3  $\times$  fk + nu2  $\times$  a2
                    end;
                    derivative(f);
                    a0:= step  $\times$  mu10; a1:= h2  $\times$  mu11;
                    for k:= 1 step 1 until m do
                        f1[k]:= a0  $\times$  f[k] + a1  $\times$  matvec(1, m, k, j, f) + f0[k];
                    sol(j1, m, p, f1);
                    if  $\neg$  linear  $\wedge$  hmin  $\neq$  hmax then sol(j1, m, p, f2);
                    for k:= 1 step 1 until m do
                        begin a0:= y[k]; y[k]:= f[k]:= a0 + f1[k];
                            f2[k]:= a0 + f2[k]
                        end
                    end
                end difference scheme;

                for k:= 1 step 1 until m do f[k]:= y[k];
                n:= 0; output; step:= 0;
            next step: n:= n + 1;
            difference scheme; x:= x + step; output;
            if x < xe then goto next step
        end effosirk;

```

3.3. De procedure stepsize

Deze procedure berekent de nieuwe integratiestap met behulp van een benadering van formule (2.24):

$$h_n = h_{n-1} \left(\frac{\frac{4}{3} \text{tol}}{\text{tol} + \|y_n - y_{n-1}\|_2} + \frac{1}{3} \right),$$

waarin $\| \cdot \|_2$ de Euclidische norm aanduidt. De tolerantie tol wordt bepaald uit de door de gebruiker op te geven absolute en relative lokale tolerantie:

$$\text{tol} = \text{aeta} + \text{reta} \|y_n\|_2.$$

3.4. De procedure coefficient

In de procedure coefficient wordt de nog vrije parameter α_3 in schema (2.20) bepaald door de methode exponentieel aan te passen in

$z_1 = h_n \times \text{delta}$, met delta de door de gebruiker op te geven geschatte eigenwaarde van de Jacobiaan. Voor $|z_1| \rightarrow 0$ gaat formule (2.21) over in

$$\alpha_3 = -\frac{1}{60} \left(1 - \frac{1}{10} z_1 + \frac{71}{350} z_1^2 + \dots \right).$$

Deze betrekking wordt gebruikt voor $|z_1| < 0.075$. Voor $z_1 < -30$ is $\exp(z_1) < 10^{-12}$ en wordt in plaats van (2.21)

$$\alpha_3 = \frac{-(z_1^2 + 6z_1 + 12)}{12z_1(2z_1 + 6)}$$

berekend.

3.5. De procedure difference scheme

In difference scheme wordt de numerieke oplossing y_{n+1} berekend volgens schema (2.20). Indien `linear = true` wordt de Jacobiaan van het stelsel

alleen berekend in de eerste integratiestap. Een nieuwe LU-ontbinding van de noemer

$$I + \frac{1}{2}(12\alpha_3 - 1)h_n J_n + \frac{1}{12}(1 - 48\alpha_3)h_n^2 J_n^2 + \alpha_3 h_n^3 J_n^3$$

wordt alleen bepaald als $\text{linear} = \text{false}$ of $h_n \neq h_{n-1}$. In difference scheme wordt tevens het eerste gedeelte van de referentie-oplossing (2.22) berekend.

3.6. Bibliotheekprocedures

De procedure `effosirk` maakt gebruik van de volgende bibliotheekprocedures: `vecvec`, `matmat`, `matvec`, `det` en `sol` (zie Hemker [1973]).

4. TESTRESULTATEN

In dit hoofdstuk bespreken we vier problemen waarmee `effosirk` is getest: een lineair stelsel differentiaalvergelijkingen; een niet-lineair stelsel; een stelsel met een positieve eigenwaarde en een eenvoudige, niet-stijve, 2^e orde vergelijking.

4.1. Een lineair stelsel differentiaalvergelijkingen

Fowler & Warten [1967] geven het gekoppelde stelsel stijve differentiaalvergelijkingen

$$(4.1) \quad y' = \begin{pmatrix} -500.5 & 499.5 \\ 499.5 & -500.5 \end{pmatrix} y + \begin{pmatrix} 2 \\ 2 \end{pmatrix}, \quad y(0) = \begin{pmatrix} -0.1 \\ 0.1 \end{pmatrix}.$$

De analytische oplossing van (4.1) luidt:

$$y(x) = 2(1 - e^{-x}) \begin{pmatrix} 1 \\ 1 \end{pmatrix} + e^{-1000x} \begin{pmatrix} -0.1 \\ 0.1 \end{pmatrix}.$$

De Jacobiaan van het stelsel heeft de eigenwaarden -1000 en -1. Het probleem is opgelost met de volgende waarden van de parameters:

$x_e = 1, 2, 10,$
 $\text{delta} = -1000,$
 $h_{\min} = h_{\max} = 10, 5, 1, 0.5, 0.1, 0.05, 0.01,$
 $\text{linear} = \text{true}.$

In tabel 4.1 geven we voor de verschillende stapgrootten een overzicht van het aantal goede cijfers van de oplossing ($= -^{10}\log |\text{relatieve fout}|$).

Tabel 4.1

Aantal goede cijfers voor probleem (4.1) opgelost met effosirk

$\begin{matrix} h \\ x \end{matrix}$	10	5	1	0.5	0.1	0.05	0.01
1			2.6	3.9	6.7	7.8	10.6
2			3.1	4.7	7.1	8.2	11.0
10	0.8	2.7	6.1	7.5	10.2	10.7	10.8

Probleem (4.1) is ook getest met de analoge semi-impliciete Runge-Kutta methode efsirk. Deze methode is niet 4^e orde maar 3^e orde exact en is beschreven in Beentjes e.a. [1973]. Een overzicht van deze resultaten geven we in tabel 4.2.

Tabel 4.2

Aantal goede cijfers voor probleem (4.1) opgelost met efsirk

$\begin{array}{c} h \\ x \end{array}$	10	5	1	0.5	0.1	0.05	0.01
1			2.2	3.0	5.1	6.0	8.3
2			2.4	3.3	5.4	6.3	8.6
10	1.0	2.2	5.3	6.2	8.2	9.1	11.7

Bij vergelijking van tabel 4.1 en 4.2 zien we dat voor dit lineaire probleem de hogere orde van efsirk tot betere resultaten leidt.

4.2. Een niet-lineair stelsel differentiaalvergelijkingen

Gear [1968] geeft het volgende stelsel niet-lineaire differentiaalvergelijkingen:

$$(4.2) \quad \begin{cases} y_1' = -1000 y_1 (y_1 + y_2 - 1.999987), \\ y_2' = -2500 y_1 (y_1 + y_2 - 2), \end{cases} \quad y(0) = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Van deze vergelijking is geen analytische oplossing bekend en we vergelijken daarom de berekende oplossing in $x = 50$ met de referentie-oplossing:

$$y_1(50) = 0.59765 \ 46988,$$

$$y_2(50) = 1.40234 \ 34075.$$

De eigenwaarden δ_1 en δ_2 van de Jacobiaan van het stelsel zijn in $x = 0$:

$$\delta_1 \approx -3500,$$

$$\delta_2 \approx 0, \quad \delta_2 < 0,$$

en in $x = 50$:

$$\delta_1 \approx -4100,$$

$$\delta_2 \approx 0, \quad \delta_2 < 0.$$

Het stelsel werd geïntegreerd over het interval $(0,50)$ met de volgende waarden van de parameters:

$$\text{delta} = \delta_1 = 0.5(j_{11} + j_{22} - \sqrt{(j_{11} - j_{22})^2 + 4j_{12}j_{21}}), \quad j_{kl} = \frac{\partial}{\partial y_l} y_k',$$

$$\text{aeta} = \text{reta} = 10^{-2}, 10^{-3}, \dots, 10^{-9},$$

$$\text{hmin} = 0.0005,$$

$$\text{hmax} = 0.3,$$

$$\text{linear} = \underline{\text{false}}.$$

In figuur 4.1 zijn bij de verschillende toleranties het aantal goede cijfers van $y_1(50)$ aangegeven. De 2^e component y_2 gaf steeds een iets beter resultaat. In de figuur is bovendien het aantal integratiestappen uitgezet.

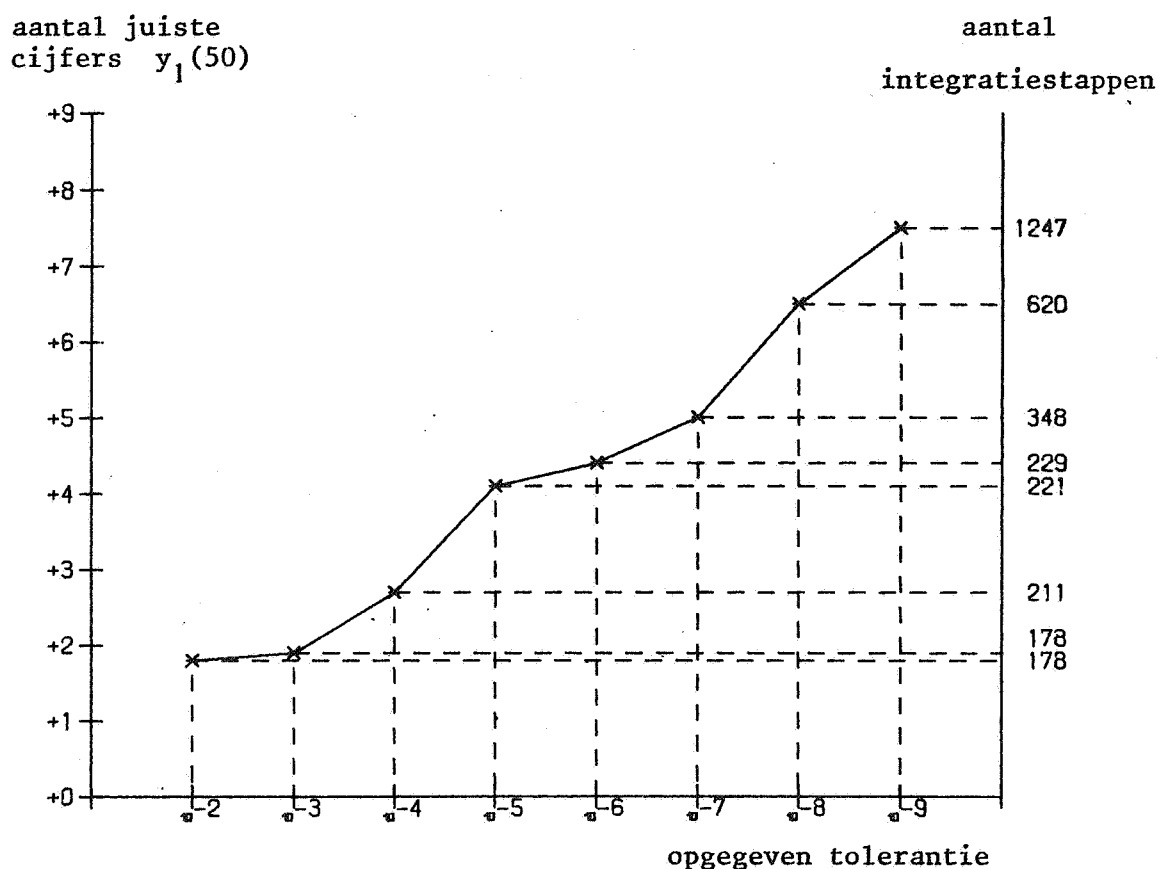


Fig. 4.1

Aantal goede cijfers van $y_1(50)$ en aantal integratiestappen voor probleem (4.2) opgelost met effosirk.

Probleem (4.2) is ook opgelost met de 3^e orde semi-impliciete methode efsirk, waarvan de resultaten uitgezet zijn in figuur 4.2. Vergelijken we beide methoden, dan zien we dat voor deze niet-lineaire differentiaalvergelijking effosirk slechtere resultaten geeft dan efsirk, in tegenstelling tot de resultaten van het lineaire stelsel differentiaalvergelijkingen (4.1). Alleen voor een kleine stapgrootte, c.q. $aeta = reta = 10^{-9}$ zijn de resultaten vergelijkbaar.

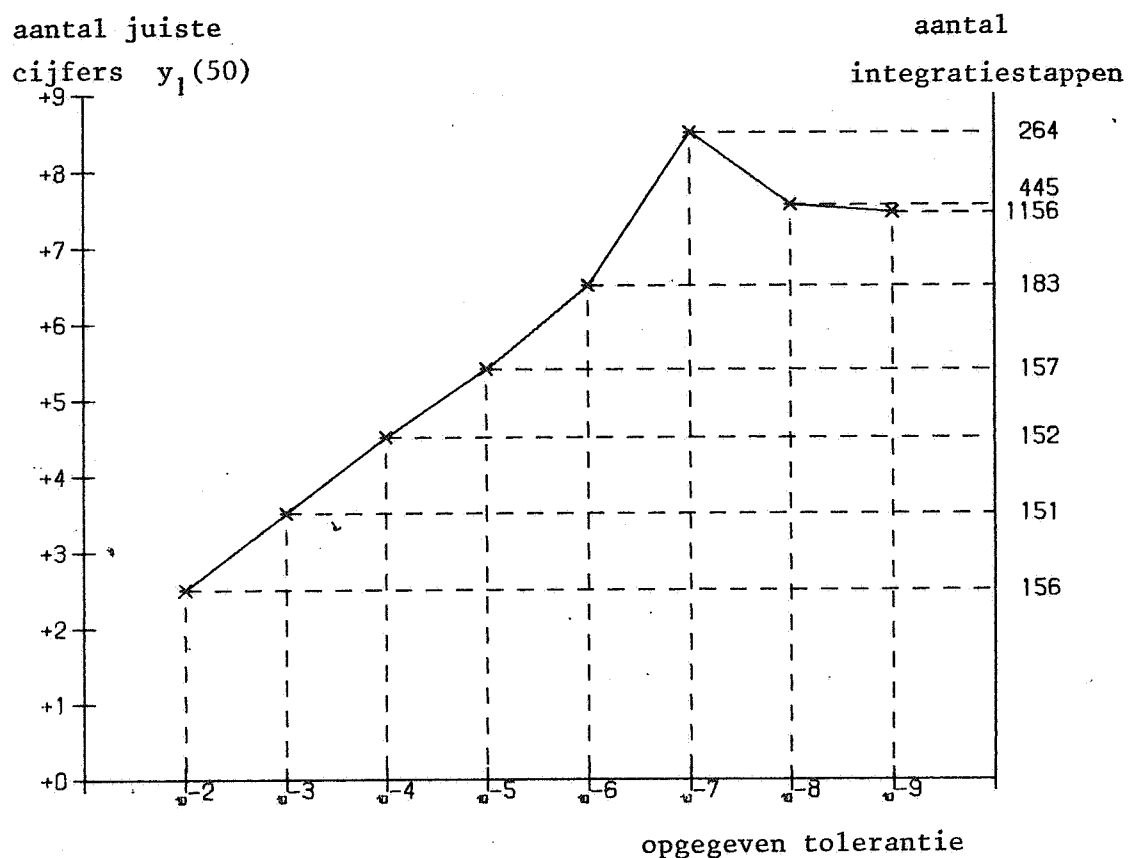


Fig. 4.2

Aantal goede cijfers en aantal integratiestappen van $y_1(50)$ voor probleem (4.2) opgelost met efsirk.

4.3. Een stelsel stijve differentiaalvergelijkingen met een positieve eigenwaarde

Krogh [1973] behandelt het volgende stelsel differentiaalvergelijkingen.

Definieer

$$z_i' = \beta_i z_i + z_i^2, \quad z_i(0) = -1, \quad i=1,2,3,4,$$

met β_i constant en ongelijk nul. De analytische oplossing van dit stelsel is

$$z_i = \frac{\beta_i}{1 - (1 + \beta_i)e^{\beta_i x}}.$$

We definiëren de vector $y = Uz$, $z = (z_1, z_2, z_3, z_4)^T$ en U de unitaire matrix

$$U = \frac{1}{2} \begin{pmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{pmatrix}.$$

We lossen nu op het beginwaardeprobleem

$$(4.3) \quad y' = Ay + Uw, \quad y(0) = (-1, -1, -1, -1)^T,$$

met $A = U \operatorname{diag}(-\beta_i) U$ en $w = (z_1^2, z_2^2, z_3^2, z_4^2)^T$. De Jacobiaan van het stelsel is $A + U \operatorname{diag}(2z_i) U$ en heeft dus de eigenwaarden $2z_i - \beta_i$.

Gear [1971] lost dit probleem op met $\beta_1 = 1000$, $\beta_2 = 800$, $\beta_3 = -10$ en $\beta_4 = 0.0001$. Bij deze keuze zijn de eigenwaarden in $x = 0$: -1002 , -802 , $+8$ en -2.001 . Voor $x \gg 1000$ zijn de eigenwaarden: -1000 , -800 , -10 en -0.0001 .

Probleem (4.3) lossen we op met effosirk met de volgende waarden van de parameters:

$$\text{delta} = 2z_1 - \beta_1 \approx -1000,$$

$$\text{hmin} = 0.0001,$$

$$\text{hmax} = 20,$$

$$\text{linear} = \underline{\text{false}}.$$

In tabel 4.3 geven we in het eerste integratiepunt na $x = 0.01, 0.1, 1, 10, 100$ en 1000 de maximale relatieve fout in de componenten van y , het aantal integratiestappen, functie-evaluaties, Jacobiaanevaluaties en de

gemiddelde stapgrootte over het gehele interval. De resultaten zijn verkregen door integratie met een tolerantie 10^{-3} . Het is duidelijk dat in de beginfase de stapgrootte sterk beperkt wordt door de stijve componenten en de positieve eigenwaarde. In het eerste integratie-interval bijvoorbeeld, is de gemiddelde stapgrootte 0.0016 terwijl op (100,1000) de gemiddelde stap 15.46 bedraagt.

In tabel 4.4 geven we de resultaten die Gear bereikte met zijn multistep methode DIFSUB (Gear [1971], p.219). Bij een vergelijkbare nauwkeurigheid lost effosirk probleem (4.3) op met een kleiner aantal integratiestappen en functie-evaluaties. Het aantal matrix-evaluaties met Gear's methode is echter kleiner dan het aantal Jacobiaan-evaluaties met effosirk.

Tabel 4.3

Resultaten van effosirk voor probleem (4.3)

maximale fout	stappen	functie- evaluaties	Jacobiaan- evaluaties	gemiddelde stap	x
$0.1842 \cdot 10^{-4}$	9	18	9	$0.1667 \cdot 10^{-2}$	0.015
$0.3216 \cdot 10^{-5}$	15	30	15	$0.7533 \cdot 10^{-2}$	0.113
$0.4887 \cdot 10^{-5}$	41	82	41	$0.2578 \cdot 10^{-1}$	1.057
$0.2202 \cdot 10^{-6}$	61	122	61	$0.1703 \cdot 10^0$	10.391
$0.4813 \cdot 10^{-6}$	87	174	87	$0.1158 \cdot 10^1$	100.750
$0.3152 \cdot 10^{-5}$	146	292	146	$0.6938 \cdot 10^1$	1012.896

Tabel 4.4

Resultaten van DIFSUB voor probleem (4.3)

maximale fout	stappen	functie- evaluaties	matrix- inversies	gemiddelde stap	x
0.9100 10^{-7}	70	179	7	0.1463 10^{-3}	0.010
0.2667 10^{-5}	110	262	12	0.9535 10^{-3}	0.105
0.2208 10^{-5}	168	405	15	0.6025 10^{-2}	1.012
0.2870 10^{-5}	216	523	20	0.4635 10^{-1}	10.011
0.2984 10^{-5}	252	616	25	0.4067 10^0	102.477
0.1199 10^{-5}	283	693	29	0.3625 10^1	1025.777

We hebben probleem (4.3) ook opgelost met verschillende toleranties. Figuur 4.3 toont het minimum aantal goede cijfers in de componenten van y in $x = 100$. Vergelijken we dit met figuur 4.4 met de resultaten verkregen met de 3^e orde semi-impliciete methode efsirk, dan zien we dat efsirk een betere precisie bereikt in een kleiner aantal stappen.

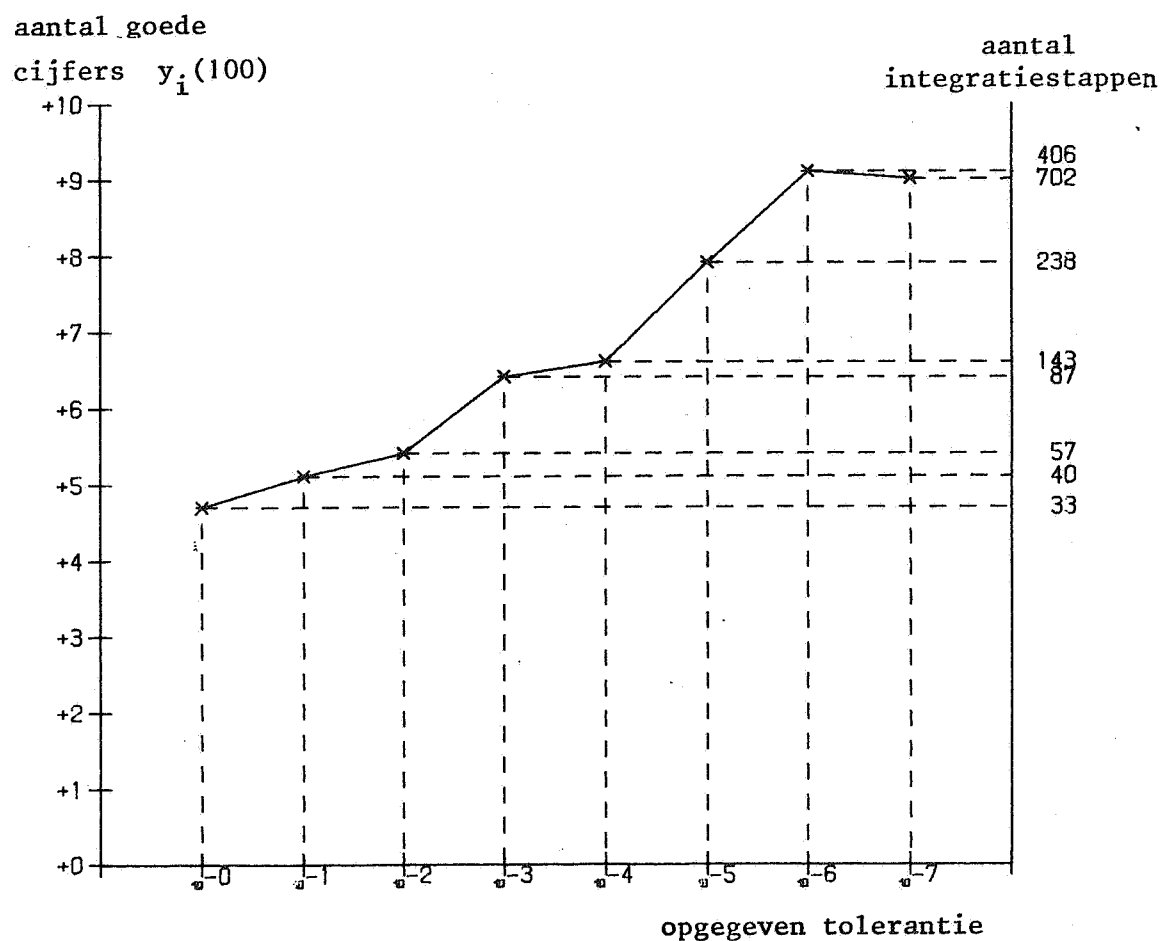


Fig. 4.3

Minimum aantal goede cijfers van $y_i(100)$, $i=1,2,3,4$, en aantal integratiestappen voor probleem (4.3) opgelost met effosirk.

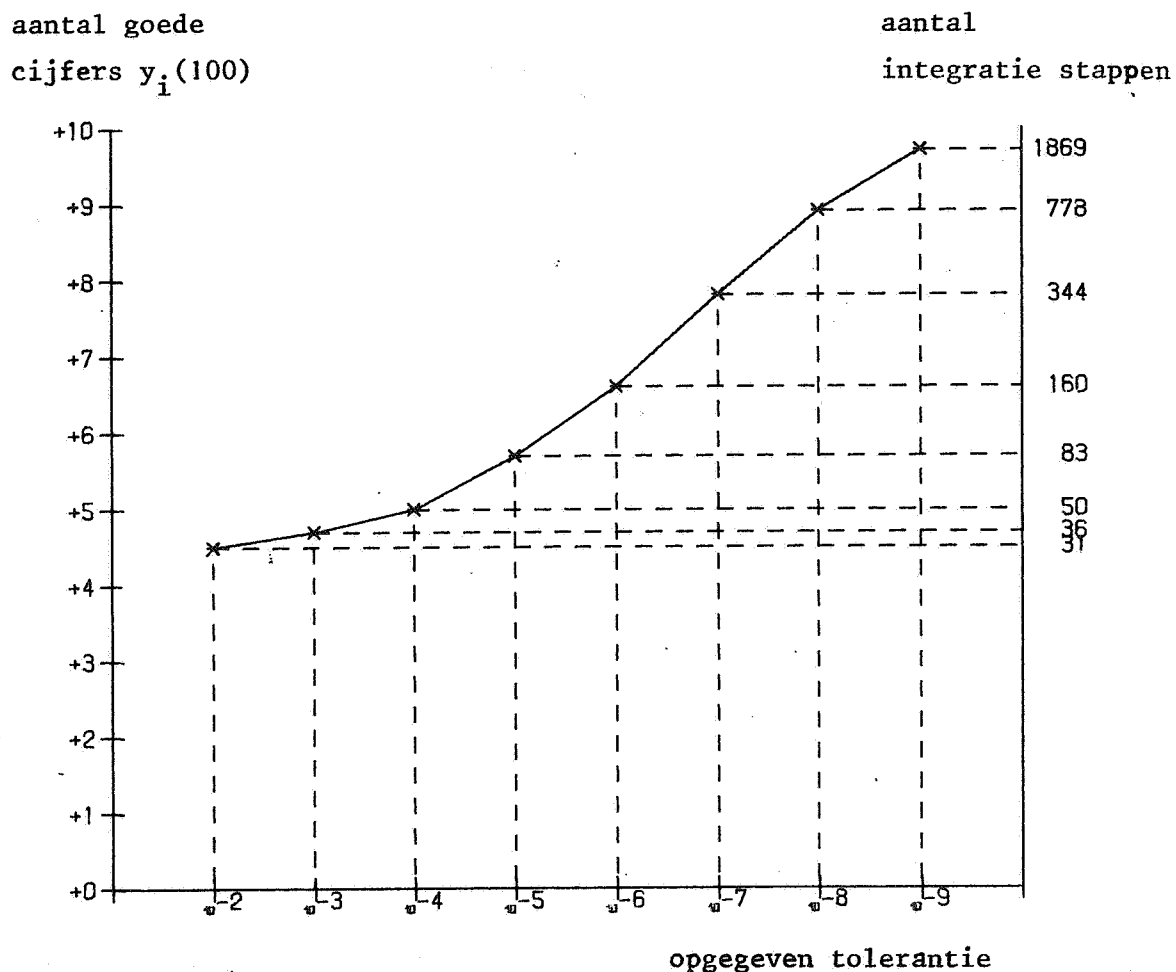


Fig. 4.4

Minimum aantal goede cijfers van $y_i(100)$, $i=1,2,3,4$ en aantal integratiestappen voor probleem (4.3) opgelost met efsirk.

4.4. Een eenvoudige, niet-stijve, 2^e orde vergelijking

We beschouwen de niet-autonome 2^e orde differentiaalvergelijking

$$(4.4) \quad y'' = -y + x, \quad y(0) = 0, \quad y'(0) = 2,$$

met analytische oplossing

$$y(x) = \sin(x) + x.$$

Dit stelsel kunnen we in de vorm (1.1) brengen door te definiëren

$$y_1 = y ,$$

$$y_2 = y' ,$$

$$y_3 = x .$$

Hierdoor gaat (4.4) over in het autonome stelsel 1^e orde vergelijkingen

$$(4.5) \quad \begin{cases} y_1' = y_2 , \\ y_2' = -y_1 + y_3 \\ y_3' = 1 , \end{cases} \quad , \quad y(0) = \begin{pmatrix} 0 \\ 2 \\ 0 \end{pmatrix} .$$

De Jacobiaan van dit stelsel heeft de eigenwaarden i , $-i$ en 0 . Daarom is het de beste strategie de methode exponentieel aan te passen in 0 . Voor dit lineaire stelsel wordt effosirk hierdoor 5^e orde nauwkeurig en heeft dienvolgende een foutterm $y_n - y_n(x)$ van de vorm $ch^6 y^{(6)}(x)$. Deze term is voor y_1 gelijk aan $ch^6 \sin(x)$ en is dus 0 voor $x = k\pi$. De foutterm voor de 2^e component is 0 voor $x = (k+\frac{1}{2})\pi$. In deze punten zullen de resultaten sterk positief beïnvloed worden.

Om deze reden kiezen we als eindpunt van het integratie-interval $\pi/4$. Het probleem is opgelost met de volgende waarden van de parameters

$$xe = \pi/4 ,$$

$$\text{delta} = 0 ,$$

$$hmin = hmax = xe, \quad xe/2, \quad xe/5, \quad xe/10, \quad xe/25, \quad xe/100 ,$$

$$\text{linear} = \underline{\text{true}} .$$

Voor stelsel (4.5) geven we ook de procedures derivative en jacobian zoals die meegegeven moeten worden in effosirk.

```

procedure derivative (y); array y;
begin real y1; y1 := y [1];
      y [1] := y [2];
      y [2] := - y1 + y [3];
      y [3] := 1
end derivative;

```

```

procedure jacobian (j,y); array j,y;
begin
      j [1,1] := 0; j [1,2] :=1; j [1,3] := 0;
      j [2,1] :=-1; j [2,2] :=0; j [2,3] := 1;
      j [3,1] := j [3,2] := j [3,3] = 0
end jacobian;

```

In tabel 4.5 geven we het aantal goede cijfers in y van de oplossing van (4.4) met effosirk. Voor de volledigheid geven we in tabel 4.6 ook de resultaten van efsirk.

Tabel 4.5

Aantal goede cijfers in y voor probleem (4.4) opgelost met effosirk

$\begin{array}{c} \backslash \\ x \end{array} \begin{array}{c} h \end{array}$	$\frac{\pi}{4}$	$\frac{\pi}{8}$	$\frac{\pi}{20}$	$\frac{\pi}{40}$	$\frac{\pi}{100}$	$\frac{\pi}{200}$
$\frac{\pi}{4}$	4.8	6.3	8.3	9.8	11.3	11.3

Tabel 4.6

Aantal goede cijfers in y voor probleem (4.4) opgelost met efsirk

$\begin{array}{c} \backslash \\ x \end{array} \begin{array}{c} h \end{array}$	$\frac{\pi}{4}$	$\frac{\pi}{8}$	$\frac{\pi}{20}$	$\frac{\pi}{40}$	$\frac{\pi}{100}$	$\frac{\pi}{200}$
$\frac{\pi}{4}$	3.7	4.9	6.5	7.7	9.3	10.5

5. CONCLUSIE

Het blijkt mogelijk bij de in sectie 2 afgeleide 2-punts semi-impliciete Runge-Kutta methode, de stabiliteitsfunctie zo te kiezen dat de methode dezelfde gunstige eigenschappen krijgt als een gewone impliciete Runge-Kutta methode. De methode is tevens speciaal geschikt gemaakt voor het oplossen van stijve differentiaalvergelijkingen.

Uit de testvoorbeelden blijkt dat de gekozen methode goede resultaten bereikt; vooral als de differentiaalvergelijking lineair of bijna lineair is. Bij sterk niet-lineaire vergelijkingen zal de nauwkeurigheid van de oplossing slechter worden en kan bij grote integratiestappen de methode zelfs een instabiel gedrag gaan vertonen. In dit geval is het raadzaam de analoge 3^e orde methode efsirk te gebruiken (zie Beentjes e.a. [1973] en Hemker [1973]).

REFERENTIES

- Beentjes, P.A., K. Dekker, H.C. Hemker, S.P.N. van Kampen en G.M. Willems [1973], *Colloquium Stijve differentiaalvergelijkingen*, deel 2, MC Syllabus 15.2, Mathematisch Centrum, Amsterdam.
- Butcher, J.C. [1963], *Coefficients for the study of Runge-Kutta integration processes*. J.Austral. Math. Soc., 3 (1963) 185.
- Butcher, J.C. [1964], *Implicit Runge-Kutta processes*. Math. Comp., 18 (1964) 50.
- Calahan, D.A. [1968], *A stable, accurate method of numerical integration for non-linear systems*. Proc. IEEE 56 (1968) 744.
- Dahlquist, G.G. [1963], *A special stability problem for linear multistep problems*. BIT, 3 (1963), 27.
- Fowler, M.E. & R.M. Warten [1967], *A numerical integration technique for ordinary differential equations with widely separated eigenvalues*, IBM Journal, 537-543.
- Gear, C.W. [1968], *The automatic integration of stiff ordinary differential equations*. Proc. of IFIP Congress, Vol. 1, p.187.
- Gear, C.W. [1971], *Numerical initial value problems in ordinary differential equations*. Prentice-Hall inc., Englewood Cliffs, New Jersey.
- Hemker, P.W. [1973], *NUMAL, a library of numerical procedures in ALGOL 60, index and KWIC index*. Mathematical Centre Report NW 8/73, Amsterdam.
- Houwen, P.J. van der [1973], *One-step methods with adaptive stability functions for the integration of differential equations*. Numerische, insbesondere approximations theoretische Behandlung von Funktionalgleichungen, 164 Lecture Notes in Mathematics 333, Springer Verlag, Heidelberg.
- Lapidus, L. & J.H. Seinfeld [1971], *Numerical solution of ordinary differential equations*. Academic Press, New York.

- Rosenbrock, H.H. [1963], *Some general implicit processes for the numerical solution of differential equations*. Comput. J., 5 (1963) 329.
- Krogh, F.T. [1973], *On testing a subroutine for the numerical integration of ordinary differential equations*. J. Assoc. Comput. Mach., 20 (1973) 545.

