

Numerieke methoden
voor stelsels
gewone differentiaalvergelijkingen

Prof. Dr. Marnix Van Daele

Deel III

Runge–Kutta-methoden

Hoofdstuk 10

Impliciete en semi-impliciete methoden

10.1 Inleiding

Zoals reeds opgemerkt in paragraaf 7.1 zijn IRKMn en zelfs DIRKMn, omwille van de noodzaak om niet-lineaire stelsels op te lossen, zeer kostelijk om te implementeren. Niettegenstaande ze doorgaans aanleiding geven tot methoden van hogere orde dan ERKMn, blijft hun gebruik hoofdzakelijk beperkt tot zgn. *stijve* problemen, omdat in die omstandigheden hun interessante stabiliteitseigenschappen de hogere implementatiekostprijs rechtvaardigen. In dit hoofdstuk zullen we een aantal welbekende IRKMn en DIRKMn nader bestuderen.

Als de RKM (7.2) toegepast wordt op het scalaire probleem $y' = f(x)$, dan krijgen we als resultaat een kwadratuurformule met als knooppunten $x_n + c_j h$ en als gewichten b_j , $j = 1, 2, \dots, s$:

$$\int_{x_n}^{x_{n+1}} f(x) dx \approx y_{n+1} - y_n = h \sum_{j=1}^s b_j f(x_n + c_j h) . \quad (10.1)$$

Vaak worden IRKMn genoemd naar de klasse van kwadratuurformules waarop ze terugvallen als we $f(x, y) = f(x)$ kiezen. In de volgende paragrafen zullen we op een aantal van deze klassen dieper ingaan. Eerst echter zullen we dieper ingaan op de constructie van impliciete methoden.

10.2 Vereenvoudigende veronderstellingen

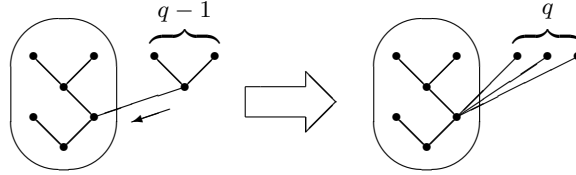
Het oplossen van het niet-lineaire stelsel dat volgt uit de ordevoorwaarden lijkt helemaal niet zo eenvoudig. Toch bestaan er elegante manieren om oplossingen te vinden. De basis hiertoe

wordt gevormd door enkele *vereenvoudigende veronderstellingen* :

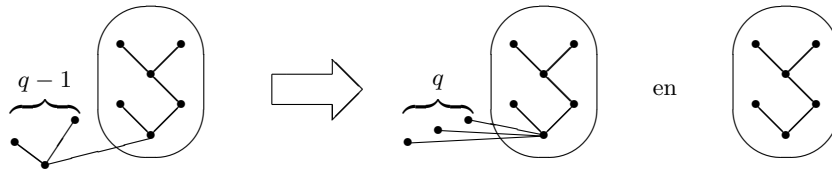
$$C(\eta) : \sum_{j=1}^s a_{ij} c_j^{q-1} = \frac{c_i^q}{q}, \quad i = 1, 2, \dots, s, \quad q = 1, 2, \dots, \eta, \quad (10.2)$$

$$D(\xi) : \sum_{i=1}^s b_i c_i^{q-1} a_{ij} = \frac{b_j}{q} (1 - c_j^q), \quad j = 1, 2, \dots, s, \quad q = 1, 2, \dots, \xi. \quad (10.3)$$

Deze veronderstellingen zorgen voor een reductie in de ordevoorwaarden voor bomen zoals getoond in Figuur 10.1 en Figuur 10.2.



Figuur 10.1: Reductie met $C(q)$



Figuur 10.2: Reductie met $D(q)$

Het resultaat van de reductie is uiteindelijk dat indien η en ξ groot genoeg zijn alle ordevoorwaarden herleiden tot de kwadratuurvoorwaarde

$$B(p) : \sum_{i=1}^s b_i c_i^{q-1} = \frac{1}{q}, \quad q = 1, 2, \dots, p. \quad (10.4)$$

Van fundamenteel belang is de volgende stelling, die bewezen werd door Butcher :

Stelling 10.2.1 *Als $C(\eta)$, $D(\xi)$ en $B(p)$ geldig zijn voor $p \leq \xi + \eta + 1$ en $p \leq 2\eta + 2$, dan is de methode van orde p . \square*

Het bewijs van deze stelling steunt o.a. op het volgende lemma.

Lemma 1 *Zij c_1, c_2, \dots, c_s allemaal verschillend. Dan volgt $D(\nu)$ uit $C(s)$ en $B(s + \nu)$. \square*

Bewijs. Stel

$$d_j^{(q)} = \sum_{i=1}^s b_i c_i^{q-1} a_{ij} - \frac{b_j}{q} (1 - c_j^q).$$

Uit de voorwaarden $C(s)$ en $B(s + \nu)$ volgt voor $q = 1, 2, \dots, \nu$ en $k = 1, 2, \dots, s$:

$$\begin{aligned} \sum_{j=1}^s c_j^{k-1} d_j^{(q)} &= \sum_{i=1}^s b_i c_i^{q-1} \sum_{j=1}^s a_{ij} c_j^{k-1} - \frac{1}{q} \sum_{j=1}^s b_j (c_j^{k-1} - c_j^{q+k-1}) \\ &= \frac{1}{k} \sum_{i=1}^s b_i c_i^{q+k-1} - \frac{1}{q} \left(\frac{1}{k} - \frac{1}{q+k} \right) \\ &= \frac{1}{k(q+k)} - \frac{1}{qk} + \frac{1}{q(q+k)} \\ &= 0. \end{aligned}$$

Dit betekent dat voor elke q de vector $(d_1^{(q)}, d_2^{(q)}, \dots, d_s^{(q)})$ de oplossing is van een homogeen stelsel van s lineaire vergelijkingen, waarvan de determinant van het Vandermonde type is. Deze vector moet dus noodzakelijk de nulvector zijn, hetgeen $D(\nu)$ aantoont. ■

10.3 Gauss–Legendre-methoden of Kuntzmann–Butcher-methoden

De eerste klasse van IRKMn zijn de Gauss–Legendre-methoden. Kuntzmann en Butcher toonden aan dat de s -traps IRKMn van dit type de hoogst mogelijke orde bezitten, nl. $2s$. Butcher vertrok van de keuze $p = 2s$ en $\eta = s$. Uit Lemma 1 volgt dan dat $D(\xi)$ geldt met $\xi = s$, zodat de voorwaarden van Stelling 10.2.1 voldaan zijn.

De keuze $p = 2s$ legt vast dat de b_i - en de c_i -waarden corresponderen met de gewichten en knooppunten van de Gauss–Legendre-veelterm $P_s(t)$. Deze veeltermen worden recursief gedefinieerd door de formule van Rodrigues :

$$P_n(t) = \frac{(-1)^n}{2^n n!} \frac{d^n}{dt^n} [(1-t^2)^n], \quad n \geq 1 \text{ en } P_0(t) = 1. \quad (10.5)$$

Stelling 10.3.1 *Als $f(t) \in C^{2n}[-1, +1]$ dan is*

$$\int_{-1}^1 f(t) dt = \sum_{j=1}^n w_j f(t_j) + \frac{2^{2n+1} (n!)^4}{[(2n)!]^3 (2n+1)} f^{(2n)}(\zeta), \quad (10.6)$$

met $-1 < \zeta < 1$ en waarbij t_j de j -de wortel is van $P_n(t) = 0$ en waarbij

$$w_j = \frac{2}{n} \frac{1}{P'_n(t_j) P_{n-1}(t_j)}. \quad (10.7)$$

□

Om nu de integraal die optreedt in (10.1) te benaderen, dienen we d.m.v. een verandering van onafhankelijke veranderlijke over te gaan naar het interval $[x_n, x_{n+1}]$. Dit kan door de transformatie

$$x = \frac{1}{2} [(x_{n+1} - x_n) t + x_{n+1} + x_n] = x_n + \frac{h}{2} (1 + t). \quad (10.8)$$

Dit introducerend in (10.1) en rekening houdend met (10.6) levert :

$$\int_{x_n}^{x_{n+1}} f(x) dx = \frac{h}{2} \int_{-1}^1 f(x_n + \frac{h}{2}(1+t)) dt \approx h \sum_{j=1}^n \frac{w_j}{2} f(x_n + \frac{h}{2}(1+t_j)),$$

waarbij

$$b_j = \frac{w_j}{2} \quad \text{en} \quad c_j = \frac{1}{2}(1+t_j). \quad (10.9)$$

Laten we enkele specifieke s -traps methoden van het Gauss-Legendre-type onderzoeken.

Geval $s = 1$

Uit (10.5) volgt dat $P_1(t) = t$, zodat $t_1 = 0$ en derhalve $c_1 = \frac{1}{2}$. Bovendien is wegens (10.7) en (10.9)

$$b_1 = \frac{w_1}{2} = \frac{1}{P_1'(t_1)P_0(t_1)} = 1.$$

Met die waarden voor b_1 en c_1 is duidelijk voldaan aan de ordevoorwaarden (9.2), (1) en (2), zodat we kunnen vaststellen dat de 1-traps methode

$$\begin{array}{c|c} \frac{1}{2} & \frac{1}{2} \\ \hline & 1 \end{array}$$

orde $p = 2$ bezit. Hierbij is $a_{11} = c_1$ wegens de rijsum-voorwaarde die uitgedrukt wordt door (10.2) met $\eta = 1$.

Geval $s = 2$

Uit (10.5) volgt dat $P_2(t) = \frac{1}{2}(3t^2 - 1)$, waaruit o.a. $t_1 = -t_2 = -\frac{1}{\sqrt{3}}$, zodat

$$c_1 = \frac{1}{2} - \frac{\sqrt{3}}{6} \quad \text{en} \quad c_2 = \frac{1}{2} + \frac{\sqrt{3}}{6}.$$

Bovendien is dan

$$w_j = \frac{1}{P_2'(t_j)P_1(t_j)} = \frac{1}{3t_j^2} = 1 \quad \text{voor } j = 1, 2,$$

waaruit $b_1 = b_2 = \frac{1}{2}$.

De coëfficiënten a_{ij} worden bepaald uit de voorwaarde $C(\eta)$ met $\eta = 2$. Men vindt bvb. voor $i = 1$:

$$\begin{cases} a_{11} + a_{12} = c_1 \\ a_{11}c_1 + a_{12}c_2 = \frac{c_1^2}{2} \end{cases}$$

waaruit

$$\begin{cases} a_{11} = \frac{c_1^2 - 2c_1c_2}{2(c_1 - c_2)} = \frac{1}{4} \\ a_{12} = \frac{c_1^2}{2(c_1 - c_2)} = \frac{1}{4} - \frac{\sqrt{3}}{6} \end{cases}$$

Uiteindelijk bekomt men de volgende Butcher-matrix :

$$\begin{array}{c|cc} \frac{1}{2} - \frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\ \frac{1}{2} + \frac{\sqrt{3}}{6} & \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

Geval $s = 3$

Uit (10.5) volgt dat $P_3(t) = \frac{1}{2}(5t^3 - 3t)$ waaruit

$$t_1 = -\sqrt{\frac{3}{5}} = -t_3, \quad t_2 = 0.$$

zodat

$$c_1 = \frac{5 - \sqrt{15}}{10}, \quad c_2 = \frac{1}{2}, \quad c_3 = \frac{5 + \sqrt{15}}{10}.$$

Bovendien is dan

$$w_j = \frac{2}{3P_3'(t_j)P_2(t_j)} = \frac{8}{9(5t_j^2 - 1)(3t_j^2 - 1)},$$

waaruit

$$w_1 = w_3 = \frac{5}{9}, \quad w_2 = \frac{8}{9},$$

of

$$b_1 = b_3 = \frac{5}{18}, \quad b_2 = \frac{4}{9}.$$

De a_{ij} factoren ($i, j = 1, 2, 3$) volgen dan uit het oplossen van $C(\eta)$ voor $\eta = 3$. De volgende Butcher-matrix wordt dan gevonden :

$$\begin{array}{c|ccc} \frac{1}{2} - \frac{\sqrt{15}}{10} & \frac{5}{36} & \frac{2}{9} - \frac{\sqrt{15}}{15} & \frac{5}{36} - \frac{\sqrt{15}}{30} \\ \frac{1}{2} & \frac{5}{36} + \frac{\sqrt{15}}{24} & \frac{2}{9} & \frac{5}{36} - \frac{\sqrt{15}}{24} \\ \frac{1}{2} + \frac{\sqrt{15}}{10} & \frac{5}{36} + \frac{\sqrt{15}}{30} & \frac{2}{9} + \frac{\sqrt{15}}{15} & \frac{5}{36} \\ \hline & \frac{5}{18} & \frac{4}{9} & \frac{5}{18} \end{array}$$

Geval $s = 4$

Op analoge wijze is het mogelijk 4-traps en 5-traps methoden op te bouwen. Steeds kan men controleren dat de orde $p = 2s$ is. Voor verder gebruik geven we in beknopte vorm de Butcher-matrix voor $s = 4$ en $p = 8$:

$$\begin{array}{c|cccc}
 \frac{1}{2} - \omega_2 & & \omega_1 & \omega'_1 - \omega_3 + \omega'_4 & \omega'_1 - \omega_3 - \omega'_4 & \omega_1 - \omega_5 \\
 \frac{1}{2} - \omega'_2 & \omega_1 - \omega'_3 + \omega_4 & & \omega'_1 & \omega'_1 - \omega'_5 & \omega_1 - \omega'_3 - \omega_4 \\
 \frac{1}{2} + \omega'_2 & \omega_1 + \omega'_3 + \omega_4 & & \omega'_1 + \omega'_5 & \omega'_1 & \omega_1 + \omega'_3 - \omega_4 \\
 \frac{1}{2} + \omega_2 & & \omega_1 + \omega_5 & \omega'_1 + \omega_3 + \omega'_4 & \omega'_1 + \omega_3 - \omega'_4 & \omega_1 \\
 \hline
 & & 2\omega_1 & 2\omega'_1 & 2\omega'_1 & 2\omega_1
 \end{array}$$

met

$$\begin{aligned}
 \omega_1 &= \frac{1}{8} - \frac{\sqrt{30}}{144}, & \omega'_1 &= \frac{1}{8} + \frac{\sqrt{30}}{144}, \\
 \omega_2 &= \frac{1}{2} \sqrt{\frac{15 + 2\sqrt{30}}{35}}, & \omega'_2 &= \frac{1}{2} \sqrt{\frac{15 - 2\sqrt{30}}{35}}, \\
 \omega_3 &= \omega_2 \left(\frac{1}{6} + \frac{\sqrt{30}}{24} \right), & \omega'_3 &= \omega'_2 \left(\frac{1}{6} - \frac{\sqrt{30}}{24} \right), \\
 \omega_4 &= \omega_2 \left(\frac{1}{21} + 5 \frac{\sqrt{30}}{168} \right), & \omega'_4 &= \omega'_2 \left(\frac{1}{21} - 5 \frac{\sqrt{30}}{168} \right), \\
 \omega_5 &= \omega_2 - 2\omega_3, & \omega'_5 &= \omega'_2 - 2\omega'_3.
 \end{aligned}$$

Opmerking 10.3.1

Vermits $c_j = \frac{1}{2}(1 + t_j)$, waarbij de t_j nulpunten zijn van $P_s(t)$, m.a.w.

$$\frac{d^s}{dt^s} [(1 - t^2)^s] = 0,$$

worden de c_j geleverd als wortels van $P_s(2c - 1)$, d.w.z. c_1, c_2, \dots, c_s zijn de oplossingen van

$$\frac{d^s}{dc^s} [(1 - (2c - 1)^2)^s] = \frac{d^s}{dc^s} [c^s (1 - c)^s] = 0.$$

□

Opmerking 10.3.2

Men kan de 1-traps Gauss–Legendre-methode schrijven als

$$\begin{aligned}
 y_{n+1} &= y_n + h k_1 \\
 &= y_n + h f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2} k_1\right) \\
 &= y_n + h f\left(x_n + \frac{h}{2}, y_n + \frac{1}{2}(y_{n+1} - y_n)\right) \\
 &= y_n + h f\left(x_n + \frac{h}{2}, \frac{1}{2}(y_n + y_{n+1})\right),
 \end{aligned}$$

in welke vorm de methode gekend is als de *impliciete middelpuntformule*. □

In het begin van deze paragraaf werd opgemerkt dat de s -traps Gauss–Legendre-methoden, die steunen op de nulpunten van de Legendre veelterm $P_s(2t - 1)$, de hoogst mogelijk bereikbare orde bezitten. Zoals nog zal blijken in volgende hoofdstukken, is het ook interessant methoden te beschouwen van orde $2s - 1$ en $2s - 2$. Algemeen kan men voor deze methoden aantonen :

Stelling 10.3.2 *Heeft een IRKM orde $2s - 1$, dan zijn c_1, c_2, \dots, c_s nulpunten van de veelterm*

$$P_s(2t - 1) + \lambda P_{s-1}(2t - 1), \quad \lambda \in \mathbb{R}. \quad (10.10)$$

Bezit een IRKM orde $2s - 2$, dan zijn c_1, c_2, \dots, c_s nulpunten van de veelterm

$$P_s(2t - 1) + \mu P_{s-1}(2t - 1) + \nu P_{s-2}(2t - 1), \quad \mu, \nu \in \mathbb{R}. \quad (10.11)$$

□

In de volgende paragrafen zullen we enkele bijzondere gevallen van deze methoden bespreken. Eerst zullen we echter aandacht besteden aan zgn. *reflecties*.

10.4 Reflecties of toegevoegde methoden

Beschouw de RKM (7.5) :

$$\begin{cases} y_{n+1} = y_n + h \sum_{i=1}^s b_i f(x_n + c_i h, Y_i), \\ Y_i = y_n + h \sum_{j=1}^s a_{ij} f(x_n + c_j h, Y_j), \quad i = 1, 2, \dots, s \end{cases} \quad (10.12)$$

De methode die y_n uitdrukt in termen van y_{n+1} is dan

$$\begin{aligned} y_n &= y_{n+1} - h \sum_{i=1}^s b_i f(x_n + c_i h, Y_i) \\ &= y_{n+1} - h \sum_{i=1}^s b_i f(x_{n+1} + (1 - c_i)(-h), Y_i) \end{aligned} \quad (10.13)$$

en

$$Y_i = y_{n+1} - h \sum_{j=1}^s (b_j - a_{ij}) f(x_{n+1} + (1 - c_j)(-h), Y_j), \quad i = 1, 2, \dots, s. \quad (10.14)$$

Vervangen we nu h door $-h$, dan is de corresponderende Butcher-matrix :

$$\begin{array}{c|cccc} 1 - c_1 & b_1 - a_{11} & b_2 - a_{12} & \cdots & b_s - a_{1s} \\ 1 - c_2 & b_1 - a_{21} & b_2 - a_{22} & \cdots & b_s - a_{2s} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 - c_s & b_1 - a_{s1} & b_2 - a_{s2} & \cdots & b_s - a_{ss} \\ \hline & b_1 & b_2 & \cdots & b_s \end{array}$$

Vermits we de gewoonte aannemen de knooppunten in niet-dalende volgorde te noteren, luidt deze matrix :

$$\begin{array}{c|cccc}
 1 - c_s & b_s - a_{ss} & b_{s-1} - a_{s s-1} & \cdots & b_1 - a_{s1} \\
 1 - c_{s-1} & b_s - a_{s-1 s} & b_{s-1} - a_{s-1 s-1} & \cdots & b_1 - a_{s-1 1} \\
 \vdots & \vdots & \vdots & \vdots & \vdots \\
 1 - c_1 & b_s - a_{1s} & b_{s-1} - a_{1 s-1} & \cdots & b_1 - a_{11} \\
 \hline
 & b_s & b_{s-1} & \cdots & b_1
 \end{array}$$

De methode die met deze Butcher-matrix overeenkomt is meteen de *reflectie (toegevoegde)* van de methode gegeven door (10.12).

10.5 Radau-methoden

Een tweede klasse van IRKMn die steunt op kwadratuurformules met niet-equidistante punten valt terug op de Radau-formules. We formuleren hieronder in de vorm van een stelling, die we niet bewijzen, de opbouw van dergelijke kwadratuurformules. Zij zijn gekenmerkt door het feit dat één van de knooppunten ofwel de ondergrens ofwel de bovengrens van het integratie-interval moet zijn. De overige knooppunten en gewichten worden dan, net zoals voor de Gauss-formules, zo gekozen dat de *graad van algebraïsche nauwkeurigheid* (GVAN) zo groot mogelijk is.

Stelling 10.5.1 (*Radau I formule*) Als $f(t) \in C^{2n-1}[-1, 1]$ dan is

$$\int_{-1}^1 f(t) dt = \sum_{j=1}^n w_j f(t_j) + \frac{2^{2n-1} n[(n-1)!]^4}{[(2n-1)!]^3} f^{(2n-1)}(\zeta), \quad (10.15)$$

met $-1 < \zeta < 1$ en waarbij t_j de j -de wortel is van

$$P_{n-1}(t) + P_n(t) = 0 \quad (10.16)$$

en

$$w_j = \frac{1}{n^2} \frac{1 - t_j}{[P_{n-1}(t_j)]^2}. \quad (10.17)$$

□

Een tweede soort kwadratuurformules, Radau II genaamd, ontstaat uit de spiegeling van de Radau I formule t.o.v. nul :

Stelling 10.5.2 (*Radau II formule*) Als $f(t) \in C^{2n-1}[-1, 1]$ dan is

$$\int_{-1}^1 f(t) dt = \sum_{j=1}^n w'_j f(t'_j) - \frac{2^{2n-1} n[(n-1)!]^4}{[(2n-1)!]^3} f^{(2n-1)}(\zeta), \quad (10.18)$$

met $-1 < \zeta < 1$ waarbij t'_j de j -de wortel is van

$$P_{n-1}(t') - P_n(t') = 0 \quad (10.19)$$

en

$$w'_j = \frac{1}{n^2} \frac{1 + t'_j}{[P_{n-1}(t'_j)]^2}. \quad (10.20)$$

□

Om nu die resultaten te kunnen gebruiken ter benadering van (10.1) moet opnieuw de transformatie (10.8) doorgevoerd worden, waaruit tevens volgt voor Radau I

$$b_j = \frac{w_j}{2} \quad \text{en} \quad c_j = \frac{1}{2} (1 + t_j) \quad (10.21)$$

en voor Radau II

$$b_j = \frac{w'_j}{2} \quad \text{en} \quad c_j = \frac{1}{2} (1 + t'_j). \quad (10.22)$$

De formules (10.21) en (10.22) leggen alleen de knooppunten c_j en de gewichten b_j vast. De structuur van de foutterm in (10.15) en (10.18) toont aan dat de n -puntskwadratuurformules de GVAN $2n - 2$ bezitten, d.i. 1 lager dan bij Gauss–Legendre-methoden. Uit nader onderzoek is gebleken dat het mogelijk is verschillende families IRKM $_n$, gebaseerd op Radau-kwadratuur, te construeren die van 1 orde lager zijn dan de de Gauss–Legendre IRKM $_n$, afhankelijk van de condities die gesteld worden aan de coëfficiënten a_{ij} :

Radau I : $C(s)$

Radau II : $D(s)$

Radau IA : De reflectie van Radau II

Radau IIA : De reflectie van Radau I

Geval $s = 1$

Vermits $P_1(t) = t$ en $P_0(t) = 1$ volgt uit (10.17) en (10.20) dat resp. $t_1 = -1$ en $t'_1 = +1$. Hieruit volgt meteen dat $w_1 = w'_1 = 2$, zodat, rekening houdend met (10.21)-(10.22), volgt dat

Radau I	Radau II
$\frac{0}{\left \begin{array}{c} 1 \end{array} \right.}$	$\frac{1}{\left \begin{array}{c} 1 \end{array} \right.}$

Voor de Radau I methode wordt $C(\eta)$ geëist met $\eta = 1$, d.i.

$$C(1) : \sum_{j=1}^s a_{ij} = c_j, \quad i = s = 1$$

m.a.w. dit is de rijsom-voorwaarde. De Radau II methoden daarentegen vertrekken van $D(1)$:

$$\sum_{i=1}^s b_i a_{ij} = b_j (1 - c_j), \quad j = s = 1,$$

zodat $a_{11} = 0$. Beschouwen we nu de reflecties van beide methoden, dan bekomen we

Radau I	Radau IIA	Radau II	Radau IA
$\begin{array}{c c} 0 & 0 \\ \hline & 1 \end{array}$	$\begin{array}{c c} 1 & 1 \\ \hline & 1 \end{array}$	$\begin{array}{c c} 1 & 0 \\ \hline & 1 \end{array}$	$\begin{array}{c c} 0 & 1 \\ \hline & 1 \end{array}$

Geval $s = 2$

Met $P_2(t) = \frac{1}{2}(3t^2 - 1)$ en $P_1(t) = t$ zijn de t_j de wortels van de volgende vierkantsvergelijking:

$$\frac{3}{2}t^2 + t - \frac{1}{2} = \frac{1}{2}(3t - 1)(t + 1) = 0,$$

i.e. $t_1 = -1$ en $t_2 = \frac{1}{3}$. Hieruit volgt dan onmiddellijk dat $t'_1 = -\frac{1}{3}$ en $t'_2 = 1$. Met die gegevens volgt uit (10.17) en (10.20) dat

$$w_1 = \frac{1}{2}, \quad w_2 = \frac{3}{2}, \quad w'_1 = \frac{3}{2}, \quad w'_2 = \frac{1}{2},$$

waaruit wegens (10.21)-(10.22)

Radau I	Radau II
$\begin{array}{c cc} 0 & & \\ & & \\ \frac{2}{3} & & \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array}$	$\begin{array}{c cc} \frac{1}{3} & & \\ & & \\ 1 & & \\ \hline & \frac{3}{4} & \frac{1}{4} \end{array}$

Voor Radau I worden de a_{ij} -waarden bepaald door $C(2)$. Dit leidt, wegens $c_1 = 0$, tot

$$\left\{ \begin{array}{l} a_{11} + a_{12} = 0 \\ a_{12} \frac{2}{3} = 0 \\ a_{21} + a_{22} = \frac{2}{3} \\ a_{22} \frac{2}{3} = \frac{2}{9} \end{array} \right. \iff \left\{ \begin{array}{l} a_{11} = 0 \\ a_{12} = 0 \\ a_{21} = \frac{1}{3} \\ a_{22} = \frac{1}{3} \end{array} \right.$$

Voor Radau II luiden de voorwaarden, gegeven door $D(2)$:

$$\begin{cases} 3a_{11} + a_{21} = 2 \\ a_{11} + a_{21} = \frac{4}{3} \\ 3a_{12} + a_{22} = 0 \\ a_{12} + a_{22} = 0 \end{cases} \iff \begin{cases} a_{11} = \frac{1}{3} \\ a_{12} = 0 \\ a_{21} = 1 \\ a_{22} = 0 \end{cases}$$

Radau I	Radau IIA	Radau II	Radau IA
$\begin{array}{c cc} 0 & 0 & 0 \\ \frac{2}{3} & \frac{1}{3} & \frac{1}{3} \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array}$	$\begin{array}{c cc} \frac{1}{3} & \frac{5}{12} & -\frac{1}{12} \\ 1 & \frac{3}{4} & \frac{1}{4} \\ \hline & \frac{3}{4} & \frac{1}{4} \end{array}$	$\begin{array}{c cc} \frac{1}{3} & \frac{1}{3} & 0 \\ 1 & 1 & 0 \\ \hline & \frac{3}{4} & \frac{1}{4} \end{array}$	$\begin{array}{c cc} 0 & \frac{1}{4} & -\frac{1}{4} \\ \frac{2}{3} & \frac{1}{4} & \frac{5}{12} \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array}$

Geval $s = 3$

Op analoge wijze verder werkend, vindt men de Radau-methoden voor hogere s waarden. Hieronder geven we de matrices corresponderend met de reflecties Radau IA en Radau IIA.

Radau IA

0	$\frac{1}{9}$	$-\frac{1 + \sqrt{6}}{18}$	$\frac{-1 + \sqrt{6}}{18}$
$\frac{6 - \sqrt{6}}{10}$	$\frac{1}{9}$	$\frac{88 + 7\sqrt{6}}{360}$	$\frac{88 - 43\sqrt{6}}{360}$
$\frac{6 + \sqrt{6}}{10}$	$\frac{1}{9}$	$\frac{88 + 43\sqrt{6}}{360}$	$\frac{88 - 7\sqrt{6}}{360}$
	$\frac{1}{9}$	$\frac{16 + \sqrt{6}}{36}$	$\frac{16 - \sqrt{6}}{36}$

Radau IIA

$\frac{4 - \sqrt{6}}{10}$	$\frac{88 - 7\sqrt{6}}{360}$	$\frac{296 - 169\sqrt{6}}{1800}$	$\frac{-2 + 3\sqrt{6}}{225}$
$\frac{4 + \sqrt{6}}{10}$	$\frac{296 + 169\sqrt{6}}{1800}$	$\frac{88 + 7\sqrt{6}}{360}$	$\frac{-2 - 3\sqrt{6}}{225}$
1	$\frac{16 - \sqrt{6}}{36}$	$\frac{16 + \sqrt{6}}{36}$	$\frac{1}{9}$
	$\frac{16 - \sqrt{6}}{36}$	$\frac{16 + \sqrt{6}}{36}$	$\frac{1}{9}$

Opmerking 10.5.1

Op analoge wijze als in Opmerking 10.3.1 kunnen we de c_k van de beide Radau-methoden berekenen als wortels van uitdrukkingen, die gebaseerd zijn op de Rodrigues-vorm van de Legendre-veeltermen. Vertrekkend van (10.5) vinden we

$$P_n(t) = \frac{(-1)^{n-1}}{2^{n-1} (n-1)!} \frac{d^{n-1}}{dt^{n-1}} [(1-t^2)^{n-1} t]$$

en

$$P_{n-1}(t) = \frac{(-1)^{n-1}}{2^{n-1} (n-1)!} \frac{d^{n-1}}{dt^{n-1}} [(1-t^2)^{n-1}],$$

waaruit

$$P_n(t) \pm P_{n-1}(t) = \frac{(-1)^{n-1}}{2^{n-1} (n-1)!} \frac{d^{n-1}}{dt^{n-1}} [(1-t^2)^{n-1} (t \pm 1)].$$

Aldus geldt, met $t = 2c - 1$

$$P_n(2c-1) + P_{n-1}(2c-1) = 0 \quad \text{a.s.a.} \quad \frac{d^{n-1}}{dc^{n-1}} [c^n (1-c)^{n-1}] = 0$$

en

$$P_n(2c-1) - P_{n-1}(2c-1) = 0 \quad \text{a.s.a.} \quad \frac{d^{n-1}}{dc^{n-1}} [c^{n-1} (1-c)^n] = 0.$$

□

Opmerking 10.5.2

De 1-traps Radau IIA methode kan net zoals de 1-traps Gauss-methode herschreven worden in de volgende vorm

$$y_{n+1} = y_n + h f(x_n + h, y_{n+1}).$$

Deze methode, die ook bekomen wordt als de eerste-orde Adams–Moulton-methode, is in de literatuur meer bekend als de *impliciete Euler-methode* of de *achterwaartse Euler-methode*. □

10.6 Lobatto-methoden

De laatste categorie van IRKMn is geassocieerd aan de Lobatto-kwadratuurformules, waarvoor de verzameling van knooppunten o.m. beide integratiegrenzen bevat. De corresponderende IRK methoden (die minstens trapgetal 2 moeten hebben) bezitten $c_1 = 0$ en $c_s = 1$. De maximaal bereikbare orde is $p = 2s - 2$. De Lobatto-kwadratuurformules zelf zullen we hier niet afleiden. Aan de hand van een stelling geven we hieronder de bruikbare formule.

Stelling 10.6.1 Als $f(t) \in C^{2n-2}[-1, 1]$ dan is

$$\int_{-1}^1 f(t) dt = \frac{2}{n(n-1)} [f(-1) + f(1)] + \sum_{j=2}^{n-1} w_j f(t_j) - \frac{n(n-1)^3 2^{2n-1} [(n-2)!]^4}{(2n-1) [(2n-2)!]^3} f^{(2n-2)}(\zeta), \quad (10.23)$$

met $-1 < \zeta < 1$ en waarbij t_j de $(j-1)$ -de wortel is van $P'_{n-1}(t) = 0$ en

$$w_j = \frac{2}{n(n-1) [P_{n-1}(t_j)]^2}, \quad t_j \neq \pm 1. \quad (10.24)$$

□

Uiteraard zijn ook hier transformaties van de vorm (10.9) of (10.21) van kracht.

Opmerking 10.6.1

Men kan aantonen dat de Legendre-veeltermen voldoen aan de betrekkingen

$$n P_n(t) = (2n - 1)t P_{n-1}(t) - (n - 1) P_{n-2}(t) \tag{10.25}$$

en

$$(1 - t^2) P'_{n-1}(t) + (n - 1)t P_{n-1}(t) - (n - 1) P_{n-2}(t) = 0. \tag{10.26}$$

Uit Stelling 10.6.1 volgt dat de s knooppunten t_j van de Lobatto-kwadratuurformule voldoen aan

$$(1 - t^2) P'_{n-1}(t) = 0,$$

waaruit m.b.v. (10.26) en (10.25) volgt dat deze knooppunten voldoen aan

$$0 = (n - 1)t P_{n-1}(t) - (n - 1) P_{n-2}(t) = n P_n(t) - n P_{n-2}(t),$$

hetgeen aantoont dat de Lobatto IRKMn voldoen aan (10.11) voor $\mu = 0$ en $\nu = -1$. □

Opnieuw zijn er voor de invulling van de a_{ij} coëfficiënten, naast de ordevoorwaarden, bijkomende afspraken nodig :

Lobatto III : $C(s - 1)$ en $a_{1s} = a_{2s} = \dots = a_{ss} = 0$

Lobatto IIIA : $C(s)$

Lobatto IIIB : $D(s)$

Lobatto IIIC : De reflectie van Lobatto III

Geval $s = 2$

In dit geval is $c_1 = 0$ en $c_2 = 1$. Rekening houdend met de transformatie (10.8) en (10.9) volgt dan dat

$$b_1 = \frac{1}{2}, \quad b_2 = \frac{1}{2}.$$

Het is onmiddellijk controleerbaar dat met die b_i en c_i waarden reeds voldaan is aan de ordevoorwaarden, wat de orde p op $2 = 2s - 2$ brengt.

Voor de Lobatto III methode en haar reflectie vindt men :

Lobatto III	Lobatto IIIC
$\begin{array}{c cc} 0 & 0 & 0 \\ 1 & 1 & 0 \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$	$\begin{array}{c cc} 0 & \frac{1}{2} & -\frac{1}{2} \\ 1 & \frac{1}{2} & \frac{1}{2} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$

Voor Lobatto IIIA en Lobatto IIIB vindt men na enig rekenwerk :

Lobatto IIIA			
0	0	0	
1	$\frac{1}{2}$	$\frac{1}{2}$	
	$\frac{1}{2}$	$\frac{1}{2}$	

Lobatto IIIB			
0	$\frac{1}{2}$	0	
1	$\frac{1}{2}$	0	
	$\frac{1}{2}$	$\frac{1}{2}$	

Geval $s = 3$

De methoden met $s = 3$ bezitten orde $p = 2s - 2 = 4$. De knooppuntwaarde t_2 in de kwadratuurformule (10.23) is de eerste wortel van $P_2'(t) = 3t = 0$, d.i. $t_2 = 0$, wat rekening houdend met o.a. (10.21) zich vertaalt in

$$c_1 = 0 \quad c_2 = \frac{1}{2} \quad c_3 = 1.$$

Uit (10.23) en (10.9) volgt rechtstreeks dat $b_1 = b_3 = \frac{1}{6}$ en uit (10.24) en (10.9) halen we dat

$$b_2 = \frac{w_2}{2} = \frac{2}{3}.$$

Uiteindelijk vindt men de volgende methoden :

Lobatto III				
0	0	0	0	
$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$	0	
1	0	1	0	
	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$	

Lobatto IIIC				
0	$\frac{1}{6}$	$-\frac{1}{3}$	$\frac{1}{6}$	
$\frac{1}{2}$	$\frac{1}{6}$	$\frac{5}{12}$	$-\frac{1}{12}$	
1	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$	
	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$	

Lobatto IIIA				
0	0	0	0	
$\frac{1}{2}$	$\frac{5}{24}$	$\frac{1}{3}$	$-\frac{1}{24}$	
1	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$	
	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$	

Lobatto IIIB				
0	$\frac{1}{6}$	$-\frac{1}{6}$	0	
$\frac{1}{2}$	$\frac{1}{6}$	$\frac{1}{3}$	0	
1	$\frac{1}{6}$	$\frac{5}{6}$	0	
	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$	

Opmerking 10.6.2

Op analoge wijze als in Opmerking 10.3.1 en Opmerking 10.5.1 kunnen we de punten c_j opnieuw berekenen als de wortelpunten van een Rodrigues-vorm; uit

Stelling 10.6.1 en de transformatie (10.8) volgt dat de c_j Lobatto-waarden voor alle types de wortelpunten zijn van

$$\frac{d^{s-2}}{dc^{s-2}}[c^{s-1}(c-1)^{s-1}] = 0.$$

□

Opmerking 10.6.3

De benamingen Radau I, Radau II en Lobatto III vinden hun oorsprong in het werk van Butcher. Hij klasseerde de methoden oorspronkelijk als methoden van het type I, II, of III naargelang de methode 0, 1 of beide als knooppunt bezit. De klassificaties IA, IIA, IIIA, IIIB en IIIC werden geïntroduceerd door Ehle en Chipman.

De methoden Radau I, Radau II en Lobatto III werden oorspronkelijk ingevoerd door Butcher als zijnde de methoden met het kleinste aantal impliciete stappen om een gegeven orde te bereiken. Toch zijn het de 5 andere soorten methoden die in de praktijk het meest gebruikt worden en dit omwille van stabiliteitsoverwegingen.

□

10.7 Collocatiemethoden

Tot nu toe hebben we RKMn gecatalogeerd volgens hun relatie met kwadratuurformules. Een andere manier om die catalogering door te voeren heeft betrekking op het feit of die methode al of niet een *collocatiemethode* is. Collocatie is een welbekende methode in de numerieke analyse. De methode bestaat erin een functie (meestal een veelterm) en een verzameling van zgn. *collocatiepunten* te kiezen en op te leggen dat die functie in die collocatiepunten zo goed mogelijk de onbekende functie numeriek benadert. Bij het oplossen van een IVP $y' = f(x, y)$, $y(a) = \eta$ kunnen we de numerieke oplossing laten evolueren van x_n naar x_{n+1} door een veelterm te kiezen van graad s met coëfficiënten in \mathbb{R}^m en een verzameling van verschillende collocatiepunten $\{x_n + c_i h \mid i = 1, 2, \dots, s\}$, waarbij we vragen dat

$$\begin{aligned} P(x_n) &= y_n \\ P'(x_n + c_i h) &= f(x_n + c_i h, P(x_n + c_i h)), \quad i = 1, 2, \dots, s. \end{aligned}$$

Merk op dat deze voorwaarden $P(x)$ uniek definiëren. We beëindigen dan de numerieke stap van x_n naar x_{n+1} door $y_{n+1} = P(x_n + h)$ te stellen. Er kan aangetoond worden dat dit proces identiek is aan het proces gevolgd door een s -traps IRKM. Om dit in te zien, moeten we ons realiseren dat $P'(x)$ een veelterm van graad $s - 1$ is, die de volgende s datapunten interpoleert :

$$(x_n + c_i h, P'(x_n + c_i h)), \quad i = 1, 2, \dots, s.$$

Derhalve kunnen we $P'(x)$ schrijven in de vorm van een Lagrange-interpolatieveelterm. Definiëren we nu

$$k_i := P'(x_n + c_i h), \quad i = 1, 2, \dots, s,$$

en schrijven we $x = x_n + t h$, dan hebben we

$$P'(x_n + t h) = \sum_{j=1}^s l_j(t) k_j, \quad (10.27)$$

met

$$l_j(t) = \prod_{\substack{i=1 \\ i \neq j}}^s \frac{t - c_i}{c_j - c_i}. \quad (10.28)$$

Integreren we nu (10.27) naar t van $t = 0$ (of $x = x_n$) naar $t = c_i$ (of $x = x_n + c_i h$), $i = 1, 2, \dots, s$ en van $t = 0$ ($x = x_n$) naar $t = 1$ ($x = x_{n+1}$), dan krijgen we

$$P(x_n + c_i h) - P(x_n) = h \sum_{j=1}^s \left(\int_0^{c_i} l_j(t) dt \right) k_j, \quad i = 1, 2, \dots, s, \quad (10.29)$$

en

$$P(x_n + h) - P(x_n) = h \sum_{j=1}^s \left(\int_0^1 l_j(t) dt \right) k_j. \quad (10.30)$$

Definiëren we nu, voor $j = 1, 2, \dots, s$:

$$a_{ij} := \int_0^{c_i} l_j(t) dt, \quad i = 1, 2, \dots, s, \quad b_j := \int_0^1 l_j(t) dt, \quad (10.31)$$

en houden we rekening met (10.29) en (10.30) dan verkrijgen we

$$\begin{aligned} k_i &= P'(x_n + c_i h) = f(x_n + c_i h, P(x_n + c_i h)) \\ &= f(x_n + c_i h, y_n + h \sum_{j=1}^s a_{ij} k_j), \quad i = 1, 2, \dots, s \end{aligned}$$

en

$$y_{n+1} = y_n + h \sum_{j=1}^s b_j k_j.$$

We bekomen dus een IRKM met als collocatiepunten de componenten van c en waarbij b en A gegeven worden door (10.31). De *klasse van collocatiemethoden* bestaat uit die IRKMn die op deze wijze kunnen afgeleid worden.

Er kan ook een andere interpretatie gehecht worden aan collocatiemethoden. Daartoe trekken we na wat er gebeurt met de alternatieve vorm (7.5) van de algemene RKM als we hierin $f(x, y) = f(x)$ kiezen. De tweede betrekking van (7.5) kan dan geïnterpreteerd worden als een kwadratuurformule voor

$$\int_{x_n}^{x_n + c_i h} f(x) dx, \quad i = 1, 2, \dots, s.$$

Vermits

$$t^{r-1} = \sum_{j=1}^s l_j(t) c_j^{r-1} \quad r = 1, \dots, s,$$

is

$$\int_0^{c_i} t^{r-1} dt = \sum_{j=1}^s \int_0^{c_i} l_j(t) dt c_j^{r-1} \quad r = 1, \dots, s$$

en

$$\int_0^1 t^{r-1} dt = \sum_{j=1}^s \int_0^1 l_j(t) dt c_j^{r-1} \quad r = 1, \dots, s,$$

zodat, gelet op (10.31), geldt

$$\sum_{j=1}^s a_{ij} c_j^{r-1} = \frac{c_i^r}{r}, \quad i = 1, 2, \dots, s, \quad r = 1, 2, \dots, s \quad (10.32)$$

en

$$\sum_{j=1}^s b_j c_j^{r-1} = \frac{1}{r}, \quad i = 1, 2, \dots, s, \quad r = 1, 2, \dots, s. \quad (10.33)$$

Deze voorwaarden zijn niets anders dan voorwaarden $C(\eta)$ en $B(p)$ uit (10.2) met $\eta = p = s$. Aan de hand van bovenstaande uiteenzetting, kunnen we de volgende stelling omschrijven :

Stelling 10.7.1 *De nodige en voldoende voorwaarde opdat een s -traps RKM van orde tenminste s en met verschillende c_i -waarden een collocatiemethode zou zijn, is dat de voorwaarde (10.32) voldaan is. \square*

We kunnen de families van IRKMn uit vorige paragrafen dan ook opsplitsen in twee groepen :

- de Gauss, Radau I, Radau IIA en Lobatto IIIA methoden zijn collocatiemethoden.
- de Radau II, Radau IA, Lobatto III, Lobatto IIIB en Lobatto IIIC methoden zijn geen collocatiemethoden.

Voorbeeld 10.7.1

De twee-traps Gauss–Legendre-methode werd opgesteld aan de hand van relatie (10.32) en is dus een collocatiemethode. Hieronder stellen we deze methode opnieuw op, maar dan door te vertrekken van de collocatievoorwaarden (10.31). We vinden :

$$l_1(t) = -\sqrt{3} \left(t - \frac{1}{2} - \frac{\sqrt{3}}{6} \right) \quad \text{en} \quad l_2(t) = \sqrt{3} \left(t - \frac{1}{2} + \frac{\sqrt{3}}{6} \right)$$

en

$$\begin{aligned}
 a_{11} &= -\int_0^{\frac{1}{2}-\frac{\sqrt{3}}{6}} \sqrt{3} \left(t - \frac{1}{2} - \frac{\sqrt{3}}{6} \right) dt = \frac{1}{4} \\
 a_{12} &= \int_0^{\frac{1}{2}-\frac{\sqrt{3}}{6}} \sqrt{3} \left(t - \frac{1}{2} + \frac{\sqrt{3}}{6} \right) dt = \frac{1}{4} - \frac{\sqrt{3}}{6} \\
 a_{21} &= -\int_0^{\frac{1}{2}+\frac{\sqrt{3}}{6}} \sqrt{3} \left(t - \frac{1}{2} - \frac{\sqrt{3}}{6} \right) dt = \frac{1}{4} + \frac{\sqrt{3}}{6} \\
 a_{22} &= \int_0^{\frac{1}{2}+\frac{\sqrt{3}}{6}} \sqrt{3} \left(t - \frac{1}{2} + \frac{\sqrt{3}}{6} \right) dt = \frac{1}{4} \\
 b_1 &= -\int_0^1 \sqrt{3} \left(t - \frac{1}{2} - \frac{\sqrt{3}}{6} \right) dt = \frac{1}{2} \\
 b_2 &= \int_0^1 \sqrt{3} \left(t - \frac{1}{2} + \frac{\sqrt{3}}{6} \right) dt = \frac{1}{2}
 \end{aligned}$$

□

10.8 Semi-impliciete methoden

Zoals reeds aangestipt in paragraaf 7.1 vraagt het minder numerieke inspanning om DIRKMn op te lossen dan volledige IRKMn. Dit verschil wordt nog groter in het geval van zgn. SDIRKMn (*singly diagonal*), waarbij alle elementen op de hoofddiagonaal dezelfde numerieke waarde hebben. Inderdaad, in paragraaf 10.9 wordt aangetoond dat indien bij de bepaling van k_i gebruik gemaakt wordt van een Newtoniaans iteratieschema, men een lineair stelsel oplost met een coëfficiëntmatrix van de vorm

$$I - h a_{ii} \frac{\partial f}{\partial y}.$$

Zijn alle a_{ii} gelijk, dan kan men herhaaldelijk gebruik maken van de LU-factorisatie van één van zo'n matrices.

Voorbeeld 10.8.1

We zoeken naar 2-traps DIRKMn (d.i. $a_{12} = 0$) van zo hoog mogelijke orde. De unieke 2-traps methode van orde 4 is niet van het DIRK-type, wat betekent dat hoogstens orde 3 kan bereikt worden. De volgende voorwaarden moeten dan gelden :

$$\left\{ \begin{array}{l}
 (1) \quad b_1 + b_2 = 1 \\
 (2) \quad b_1 c_1 + b_2 c_2 = \frac{1}{2} \\
 (3) \quad b_1 c_1^2 + b_2 c_2^2 = \frac{1}{3} \\
 (4) \quad b_1 a_{11} c_1 + b_2 (a_{21} c_1 + a_{22} c_2) = \frac{1}{6}
 \end{array} \right.$$

De eerste 3 voorwaarden zijn alleen voldaan als

$$\begin{vmatrix} 1 & 1 & 1 \\ c_1 & c_2 & \frac{1}{2} \\ c_1^2 & c_2^2 & \frac{1}{3} \end{vmatrix} = 0 \quad \text{of} \quad (c_2 - c_1) \left[\frac{1}{3} - \frac{1}{2}(c_1 + c_2) + c_1 c_2 \right] = 0.$$

Is $c_1 = c_2$, dan moet wegens (1) en (2) $c_1 = c_2 = \frac{1}{2}$ en bedraagt de orde slechts 2. We stellen aldus

$$c_2 = \frac{2 - 3c_1}{3(1 - 2c_1)},$$

waaruit wegens (1) en (2) ook

$$\begin{aligned} b_1 &= \frac{1 - 2c_2}{2(c_1 - c_2)} = \frac{1}{4(1 - 3c_1 + 3c_1^2)} \\ b_2 &= \frac{1 - 2c_1}{2(c_2 - c_1)} = \frac{3(1 - 2c_1)^2}{4(1 - 3c_1 + 3c_1^2)} \end{aligned}$$

Om naast (4) nog extra voorwaarden op te leggen op de 3 a_{ij} -waarden gebruiken we de rijsum-voorwaarden. We vinden aldus $a_{11} = c_1$ en

$$\begin{aligned} \frac{1}{6} &= b_1 c_1^2 + b_2 [(c_2 - a_{22}) c_1 + a_{22} c_2] \\ &= c_1 (b_1 c_1 + b_2 c_2) + b_2 (c_2 - c_1) a_{22} \\ &= \frac{c_1}{2} + b_2 (c_2 - c_1) a_{22}, \end{aligned}$$

waaruit

$$a_{22} = \frac{1 - 3c_1}{3(1 - 2c_1)}, \quad \text{en} \quad a_{12} = c_2 - a_{22} = \frac{1}{3(1 - 2c_1)}.$$

We verkrijgen aldus een 1-parameter familie van DIRKMn van orde 3 :

λ	λ	0
$\frac{2 - 3\lambda}{3(1 - 2\lambda)}$	$\frac{1}{3(1 - 2\lambda)}$	$\frac{1 - 3\lambda}{3(1 - 2\lambda)}$
$\frac{1}{4(1 - 3\lambda + 3\lambda^2)}$	$\frac{3(1 - 2\lambda)^2}{4(1 - 3\lambda + 3\lambda^2)}$	

Binnen deze familie kunnen we nu op zoek gaan naar de SDIRKMn. We bekommen

$$\lambda = \frac{1 - 3\lambda}{3(1 - 2\lambda)} \iff -6\lambda^2 + 6\lambda - 1 = 0$$

waaruit

$$\lambda = \frac{3 \pm \sqrt{3}}{6}.$$

We verkrijgen aldus het volgende paar SDIRK-methoden :

$$\begin{array}{c|cc} \frac{3 \pm \sqrt{3}}{6} & \frac{3 \pm \sqrt{3}}{6} & 0 \\ \frac{3 \mp \sqrt{3}}{6} & \mp \frac{\sqrt{3}}{3} & \frac{3 \pm \sqrt{3}}{6} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

Er bestaan uiteraard ook hogere orde DIRKMn en SDIRKMn. Hun opbouw steunt opnieuw op typische theorieën, die grotendeels ook ontworpen zijn door Butcher. De studie hiervan valt buiten het kader van deze nota's. \square

10.9 Implementatie van niet-expliciete methoden

Het gebruik van een niet-ERKM dwingt ons bij elke stap een niet-lineair stelsel in de k_i op te lossen. Dit kan gebeuren aan de hand van vastpunt-iteratie of m.b.v. een Newtoniaans iteratieschema.

10.9.1 Vastpunt-iteratie

Een elegant algoritme, toepasbaar op autonome problemen van de vorm (8.1), werd voorgesteld door Van der Houwen en Sommeijer. Om dit algoritme uiteen te zetten, herschrijven we de RKM

$$\begin{aligned} y_{n+1} &= y_n + h \sum_{i=1}^s b_i k_i \\ k_i &= f\left(y_n + h \sum_{j=1}^s a_{ij} k_j\right), \quad i = 1, 2, \dots, s \end{aligned}$$

in een vectornotatie. Als we de vectoren $e, k \in \mathbb{R}^s$ definiëren als $e := [1, 1, \dots, 1]^T$ en $k := [k_1, k_2, \dots, k_s]^T$ dan is de RKM te schrijven als

$$y_{n+1} = y_n + h b^T k$$

$$k = f(y_n e + h A k)$$

Verder is bij die notatie de conventie ingebouwd dat voor elke vector $v = (v_j)$, $f(v)$ de vector met waarden $f(v_j)$ voorstelt. Door de vergelijking voor k m maal te itereren en de m -de iteratiewaarde als benadering voor k te beschouwen, bekommen we de methode

$$k^{(j)} = f(y_n e + h A k^{(j-1)}), \quad j = 1, 2, \dots, m, \tag{10.34}$$

$$y_{n+1} = y_n + h b^T k^{(m)}. \tag{10.35}$$

Om dit iteratieproces te kunnen opstarten is er een methode nodig die de beginschatting $k^{(0)}$ levert; door Van der Houwen en Sommeijer wordt die methode de *predictor methode* genoemd, terwijl (10.34) als de *corrector methode* wordt betiteld. Het is evident dat bij elke iteratiestap de lopende benadering voor y_{n+1} berekend kan worden m.b.v. de formule

$$y^{(j)} = y_n + h b^T k^{(j)}, \quad j = 0, 1, \dots$$

Als $k^{(0)}$ een benadering is voor k die voldoet aan de voorwaarde

$$k^{(0)} = k + \mathcal{O}(h^q),$$

en de RKM bezit de orde p dan is er door Jackson en N rsett aangetoond in een technisch rapport dat de globale orde van (10.34) gelijk is aan

$$p^* = \min\{p, q + m\}.$$

Door een eenvoudige predictor methode te gebruiken, nl.

$$k^{(0)} = f(y_n) e = k + \mathcal{O}(h),$$

d.w.z. $q = 1$, bekomen Van der Houwen en Sommeijer de volgende stelling :

Stelling 10.9.1 *Als $\{A, b^T\}$ een (al dan niet impliciete) s -traps RKM definieert van orde p , dan definieert*

$$\begin{aligned} k^{(0)} &= f(y_n) e, \\ k^{(j)} &= f(y_n e + h A k^{(j-1)}), \quad j = 1, 2, \dots, m, \\ y_{n+1} &= y_n + h b^T k^{(m)} \end{aligned}$$

een $(m + 1)$ -traps ERKM van orde $p^ = \min\{p, m + 1\}$.* □

In de praktijk wordt in dit algoritme meestal een Gauss–Legendre-methode gebruikt en $m = p - 1$ gesteld om effectief een orde p te bereiken.

Van der Houwen en Sommeijer ontwikkelden een code rond dit algoritme om ze te laten lopen in een *geparallelliseerde omgeving*, m.a.w. op een computer met verschillende processoren die gelijktijdig verschillende jobs kunnen uitvoeren; vandaar de naam PIRK : *parallel iterated RK*.

De controle over de staplengte gebeurt als volgt : vermits we bij de iteratieprocedure voor k verschillende benaderingen $k^{(j)}$ bij toenemende orde bekomen, nl.

$$k^{(j)} - k = \mathcal{O}(h^{\min\{p, q+j\}}), \quad j = 1, 2, \dots, m,$$

kunnen we, naast de finale benadering

$$y_{n+1} = y_n + h b^T k^{(m)}$$

gemakkelijk een tussenoplossing

$$y^{(i)} = y_n + h b^T k^{(i)}, \quad \text{voor } i < m$$

construeren. Vermits $k^{(i)}$ reeds berekend is, zijn geen bijkomende functie-evaluaties nodig. Die referentieoplossing $y^{(i)}$ kan beschouwd worden als een *ingebede* oplossing. Als schatting voor de lokale fout ϵ in de stap van x_n naar $x_{n+1} = x_n + h$, nemen we

$$\epsilon = \|y_{n+1} - y^{(i)}\|.$$

Meestal gebruikt men referentieoplossingen $y^{(i)}$ zodat de ordes van y_{n+1} en $y^{(i)}$ slechts met één eenheid verschillen. In de code wordt dit idee gevolgd en $i = m - 1$ wordt gekozen, wat betekent dat

$$\epsilon = \|y_{n+1} - y^{(m-1)}\| = \mathcal{O}(h^p). \quad (10.36)$$

Zoals in paragraaf 9.3 wordt ϵ vergeleken met een voorgeschreven tolerantie TOL en de stap wordt aanvaard als $\epsilon \leq \text{TOL}$. Op die wijze laat (10.36) ons toe een maximale staplengte voorop te stellen als

$$h \left(\frac{\text{TOL}}{\epsilon} \right)^{1/p}$$

en een nieuwe stap te introduceren (cfr. (9.17)) als

$$h_{\text{nieuw}} = h \min \left(6, \max \left(\frac{1}{3}, 0.9 \left(\frac{\text{TOL}}{\epsilon} \right)^{1/p} \right) \right).$$

10.9.2 Newton-iteratie

Het vastpunt-iteratieproces zoals o.m. beschreven in de PIRKM, transformeert IRKMn in ERKMn en heeft tot gevolg dat de stabiliteitseigenschappen van de methode gewijzigd worden. Dit heeft in het bijzonder zijn weerslag wanneer de methode toegepast wordt op stijve problemen.

Inderdaad, het iteratieproces

$$k^{(j)} = f(y_n e + h A k^{(j-1)}) = \phi(k^{(j-1)}), \quad (10.37)$$

convergeert wegens Stelling A.4.1 indien de functie ϕ een Lipschitz-constante L bezit waarvoor geldt $0 \leq L < 1$. Deze Lipschitz-constante L voldoet, indien ϕ differentieerbaar is, aan

$$L = \sup \left\| \frac{\partial \phi}{\partial k} \right\| \geq \max_t |\Lambda_t|,$$

waarbij Λ_t de eigenwaarden voorstellen van de $m s \times m s$ matrix $\frac{\partial \phi}{\partial k}$. Vermits

$$\begin{bmatrix} {}^1k_1 \\ {}^2k_1 \\ \vdots \\ {}^{mk_1} \\ {}^1k_2 \\ \vdots \\ {}^{mk_s} \end{bmatrix} = \begin{bmatrix} {}^1f({}^1y_n + h \sum_{j=1}^s a_{1j} {}^1k_j, {}^2y_n + h \sum_{j=1}^s a_{1j} {}^2k_j, \dots, {}^m y_n + h \sum_{j=1}^s a_{1j} {}^{mk_j}) \\ {}^2f({}^1y_n + h \sum_{j=1}^s a_{1j} {}^1k_j, {}^2y_n + h \sum_{j=1}^s a_{1j} {}^2k_j, \dots, {}^m y_n + h \sum_{j=1}^s a_{1j} {}^{mk_j}) \\ \vdots \\ {}^m f({}^1y_n + h \sum_{j=1}^s a_{1j} {}^1k_j, {}^2y_n + h \sum_{j=1}^s a_{1j} {}^2k_j, \dots, {}^m y_n + h \sum_{j=1}^s a_{1j} {}^{mk_j}) \\ {}^1f({}^1y_n + h \sum_{j=1}^s a_{2j} {}^1k_j, {}^2y_n + h \sum_{j=1}^s a_{2j} {}^2k_j, \dots, {}^m y_n + h \sum_{j=1}^s a_{2j} {}^{mk_j}) \\ \vdots \\ {}^m f({}^1y_n + h \sum_{j=1}^s a_{sj} {}^1k_j, {}^2y_n + h \sum_{j=1}^s a_{sj} {}^2k_j, \dots, {}^m y_n + h \sum_{j=1}^s a_{sj} {}^{mk_j}) \end{bmatrix}$$

vinden we dat het $(m(i-1) + j, m(l-1) + n)$ -de element in de Jacobiaan $\frac{\partial \phi}{\partial k}$ gegeven wordt door

$$\frac{\partial^j \phi_i}{\partial {}^{nk_l}} = h a_{il} \frac{\partial^j f}{\partial {}^{ny}}$$

Vermits

$$\frac{\partial \phi}{\partial k} = \begin{bmatrix} \frac{\partial^1 \phi_1}{\partial {}^1k_1} & \frac{\partial^1 \phi_1}{\partial {}^2k_1} & \cdots & \frac{\partial^1 \phi_1}{\partial {}^{mk_1}} & \frac{\partial^1 \phi_1}{\partial {}^1k_2} & \cdots & \frac{\partial^1 \phi_1}{\partial {}^{mk_s}} \\ \frac{\partial^2 \phi_1}{\partial {}^1k_1} & \frac{\partial^2 \phi_1}{\partial {}^2k_1} & \cdots & \frac{\partial^2 \phi_1}{\partial {}^{mk_1}} & \frac{\partial^2 \phi_1}{\partial {}^1k_2} & \cdots & \frac{\partial^2 \phi_1}{\partial {}^{mk_s}} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial^m \phi_1}{\partial {}^1k_1} & \frac{\partial^m \phi_1}{\partial {}^2k_1} & \cdots & \frac{\partial^m \phi_1}{\partial {}^{mk_1}} & \frac{\partial^m \phi_1}{\partial {}^1k_2} & \cdots & \frac{\partial^m \phi_1}{\partial {}^{mk_s}} \\ \frac{\partial^1 \phi_2}{\partial {}^1k_1} & \frac{\partial^1 \phi_2}{\partial {}^2k_1} & \cdots & \frac{\partial^1 \phi_2}{\partial {}^{mk_1}} & \frac{\partial^1 \phi_2}{\partial {}^1k_2} & \cdots & \frac{\partial^1 \phi_2}{\partial {}^{mk_s}} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial^m \phi_s}{\partial {}^1k_1} & \frac{\partial^m \phi_s}{\partial {}^2k_1} & \cdots & \frac{\partial^m \phi_s}{\partial {}^{mk_1}} & \frac{\partial^m \phi_s}{\partial {}^1k_2} & \cdots & \frac{\partial^m \phi_s}{\partial {}^{mk_s}} \end{bmatrix}$$

is, als $J = \frac{\partial f}{\partial y}$,

$$\frac{\partial \phi}{\partial k} = h \begin{bmatrix} a_{11} J & a_{12} J & \dots & a_{1s} J \\ a_{21} J & a_{22} J & \dots & a_{2s} J \\ \vdots & \vdots & & \vdots \\ a_{s1} J & a_{s2} J & \dots & a_{ss} J \end{bmatrix} \quad (10.38)$$

De matrix die optreedt in (10.38) heeft een heel speciale vorm. Ze is het *direct product* \otimes van de matrices A en J , m.a.w.

$$\frac{\partial \phi}{\partial k} = h A \otimes \frac{\partial f}{\partial y}.$$

Men kan algemeen aantonen voor $p \times p$ matrices P , P_1 en P_2 en $q \times q$ matrices Q , Q_1 en Q_2 :

- (1) $(P_1 \otimes Q_1)(P_2 \otimes Q_2) = P_1 P_2 \otimes Q_1 Q_2$
- (2) $(P \otimes Q)^{-1} = P^{-1} \otimes Q^{-1}$.
- (3) als α_i , $i = 1, 2, \dots, p$ de eigenwaarden van P en β_j , $j = 1, 2, \dots, q$ deze van Q zijn, dan zijn de eigenwaarden van $P \otimes Q$ gegeven door de producten $\alpha_i \beta_j$, $i = 1, 2, \dots, p$ en $j = 1, 2, \dots, q$.

Uit eigenschap (3) volgt onmiddellijk dat als μ_i , $i = 1, 2, \dots, s$ de eigenwaarden van A en λ_j , $j = 1, 2, \dots, m$ de eigenwaarden van de Jacobiaan J zijn, de eigenwaarden van

$$\frac{\partial \phi}{\partial k}$$

gegeven worden door de ms producten $h \mu_i \lambda_j = h \Lambda_{ij}$, $i = 1, 2, \dots, s$, en $j = 1, 2, \dots, m$. De eigenwaarden μ_i zijn afhankelijk van A , d.w.z. van de methode en zijn normaal gezien niet zeer klein. De eigenwaarden λ_j daarentegen zijn afhankelijk van het probleem. Voor stijve problemen, dit zijn problemen die gekenmerkt worden door waarden λ_j met $\text{Re } \lambda_j \ll 0$, betekent dit dat dit ook geldt voor de corresponderende Λ -waarden. Vermits

$$1 > L \geq \max_t |\Lambda_t| = h \max_{i,j} |\mu_i \lambda_j|,$$

betekent dit dat convergentie slechts mogelijk is voor (zeer) kleine waarden van h .

De enige manier van dit probleem verlost te raken is gebruik te maken van Newton-iteratie i.p.v. vastpunt-iteratie. Het pseudo-schema luidt aldus volgens (1.23)

$$(I_{ms} - h A \otimes J) \Delta k^{[\nu]} = \Gamma^{[\nu]}, \quad \nu = 0, 1, \dots, \quad (10.39)$$

waarbij

$$J = \frac{\partial f}{\partial y}(k^{[0]}), \quad \Delta k^{[\nu]} = k^{[\nu+1]} - k^{[\nu]}, \quad \Gamma^{[\nu]} = -k^{[\nu]} + \phi(k^{[\nu]}).$$

Het meest energie wretende numerieke werk bestaat er uit in elke stap een LU decompositie uit te voeren van een $m s \times m s$ matrix, wat neerkomt op $\mathcal{O}((m s)^3)$ bewerkingen. Soms kan men dit aantal bewerkingen gevoelig reduceren door gebruik te maken van de Jordaan-normale vorm M van de matrix A . Er bestaat voor iedere matrix A inderdaad een niet-singuliere matrix H zodat

$$M = H^{-1} A H = \begin{bmatrix} \mu_1 & \omega_1 & 0 & 0 & \dots & 0 \\ 0 & \mu_2 & \omega_2 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & & \vdots \\ 0 & 0 & \dots & 0 & \mu_{s-1} & \omega_{s-1} \\ 0 & 0 & \dots & 0 & 0 & \mu_s \end{bmatrix},$$

en waarbij de μ_i de eigenwaarden zijn van A en waarbij elke ω_j hetzij 0, hetzij 1 is.

Beschouw nu de transformatie van $k^{[\nu]}$ en $\Gamma^{[\nu]}$ naar $\hat{k}^{[\nu]}$ en $\hat{\Gamma}^{[\nu]}$ gegeven door

$$\hat{k}^{[\nu]} := (H^{-1} \otimes I_m) k^{[\nu]}, \quad \hat{\Gamma}^{[\nu]} := (H^{-1} \otimes I_m) \Gamma^{[\nu]}.$$

Dan is wegens eigenschap (2) de inverse transformatie

$$k^{[\nu]} := (H \otimes I_m) \hat{k}^{[\nu]}, \quad \Gamma^{[\nu]} := (H \otimes I_m) \hat{\Gamma}^{[\nu]}.$$

Dan wordt (10.39) met $\Delta \hat{k}^{[\nu]} = \hat{k}^{[\nu+1]} - \hat{k}^{[\nu]}$:

$$(I_{ms} - h A \otimes J) (H \otimes I_m) \Delta \hat{k}^{[\nu]} = (H \otimes I_m) \hat{\Gamma}^{[\nu]},$$

of

$$[I_{ms} - h(H \otimes I_m)^{-1} (A \otimes J) (H \otimes I_m)] \Delta \hat{k}^{[\nu]} = \hat{\Gamma}^{[\nu]},$$

Via eigenschappen (1) en (2) bekommen we

$$\begin{aligned} (H \otimes I_m)^{-1} (A \otimes J) (H \otimes I_m) &= (H^{-1} \otimes I_m) (A H \otimes J) \\ &= H^{-1} A H \otimes J \\ &= M \otimes J, \end{aligned}$$

zodat

$$(I_{ms} - h M \otimes J) \Delta \hat{k}^{[\nu]} = \hat{\Gamma}^{[\nu]}, \quad \nu = 0, 1, \dots, \quad (10.40)$$

wat de Newtoniaanse iteratie uitdrukt in termen van de getransformeerde veranderlijken $\hat{k}^{[\nu]}$ en $\hat{\Gamma}^{[\nu]}$. Het belangrijk verschil tussen (10.39) en (10.40) is dat deze laatste de LU decompositie vereist van s blokken $I_m - h \mu_i J$, wat computationeel neerkomt op nog slechts $\mathcal{O}(s m^3)$ bewerkingen. Nog grotere winst treedt op indien de methode van het SIRK-type (*singly implicit*) is : dit zijn volledig IRKMn waarvoor de s eigenwaarden μ_i van A dezelfde numerieke waarde hebben. Op deze manier worden SIRKMn bijna even efficiënt als DIRKMn. Men kan SIRKMn construeren van willekeurige orde. Merk wel op dat de methode minder attractief wordt indien de matrix A complexe eigenwaarden bezit, wat meteen ook H en de getransformeerde variabelen $\hat{k}^{[\nu]}$ en $\hat{\Gamma}^{[\nu]}$ complex maakt.

Voorbeeld 10.9.1

Beschouw vooreerst een 2-traps RKM waarbij de matrix A twee reële verschillende eigenwaarden μ_1 en μ_2 bezit. Dan is

$$M = \begin{bmatrix} \mu_1 & 0 \\ 0 & \mu_2 \end{bmatrix}, \quad I_{2m} - h M \otimes J = \begin{bmatrix} I_m - h \mu_1 J & 0 \\ 0 & I_m - h \mu_2 J \end{bmatrix}.$$

We berekenen twee LU decomposities van $m \times m$ matrices, nl.

$$I_m - h \mu_1 J = L_1 U_1 \quad \text{en} \quad I_m - h \mu_2 J = L_2 U_2.$$

Dan is

$$I_{2m} - h M \otimes J = \begin{bmatrix} L_1 U_1 & 0 \\ 0 & L_2 U_2 \end{bmatrix} = LU,$$

met

$$L = \begin{bmatrix} L_1 & 0 \\ 0 & L_2 \end{bmatrix}, \quad U = \begin{bmatrix} U_1 & 0 \\ 0 & U_2 \end{bmatrix}.$$

We beschouwen ook een 2-traps SIRKM. De matrix A bezit dan een eigenwaarde van multiplicitéit 2 en

$$M = \begin{bmatrix} \mu & 1 \\ 0 & \mu \end{bmatrix}, \quad I_{2m} - h M \otimes J = \begin{bmatrix} I_m - h \mu J & -h J \\ 0 & I_m - h \mu J \end{bmatrix}.$$

We berekenen één LU decompositie $I_m - h \mu J = L_1 U_1$, zodat

$$I_{2m} - h M \otimes J = \begin{bmatrix} L_1 U_1 & -h J \\ 0 & L_1 U_1 \end{bmatrix} = LU,$$

met

$$L = \begin{bmatrix} L_1 & 0 \\ 0 & L_1 \end{bmatrix}, \quad U = \begin{bmatrix} U_1 & -h L_1^{-1} J \\ 0 & U_1 \end{bmatrix}.$$

□

Net zoals voor ERKMn vormt het wijzigen van de stap h geen probleem en foutschatting kan gebeuren op basis van hetzij Richardson-extrapolatie of door inbedding. Voor ERKMn hebben we gezien dat (in termen van het aantal functie-evaluaties per stap) inbedding efficiënter was dan extrapolatie. Dit argument is van significant minder belang voor niet-ERKMn vermits het grote rekenwerk sluipt in de LU decompositie. Om impliciete RK-algoritmen werkelijk competitief te maken moet men de kosten ten gevolge van het impliciet zijn zoveel mogelijk beperken. De Butcher-transformatie in combinatie met SIRKMn beperken de kosten bvb. tot een niveau dat vergelijkbaar is met dat van de VSVO codes gebaseerd op LMM van het BDF-type. De code STRIDE (Burrage, Butcher en Chipman, 1979) is een code waarin deze technieken worden gebruikt.

10.9.3 Adaptieve methoden

Het gebruik van impliciete methoden in het kader van stijve problemen noopt tot het gebruik van Newton-iteratie, wat op zijn beurt leidt tot de introductie van de Jacobiaan. Uit deze observatie is het idee ontstaan om *adaptieve* methoden te construeren, waarbij het gebruik van de Jacobiaan reeds in de methode ingebakken zit.

In het kader van RKMn moeten hier de *Rosenbrock*-methoden vermeld worden. Naar deze methoden is de laatste jaren veel onderzoek verricht en vele vormen zijn dan ook bestudeerd. De meest bekende vorm is :

Definitie 10.9.1 *De s-traps Rosenbrock-methode voor autonome problemen is gegeven door*

$$k_i = f\left(y_n + h \sum_{j=1}^{i-1} \alpha_{ij} k_j\right) + h J \sum_{j=1}^i \gamma_{ij} k_j, \quad i = 1, 2, \dots, s$$

$$y_{n+1} = y_n + \sum_{j=1}^s b_j k_j$$

Hierin zijn α_{ij} , γ_{ij} en b_i de bepalende coëfficiënten en $J = \frac{\partial f}{\partial y}(y_n)$. □

Rosenbrock-methoden vereisen aldus de oplossing van een lineair $m \times m$ stelsel in k_i voor elk van de s trappen. Het is opnieuw evident dat indien $\gamma_{11} = \gamma_{22} = \dots = \gamma_{ss}$ slechts 1 LU decompositie per stap moet doorgevoerd worden.

Onder de adaptieve methoden van het meerstapstype moeten we zeker de *Obreshkov*-methoden vermelden :

Definitie 10.9.2 *De algemeenste k-stap Obreshkov-methode voor autonome problemen die hoogstens tweede orde afgeleiden bevat, is gegeven door*

$$\sum_{j=0}^k \alpha_j y_{n+j} = h \sum_{j=0}^k \beta_j f_{n+j} + h^2 \sum_{j=0}^k \gamma_j g_{n+j},$$

met $g(y) := \frac{\partial f}{\partial y}(y) f(y)$, $\alpha_k = 1$ en $|\alpha_0| + |\beta_0| + |\gamma_0| \neq 0$. □