Numerical Solution of Linear, Nonhomogeneous Differential Equation Systems via Padé Approximation

Kenneth C. Johnson

KJ Innovation

kjinnovation@earthlink.net

(Posted October 31, 2016.)

Abstract

This paper generalizes an earlier investigation of linear differential equation solutions via Padé approximation (viXra:1509.0286), for the case of nonhomogeneous equations. Formulas are provided for approximation orders 2, 4, 6, and 8, for both constant-coefficient and functional-coefficient cases. The scale-and-square algorithm for the constant-coefficient case is generalized for nonhomogeneous equations. Implementation details including step size initialization and tolerance control are discussed.

1. Introduction

An earlier study [1] investigated solutions of the linear differential equation $F'[x] = D[x] F[x]$ via Padé approximation: $F[h] \approx Q[h]^{-1} Q[-h] F[-h]$, where $Q[h]$ is a polynomial. (D and Q are square matrices; F may be a column vector or a multi-column matrix. Square braces "$[\ldots]$" delimit function arguments, while round braces "$(\ldots)$" are reserved for grouping.)

We consider here the more general nonhomogeneous equation,

$$F'[x] = D[x] F[x] + C[x] \,. \tag{1}$$

(C is a vector or matrix, size-matched to F.) For this case, Eq. (7) in [1] is generalized by including an additional matrix polynomial R on the left,

$$\tfrac{1}{2}(R[h] - R[-h]) + Q[h] F[h] - Q[-h] F[-h] = O\, h^{2n+1} \,, \tag{2}$$

The Q polynomial has the form given in [1]; it is determined from D and has no dependence on C. The R polynomial depends on both D and C and has a linear dependence on C. In some cases $R[h]$ is an odd function of h ($R[-h] = -R[h]$), in which case the $\tfrac{1}{2}(R[h] - R[-h])$ term in Eq. (2) is replaced by $R[h]$.

The homogeneous equation ($C = \mathbf{0}$ in Eq. (1)) has solutions of the form $F[x] = \Phi[x] F[0]$, where $\Phi[x]$ is the solution of the initial value problem,

$$\Phi'[x] = D[x] \Phi[x], \quad \Phi[0] = \mathbf{I} \,, \tag{3}$$

where $\mathbf{I}$ is an identity matrix. For the nonhomogeneous case, general solutions of Eq. (1) (with $F[x]$ specified at $x = x_0$) are of the form

$$F[x] = \Phi[x]\left(\int_{x_0}^{x} \Phi[t]^{-1}\, C[t]\, dt + \Phi[x_0]^{-1}\, F[x_0]\right). \tag{4}$$

For the special case of constant D, $\Phi[x]$ is an exponential matrix,

$$\Phi[x] = \exp[D\,x] \quad (\text{constant } D). \tag{5}$$

If C is also constant, Eq. (4) reduces to

$$F[x] = D^{-1}(\exp[D\,(x - x_0)] - \mathbf{I})\,C + \exp[D\,(x - x_0)]\,F[x_0] \quad (\text{constant } D \text{ and } C). \tag{6}$$

This formula cannot be used when D is singular (even though the first left-hand term is well defined by its Taylor series), and it has poor numerical precision will D is near-singular or $x - x_0$ is very small. But the Padé approximation method based on Eq. (2) does not have these limitations. The method can be used, for example, to robustly calculate $D^{-1}(\exp[D] - \mathbf{I})$ (even for singular D) by setting $F[x_0] = \mathbf{0}$, $C = \mathbf{I}$, and $x - x_0 = 1$.

Eq. (2) is used to integrate $F[x]$ across a small interval, from $x = -h$ to $x = h$. The independent variable x can be scaled and shifted to convert this to an integration from $x = x_0$ to $x = x_0 + \Delta x$ for a sufficiently small Δx, and multiple such integrations are concatenated to calculate $F[x]$ over a large integration interval. For the homogeneous, constant-coefficient case (D constant, $C = \mathbf{0}$), the concatenation can be efficiently implemented using a "scale-and-square" technique based on the relation

$$\exp[D\,x] = (\ldots((\exp[2^{-j}\,D\,x])\overbrace{^{2})^{2}\ldots)^{2}}^{j\times}\,. \tag{7}$$

(For some sufficiently large integer j, a Padé approximant is used to calculate $\exp[2^{-j}\,D\,x]$, and the result is squared j times to obtain $\exp[D\,x]$.) This algorithm can be generalized for the nonhomogeneous case with constant D and C.

Section 2 lists polynomial functions Q and R in Eq. (2) for various approximation orders. Section 3 outlines the scale-and-square algorithm, generalized for the nonhomogeneous case. Section 4 discusses the choice of integration interval size. The Appendix provides Mathematica code validating the results of section 2.

2. Padé-approximation formulas

The Q and R polynomials in Eq. (2) are listed below for approximation orders 2, 4, 6, and 8, first for the case of constant D and C and then for the non-constant case. The constant-coefficient formulas include an estimate of the approximation error, which is useful for determining the integration step size. For the non-constant case similar error approximations would be too complex to be of much use, but the constant-coefficient formulas can be used for step size initialization as described in section 4.

$$\begin{aligned}
&\text{order-2 accuracy, constant } D, C:\\
&Q[h] = \mathbf{I} - h\,D\\
&R[h] = -2\,h\,C\\
&R[h] + Q[h]\,F[h] - Q[-h]\,F[-h] = -\tfrac{2}{3}h^3\,D^2\,(C + D\,F[0]) + O\,h^5
\end{aligned} \tag{8}$$

$$\begin{aligned}
&\text{order-4 accuracy, constant } D, C:\\
&Q[h] = \mathbf{I} - h\,D + \tfrac{1}{3}h^2\,D^2\\
&R[h] = -2\,h\,C\\
&R[h] + Q[h]\,F[h] - Q[-h]\,F[-h] = \tfrac{2}{45}h^5\,D^4\,(C + D\,F[0]) + O\,h^7
\end{aligned} \tag{9}$$

$$\begin{aligned}
&\text{order-6 accuracy, constant } D, C:\\
&Q[h] = \mathbf{I} - h\,D + \tfrac{2}{5}h^2\,D^2 - \tfrac{1}{15}h^3\,D^3\\
&R[h] = -2\,h\,C - \tfrac{2}{15}h^3\,D^2\,C\\
&R[h] + Q[h]\,F[h] - Q[-h]\,F[-h] = -\tfrac{2}{1575}h^7\,D^6\,(C + D\,F[0]) + O\,h^9
\end{aligned} \tag{10}$$

$$\begin{aligned}
&\text{order-8 accuracy, constant } D, C:\\
&Q[h] = \mathbf{I} - h\,D + \tfrac{3}{7}h^2\,D^2 - \tfrac{2}{21}h^3\,D^3 + \tfrac{1}{105}h^4\,D^4\\
&R[h] = -2\,h\,C - \tfrac{4}{21}h^3\,D^2\,C\\
&R[h] + Q[h]\,F[h] - Q[-h]\,F[-h] = \tfrac{2}{99225}h^9\,D^8\,(C + D\,F[0]) + O\,h^{11}
\end{aligned} \tag{11}$$

Eq’s. (8)-(11) are specializations of the following general formula, in which the n subscript is applied to the Q and R matrices to identify the accuracy order ($2\,n$):

$$\begin{aligned}
&\text{order-}2\,n\text{ accuracy, constant } D, C:\\
&Q_n[h] = \sum_{j=0}^{n} \frac{(2\,n - j)!\,n!}{j\,!(2\,n)!(n-j)!}(-2\,h\,D)^j\\
&R_n[h] = 2\sum_{\substack{j\ \text{odd},\\ 1\le j\le n}} \frac{(2\,n - j)!\,n!}{j\,!(2\,n)!(n-j)!}(-2\,h\,D)^{j-1}\,(-2\,h\,C)\\
&residual_n[h] = \frac{(-1)^n\,(n!)^2\,(2\,h)^{2n+1}}{(2\,n)!(2\,n+1)!}D^{2n}\,(C + D\,F[0])\\
&R_n[h] + Q_n[h]\,F[h] - Q_n[-h]\,F[-h] = residual_n[h] + O\,h^{2n+3}
\end{aligned} \tag{12}$$

The subscripted functions in Eq’s. (12) can be efficiently calculated by using the following recursion relations,

$$
\begin{aligned}
&Q_0[h] = \mathbf{I}\,, \\
&Q_1[h] = \mathbf{I} - h\,D\,, \\
&Q_{n+1}[h] = Q_n[h] + \frac{h^2\,D^2}{(2\,n+1)\,(2\,n-1)}\,Q_{n-1}[h]
\end{aligned}
\tag{13}
$$

$$
\begin{aligned}
&R_0[h] = \mathbf{0}\,, \\
&R_1[h] = -2\,h\,C\,, \\
&R_{n+1}[h] = R_n[h] + \frac{h^2\,D^2}{(2\,n+1)\,(2\,n-1)}\,R_{n-1}[h]
\end{aligned}
\tag{14}
$$

$$
\begin{aligned}
&residual_0[h] = 2\,h\,(C + D\,F[0]) \\
&residual_{n+1}[h] = \frac{-h^2\;D^2}{(2\,n+1)\,(2\,n+3)}\,residual_n[h]
\end{aligned}
\tag{15}
$$

For non-constant D and C general formulas such as Eq. (12) have not been developed, but several special cases are listed below.

$$
\begin{aligned}
&\text{order-2 accuracy, non-constant } D, C: \\
&Q[h] = \mathbf{I} - h\,D[0] \\
&R[h] = -2\,h\,C[0] \\
&R[h] + Q[h]\,F[h] - Q[-h]\,F[-h] = O\,h^3
\end{aligned}
\tag{16}
$$

$$
\begin{aligned}
&\text{order-4 accuracy, non-constant } D, C: \\
&Q[h] = \mathbf{I} - h\,(-\tfrac{1}{6}\,D[-h] + \tfrac{2}{3}\,D[0] + \tfrac{1}{2}\,D[h]) + \tfrac{1}{3}\,h^2\;D[h]^2 \\
&R[h] = -2\,h\,(-\tfrac{1}{6}\,C[-h] + \tfrac{2}{3}\,C[0] + \tfrac{1}{2}\,C[h]) + \tfrac{2}{3}\,h^2\;D[h]\,C[h] \\
&\tfrac{1}{2}\,(R[h] - R[-h]) + Q[h]\,F[h] - Q[-h]\,F[-h] = O\,h^5
\end{aligned}
\tag{17}
$$

$$
\begin{aligned}
&\text{order-6 accuracy, non-constant } D, C: \\
&Q_3[h] = \mathbf{I} - h\,(\tfrac{2}{45}\,D[-\tfrac{1}{2}\,h] + \tfrac{2}{15}\,D[0] + \tfrac{2}{3}\,D[\tfrac{1}{2}\,h] + \tfrac{7}{45}\,D[h]) + \\
&\quad (\tfrac{1}{15}\,D[-\tfrac{1}{2}\,h] + \tfrac{1}{5}\,D[0] + \tfrac{11}{15}\,D[\tfrac{1}{2}\,h]) \\
&\quad (\tfrac{2}{5}\,h^2\,(\tfrac{1}{9}\,D[-\tfrac{1}{2}\,h] - \tfrac{1}{2}\,D[0] + D[\tfrac{1}{2}\,h] + \tfrac{7}{18}\,D[h]) - \tfrac{1}{15}\,h^3\;D[h]^2) \\
&R[h] = -2\,h\,(\tfrac{2}{45}\,C[-\tfrac{1}{2}\,h] + \tfrac{2}{15}\,C[0] + \tfrac{2}{3}\,C[\tfrac{1}{2}\,h] + \tfrac{7}{45}\,C[h]) + \\
&\quad 2\,(\tfrac{1}{15}\,D[-\tfrac{1}{2}\,h] + \tfrac{1}{5}\,D[0] + \tfrac{11}{15}\,D[\tfrac{1}{2}\,h]) \\
&\quad (\tfrac{2}{5}\,h^2\,(\tfrac{1}{9}\,C[-\tfrac{1}{2}\,h] - \tfrac{1}{2}\,C[0] + C[\tfrac{1}{2}\,h] + \tfrac{7}{18}\,C[h]) - \tfrac{1}{15}\,h^3\;D[h]\,C[h]) \\
&\tfrac{1}{2}\,(R[h] - R[-h]) + Q[h]\,F[h] - Q[-h]\,F[-h] = O\,h^7
\end{aligned}
\tag{18}
$$

order-8 accuracy, non-constant D, C :

$$\begin{aligned}
&L_1[h] = \tfrac{403}{16800} D[-h] - \tfrac{279}{2800} D[-\tfrac{2}{3} h] + \tfrac{99}{800} D[-\tfrac{1}{3} h] \\
&\quad + \tfrac{34}{105} D[0] - \tfrac{333}{5600} D[\tfrac{1}{3} h] + \tfrac{1719}{2800} D[\tfrac{2}{3} h] + \tfrac{1237}{16800} D[h] \\
&L_2[h] = \tfrac{57}{1120} D[-h] - \tfrac{243}{560} D[-\tfrac{2}{3} h] + \tfrac{1269}{1120} D[-\tfrac{1}{3} h] - \tfrac{3}{4} D[0] \\
&\quad + \tfrac{891}{1120} D[\tfrac{1}{3} h] + \tfrac{27}{112} D[\tfrac{2}{3} h] - \tfrac{41}{1120} D[h] \\
&L_3[h] = -\tfrac{2067}{9680} D[-h] + \tfrac{6021}{4840} D[-\tfrac{2}{3} h] - \tfrac{5805}{1936} D[-\tfrac{1}{3} h] + \tfrac{1863}{484} D[0] \\
&\quad - \tfrac{5697}{1936} D[\tfrac{1}{3} h] + \tfrac{10341}{4840} D[\tfrac{2}{3} h] - \tfrac{727}{9680} D[h] \\
&L_4[h] = \tfrac{63}{16} D[-h] - \tfrac{1809}{40} D[-\tfrac{2}{3} h] + \tfrac{2295}{16} D[-\tfrac{1}{3} h] - \tfrac{801}{4} D[0] \\
&\quad + \tfrac{2133}{16} D[\tfrac{1}{3} h] - \tfrac{297}{8} D[\tfrac{2}{3} h] + \tfrac{233}{80} D[h] \\
&L_5[h] = \tfrac{123}{160} D[-h] - \tfrac{135}{8} D[-\tfrac{2}{3} h] + \tfrac{2295}{32} D[-\tfrac{1}{3} h] - 132\, D[0] \\
&\quad + \tfrac{3861}{32} D[\tfrac{1}{3} h] - \tfrac{1917}{40} D[\tfrac{2}{3} h] + \tfrac{149}{32} D[h] \\
&L_6[h] = -\tfrac{6}{35} D[-h] + \tfrac{27}{10} D[-\tfrac{2}{3} h] - \tfrac{1053}{112} D[-\tfrac{1}{3} h] + \tfrac{57}{4} D[0] \\
&\quad - \tfrac{621}{56} D[\tfrac{1}{3} h] + \tfrac{729}{140} D[\tfrac{2}{3} h] - \tfrac{277}{560} D[h] \\
&L_7[h] = \tfrac{403}{16800} C[-h] - \tfrac{279}{2800} C[-\tfrac{2}{3} h] + \tfrac{99}{800} C[-\tfrac{1}{3} h] + \tfrac{34}{105} C[0] \\
&\quad - \tfrac{333}{5600} C[\tfrac{1}{3} h] + \tfrac{1719}{2800} C[\tfrac{2}{3} h] + \tfrac{1237}{16800} C[h] \\
&L_8[h] = -\tfrac{2067}{9680} C[-h] + \tfrac{6021}{4840} C[-\tfrac{2}{3} h] - \tfrac{5805}{1936} C[-\tfrac{1}{3} h] + \tfrac{1863}{484} C[0] \\
&\quad - \tfrac{5697}{1936} C[\tfrac{1}{3} h] + \tfrac{10341}{4840} C[\tfrac{2}{3} h] - \tfrac{727}{9680} C[h] \\
&L_9[h] = \tfrac{123}{160} C[-h] - \tfrac{135}{8} C[-\tfrac{2}{3} h] + \tfrac{2295}{32} C[-\tfrac{1}{3} h] - 132\, C[0] \\
&\quad + \tfrac{3861}{32} C[\tfrac{1}{3} h] - \tfrac{1917}{40} C[\tfrac{2}{3} h] + \tfrac{149}{32} C[h] \\
&Q[h] = \mathbf{I} - h\, L_1[h] + L_2[h](\tfrac{121}{315} h^2\, L_3[h] - \tfrac{2}{315} h^3\, L_4[h]\, L_5[h]) \\
&\quad + \left(\tfrac{2}{45} h^2\, L_6[h] + L_2[h](-\tfrac{4}{45} h^3\, L_6[h] + \tfrac{1}{105} h^4\, D[h]^2)\right) D[h] \\
&R[h] = -2\, h\, L_7[h] + 2\, L_2[h](\tfrac{121}{315} h^2\, L_8[h] - \tfrac{2}{315} h^3\, L_4[h]\, L_9[h]) \\
&\quad + 2\left(\tfrac{2}{45} h^2\, L_6[h] + L_2[h](-\tfrac{4}{45} h^3\, L_6[h] + \tfrac{1}{105} h^4\, D[h]^2)\right) C[h] \\
&\tfrac{1}{2}(R[h] - R[-h]) + Q[h]\, F[h] - Q[-h]\, F[-h] = O\, h^9
\end{aligned} \tag{19}$$

Note the commonality of subexpressions in the $Q[h]$ and $R[h]$ formulas in Eq. (19). Also, the product factors $D[h]^2$ in Eq's. (17)-(19) can be re-used in the next integration step as $D[-h]^2$ for calculating $Q[-h]$ and $R[-h]$. The products $D[h]C[h]$ in Eq's. (17) and (18) can similarly be carried over to the next step.

3. Scale-and-square algorithm

For the constant-coefficient case, Eq. (6) can be formulated as

$$F[x_0 + \Delta x] = \Gamma_1\, C + \Phi\, F[x_0] \tag{20}$$

where

$$\Gamma_1 = D^{-1}(\exp[D\,\Delta x] - \mathbf{I}), \tag{21}$$

$$\Phi = \exp[D\,\Delta x]. \tag{22}$$

The Γ_1 and Φ matrices, which depend on Δx but not on x_0, can be obtained from the Padé approximation, Eq. (2), for small Δx,

$$F[h] \approx Q[h]^{-1}(Q[-h]\,F[-h] - R[h]), \tag{23}$$

(The term $\frac{1}{2}(R[h] - R[-h])$ in Eq. (2) has been replaced by $R[h]$ because $R[h]$ is an odd function of h for the constant-coefficient case.) Eq. (23) is applied with $\Delta x = 2\,h$ and with the x coordinate origin shifted so that $x_0 = -h$. All of the R matrices in Eq's. (8)-(12) have a right-factor of C, which can be separated out to obtain Γ_1 in Eq. (20):

$$\Gamma_1\,C = -Q[h]^{-1}\,R[h], \tag{24}$$

$$\Phi = Q[h]^{-1}\,Q[-h], \tag{25}$$

Eq. (20) is applied recursively to integrate $F[x]$ over a large, m-step integration interval,

$$F[x_0 + m\,\Delta x] = \Gamma_m\,C + \Phi^m\,F[x_0] \tag{26}$$

where

$$\Gamma_m = (\mathbf{I} + \Phi + \Phi^2 + \ldots + \Phi^{m-1})\,\Gamma_1. \tag{27}$$

Given Γ_m and Φ^m for any particular integer m, Γ_{2m} and Φ^{2m} are obtained as

$$\Gamma_{2m} = \Gamma_m + \Phi^m\,\Gamma_m, \tag{28}$$

$$\Phi^{2m} = (\Phi^m)^2. \tag{29}$$

Eq. (29) is the basis of the standard scale-and-square algorithm for homogeneous linear differential equations, and Eq. (28) generalizes the method for nonhomogeneous equations.

In implementing the Padé approximation it is advantageous to calculate the even and odd parts of $Q[h]$ separately so that $Q[h]$ and $Q[-h]$ can both be generated with little computational overhead,

$$Q[\pm h] = Q^{[\text{even}]}[h] \pm Q^{[\text{odd}]}[h], \tag{30}$$

For small Δx the matrix Γ_1 (Eq. (21)) is approximately proportional to Δx, but Φ is approximately equal to $\mathbf{I}$ with a small Δx-proportional increment. To avoid possible precision loss in the Φ diagonal elements, the matrix can be calculated with $\mathbf{I}$ subtracted off. Eq. (25) is modified as

$$\Phi - \mathbf{I} = Q[h]^{-1}(Q[-h] - Q[h]) = -2\,Q[h]^{-1}\,Q^{[\text{odd}]}[h], \tag{31}$$

The $\mathbf{I}$ separation is preserved through the scale-and-square process by modifying Eq's. (28) and (29) as follows,

$$\Gamma_{2m} = 2\,\Gamma_m + (\Phi^m - \mathbf{I})\,\Gamma_m\,, \tag{32}$$

$$(\Phi^{2m} - \mathbf{I}) = (\Phi^m - \mathbf{I})^2 + 2\,(\Phi^m - \mathbf{I})\,. \tag{33}$$

4. Error analysis and tolerance control

Continuing with the constant-coefficient case, the inaccuracy of the Padé approximation will result in errors in Eq. (20). Denoting error terms by the prefix "¢", the calculated error in $F[x_0 + \Delta x]$ is

$$\text{¢}\,F[x_0 + \Delta x] = (\text{¢}\Gamma_1)\,C + (\text{¢}\Phi)\,F[x_0] \tag{34}$$

The errors in Γ_1 and Φ can be obtained from Eq. (12), in which $F[h]$ is calculated by ignoring the residual term ($residual_n[h]$),

$$Q_n[h]\,\text{¢}\,F[h] \approx -residual_n[h] \tag{35}$$

For small h, $Q_n[h]$ is close to $\mathbf{I}$ (i.e., $Q_n[h] = \mathbf{I} + O\,h$) and Eq. (35) simplifies to

$$\text{¢}\,F[h] \approx -residual_n[h] \tag{36}$$

A comparison of Eq's. (34) and (36), with $x_0 = -h$, $\Delta x = 2\,h$, and with $residual_n[h]$ defined in Eq's. (12), yields the following expressions for $\text{¢}\Gamma_1$ and $\text{¢}\Phi$ (from Eq. (12))

$$\text{¢}\Gamma_1 = -\frac{(-1)^n\,(n!)^2\,(\Delta x)^{2n+1}}{(2\,n)!(2\,n+1)!}\,D^{2n} \quad (\Delta x = 2\,h) \tag{37}$$

$$\text{¢}\Phi = \text{¢}\Gamma_1\,D \tag{38}$$

The cumulative errors in m integration steps (Eq. (26)) are represented as

$$\text{¢}\,F[x_0 + m\,\Delta x] = (\text{¢}\Gamma_m)\,C + (\text{¢}\,(\Phi^m))\,F[x_0] \tag{39}$$

The $\text{¢}\,(\Phi^m)$ error has the approximate form

$$\text{¢}\,(\Phi^m) \approx (\text{¢}\Phi)\,\Phi^{m-1} + \Phi\,(\text{¢}\Phi)\,\Phi^{m-2} + \ldots + \Phi^{m-1}\,(\text{¢}\Phi) \tag{40}$$

Φ is close to $\mathbf{I}$ ($\Phi = \mathbf{I} + O\,\Delta x$, Eq. (22)), so Eq. (40) simplifies to

$$\text{¢}\,(\Phi^m) \approx m\,\text{¢}\Phi \tag{41}$$

A similar relation is applied to the Φ powers on the right side of Eq. (27),

$$\text{¢}\Gamma_m \approx (\text{¢}\Phi + 2\,\text{¢}\Phi + \ldots + (m-1)\,\text{¢}\Phi)\,\Gamma_1 + (\mathbf{I} + \Phi + \Phi^2 + \ldots + \Phi^{m-1})\,\text{¢}\Gamma_1\,. \tag{42}$$

Making the approximations $\Phi \approx \mathbf{I}$ and $\Gamma_1 \approx \mathbf{I}\,\Delta x$ (from Eq. (21)), Eq. (42) simplifies to

$$\text{¢}\Gamma_m \approx \tfrac{1}{2}\,m\,(m-1)\,(\text{¢}\Phi)\,\mathbf{I}\,\Delta x + m\,\text{¢}\Gamma_1\,. \tag{43}$$

With substitution from Eq. (38), this further simplifies to

$$\text{¢}\Gamma_m \approx \tfrac{1}{2}\, m\,(m-1)\,(\text{¢}\Gamma_1)\, D\,\Delta x + m\,\text{¢}\Gamma_1 \approx m\,\text{¢}\Gamma_1 . \tag{44}$$

(The $D\,\Delta x$ is small in relation to unity and can therefore be neglected in Eq. (44).)

Eq's. (41), (44), and (38) are substituted in Eq. (39),

$$\text{¢}\, F[x_0 + m\,\Delta x] \approx m\,(\text{¢}\Gamma_1)\,(C + D\,F[x_0]) \tag{45}$$

(To a first-order approximation the single-step error $\text{¢}\, F[x_0 + \Delta x]$ is simply multiplied by m in taking m integration steps.) Eq. (45) includes two error terms: a dimensionless relative error factor $m\,(\text{¢}\Gamma_1)\,D$, which is applied to $F[x_0]$, and an absolute error term $m\,(\text{¢}\Gamma_1)\,C$, which has the same dimensional units as F. Δx can be chosen to impose an approximate tolerance bound on both error terms,

$$m\left\|(\text{¢}\Gamma_1)\,C\right\| \le abs\_tol, \quad m\left\|(\text{¢}\Gamma_1)\,D\right\| \le rel\_tol \tag{46}$$

where abs_tol and rel_tol are specified absolute and relative tolerance bounds and $\|\ldots\|$ is the Frobenius norm. With $\text{¢}\Gamma_1$ substituted from Eq. (37), the following conditions are obtained from Eq. (46),

$$m\,\frac{(n!)^2\,(x_{range}\,/\,m)^{2n+1}}{(2\,n)!(2\,n+1)!}\left\|D^{2n}\,C\right\| \le abs\_tol, \quad m\,\frac{(n!)^2\,(x_{range}\,/\,m)^{2n+1}}{(2\,n)!(2\,n+1)!}\left\|D^{2n+1}\right\| \le rel\_tol \tag{47}$$

where x_{range} is the total integration range,

$$x_{range} = m\,\Delta x \tag{48}$$

Eq. (47) implies the following limit on m,

$$m \ge \left(\frac{(n!)^2\, {x_{range}}^{2n+1}}{(2\,n)!(2\,n+1)!}\max\left[\frac{\left\|D^{2n}\,C\right\|}{abs\_tol}, \frac{\left\|D^{2n+1}\right\|}{rel\_tol}\right]\right)^{1/(2n)} \tag{49}$$

With the scale-and-square algorithm, m is a power of 2 and Eq. (49) translates to

$$m = 2^j, \quad j \ge \frac{1}{2\,n}\log_2\left[\frac{(n!)^2\, {x_{range}}^{2n+1}}{(2\,n)!(2\,n+1)!}\max\left[\frac{\left\|D^{2n}\,C\right\|}{abs\_tol}, \frac{\left\|D^{2n+1}\right\|}{rel\_tol}\right]\right] \tag{50}$$

The above formulas are not directly applicable to the non-constant-coefficient case, but Eq. (49) can be used to obtain an initial integration step size Δx, using values of $D[x]$ and $C[x]$ at the beginning of the integration interval. Then, at each integration step, $F[x + \Delta x]$ is determined from $F[x]$ by two estimation methods to obtain an estimated integration error via Richardson extrapolation. A first estimate $F_1[x + \Delta x]$ is obtained by making a single-step Padé approximation, and a second estimate $F_2[x + \Delta x]$ is obtained by making two Padé approximation steps with step size $\tfrac{1}{2}\Delta x$. The errors in these estimates are approximately

$$\text{¢}\, F_1[x + \Delta x] \approx A\,\Delta x^{2n+1}, \quad \text{¢}\, F_2[x + \Delta x] \approx 2\,A\,(\tfrac{1}{2}\Delta x)^{2n+1} \tag{51}$$

where the accuracy order is $2n$, A is an undetermined matrix, and the factor of 2 is included in the second equality to account for the two steps. The following relation is obtained by eliminating A between Eq's. (51),

$$¢\, F_1[x+\Delta x] \approx 2^{2n}\, ¢\, F_2[x+\Delta x] \tag{52}$$

Applying the error correction to both estimates should give the same result,

$$F_1[x+\Delta x] - ¢\, F_1[x+\Delta x] = F_2[x+\Delta x] - ¢\, F_2[x+\Delta x] \tag{53}$$

$¢\, F_1[x+\Delta x]$ is eliminated from Eq's. (52) and (53) to obtain

$$¢\, F_2[x+\Delta x] = \frac{F_1[x+\Delta x] - F_2[x+\Delta x]}{2^{2n} - 1} \tag{54}$$

The integration step Δx is decreased or increased by factors of 2 to keep this estimated error within allowed tolerance bounds (i.e. the step is decreased if the error significantly exceeds the tolerance, and is increased if the error times 2^{2n+1} is well within the tolerance). Some excursion of the error over the tolerance limit can be allowed because the calculated $F_2[x+\Delta x]$ can be decremented by $¢\, F_2[x+\Delta x]$ from Eq. (54) to improve its accuracy.

The $F[h]$ value calculated from Eq. (2) can be separated into two components: an R-dependent term $F^{[R]}$ (which is not dependent on $F[-h]$), and a term $\Phi\, F[-h]$ (which is not dependent on R or C),

$$F[h] = F^{[R]} + \Phi\, F[-h]\,, \tag{55}$$

where

$$F^{[R]} = -\tfrac{1}{2} Q[h]^{-1} \, (R[h] - R[-h])\,, \tag{56}$$

$$\Phi = Q[h]^{-1}\, Q[-h]\,. \tag{57}$$

Similarly, the $F_1[x+\Delta x]$ term in Eq. (54) can be separated as

$$F_1[x+\Delta x] = F_1^{[R]} + \Phi_1\, F[x]\,. \tag{58}$$

The same separation is made for $F_2[x+\Delta x]$ in two steps,

$$\begin{aligned} &F_2[x+\tfrac{1}{2}\Delta x] = F_{2,1}^{[R]} + \Phi_{2,1}\, F[x], \\ &F_2[x+\Delta x] = F_{2,2}^{[R]} + \Phi_{2,2}\, F[x+\tfrac{1}{2}\Delta x] = F_{2,2}^{[R]} + \Phi_{2,2}\, (F_{2,1}^{[R]} + \Phi_{2,1}\, F[x]). \end{aligned} \tag{59}$$

This expression is of the form

$$F_2[x+\Delta x] = F_2^{[R]} + \Phi_2\, F[x] \quad (\text{with } F_2^{[R]} = F_{2,2}^{[R]} + \Phi_{2,2}\, F_{2,1}^{[R]},\ \Phi_2 = \Phi_{2,2}\, \Phi_{2,1})\,. \tag{60}$$

The estimated error $¢\, F_2[x+\Delta x]$ in Eq. (54) is correspondingly separated into $F[x]$-proportionate and $F[x]$-independent terms,

$$\text{¢}\,F_2[x+\Delta x]=\text{¢}\,F_2^{[R]}+\text{¢}\Phi_2\,F[x]\quad(\text{with }\text{¢}\,F_2^{[R]}=\frac{F_1^{[R]}-F_2^{[R]}}{2^{2n}-1},\ \text{¢}\Phi_2=\frac{\Phi_1-\Phi_2}{2^{2n}-1})\,. \qquad (61)$$

The following tolerance specifications are analogous to the homogeneous-case conditions (Eq. (46)), and are equivalent when D and C are constant,

$$\frac{x_{\text{range}}}{\Delta x}\left\|\text{¢}\,F_2^{[R]}\right\|\le abs\_tol,\quad \frac{x_{\text{range}}}{\Delta x}\left\|\text{¢}\Phi_2\right\|\le rel\_tol\,. \qquad (62)$$

These conditions can be used to control the integration step size.

Reference

[1] Johnson, K. “Numerical Solution of Linear, Homogeneous Differential Equation Systems via Padé Approximation.” (v2, posted April 22, 2016) http://vixra.org/abs/1509.0286.

Appendix: Mathematica verification of Eq’s. (8)-(11) and (16)-(19)

The calculations underlying Eq’s. (8)-(11) and (16)-(19) require non-commutative symbolic algebra. The following results are obtained using the NCAlgebra package for Mathematica, from the University of California, San Diego (http://math.ucsd.edu/~ncalg/). The Mathematica code loads the NCAlgebra package, adds some additional functionality, and verifies the equations.

```
(* Load NCAlgebra package (http://math.ucsd.edu/~ncalg/) *)
<< NC`
<< NCAlgebra`

(* Make all variables commutative by default.
  (Override the default noncommutativity of single-letter lowercase variables.) *)
Remove[a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z]

(* D0, C0, Dfn, Cfn, F, Q, and R represent matrices. D0 and C0 represent constants;
Dfn, Cfn, F, Q, and R represent functions, and "1" represents the identity matrix. *)
SetNonCommutative[D0, C0, Dfn, Cfn, F, Q, R];

(* Series and O (e.g. O[h]^n) do not work with NC types
 (e.g.: try Dfn[h]**F[h]+O[h]^2 or Series[Dfn[h]**F[h],{h,0,1}]). Define a variant that does work. *)
NCSeries[f_, {x_, x0_, n_}] := NCExpand[Sum[(D[f, {x, j}] / j ! /. x → x0) (x - x0) ^j, {j, 0, n}]] + O[x - x0] ^ (n + 1);

(* substD is a substitution rule for reducing derivatives of F using the relation F'[h]==Dfn[h]**F[h]+Cfn[h].
    Use "... //. substD" to eliminate all F derivatives.
    (The substD definition uses ":>",
     not "->" otherwise the substitutions will not work when x or n has a preassigned value.) *)
substD = Derivative[n_][F][x_] :> Derivative[n - 1][Dfn[#] ** F[#] + Cfn[#] &][x];

(* substD0 is a substitution rule for reducing derivatives of F using the relation F'[h]==
 D0**F[h]+C0. This specializes substD for the case where Dfn and Cfn are constant. *)
substD0 = Derivative[n_][F][x_] :> Derivative[n - 1][D0 ** F[#] + C0 &][x];


(* Eq 8 *)
Q[h_] := 1 - h D0;
R[h_] := -2 h C0;
Factor[NCExpand[Normal[NCSeries[R[h] + Q[h] ** F[h] - Q[-h] ** F[-h], {h, 0, 4}]] //. substD0]]

 2
- ─ h^3 (D0 ** D0 ** C0 + D0 ** D0 ** D0 ** F[0])
 3


(* Eq 9 *)
                      1
Q[h_] := 1 - h D0 + ─ h^2 D0 ** D0;
                      3
R[h_] := -2 h C0;
Factor[NCExpand[Normal[NCSeries[R[h] + Q[h] ** F[h] - Q[-h] ** F[-h], {h, 0, 6}]] //. substD0]]

 2
── h^5 (D0 ** D0 ** D0 ** D0 ** C0 + D0 ** D0 ** D0 ** D0 ** D0 ** F[0])
45


(* Eq 10 *)
                      2                     1
Q[h_] := 1 - h D0 + ─ h^2 D0 ** D0 - ── h^3 D0 ** D0 ** D0;
                      5                    15
                      2
R[h_] := -2 h C0 - ── h^3 D0 ** D0 ** C0;
                     15
Factor[NCExpand[Normal[NCSeries[R[h] + Q[h] ** F[h] - Q[-h] ** F[-h], {h, 0, 8}]] //. substD0]]

  2 h^7 (D0 ** D0 ** D0 ** D0 ** D0 ** D0 ** C0 + D0 ** D0 ** D0 ** D0 ** D0 ** D0 ** D0 ** F[0])
- ──────────────────────────────────────────────────────────────────────────────────────────────
                                              1575


(* Eq 11 *)
                      3                    2                            1
Q[h_] := 1 - h D0 + ─ h^2 D0 ** D0 - ── h^3 D0 ** D0 ** D0 + ─── h^4 D0 ** D0 ** D0 ** D0;
                      7                   21                          105
                      4
R[h_] := -2 h C0 - ── h^3 D0 ** D0 ** C0;
                     21
Factor[NCExpand[Normal[NCSeries[R[h] + Q[h] ** F[h] - Q[-h] ** F[-h], {h, 0, 10}]] //. substD0]]

   1
──────2 h^9 (D0 ** D0 ** D0 ** D0 ** D0 ** D0 ** D0 ** D0 ** C0 + D0 ** D0 ** D0 ** D0 ** D0 ** D0 ** D0 ** D0 ** D0 ** F[0])
99 225
```

```
(* Eq 16 *)
Q[h_] := 1 - h Dfn[0];
R[h_] := -2 h Cfn[0];
NCExpand[Normal[NCSeries[R[h] + Q[h] ** F[h] - Q[-h] ** F[-h], {h, 0, 2}]] //. substD]
```

```
0
```

```
(* Eq 17 *)
Q[h_] := 1 - h (-1/6 Dfn[-h] + 2/3 Dfn[0] + 1/2 Dfn[h]) + 1/3 h^2 Dfn[h] ** Dfn[h];
R[h_] := -2 h (-1/6 Cfn[-h] + 2/3 Cfn[0] + 1/2 Cfn[h]) + 2/3 h^2 Dfn[h] ** Cfn[h];
NCExpand[Normal[NCSeries[1/2 (R[h] - R[-h]) + Q[h] ** F[h] - Q[-h] ** F[-h], {h, 0, 4}]] //. substD]
```

```
0
```

```
(* Eq 18 *)
Q[h_] := 1 - h (2/45 Dfn[-h/2] + 2/15 Dfn[0] + 2/3 Dfn[h/2] + 7/45 Dfn[h]) +
   (1/15 Dfn[-h/2] + 1/5 Dfn[0] + 11/15 Dfn[h/2]) **
    (2/5 h^2 (1/9 Dfn[-h/2] - 1/2 Dfn[0] + Dfn[h/2] + 7/18 Dfn[h]) - 1/15 h^3 Dfn[h] ** Dfn[h]);
R[h_] := -2 h (2/45 Cfn[-h/2] + 2/15 Cfn[0] + 2/3 Cfn[h/2] + 7/45 Cfn[h]) +
   2 (1/15 Dfn[-h/2] + 1/5 Dfn[0] + 11/15 Dfn[h/2]) **
     (2/5 h^2 (1/9 Cfn[-h/2] - 1/2 Cfn[0] + Cfn[h/2] + 7/18 Cfn[h]) - 1/15 h^3 Dfn[h] ** Cfn[h]);
NCExpand[Normal[NCSeries[1/2 (R[h] - R[-h]) + Q[h] ** F[h] - Q[-h] ** F[-h], {h, 0, 6}]] //. substD]
```

```
0
```

```
(* Eq 19 *)
L1[h_] := 403/16800 Dfn[-h] - 279/2800 Dfn[-2 h/3] + 99/800 Dfn[-h/3] + 34/105 Dfn[0] - 333/5600 Dfn[h/3] + 1719/2800 Dfn[2 h/3] + 1237/16800 Dfn[h];
L2[h_] := 57/1120 Dfn[-h] - 243/560 Dfn[-2 h/3] + 1269/1120 Dfn[-h/3] - 3/4 Dfn[0] + 891/1120 Dfn[h/3] + 27/112 Dfn[2 h/3] - 41/1120 Dfn[h];
L3[h_] := -2067/9680 Dfn[-h] + 6021/4840 Dfn[-2 h/3] - 5805/1936 Dfn[-h/3] + 1863/484 Dfn[0] - 5697/1936 Dfn[h/3] + 10341/4840 Dfn[2 h/3] - 727/9680 Dfn[h];
L4[h_] := 63/16 Dfn[-h] - 1809/40 Dfn[-2 h/3] + 2295/16 Dfn[-h/3] - 801/4 Dfn[0] + 2133/16 Dfn[h/3] - 297/8 Dfn[2 h/3] + 233/80 Dfn[h];
L5[h_] := 123/160 Dfn[-h] - 135/8 Dfn[-2 h/3] + 2295/32 Dfn[-h/3] - 132 Dfn[0] + 3861/32 Dfn[h/3] - 1917/40 Dfn[2 h/3] + 149/32 Dfn[h];
L6[h_] := -6/35 Dfn[-h] + 27/10 Dfn[-2 h/3] - 1053/112 Dfn[-h/3] + 57/4 Dfn[0] - 621/56 Dfn[h/3] + 729/140 Dfn[2 h/3] - 277/560 Dfn[h];
L7[h_] := 403/16800 Cfn[-h] - 279/2800 Cfn[-2 h/3] + 99/800 Cfn[-h/3] + 34/105 Cfn[0] - 333/5600 Cfn[h/3] + 1719/2800 Cfn[2 h/3] + 1237/16800 Cfn[h];
L8[h_] := -2067/9680 Cfn[-h] + 6021/4840 Cfn[-2 h/3] - 5805/1936 Cfn[-h/3] + 1863/484 Cfn[0] - 5697/1936 Cfn[h/3] + 10341/4840 Cfn[2 h/3] - 727/9680 Cfn[h];
L9[h_] := 123/160 Cfn[-h] - 135/8 Cfn[-2 h/3] + 2295/32 Cfn[-h/3] - 132 Cfn[0] + 3861/32 Cfn[h/3] - 1917/40 Cfn[2 h/3] + 149/32 Cfn[h];
Q[h_] := 1 - h L1[h] + L2[h] ** (121/315 h^2 L3[h] - 2/315 h^3 L4[h] ** L5[h]) +
   (2/45 h^2 L6[h] + L2[h] ** (-4/45 h^3 L6[h] + 1/105 h^4 Dfn[h] ** Dfn[h])) ** Dfn[h];
R[h_] := -2 h L7[h] + 2 L2[h] ** (121/315 h^2 L8[h] - 2/315 h^3 L4[h] ** L9[h]) +
   2 (2/45 h^2 L6[h] + L2[h] ** (-4/45 h^3 L6[h] + 1/105 h^4 Dfn[h] ** Dfn[h])) ** Cfn[h];
NCExpand[Normal[NCSeries[1/2 (R[h] - R[-h]) + Q[h] ** F[h] - Q[-h] ** F[-h], {h, 0, 6}]] //. substD]

0
```