Analyzing and extending existing classes of methods by means of the theoretical framework of GLMs

Giuseppe IZZO [1]

University of Naples Federico II, Italy

February 21, 2023

Part of this work is joint with Z.Jackiewicz (ASU, USA) and S.Boscarino (Unict, Italy)

Università degli Studi di Catania, Italy

[1] e-mail: giuseppe.izzo@unina.it.

1 General Linear Methods

- Formulation of GLMs
- RK, LMM and BDF represented as GLMs
- GLMs as framework to analyze and generalize
- MEBDF represented as GLMs
- Generalized Linear Multistep Methods

2 Self Starting GLMs

- Introduction
- Singly Diagonally-Implicit Methods
- Explicit Methods
- Implicit-Explicit Methods

3 Work in progress and future work

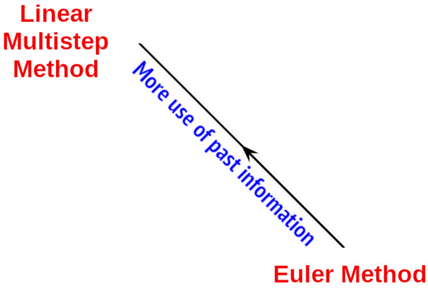
Università degli Studi di Napoli Federico II

Introduction

Euler Method

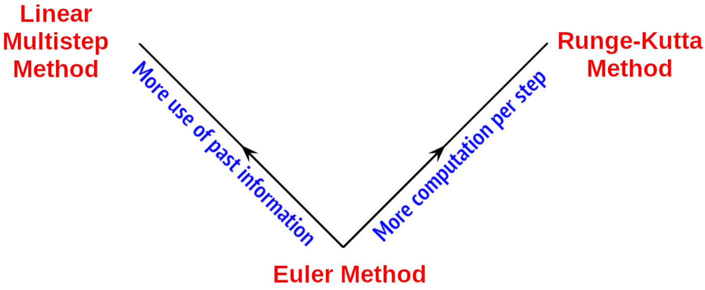
Author: J.C.Butcher

Introduction



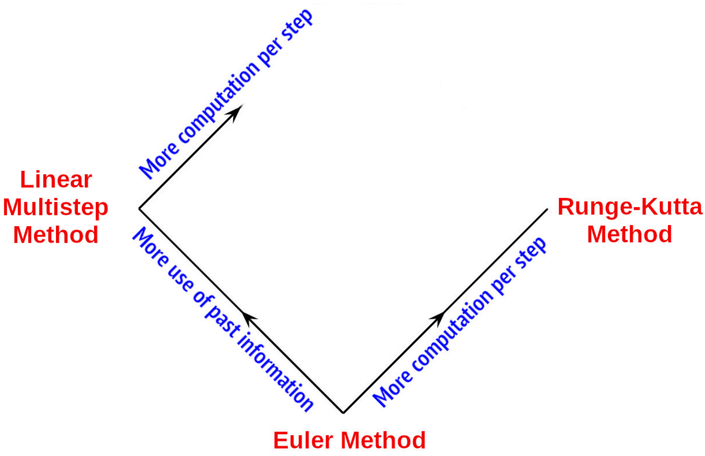
Author: J.C.Butcher

Introduction



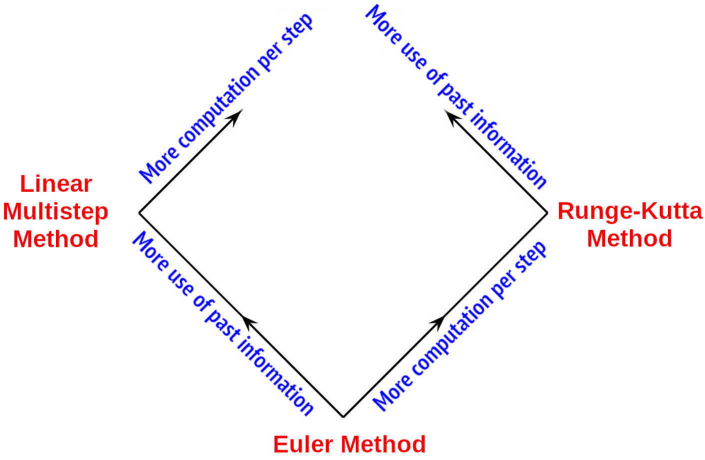
Author: J.C.Butcher

Introduction



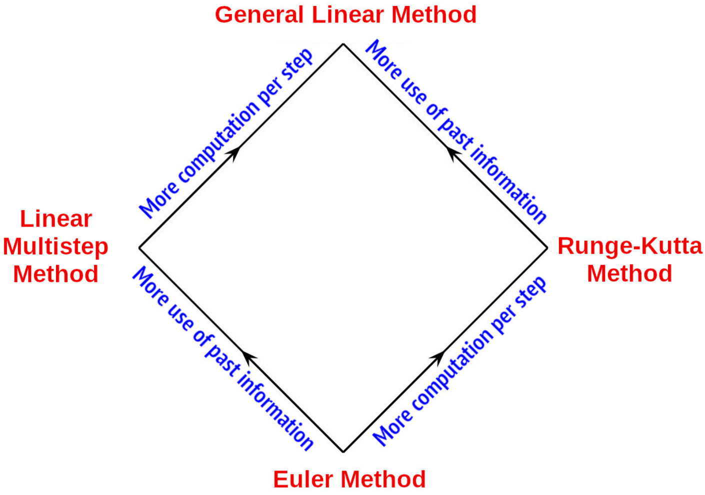
Author: J.C.Butcher

Introduction



Author: J.C.Butcher

Introduction



Author: J.C.Butcher

Introduction

Let us consider an initial value problem (IVP)

$$\begin{cases} y'(t) = f\big(y(t)\big), \quad t \in [t_0, T], \\ y(t_0) = y_0. \end{cases}$$

where $f : \mathbb{R}^m \to \mathbb{R}^m$.

Introduction

Let us consider an initial value problem (IVP)

$$\begin{cases} y'(t) = f\big(y(t)\big), & t \in [t_0, T], \\ y(t_0) = y_0. \end{cases}$$

where $f : \mathbb{R}^m \to \mathbb{R}^m$.

The usual General linear methods (GLMs) formulation is

$$\begin{cases} Y_i^{[n]} = h \sum_{j=1}^{s} a_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} u_{ij} y_j^{[n-1]}, & i = 1, 2, \dots, s, \\ y_i^{[n]} = h \sum_{j=1}^{s} b_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} v_{ij} y_j^{[n-1]}, & i = 1, 2, \dots, r, \end{cases}$$

$n = 1, 2, \dots, N$, where $Nh = T - t_0$.

General Linear Methods

$$\begin{cases} Y_i^{[n]} = h\displaystyle\sum_{j=1}^{s} a_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} u_{ij} y_j^{[n-1]}, & i = 1, 2, \ldots, s, \\ y_i^{[n]} = h\displaystyle\sum_{j=1}^{s} b_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} v_{ij} y_j^{[n-1]}, & i = 1, 2, \ldots, r, \end{cases}$$

for $n = 1, 2, \ldots, N$, where $Nh = T - t_0$.

General Linear Methods

$$\begin{cases} Y_i^{[n]} = h\sum_{j=1}^{s} a_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} u_{ij} y_j^{[n-1]}, & i = 1, 2, \ldots, s, \\ y_i^{[n]} = h\sum_{j=1}^{s} b_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} v_{ij} y_j^{[n-1]}, & i = 1, 2, \ldots, r, \end{cases}$$

for $n = 1, 2, \ldots, N$, where $Nh = T - t_0$.

Internal stages:

$$Y_i^{[n]} = y(t_{n-1} + c_i h) + O(h^{q+1}), \quad i = 1, 2, \ldots, s,$$

General Linear Methods

$$\begin{cases} Y_i^{[n]} = h\sum_{j=1}^{s} a_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} u_{ij} y_j^{[n-1]}, & i = 1, 2, \ldots, s, \\ y_i^{[n]} = h\sum_{j=1}^{s} b_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} v_{ij} y_j^{[n-1]}, & i = 1, 2, \ldots, r, \end{cases}$$

for $n = 1, 2, \ldots, N$, where $Nh = T - t_0$.

Internal stages:

$$Y_i^{[n]} = y(t_{n-1} + c_i h) + O(h^{q+1}), \quad i = 1, 2, \ldots, s,$$

External stages:

$$y_i^{[n]} = \sum_{k=0}^{p} q_{ik} h^k y^{(k)}(t_n) + O(h^{p+1}), \quad i = 1, 2, \ldots, r.$$

General Linear Methods

$$\begin{cases} Y_i^{[n]} = h\sum_{j=1}^{s} \boxed{a_{ij}} f(Y_j^{[n]}) + \sum_{j=1}^{r} \boxed{u_{ij}} y_j^{[n-1]}, & i = 1, 2, \ldots, s, \\ y_i^{[n]} = h\sum_{j=1}^{s} \boxed{b_{ij}} f(Y_j^{[n]}) + \sum_{j=1}^{r} \boxed{v_{ij}} y_j^{[n-1]}, & i = 1, 2, \ldots, r, \end{cases}$$

for $n = 1, 2, \ldots, N$, where $Nh = T - t_0$.

Internal stages:

$$Y_i^{[n]} = y(t_{n-1} + \boxed{c_i} h) + O(h^{q+1}), \quad i = 1, 2, \ldots, s,$$

External stages:

$$y_i^{[n]} = \sum_{k=0}^{p} \boxed{q_{ik}} h^k y^{(k)}(t_n) + O(h^{p+1}), \quad i = 1, 2, \ldots, r.$$

General Linear Methods

$$\begin{cases} Y_i^{[n]} = h\sum_{j=1}^{s} a_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} u_{ij} y_j^{[n-1]}, & i = 1, 2, \ldots, s, \\ y_i^{[n]} = h\sum_{j=1}^{s} b_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} v_{ij} y_j^{[n-1]}, & i = 1, 2, \ldots, r, \end{cases}$$

for $n = 1, 2, \ldots, N$, where $Nh = T - t_0$.

General Linear Methods

$$\begin{cases} Y_i^{[n]} = h\sum_{j=1}^{s} a_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} u_{ij} y_j^{[n-1]}, & i = 1, 2, \dots, s, \\ y_i^{[n]} = h\sum_{j=1}^{s} b_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} v_{ij} y_j^{[n-1]}, & i = 1, 2, \dots, r, \end{cases}$$

for $n = 1, 2, \dots, N$, where $Nh = T - t_0$.

$$\mathbf{A} = [a_{ij}] \in \mathbb{R}^{s\times s}, \quad \mathbf{U} = [u_{ij}] \in \mathbb{R}^{s\times r},$$

$$\mathbf{B} = [b_{ij}] \in \mathbb{R}^{r\times s}, \quad \mathbf{V} = [v_{ij}] \in \mathbb{R}^{r\times r},$$

General Linear Methods

$$\begin{cases} Y_i^{[n]} = h\sum_{j=1}^{s} a_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} u_{ij} y_j^{[n-1]}, & i = 1, 2, \ldots, s, \\ y_i^{[n]} = h\sum_{j=1}^{s} b_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} v_{ij} y_j^{[n-1]}, & i = 1, 2, \ldots, r, \end{cases}$$

for $n = 1, 2, \ldots, N$, where $Nh = T - t_0$.

$$\mathbf{A} = [a_{ij}] \in \mathbb{R}^{s\times s}, \quad \mathbf{U} = [u_{ij}] \in \mathbb{R}^{s\times r},$$

$$\mathbf{B} = [b_{ij}] \in \mathbb{R}^{r\times s}, \quad \mathbf{V} = [v_{ij}] \in \mathbb{R}^{r\times r},$$

$$\mathbf{c} = [c_i] \in \mathbb{R}^{s}, \quad \mathbf{W} = [\mathbf{q}_0, \mathbf{q}_1, \ldots, \mathbf{q}_p] = [q_{ij}] \in \mathbb{R}^{r\times(p+1)}$$

General Linear Methods - Matrix Form

Set

$$Y^{[n]} = \begin{bmatrix} Y_1^{[n]} \\ \vdots \\ Y_s^{[n]} \end{bmatrix} \in \mathbb{R}^{sm}, \quad F^{[n]} = \begin{bmatrix} F_1^{[n]} \\ \vdots \\ F_s^{[n]} \end{bmatrix} \in \mathbb{R}^{sm}, \quad y^{[n]} = \begin{bmatrix} y_1^{[n]} \\ \vdots \\ y_r^{[n]} \end{bmatrix} \in \mathbb{R}^{rm},$$

GLMs can be written in matrix form as

$$\left[\begin{array}{c} Y^{[n]} \\ \hline y^{[n]} \end{array}\right] = \left[\begin{array}{c|c} \mathbf{A} \otimes \mathbf{I} & \mathbf{U} \otimes \mathbf{I} \\ \hline \mathbf{B} \otimes \mathbf{I} & \mathbf{V} \otimes \mathbf{I} \end{array}\right] \left[\begin{array}{c} hf(Y^{[n]}) \\ \hline y^{[n-1]} \end{array}\right].$$

Runge–Kutta represented as GLMs

$$\begin{cases} Y_i = y_n + h \displaystyle\sum_{j=1}^{s} a_{ij} f(Y_j), \qquad i = 1, 2, \ldots, s, \\ y_{n+1} = y_n + h \displaystyle\sum_{j=1}^{s} b_j f(Y_j) \end{cases}$$

$$\left[\begin{array}{c|c} \mathbf{A} & \mathbf{U} \\ \hline \mathbf{B} & \mathbf{V} \end{array}\right] = \left[\begin{array}{ccc|c} a_{11} & \cdots & a_{1s} & 1 \\ \vdots & \ddots & \vdots & \vdots \\ a_{s1} & \cdots & a_{ss} & 1 \\ \hline b_1 & \cdots & b_s & 1 \end{array}\right]$$

Linear Multistep Methods represented as GLMs

$$y_n = \sum_{j=1}^{k} \alpha_j y_{n-j} + h \sum_{j=0}^{k} \beta_j f(y_{n-j})$$

$$\left[\begin{array}{c|c} \mathbf{A} & \mathbf{U} \\ \hline \mathbf{B} & \mathbf{V} \end{array}\right] = \left[\begin{array}{c|cccccc} \beta_0 & 0 & 0 & 0 & \ldots & 0 & 1 \\ \hline \alpha_k\beta_0 + \beta_k & 0 & 0 & 0 & \ldots & 0 & \alpha_k \\ \alpha_{k-1}\beta_0 + \beta_{k-1} & 1 & 0 & 0 & \ldots & 0 & \alpha_{k-1} \\ \alpha_{k-2}\beta_0 + \beta_{k-2} & 0 & 1 & 0 & \ldots & 0 & \alpha_{k-2} \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ \alpha_2\beta_0 + \beta_2 & 0 & 0 & 0 & \ldots & 0 & \alpha_2 \\ \alpha_1\beta_0 + \beta_1 & 0 & 0 & 0 & \ldots & 1 & \alpha_1 \end{array}\right],$$

J.C. Butcher and A.T. Hill, BIT, 2006.

BDF represented as GLMs

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h\beta_k f(y_{n+k})$$

$$\left[\begin{array}{c|c} \mathbf{A} & \mathbf{U} \\ \hline \mathbf{B} & \mathbf{V} \end{array}\right] = \left[\begin{array}{c|cccccc} \beta_k & -\alpha_{k-1} & -\alpha_{k-2} & \cdots & -\alpha_1 & -\alpha_0 \\ \hline \beta_k & -\alpha_{k-1} & -\alpha_{k-2} & \cdots & -\alpha_1 & -\alpha_0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 1 & 0 \end{array}\right],$$

GLMs as framework to analyze and generalize

We can use general linear methods as a framework to analyze and generalize existing classes of numerical methods.

Example:

Modified Extended Backward Differentiation Formulae

$\Downarrow$

Generalized Linear Multistep Methods

[1] G.Izzo, Z.Jackiewicz, *Generalized linear multistep methods for ordinary differential equations*, Applied Numerical Mathematics 114 (2017) 165–178.

Extended Backward Differentiation Formulae

Consider the classical BDF method

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h\beta_k f_{n+k}$$

where $f_{n+k} = f(t_{n+k}, y_{n+k})$,

Extended Backward Differentiation Formulae

Extend the classical BDF method

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h\beta_k f_{n+k} + h\beta_{k+1} f_{n+k+1},$$

where $f_{n+k} = f(t_{n+k}, y_{n+k})$, $f_{n+k+1} = f(t_{n+k+1}, y_{n+k+1})$.

Extended Backward Differentiation Formulae

Extend the classical BDF method

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h\beta_k f_{n+k} + h\beta_{k+1} f_{n+k+1},$$

where $f_{n+k} = f(t_{n+k}, y_{n+k})$, $f_{n+k+1} = f(t_{n+k+1}, y_{n+k+1})$.

- Based on the idea of using an approximation of the solution at a future point t_{n+k+1}.

Extended Backward Differentiation Formulae

Extend the classical BDF method

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h\beta_k f_{n+k} + h\beta_{k+1} f_{n+k+1},$$

where $f_{n+k} = f(t_{n+k}, y_{n+k}), f_{n+k+1} = f(t_{n+k+1}, y_{n+k+1})$.

- Based on the idea of using an approximation of the solution at a future point t_{n+k+1}.
- It is needed to have a suitable estimation of y_{n+k+1}.

Extended Backward Differentiation Formulae

(i) Compute $\overline{y}_{n+k}$ as the solution of the conventional BDF method

$$\overline{y}_{n+k} + \sum_{j=0}^{k-1} \widehat{\alpha}_j y_{n+j} = h\widehat{\beta}_k \overline{f}_{n+k},$$

$\overline{f}_{n+k} = f(t_{n+k}, \overline{y}_{n+k})$.

Extended Backward Differentiation Formulae

$$(i) \qquad \bar{y}_{n+k} + \sum_{j=0}^{k-1} \widehat{\alpha}_j y_{n+j} = h\widehat{\beta}_k \bar{f}_{n+k},$$

Extended Backward Differentiation Formulae

$$(i) \qquad \overline{y}_{n+k} + \sum_{j=0}^{k-1} \widehat{\alpha}_j y_{n+j} = h\widehat{\beta}_k \overline{f}_{n+k},$$

(ii) Compute $\overline{y}_{n+k+1}$ as the solution of the same BDF advanced one step, that is,

$$\overline{y}_{n+k+1} + \widehat{\alpha}_{k-1}\overline{y}_{n+k} + \sum_{j=0}^{k-2} \widehat{\alpha}_j y_{n+j+1} = h\widehat{\beta}_k \overline{f}_{n+k+1},$$

where $\overline{f}_{n+k+1} = f(t_{n+k+1}, \overline{y}_{n+k+1})$.

Extended Backward Differentiation Formulae

$$(i) \qquad \overline{y}_{n+k} + \sum_{j=0}^{k-1} \widehat{\alpha}_j y_{n+j} = h\widehat{\beta}_k \overline{f}_{n+k},$$

$$(ii) \qquad \overline{y}_{n+k+1} + \widehat{\alpha}_{k-1}\overline{y}_{n+k} + \sum_{j=0}^{k-2} \widehat{\alpha}_j y_{n+j+1} = h\widehat{\beta}_k \overline{f}_{n+k+1},$$

Extended Backward Differentiation Formulae

$$(i) \qquad \overline{y}_{n+k} + \sum_{j=0}^{k-1} \widehat{\alpha}_j y_{n+j} = h\widehat{\beta}_k \overline{f}_{n+k},$$

$$(ii) \qquad \overline{y}_{n+k+1} + \widehat{\alpha}_{k-1}\overline{y}_{n+k} + \sum_{j=0}^{k-2} \widehat{\alpha}_j y_{n+j+1} = h\widehat{\beta}_k \overline{f}_{n+k+1},$$

(iii) Discard $\overline{y}_{n+k}$, compute $\overline{f}_{n+k+1}$ and insert it into EBDF method, to solve for y_{n+k}:

$$y_{n+k} + \sum_{j=0}^{k-1} \alpha_j y_{n+j} = h\beta_k f_{n+k} + h\beta_{k+1}\overline{f}_{n+k+1}.$$

Extended Backward Differentiation Formulae

$$(i) \quad \bar{y}_{n+k} + \sum_{j=0}^{k-1} \widehat{\alpha}_j y_{n+j} = h\widehat{\beta}_k \bar{f}_{n+k},$$

$$(ii) \quad \bar{y}_{n+k+1} + \widehat{\alpha}_{k-1}\bar{y}_{n+k} + \sum_{j=0}^{k-2} \widehat{\alpha}_j y_{n+j+1} = h\widehat{\beta}_k \bar{f}_{n+k+1},$$

$$(iii) \quad y_{n+k} + \sum_{j=0}^{k-1} \alpha_j y_{n+j} = h\beta_k f_{n+k} + h\beta_{k+1}\bar{f}_{n+k+1}.$$

Extended Backward Differentiation Formulae

$$(i)\qquad \bar{y}_{n+k} + \sum_{j=0}^{k-1} \widehat{\alpha}_j y_{n+j} = h\widehat{\beta}_k \bar{f}_{n+k},$$

$$(ii)\qquad \bar{y}_{n+k+1} + \widehat{\alpha}_{k-1}\bar{y}_{n+k} + \sum_{j=0}^{k-2} \widehat{\alpha}_j y_{n+j+1} = h\widehat{\beta}_k \bar{f}_{n+k+1},$$

$$(iii)\qquad y_{n+k} + \sum_{j=0}^{k-1} \alpha_j y_{n+j} = h\beta_k f_{n+k} + h\beta_{k+1}\bar{f}_{n+k+1}.$$

If the EBDF method used in (iii) is of order $k+1$
and BDF methods in (i) and (ii) are of order k,
then the overall algorithm (i)-(iii) has order $k+1$.

Extended Backward Differentiation Formulae

$$(i) \qquad \bar{y}_{n+k} + \sum_{j=0}^{k-1} \widehat{\alpha}_j y_{n+j} = h\widehat{\beta}_k \bar{f}_{n+k},$$

$$(ii) \qquad \bar{y}_{n+k+1} + \widehat{\alpha}_{k-1}\bar{y}_{n+k} + \sum_{j=0}^{k-2} \widehat{\alpha}_j y_{n+j+1} = h\widehat{\beta}_k \bar{f}_{n+k+1},$$

$$(iii) \qquad y_{n+k} + \sum_{j=0}^{k-1} \alpha_j y_{n+j} = h\beta_k f_{n+k} + h\beta_{k+1}\bar{f}_{n+k+1}.$$

Modified Extended Backward Differentiation Formulae

$$(i) \qquad \bar{y}_{n+k} + \sum_{j=0}^{k-1} \widehat{\alpha}_j y_{n+j} = h\widehat{\beta}_k \bar{f}_{n+k},$$

$$(ii) \qquad \bar{y}_{n+k+1} + \widehat{\alpha}_{k-1}\bar{y}_{n+k} + \sum_{j=0}^{k-2} \widehat{\alpha}_j y_{n+j+1} = h\widehat{\beta}_k \bar{f}_{n+k+1},$$

$$(iii) \qquad y_{n+k} + \sum_{j=0}^{k-1} \alpha_j y_{n+j} = h\beta_k f_{n+k} + h\beta_{k+1}\bar{f}_{n+k+1}.$$

Modified Extended Backward Differentiation Formulae

$$(i) \quad \bar{y}_{n+k} + \sum_{j=0}^{k-1} \widehat{\alpha}_j y_{n+j} = h\widehat{\beta}_k \bar{f}_{n+k},$$

$$(ii) \quad \bar{y}_{n+k+1} + \widehat{\alpha}_{k-1}\bar{y}_{n+k} + \sum_{j=0}^{k-2} \widehat{\alpha}_j y_{n+j+1} = h\widehat{\beta}_k \bar{f}_{n+k+1},$$

$$(iii) \quad \sum_{j=0}^{k} \alpha_j y_{n+j} = h\widehat{\beta}_k f_{n+k} + h(\beta_k - \widehat{\beta}_k)\bar{f}_{n+k} + h\beta_{k+1}\bar{f}_{n+k+1}.$$

Modified Extended BDF represented as GLMs

We can represent the MEBDF

$$(i) \quad \bar{y}_{n+k} + \sum_{j=0}^{k-1} \widehat{\alpha}_j y_{n+j} = h\widehat{\beta}_k \bar{f}_{n+k},$$

$$(ii) \quad \bar{y}_{n+k+1} + \widehat{\alpha}_{k-1}\bar{y}_{n+k} + \sum_{j=0}^{k-2} \widehat{\alpha}_j y_{n+j+1} = h\widehat{\beta}_k \bar{f}_{n+k+1},$$

$$(iii) \quad \sum_{j=0}^{k} \alpha_j y_{n+j} = h\widehat{\beta}_k f_{n+k} + h(\beta_k - \widehat{\beta}_k)\bar{f}_{n+k} + h\beta_{k+1}\bar{f}_{n+k+1}.$$

as a General Linear Method

MEBDF represented as GLMs

We can represent the MEBDF

$$(i) \quad \bar{y}_{n+k} + \sum_{j=0}^{k-1} \widehat{\alpha}_j y_{n+j} = h\widehat{\beta}_k \bar{f}_{n+k},$$

$$(ii) \quad \bar{y}_{n+k+1} + \widehat{\alpha}_{k-1}\bar{y}_{n+k} + \sum_{j=0}^{k-2} \widehat{\alpha}_j y_{n+j+1} = h\widehat{\beta}_k \bar{f}_{n+k+1},$$

$$(iii) \quad y_{n+k} + \sum_{j=0}^{k-1} \alpha_j y_{n+j} = h\widehat{\beta}_k f_{n+k} + h(\beta_k - \widehat{\beta}_k)\bar{f}_{n+k} + h\beta_{k+1}\bar{f}_{n+k+1}.$$

as a General Linear Method

MEBDF represented as GLMs

We can represent the MEBDF

$$(i) \quad \bar{y}_{n+k} + \sum_{j=0}^{k-1} \widehat{\alpha}_j y_{n+j} = h\widehat{\beta}_k \bar{f}_{n+k},$$

$$(ii) \quad \bar{y}_{n+k+1} + \widehat{\alpha}_{k-1}\bar{y}_{n+k} + \sum_{j=0}^{k-2} \widehat{\alpha}_j y_{n+j+1} = h\widehat{\beta}_k \bar{f}_{n+k+1},$$

$$(iii) \quad y_{n+k} + \sum_{j=0}^{k-1} \alpha_j y_{n+j} = h\widehat{\beta}_k f_{n+k} + h(\beta_k - \widehat{\beta}_k)\bar{f}_{n+k} + h\beta_{k+1}\bar{f}_{n+k+1}.$$

as a General Linear Method

MEBDF represented as GLMs

$$Y^{[n]} = \begin{bmatrix} \bar{y}_{n+k} \\ \bar{y}_{n+k+1} \\ y_{n+k} \end{bmatrix}, \quad f(Y^{[n]}) = \begin{bmatrix} \bar{f}_{n+k} \\ \bar{f}_{n+k+1} \\ f_{n+k} \end{bmatrix},$$

$$\mathbf{c} = \begin{bmatrix} k+1 & k+2 & k+1 \end{bmatrix}^T,$$

MEBDF represented as GLMs

$$Y^{[n]} = \begin{bmatrix} \bar{y}_{n+k} \\ \bar{y}_{n+k+1} \\ y_{n+k} \end{bmatrix}, \quad f(Y^{[n]}) = \begin{bmatrix} \bar{f}_{n+k} \\ \bar{f}_{n+k+1} \\ f_{n+k} \end{bmatrix}, \quad y^{[n]} = \begin{bmatrix} y_{n+k} \\ y_{n+k-1} \\ \vdots \\ y_{n+1} \end{bmatrix},$$

$$\mathbf{c} = \begin{bmatrix} k+1 & k+2 & k+1 \end{bmatrix}^T,$$

MEBDF represented as GLMs

$$Y^{[n]} = \begin{bmatrix} \bar{y}_{n+k} \\ \bar{y}_{n+k+1} \\ y_{n+k} \end{bmatrix}, \quad f(Y^{[n]}) = \begin{bmatrix} \bar{f}_{n+k} \\ \bar{f}_{n+k+1} \\ f_{n+k} \end{bmatrix}, \quad y^{[n]} = \begin{bmatrix} y_{n+k} \\ y_{n+k-1} \\ \vdots \\ y_{n+1} \end{bmatrix},$$

$$\mathbf{c} = \begin{bmatrix} k+1 & k+2 & k+1 \end{bmatrix}^T,$$

and, since we have to satisfy

$$y_i^{[n]} = \sum_{k=0}^{p} q_{ik} h^k y^{(k)}(t_n) + O(h^{p+1}) = y(t_{n+k-i+1}) + O(h^{p+1}), \quad i = 1, 2, \ldots, k.$$

MEBDF represented as GLMs

$$Y^{[n]} = \begin{bmatrix} \bar{y}_{n+k} \\ \bar{y}_{n+k+1} \\ y_{n+k} \end{bmatrix}, \quad f(Y^{[n]}) = \begin{bmatrix} \bar{f}_{n+k} \\ \bar{f}_{n+k+1} \\ f_{n+k} \end{bmatrix}, \quad y^{[n]} = \begin{bmatrix} y_{n+k} \\ y_{n+k-1} \\ \vdots \\ y_{n+1} \end{bmatrix},$$

$$\mathbf{c} = \begin{bmatrix} k+1 & k+2 & k+1 \end{bmatrix}^T,$$

and, since we have to satisfy

$$y_i^{[n]} = \sum_{k=0}^{p} q_{ik} h^k y^{(k)}(t_n) + O(h^{p+1}) = y(t_{n+k-i+1}) + O(h^{p+1}), \quad i = 1, 2, \ldots, k.$$

we choose

$$\mathbf{q_j} = \left[\frac{(k-i+1)^j}{j!} \right]_{i=1,\ldots,k}, \quad j = 0, \ldots, k+1,$$

MEBDF represented as GLMs

$$\mathbf{A} = \begin{bmatrix} \widehat{\beta}_k & 0 & 0 \\ -\widehat{\alpha}_{k-1}\widehat{\beta}_k & \widehat{\beta}_k & 0 \\ \beta_k - \widehat{\beta}_k & \beta_{k+1} & \widehat{\beta}_k \end{bmatrix},$$

$$\mathbf{U} = \begin{bmatrix} -\widehat{\alpha}_{k-1} & -\widehat{\alpha}_{k-2} & \cdots & -\widehat{\alpha}_1 & -\widehat{\alpha}_0 \\ \widehat{\alpha}_{k-1}\widehat{\alpha}_{k-1}-\widehat{\alpha}_{k-2} & \widehat{\alpha}_{k-1}\widehat{\alpha}_{k-2}-\widehat{\alpha}_{k-3} & \cdots & \widehat{\alpha}_{k-1}\widehat{\alpha}_1-\widehat{\alpha}_0 & \widehat{\alpha}_{k-1}\widehat{\alpha}_0 \\ -\alpha_{k-1} & -\alpha_{k-2} & \cdots & -\alpha_1 & -\alpha_0 \end{bmatrix},$$

$$\mathbf{B} = \begin{bmatrix} \beta_k - \widehat{\beta}_k & \beta_{k+1} & \widehat{\beta}_k \\ 0 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{V} = \begin{bmatrix} -\alpha_{k-1} & -\alpha_{k-2} & \cdots & -\alpha_1 & -\alpha_0 \\ 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix},$$

Generalization

$$\mathbf{A} = \begin{bmatrix} \lambda & 0 & 0 \\ a_{21} & \lambda & 0 \\ a_{31} & a_{32} & \lambda \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1,k-1} & u_{1k} \\ u_{21} & u_{22} & \cdots & u_{2,k-1} & u_{2k} \\ u_{31} & u_{32} & \cdots & u_{3,k-1} & u_{3k} \end{bmatrix},$$

$$\mathbf{B} = \begin{bmatrix} a_{31} & a_{32} & \lambda \\ 0 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{V} = \begin{bmatrix} u_{31} & u_{32} & \cdots & u_{3,k-1} & u_{3k} \\ 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}.$$

Generalization

We keep

$$y^{[n]} = \begin{bmatrix} y_{n+k}, & y_{n+k-1}, & \dots, & y_{n+1} \end{bmatrix}^T,$$

and

$$\mathbf{q_j} = \left[\frac{(k-i+1)^j}{j!}\right]_{i=1,\dots,k}, \quad j = 0, \dots, k+1, \qquad (1)$$

Generalization

We keep

$$y^{[n]} = \left[\begin{array}{cccc} y_{n+k}, & y_{n+k-1}, & \dots, & y_{n+1} \end{array} \right]^T,$$

and

$$\mathbf{q_j} = \left[\frac{(k-i+1)^j}{j!} \right]_{i=1,\dots,k}, \quad j = 0, \dots, k+1, \tag{1}$$

but, we assume that the abscissa vector is given by

$$\mathbf{c} = \left[\begin{array}{ccc} k+1+\delta_1, & k+1+\delta_2, & k+1 \end{array} \right]^T, \tag{2}$$

Generalization

We keep

$$y^{[n]} = \left[\begin{array}{cccc} y_{n+k}, & y_{n+k-1}, & \dots, & y_{n+1} \end{array} \right]^T,$$

and

$$\mathbf{q_j} = \left[\frac{(k-i+1)^j}{j!} \right]_{i=1,\dots,k}, \quad j = 0, \dots, k+1, \tag{1}$$

but, we assume that the abscissa vector is given by

$$\mathbf{c} = \left[\begin{array}{ccc} k+1+\delta_1, & k+1+\delta_2, & k+1 \end{array} \right]^T, \tag{2}$$

and we require the method to have stage order $q = p - 1 = k$, that is

$$Y_j^{[n]} = y(t_{n-1} + c_j h) + O(h^{k+1}), \quad j = 1, 2, 3.$$

Methods of order $p = 2, 3, 4$

k	p	δ_1	δ_2	$\|\|\mathbf{ec}_p(\delta_1, \delta_2)\|\|_1$	$\|\|\mathbf{ec}_p(0, 1)\|\|_1$	α	α_{MEBDF}
1	2	$\frac{-3-\sqrt{3}}{6}$	$-\frac{\sqrt{3}}{6}$		0.667	$90°$	$90°$
2	3	$-\frac{21}{43}$	$-\frac{13}{33}$		0.285	$90°$	$90°$
3	4	$-\frac{46}{131}$	$-\frac{53}{114}$		0.769	$90°$	$90°$

- A-stable like the MEBDF of the same order,

Methods of order $p = 2, 3, 4$

k	p	δ_1	δ_2	$\|\|\mathbf{ec}_p(\delta_1, \delta_2)\|\|_1$	$\|\|\mathbf{ec}_p(0, 1)\|\|_1$	α	α_{MEBDF}
1	2	$\frac{-3-\sqrt{3}}{6}$	$-\frac{\sqrt{3}}{6}$	0.026	0.667	90°	90°
2	3	$-\frac{21}{43}$	$-\frac{13}{33}$	0.008	0.285	90°	90°
3	4	$-\frac{46}{131}$	$-\frac{53}{114}$	0.019	0.769	90°	90°

- *A*-stable like the MEBDF of the same order,
- **Smaller error coefficients than MEBDF**.

Methods of order $p \geq 5$

k	p	δ_1	δ_2	$\|\|\mathbf{ec}_p(\delta_1,\delta_2)\|\|_1$	$\|\|\mathbf{ec}_p(0,1)\|\|_1$	α	α_{MEBDF}
4	5	$-\frac{10}{41}$	$-\frac{24}{49}$		2.626		$88.36°$
5	6	$-\frac{2}{7}$	$-\frac{5}{11}$		8.306		$83.07°$
6	7	$-\frac{9}{34}$	$-\frac{38}{83}$		24.796		$74.48°$
7	8	$-\frac{11}{39}$	$-\frac{19}{44}$		71.498		$61.98°$
8	9	$-\frac{11}{38}$	$-\frac{20}{47}$		201.797		$42.87°$

Methods of order $p \geq 5$

k	p	δ_1	δ_2	$\|\|\mathbf{ec}_p(\delta_1, \delta_2)\|\|_1$	$\|\|\mathbf{ec}_p(0, 1)\|\|_1$	α	α_{MEBDF}
4	5	$-\frac{10}{41}$	$-\frac{24}{49}$		2.626	88.24°	88.36°
5	6	$-\frac{2}{7}$	$-\frac{5}{11}$		8.306	83.41°	83.07°
6	7	$-\frac{9}{34}$	$-\frac{38}{83}$		24.796	76.21°	74.48°
7	8	$-\frac{11}{39}$	$-\frac{19}{44}$		71.498	67.21°	61.98°
8	9	$-\frac{11}{38}$	$-\frac{20}{47}$		201.797	55.47°	42.87°

- Except for $k = 4$, larger angle of $A(\alpha)$-stability than MEBDF of the same order,

Methods of order $p \geq 5$

k	p	δ_1	δ_2	$\|\|\mathbf{ec}_p(\delta_1, \delta_2)\|\|_1$	$\|\|\mathbf{ec}_p(0, 1)\|\|_1$	α	α_{MEBDF}
4	5	$-\frac{10}{41}$	$-\frac{24}{49}$	0.043	2.626	88.24°	88.36°
5	6	$-\frac{2}{7}$	$-\frac{5}{11}$	0.007	8.306	83.41°	83.07°
6	7	$-\frac{9}{34}$	$-\frac{38}{83}$	0.047	24.796	76.21°	74.48°
7	8	$-\frac{11}{39}$	$-\frac{19}{44}$	0.582	71.498	67.21°	61.98°
8	9	$-\frac{11}{38}$	$-\frac{20}{47}$	0.512	201.797	55.47°	42.87°

- Except for $k = 4$, larger angle of $A(\alpha)$-stability than MEBDF of the same order,
- **Smaller error coefficients than the corresponding MEBDF.**

Generalized Linear Multistep Methods $s = 3$

$$\mathbf{A} = \begin{bmatrix} \lambda & 0 & 0 \\ a_{21} & \lambda & 0 \\ a_{31} & a_{32} & \lambda \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1,k-1} & u_{1k} \\ u_{21} & u_{22} & \cdots & u_{2,k-1} & u_{2k} \\ u_{31} & u_{32} & \cdots & u_{3,k-1} & u_{3k} \end{bmatrix},$$

$$\mathbf{B} = \begin{bmatrix} a_{31} & a_{32} & \lambda \\ 0 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{V} = \begin{bmatrix} u_{31} & u_{32} & \cdots & u_{3,k-1} & u_{3k} \\ 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}.$$

Generalized Linear Multistep Methods $s = 2$

$$\mathbf{A} = \begin{bmatrix} \lambda & 0 \\ a_{21} & \lambda \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} u_{11} & u_{12} & \dots & u_{1,k-1} & u_{1k} \\ u_{21} & u_{22} & \dots & u_{2,k-1} & u_{2k} \end{bmatrix},$$

$$\mathbf{B} = \begin{bmatrix} a_{21} & \lambda \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{V} = \begin{bmatrix} u_{21} & u_{22} & \dots & u_{2,k-1} & u_{2k} \\ 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix},$$

and

$$\mathbf{c} = \begin{bmatrix} k+1+\delta_1, & k+1 \end{bmatrix}^T.$$

k	p	δ_1	α
1	2	0.7071067811865475	90°
2	3	0.7335258700204377	90°
3	4	0.7504244509534406	90°
4	5	0.7626121809773775	86.04°
5	6	0.7720273095289394	75.14°
6	7	0.7796319013839141	57.37°
7	8	0.7859692192050817	30.48°
8	9	0.7913743037118012	-

Table: Values of δ_1 which maximize the angles α of $A(\alpha)$-stability for GLMMs2.

Angles α of $A(\alpha)$-stability

GLMMs2			GLMMs3			MEBDF			BDF		
k	p	α	k	p	α	k	p	α	k	p	α
-	-	-	-	-	-	-	-	-	1	1	90°
1	2	90°	1	2	90°	1	2	90°	2	2	90°
2	3	90°	2	3	90°	2	3	90°	3	3	86.03°
3	4	90°	3	4	90°	3	4	90°	4	4	73.35°
4	5	86.04°	4	5	88.25°	4	5	88.36°	5	5	51.84°
5	6	75.14°	5	6	83.41°	5	6	83.07°	6	6	17.84°
6	7	57.37°	6	7	76.46°	6	7	74.48°	-	-	-
7	8	30.48°	7	8	67.23°	7	8	61.98°	-	-	-
8	9	-	8	9	55.13°	8	9	42.87°	-	-	-

1. General Linear Methods
   - Formulation of GLMs
   - RK, LMM and BDF represented as GLMs
   - GLMs as framework to analyze and generalize
   - MEBDF represented as GLMs
   - Generalized Linear Multistep Methods

2. Self Starting GLMs
   - Introduction
   - Singly Diagonally-Implicit Methods
   - Explicit Methods
   - Implicit-Explicit Methods

3. Work in progress and future work

General Linear Methods

$$\begin{cases} Y_i^{[n]} = h\sum_{j=1}^{s} a_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} u_{ij} y_j^{[n-1]}, & i = 1, 2, \ldots, s, \\ y_i^{[n]} = h\sum_{j=1}^{s} b_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} v_{ij} y_j^{[n-1]}, & i = 1, 2, \ldots, r, \end{cases}$$

for $n = 1, 2, \ldots, N$, where $Nh = T - t_0$.

Internal stages:

$$Y_i^{[n]} = y(t_{n-1} + c_i h) + O(h^{q+1}), \quad i = 1, 2, \ldots, s,$$

External approximations:

$$y_i^{[n]} = \sum_{k=0}^{p} q_{ik} h^k y^{(k)}(t_n) + O(h^{p+1}), \quad i = 1, 2, \ldots, r.$$

General Linear Methods

$$\begin{cases} Y_i^{[n]} = h\sum_{j=1}^{s} a_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} u_{ij} y_j^{[n-1]}, & i = 1, 2, \ldots, s, \\ y_i^{[n]} = h\sum_{j=1}^{s} b_{ij} f(Y_j^{[n]}) + \sum_{j=1}^{r} v_{ij} y_j^{[n-1]}, & i = 1, 2, \ldots, r, \end{cases}$$

for $n = 1, 2, \ldots, N$, where $Nh = T - t_0$.

Internal stages:

$$Y_i^{[n]} = y(t_{n-1} + c_i h) + O(h^{q+1}), \quad i = 1, 2, \ldots, s,$$

External approximations:

$$\boxed{y_i^{[n]} = \sum_{k=0}^{p} q_{ik} h^k y^{(k)}(t_n) + O(h^{p+1}), \quad i = 1, 2, \ldots, r.}$$

In the GLMs literature, attention has focused almost exclusively on methods with *high stage order*, that is $q = p$ or $q = p - 1$.

General Linear Methods can be written in matrix form as

$$\begin{bmatrix} Y^{[n]} \\ \hline y^{[n]} \end{bmatrix} = \left[\begin{array}{c|c} \mathbf{A}\otimes\mathbf{I} & \mathbf{U}\otimes\mathbf{I} \\ \hline \mathbf{B}\otimes\mathbf{I} & \mathbf{V}\otimes\mathbf{I} \end{array}\right] \begin{bmatrix} hf(Y^{[n]}) \\ \hline y^{[n-1]} \end{bmatrix}, \quad n = 1, 2, \ldots$$

where

$$Y^{[n]} = y(t_{n-1} + \mathbf{c}h) + O(h^{q+1}),$$

$$\boxed{y^{[n]} = (\mathbf{W}\otimes\mathbf{I})z(t_n, h) + O(h^{p+1}),}$$

and

$$z(t,h) = \left[y(t),\ hy'(t),\ \ldots,\ h^p y^{(p)}(t)\right]^T.$$

We consider the case $\mathbf{W} = \left[\widetilde{W}, \mathbf{0}\right]$, where $\widetilde{W} \in \mathbb{R}^{r,2}$

(e.g. $r = 2$, $\widetilde{W} = I_2 \rightarrow$ method in Nordsieck form).

We consider the case $\mathbf{W} = \left[\widetilde{W}, \mathbf{0}\right]$, where $\widetilde{W} \in \mathbb{R}^{r,2}$.

Pros

- No need for a starting procedure and very easy (or no) finishing procedure;

We consider the case $\mathbf{W} = \left[\widetilde{W}, \mathbf{0}\right]$, where $\widetilde{W} \in \mathbb{R}^{r,2}$.

PROS

- No need for a starting procedure and very easy (or no) finishing procedure;
- Multistep methods with *one-step structure*: very easy rescaling procedure in case of stepsize changing, since the input vector $y^{[n-1]}$ depends only on t_{n-1} and h;

We consider the case $\mathbf{W} = \left[\widetilde{W}, \mathbf{0}\right]$, where $\widetilde{W} \in \mathbb{R}^{r,2}$.

PROS

- No need for a starting procedure and very easy (or no) finishing procedure;
- Multistep methods with *one-step structure*: very easy rescaling procedure in case of stepsize changing, since the input vector $y^{[n-1]}$ depends only on t_{n-1} and h;
- Ability to achieve improved accuracy and stability properties;

We consider the case $\mathbf{W} = \left[\widetilde{W}, \mathbf{0}\right]$, where $\widetilde{W} \in \mathbb{R}^{r,2}$.

PROS

- No need for a starting procedure and very easy (or no) finishing procedure;
- Multistep methods with *one-step structure*: very easy rescaling procedure in case of stepsize changing, since the input vector $y^{[n-1]}$ depends only on t_{n-1} and h;
- Ability to achieve improved accuracy and stability properties;
- In some special case only one of the external stages actually requires new computation.

We consider the case $\mathbf{W} = \left[\widetilde{W}, \mathbf{0}\right]$, where $\widetilde{W} \in \mathbb{R}^{r,2}$.

Pros

- No need for a starting procedure and very easy (or no) finishing procedure;
- Multistep methods with *one-step structure*: very easy rescaling procedure in case of stepsize changing, since the input vector $y^{[n-1]}$ depends only on t_{n-1} and h;
- Ability to achieve improved accuracy and stability properties;
- In some special case only one of the external stages actually requires new computation.

Cons

- Slightly higher computational costs than RK, but no additional function evaluations are needed

Runge–Kutta represented as GLMs

$$\begin{cases} Y_i = y_n + h \sum_{j=1}^{s} a_{ij} f(Y_j), \qquad i = 1, 2, \ldots, s, \\ y_{n+1} = y_n + h \sum_{j=1}^{s} b_j f(Y_j) \end{cases}$$

$$\left[\begin{array}{c|c} \mathbf{A} & \mathbf{U} \\ \hline \mathbf{B} & \mathbf{V} \end{array}\right] = \left[\begin{array}{ccc|c} a_{11} & \cdots & a_{1s} & 1 \\ \vdots & \ddots & \vdots & \vdots \\ a_{s1} & \cdots & a_{ss} & 1 \\ \hline b_1 & \cdots & b_s & 1 \end{array}\right]$$

Singly Diagonally-Implicit Methods

DIRK

$$\left[\begin{array}{c|c} \mathbf{A} & \mathbf{U} \\ \hline \mathbf{B} & \mathbf{V} \end{array}\right] = \left[\begin{array}{cccc|c} \lambda & 0 & \cdots & 0 & 1 \\ a_{21} & \lambda & \cdots & 0 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{s1} & a_{s2} & \cdots & \lambda & 1 \\ \hline b_1 & b_2 & \cdots & b_s & 1 \end{array}\right]$$

Singly Diagonally-Implicit Methods

DIRK

$$\left[\begin{array}{c|c}\mathbf{A} & \mathbf{U}\\ \hline \mathbf{B} & \mathbf{V}\end{array}\right] = \left[\begin{array}{cccc|c}\lambda & 0 & \cdots & 0 & 1\\ a_{21} & \lambda & \cdots & 0 & 1\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ a_{s1} & a_{s2} & \cdots & \lambda & 1\\ \hline b_1 & b_2 & \cdots & b_s & 1\end{array}\right]$$

SSGLM

$$\left[\begin{array}{c|c}\mathbf{A} & \mathbf{U}\\ \hline \mathbf{B} & \mathbf{V}\end{array}\right] = \left[\begin{array}{cccc|cc}\lambda & 0 & \cdots & 0 & u_{11} & u_{12}\\ a_{21} & \lambda & \cdots & 0 & u_{21} & u_{22}\\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots\\ a_{s1} & a_{s2} & \cdots & \lambda & u_{s1} & u_{s2}\\ \hline b_{11} & b_{12} & \cdots & b_{1s} & v_{11} & v_{12}\\ b_{21} & b_{22} & \cdots & b_{2s} & v_{21} & v_{22}\end{array}\right]$$

Example, SSGLM with $p = s = 3$ and $q = 2$

Two-parameter family of methods of order $p = 3$ and stage order $q = 2$:

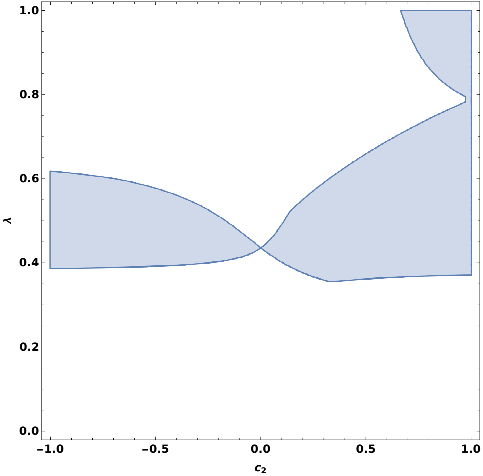
$$\mathbf{A} = \begin{bmatrix} \lambda & 0 & 0 \\ \frac{c2(c2-2\lambda)}{4\lambda} & \lambda & 0 \\ \frac{c2(3-6\lambda)+6\lambda-2}{12\lambda(c2-2\lambda)} & \frac{6\lambda^2-6\lambda+1}{3c2^2-6c2\lambda} & \lambda \end{bmatrix} \quad \mathbf{U} = \begin{bmatrix} 1 & \lambda \\ 1 & -\frac{c2^2}{4\lambda}+\frac{3c2}{2}-\lambda \\ 1 & \frac{2\left(6\lambda^2-6\lambda+1\right)-3c2\left(4\lambda^2-6\lambda+1\right)}{12c2\lambda} \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} \frac{c2(3-6\lambda)+6\lambda-2}{12\lambda(c2-2\lambda)} & \frac{6\lambda^2-6\lambda+1}{3c2^2-6c2\lambda} & \lambda \\ -\frac{(c2-1)\left(6\lambda^3-18\lambda^2+9\lambda-1\right)}{2\lambda(6\lambda^2-6\lambda+1)(c2-2\lambda)} & \frac{12\lambda^4-42\lambda^3+36\lambda^2-11\lambda+1}{c2(6\lambda^2-6\lambda+1)(c2-2\lambda)} & \frac{3\lambda\left(2\lambda^2-4\lambda+1\right)}{6\lambda^2-6\lambda+1} \end{bmatrix}$$

$$\mathbf{V} = \begin{bmatrix} 1 & \frac{2\left(6\lambda^2-6\lambda+1\right)-3c2\left(4\lambda^2-6\lambda+1\right)}{12c2\lambda} \\ 0 & -\frac{(c2-1)\left(12\lambda^4-42\lambda^3+36\lambda^2-11\lambda+1\right)}{2c2(6\lambda^3-6\lambda^2+\lambda)} \end{bmatrix}$$

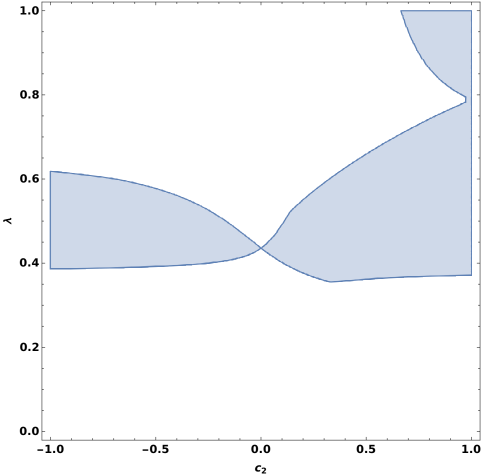
$$\mathbf{c} = \begin{bmatrix} 2\lambda & c_2 & 1 \end{bmatrix}^T$$

Example, SSGLM with $p = s = 3$ and $q = 2$



L-stable SSGLMp3 methods in the (c_2,λ)-plane.

Example, SSGLM with $p = s = 3$ and $q = 2$



L-stable SSGLMp3 methods in the (c_2,λ)-plane.

Let us show some numerical results for
$c_2 \approx 0.8495959692893016$ and $\lambda \approx 0.6177525723748765$

DIRK $p = 3$

Let us compare to L-stable DIRK with $p = s = 3$:

$$\mathbf{c} = \begin{bmatrix} \lambda & \frac{1}{2}(1+\lambda) & 1 \end{bmatrix}^T$$

$$\mathbf{A} = \begin{bmatrix} \lambda & 0 & 0 \\ -\frac{2(3\lambda^3-9\lambda^2+6\lambda-1)}{3(2\lambda^2-4\lambda+1)} & \lambda & 0 \\ \frac{4\lambda-1}{4(3\lambda^3-9\lambda^2+6\lambda-1)} & -\frac{3(2\lambda^2-4\lambda+1)^2}{4(3\lambda^3-9\lambda^2+6\lambda-1)} & \lambda \end{bmatrix}$$

$$\mathbf{b} = \begin{bmatrix} \frac{4\lambda-1}{4(3\lambda^3-9\lambda^2+6\lambda-1)} & -\frac{3(2\lambda^2-4\lambda+1)^2}{4(3\lambda^3-9\lambda^2+6\lambda-1)} & \lambda \end{bmatrix}^T,$$

where $\lambda \approx 0.4358665215...$ satisfies $\lambda^3 - 3\lambda^2 + \frac{3\lambda}{2} - \frac{1}{6} = 0$

Prothero-Robinson Equation

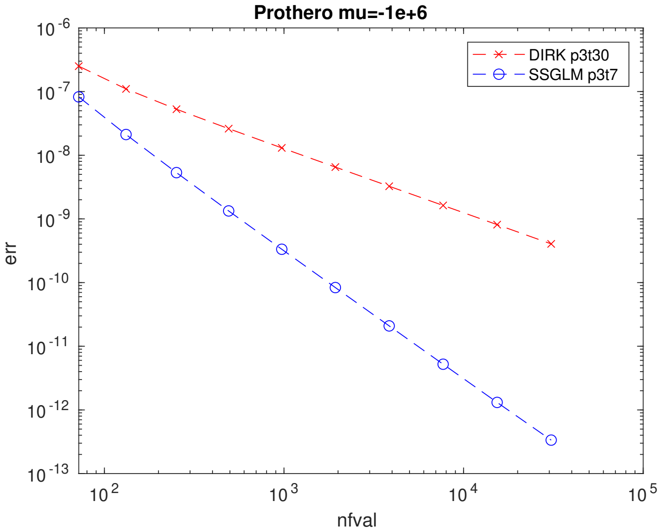
We consider the Prothero-Robinson equation

$$\begin{cases} y'(t) &= \mu(y(t) - \phi(t)) + \phi'(t), \\ y(0) &= \phi(0). \end{cases}$$

with

$$\mu = -10^6, \quad \phi(t) = \left(t + \frac{\pi}{4}\right) \quad \text{and} \quad T = 10.$$

Prothero: error vs *nfval*, for $\mu = -10^6$, $T = 10$, $p = 3$



DIRK $p = 3$: red line, SSGLM $p = 3$: blue line.

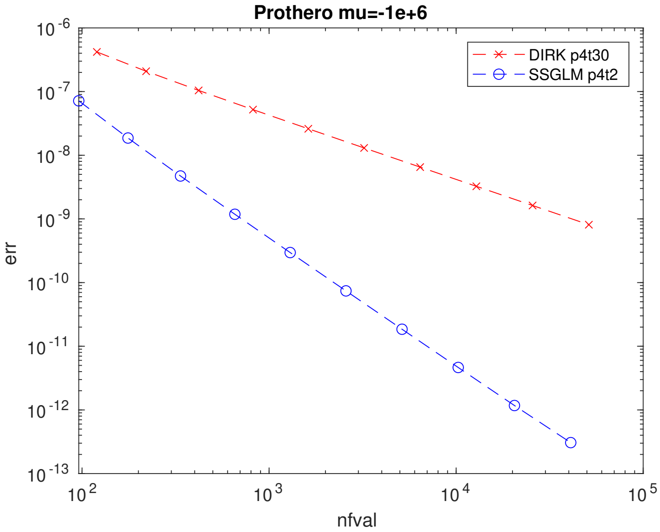
DIRK $p = 4$

Let us compare to L-stable DIRK with $p = 4, s = 5$ from Hairer & Wanner *Solving ODEs II* :

Table 6.5. *L*-stable SDIRK method of order 4

$$
\begin{array}{c|ccccc}
\frac{1}{4} & \frac{1}{4} & & & & \\
\frac{3}{4} & \frac{1}{2} & \frac{1}{4} & & & \\
\frac{11}{20} & \frac{17}{50} & -\frac{1}{25} & \frac{1}{4} & & \\
\frac{1}{2} & \frac{371}{1360} & -\frac{137}{2720} & \frac{15}{544} & \frac{1}{4} & \\
1 & \frac{25}{24} & -\frac{49}{48} & \frac{125}{16} & -\frac{85}{12} & \frac{1}{4} \\
\hline
y_1 = & \frac{25}{24} & -\frac{49}{48} & \frac{125}{16} & -\frac{85}{12} & \frac{1}{4}
\end{array}
\qquad (6.16)
$$

Prothero: error vs *nfval*, for $\mu = -10^6$, $T = 10$, $p = 4$



DIRK $p = 4$, $s = 5$: red line, SSGLM $p = 4$, $s = 5$: blue line.

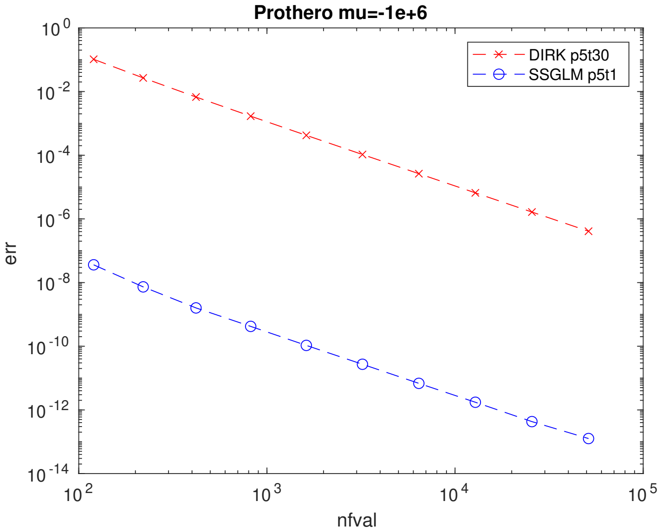
DIRK $p = 5$

Let us compare to L-stable DIRK with $p = 5, s = 5$ from Kennedy & Carpenter, *Diagonally Implicit Runge-Kutta Methods for Ordinary Differential Equations. A Review*, NASA Report TM–2016–219173 :

Table 24. SDIRK5()5L[1].

$\frac{4024571134387}{14474071345096}$	$\frac{4024571134387}{14474071345096}$	0	0	0	0
$\frac{5555633399575}{5431021154178}$	$\frac{9365021263232}{12572342979331}$	$\frac{4024571134387}{14474071345096}$	0	0	0
$\frac{5255299487392}{12852514622453}$	$\frac{2144716224527}{9320917548702}$	$\frac{-397905335951}{4008788611757}$	$\frac{4024571134387}{14474071345096}$	0	0
$\frac{3}{20}$	$\frac{-291541413000}{6267936762551}$	$\frac{226761949132}{4473940808273}$	$\frac{-1282248297070}{9697416712681}$	$\frac{4024571134387}{14474071345096}$	0
$\frac{10449500210709}{14474071345096}$	$\frac{-2481679516057}{4626464057815}$	$\frac{-197112422687}{6604378783090}$	$\frac{3952887910906}{9713059315593}$	$\frac{4906835613583}{8134926921134}$	$\frac{4024571134387}{14474071345096}$
b_i	$\frac{-2522702558582}{12162329469185}$	$\frac{1018267903655}{12907234417901}$	$\frac{4542392826351}{13702606430957}$	$\frac{5001116467727}{12224457745473}$	$\frac{1509636094297}{3891594770934}$

Prothero: error vs *nfval*, for $\mu = -10^6$, $T = 10$, $p = 5$



DIRK $p = 5, s = 5$: red line, SSGLM $p = 5, s = 5$: blue line.

Van der Pol oscillator

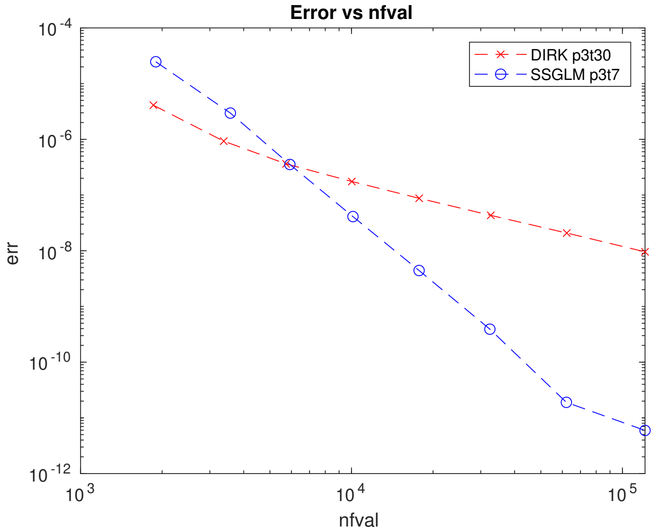
$$\begin{cases} y_1' = y_2, \\ y_2' = \dfrac{1}{\varepsilon}((1 - y_1^2)y_2 - y_1), \end{cases}$$

$t \in [0, T]$, with initial conditions

$$y_1(0) = 2, \quad y_2(0) = -\frac{2}{3} + \frac{10}{81}\varepsilon - \frac{292}{2187}\varepsilon^2 - \frac{1814}{19683}\varepsilon^3 + O(\varepsilon^4),$$

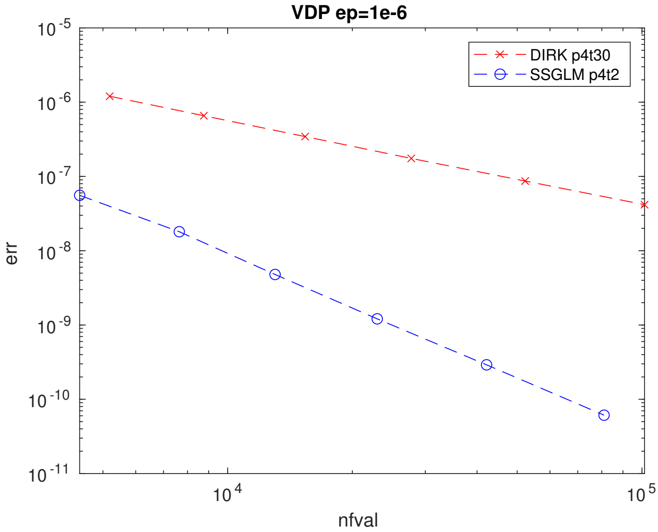
where ε represents a stiffness parameter.

VDP: error vs *nfval*, for $\lambda = 10^{-6}$, $T = 3/4$, $p = 3$



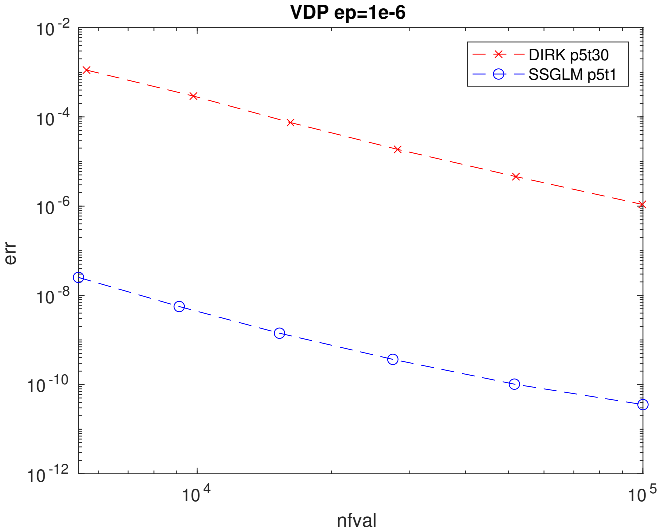
DIRK $p = 3$: red line, SSGLM $p = 3$: blue line.

VDP: error vs *nfval*, for $\lambda = 10^{-6}$, $T = 3/4$, $p = 4$



DIRK $p = 4$: red line, SSGLM $p = 4$: blue line.

VDP: error vs *nfval*, for $\lambda = 10^{-6}$, $T = 3/4$, $p = 5$



DIRK $p = 5$: red line, SSGLM $p = 5$: blue line.

Explicit Methods

Explicit RK

$$\left[\begin{array}{c|c} \mathbf{A} & \mathbf{U} \\ \hline \mathbf{B} & \mathbf{V} \end{array}\right] = \left[\begin{array}{cccc|c} 0 & 0 & \cdots & 0 & 1 \\ a_{21} & 0 & \cdots & 0 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{s1} & a_{s2} & \cdots & 0 & 1 \\ \hline b_1 & b_2 & \cdots & b_s & 1 \end{array}\right]$$

Explicit Methods

Explicit RK

$$\left[\begin{array}{c|c} \mathbf{A} & \mathbf{U} \\ \hline \mathbf{B} & \mathbf{V} \end{array}\right] = \left[\begin{array}{cccc|c} 0 & 0 & \cdots & 0 & 1 \\ a_{21} & 0 & \cdots & 0 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{s1} & a_{s2} & \cdots & 0 & 1 \\ \hline b_1 & b_2 & \cdots & b_s & 1 \end{array}\right]$$

Explicit SSGLM

$$\left[\begin{array}{c|c} \mathbf{A} & \mathbf{U} \\ \hline \mathbf{B} & \mathbf{V} \end{array}\right] = \left[\begin{array}{cccc|cc} 0 & 0 & \cdots & 0 & u_{11} & u_{12} \\ a_{21} & 0 & \cdots & 0 & u_{21} & u_{22} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ a_{s1} & a_{s2} & \cdots & 0 & u_{s1} & u_{s2} \\ \hline b_{11} & b_{12} & \cdots & b_{1s} & v_{11} & v_{12} \\ b_{21} & b_{22} & \cdots & b_{2s} & v_{21} & v_{22} \end{array}\right]$$

Explicit Methods

Explicit RK

$$\left[\begin{array}{c|c}\mathbf{A} & \mathbf{U}\\ \hline \mathbf{B} & \mathbf{V}\end{array}\right] = \left[\begin{array}{cccc|c} 0 & 0 & \cdots & 0 & 1\\ a_{21} & 0 & \cdots & 0 & 1\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ a_{s1} & a_{s2} & \cdots & 0 & 1\\ \hline b_1 & b_2 & \cdots & b_s & 1\end{array}\right]$$

Explicit SSGLM

$$\left[\begin{array}{c|c}\mathbf{A} & \mathbf{U}\\ \hline \mathbf{B} & \mathbf{V}\end{array}\right] = \left[\begin{array}{cccc|cc} 0 & 0 & \cdots & 0 & u_{11} & u_{12}\\ a_{21} & 0 & \cdots & 0 & u_{21} & u_{22}\\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots\\ a_{s1} & a_{s2} & \cdots & 0 & u_{s1} & u_{s2}\\ \hline b_{11} & b_{12} & \cdots & b_{1s} & v_{11} & v_{12}\\ 0 & 0 & \cdots & 1 & 0 & 0\end{array}\right]$$

UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II

Example, Explicit SSGLM with $p = 3$.

Four-parameter family of methods of order $p = 3$:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 \\ a_{21} & 0 & 0 \\ a_{31} & a_{32} & 0 \end{bmatrix} \quad \mathbf{U} = \begin{bmatrix} 1 & 0 \\ 1 & c_2 - a_{21} \\ 1 & -a_{31} - a_{32} + 1 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} \frac{12a_{32}c_2^3-(12a_{32}+5)c_2^2+2(a_{32}+3)c_2-1}{6(c_2-1)c_2(2a_{32}c_2-1)} & \frac{1}{6c_2-6c_2^2} & \frac{2-3c_2}{6-6c_2} \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{V} = \begin{bmatrix} 1 & \frac{-3a_{32}c_2^2+(2a_{32}+1)c_2-1}{3(c_2-1)(2a_{32}c_2-1)} \\ 0 & 0 \end{bmatrix} \quad \mathbf{c} = \begin{bmatrix} 0 & c_2 & 1 \end{bmatrix}^T$$

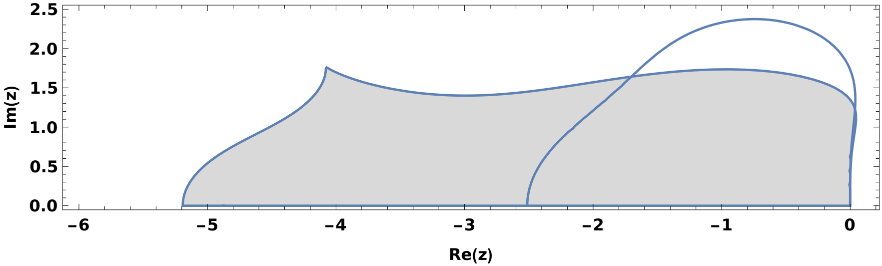
Example, Explicit SSGLM with $p = 3$

Trying to maximize the area of the Stability Region, for

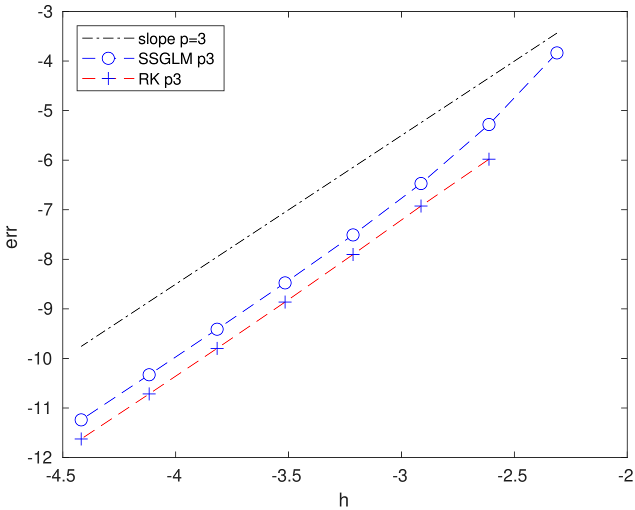
$$a_{21} = 0.2257586925723292, \quad a_{31} = -0.9077702963715302,$$

$$a_{32} = 1.5694810537893860, \quad c_2 = 0.3924017726910018.$$

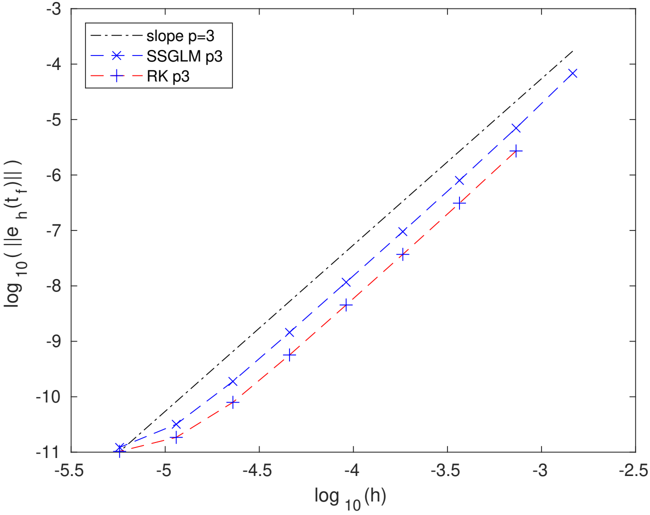
we obtain



Prothero: Explicit SSGLM, $\mu = -10^3$, $T = 10$, $p = 3$



VDP: Explicit SSGLM, $\lambda = 10^{-3}$, $T = 0.551$, $p = 3$



Implicit-explicit Self Starting General Linear Methods

Let us consider the following differential problem

$$\begin{cases} y'(t) = f\big(y(t)\big) + g\big(y(t)\big), \quad t \in [t_0, T], \\ y(t_0) = y_0 \in \mathbb{R}^m, \end{cases} \tag{3}$$

Where

- $f : \mathbb{R}^m \to \mathbb{R}^m$, represents the non-stiff processes
- $g : \mathbb{R}^m \to \mathbb{R}^m$, represents the stiff processes.

Implicit-explicit Self Starting General Linear Methods

Let us consider the following differential problem

$$\begin{cases} y'(t) = f\big(y(t)\big) + g\big(y(t)\big), \quad t \in [t_0, T], \\ y(t_0) = y_0 \in \mathbb{R}^m, \end{cases} \tag{3}$$

Where

- $f : \mathbb{R}^m \to \mathbb{R}^m$, represents the non-stiff processes $\longleftarrow$ explicit method
- $g : \mathbb{R}^m \to \mathbb{R}^m$, represents the stiff processes. $\longleftarrow$ implicit method

Implicit-Explicit Self Starting General Linear Methods

Implicit-explicit GLMs be written in matrix form

$$\begin{cases} Y^{[n+1]} = h(\mathbf{A} \otimes \mathbf{I})f\left(Y^{[n+1]}\right) + h(\mathbf{A}^* \otimes \mathbf{I})g\left(Y^{[n+1]}\right) + (\mathbf{U} \otimes \mathbf{I})y^{[n]}, \\ y^{[n+1]} = h(\mathbf{B} \otimes \mathbf{I})f\left(Y^{[n+1]}\right) + h(\mathbf{B}^* \otimes \mathbf{I})g\left(Y^{[n+1]}\right) + (\mathbf{V} \otimes \mathbf{I})y^{[n]}, \end{cases}$$

$n = 0, 1, \ldots, N-1, \mathbf{I} \in \mathbb{R}^m.$

Implicit-Explicit Self Starting General Linear Methods

Implicit-explicit GLMs be written in matrix form

$$\begin{cases} Y^{[n+1]} = h(\mathbf{A} \otimes \mathbf{I})f\left(Y^{[n+1]}\right) + h(\mathbf{A}^* \otimes \mathbf{I})g\left(Y^{[n+1]}\right) + (\mathbf{U} \otimes \mathbf{I})y^{[n]}, \\ y^{[n+1]} = h(\mathbf{B} \otimes \mathbf{I})f\left(Y^{[n+1]}\right) + h(\mathbf{B}^* \otimes \mathbf{I})g\left(Y^{[n+1]}\right) + (\mathbf{V} \otimes \mathbf{I})y^{[n]}, \end{cases}$$

$n = 0, 1, \ldots, N-1$, $\mathbf{I} \in \mathbb{R}^m$.
We assume that both methods, explicit and implicit, have the same abscissa vector $\mathbf{c}$ *and the same coefficients matrices* $\mathbf{U}$ *and* $\mathbf{V}$.

Implicit-Explicit Self Starting General Linear Methods

Implicit-explicit GLMs be written in matrix form

$$\begin{cases} Y^{[n+1]} = h(\mathbf{A} \otimes \mathbf{I})f\left(Y^{[n+1]}\right) + h(\mathbf{A}^* \otimes \mathbf{I})g\left(Y^{[n+1]}\right) + (\mathbf{U} \otimes \mathbf{I})y^{[n]}, \\ y^{[n+1]} = h(\mathbf{B} \otimes \mathbf{I})f\left(Y^{[n+1]}\right) + h(\mathbf{B}^* \otimes \mathbf{I})g\left(Y^{[n+1]}\right) + (\mathbf{V} \otimes \mathbf{I})y^{[n]}, \end{cases}$$

$n = 0, 1, \ldots, N-1$, $\mathbf{I} \in \mathbb{R}^m$.
We assume that both methods, explicit and implicit, have the same abscissa vector $\mathbf{c}$ *and the same coefficients matrices* $\mathbf{U}$ *and* $\mathbf{V}$.

For high stage order methods:

IM, order p and stage order $q = p$
EX, order p and stage order $q = p$ $\Rightarrow$ IMEX, order p and stage order $q = p$

Implicit-Explicit Self Starting General Linear Methods

Implicit-explicit GLMs be written in matrix form

$$\begin{cases} Y^{[n+1]} = h(\mathbf{A} \otimes \mathbf{I})f\left(Y^{[n+1]}\right) + h(\mathbf{A}^* \otimes \mathbf{I})g\left(Y^{[n+1]}\right) + (\mathbf{U} \otimes \mathbf{I})y^{[n]}, \\ y^{[n+1]} = h(\mathbf{B} \otimes \mathbf{I})f\left(Y^{[n+1]}\right) + h(\mathbf{B}^* \otimes \mathbf{I})g\left(Y^{[n+1]}\right) + (\mathbf{V} \otimes \mathbf{I})y^{[n]}, \end{cases}$$

$n = 0, 1, \ldots, N-1$, $\mathbf{I} \in \mathbb{R}^m$.
We assume that both methods, explicit and implicit, have the same abscissa vector $\mathbf{c}$ *and the same coefficients matrices* $\mathbf{U}$ *and* $\mathbf{V}$.

For high stage order methods:

IM, order p and stage order $q = p$
EX, order p and stage order $q = p$ $\Rightarrow$ IMEX, order p and stage order $q = p$

Here we cannot force high stage order.

Implicit-Explicit Self Starting General Linear Methods

$$\begin{cases} Y^{[n+1]} = h(\mathbf{A} \otimes \mathbf{I}) f\left(Y^{[n+1]}\right) + h(\mathbf{A}^* \otimes \mathbf{I}) g\left(Y^{[n+1]}\right) + (\mathbf{U} \otimes \mathbf{I}) y^{[n]}, \\ y^{[n+1]} = h(\mathbf{B} \otimes \mathbf{I}) f\left(Y^{[n+1]}\right) + h(\mathbf{B}^* \otimes \mathbf{I}) g\left(Y^{[n+1]}\right) + (\mathbf{V} \otimes \mathbf{I}) y^{[n]}, \end{cases}$$

$n = 0, 1, \ldots, N-1, \mathbf{I} \in \mathbb{R}^m$

Implicit-Explicit Self Starting General Linear Methods

$$\begin{cases} Y^{[n+1]} = h(\mathbf{A} \otimes \mathbf{I}) f\left(Y^{[n+1]}\right) + h(\mathbf{A}^* \otimes \mathbf{I}) g\left(Y^{[n+1]}\right) + (\mathbf{U} \otimes \mathbf{I}) y^{[n]}, \\ y^{[n+1]} = h(\mathbf{B} \otimes \mathbf{I}) f\left(Y^{[n+1]}\right) + h(\mathbf{B}^* \otimes \mathbf{I}) g\left(Y^{[n+1]}\right) + (\mathbf{V} \otimes \mathbf{I}) y^{[n]}, \end{cases}$$

$n = 0, 1, \ldots, N-1$, $\mathbf{I} \in \mathbb{R}^m$, where

$$y_i^{[n]} = q_{i0} y(t_n) + q_{i1} h f(t_n, y(t_n)) + q_{i1}^* h g(t_n, y(t_n)) + \mathcal{O}(h^{p+1}) \quad i = 1, 2.$$

Implicit-Explicit Self Starting General Linear Methods

$$\begin{cases} Y^{[n+1]} = h(\mathbf{A} \otimes \mathbf{I})f\big(Y^{[n+1]}\big) + h(\mathbf{A}^* \otimes \mathbf{I})g\big(Y^{[n+1]}\big) + (\mathbf{U} \otimes \mathbf{I})y^{[n]}, \\ y^{[n+1]} = h(\mathbf{B} \otimes \mathbf{I})f\big(Y^{[n+1]}\big) + h(\mathbf{B}^* \otimes \mathbf{I})g\big(Y^{[n+1]}\big) + (\mathbf{V} \otimes \mathbf{I})y^{[n]}, \end{cases}$$

$n = 0, 1, \ldots, N-1$, $\mathbf{I} \in \mathbb{R}^m$, where

$$y_i^{[n]} = q_{i0}y(t_n) + q_{i1}hf(t_n, y(t_n)) + q_{i1}^* hg(t_n, y(t_n)) + \mathcal{O}(h^{p+1}) \quad i = 1, 2.$$

We assume $q_{10} = 1$, $q_{20} = 0$, and $q_{11}^* = q_{11} = 0$, $q_{21}^* = 1$, so

$$y_1^{[n]} = y(t_n) + \mathcal{O}(h^{p+1}) \quad \longleftarrow \text{no finishing procedure}$$

$$y_2^{[n]} = hg(t_n, y(t_n)) + q_{21}hf(t_n, y(t_n)) + \mathcal{O}(h^{p+1})$$

IMEX SSGLMs - Numerical Experiments

We report some numerical results obtained by two IMEX SSGLMs:

- of order $p = 3$;
- with $s = 3$ and $s = 4$ stages, respectively;

IMEX SSGLMs - Numerical Experiments

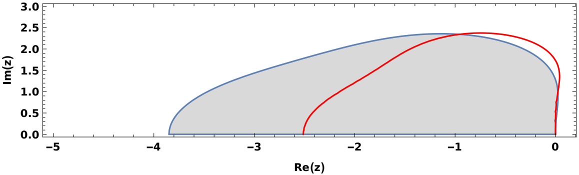
We report some numerical results obtained by two IMEX SSGLMs:

- of order $p = 3$;
- with $s = 3$ and $s = 4$ stages, respectively;
- with implicit part which
  - is Singly Diagonally-Implicit,
  - is L-stable, FSAL,
  - has stage order $q = 2$;

IMEX SSGLMs - Numerical Experiments

We report some numerical results obtained by two IMEX SSGLMs:

- of order $p = 3$;
- with $s = 3$ and $s = 4$ stages, respectively;
- with implicit part which
  - is Singly Diagonally-Implicit,
  - is L-stable, FSAL,
  - has stage order $q = 2$;
- with explicit part which has absolute stability region larger than explicit RKp3s3.



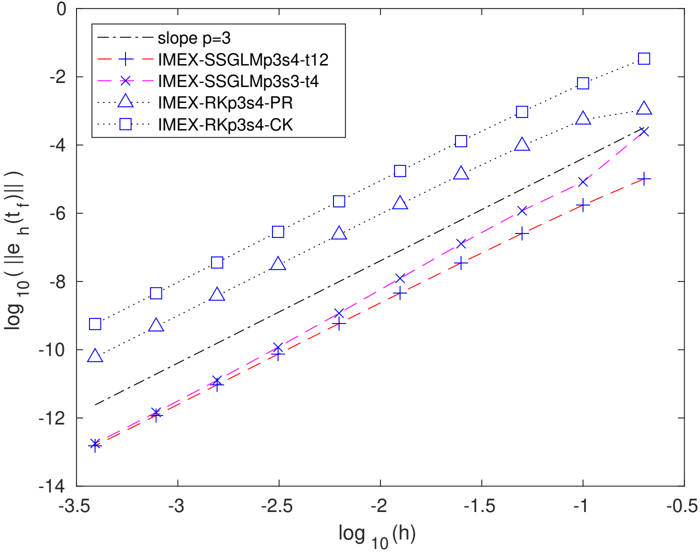
Additive Linear Test Equation

We consider the linear test equation

$$\begin{cases} y'(t) &= \lambda_0\, y(t) + \lambda_1\, y(t), \\ y(t_0) &= y_0, \end{cases}$$

$t \in [0, T]$, with $\lambda_0 = -1$, $\lambda_1 = -10$, $y_0 = 1$, $T = 1$.

Additive Linear Test Equation



Van der Pol Oscillator

We consider the van der Pol equation

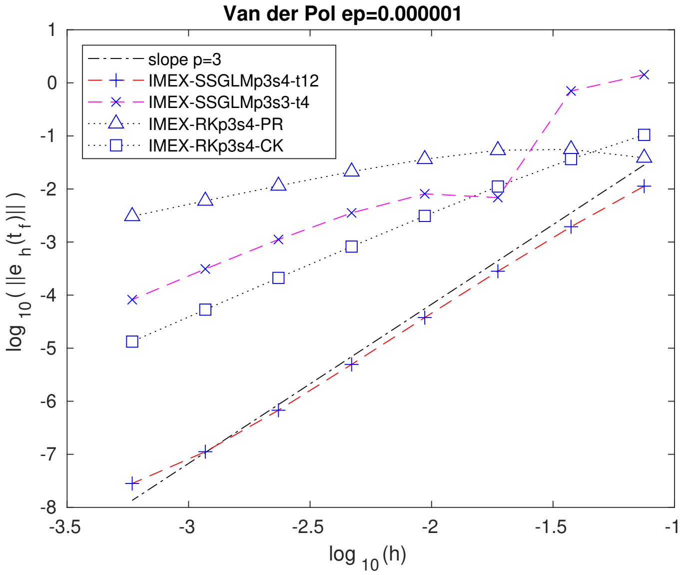
$$\begin{cases} y_1' = y_2, \\ y_2' = \dfrac{1}{\varepsilon}\left((1-y_1^2)y_2 - y_1\right), \end{cases}$$

$t \in [0, T]$, with initial conditions

$$y_1(0) = 2, \quad y_2(0) = -\frac{2}{3} + \frac{10}{81}\varepsilon - \frac{292}{2187}\varepsilon^2 - \frac{1814}{19683}\varepsilon^3 + O(\varepsilon^4),$$

where ε represents a stiffness parameter.

Van der Pol Oscillator, $\varepsilon = 10^{-6}$



Advection-reaction problem

$$\begin{cases} \dfrac{\partial u}{\partial t} + \alpha_1 \dfrac{\partial u}{\partial x} = -k_1 u + k_2 v + s_1, \\ \dfrac{\partial v}{\partial t} + \alpha_2 \dfrac{\partial v}{\partial x} = k_1 u - k_2 v + s_2, \end{cases} \qquad 0 \le x \le 1, \;\; 0 \le t \le 1$$

with parameters $\alpha_1 = 1, \;\alpha_2 = 0, \;k_1 = 10^6, \;k_2 = 2k_1, \;s_1 = 0, \;s_2 = 1$, and with initial and boundary values

$$u(x,0) = 1 + s_2 x, \quad v(x,0) = \frac{k_1}{k_2} u(x,0) + \frac{s_2}{k_2}, \quad 0 \le x \le 1,$$

$$u(0,t) = \gamma_1(t), \quad v(0,t) = \gamma_2(t), \quad 0 \le t \le 1.$$

Advection-reaction problem

$$\begin{cases} \dfrac{\partial u}{\partial t} + \alpha_1 \dfrac{\partial u}{\partial x} = -k_1 u + k_2 v + s_1, \\ \dfrac{\partial v}{\partial t} + \alpha_2 \dfrac{\partial v}{\partial x} = k_1 u - k_2 v + s_2, \end{cases} \qquad 0 \le x \le 1, \ \ 0 \le t \le 1$$

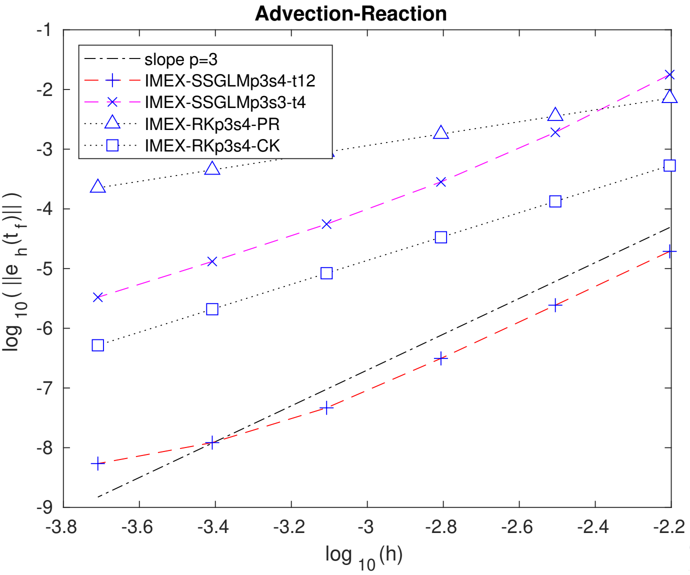
with parameters $\alpha_1 = 1, \ \alpha_2 = 0, \ k_1 = 10^6, \ k_2 = 2k_1, \ s_1 = 0, \ s_2 = 1$, and with initial and boundary values

$$u(x,0) = 1 + s_2 x, \quad v(x,0) = \frac{k_1}{k_2} u(x,0) + \frac{s_2}{k_2}, \quad 0 \le x \le 1,$$

$$u(0,t) = \gamma_1(t), \quad v(0,t) = \gamma_2(t), \quad 0 \le t \le 1.$$

Time dependent Dirichlet data $\gamma_1(t) = 1 - \sin(12t)^4$ at the left boundary. u_x is approximated by fourth-order central differences in the interior domain and third-order finite differences at the boundary.

Advection-reaction problem



Shallow water model

$$\begin{cases} \dfrac{\partial}{\partial t}h + \dfrac{\partial}{\partial x}(hv) = 0, \\ \dfrac{\partial}{\partial t}(hv) + \dfrac{\partial}{\partial x}\left(h + \dfrac{1}{2}h^2\right) = \dfrac{1}{\varepsilon}\left(\dfrac{h^2}{2} - hv\right), \end{cases}$$

where h is the water height with respect to the bottom and hv is the flux.

We use periodic boundary conditions and initial conditions at $t_0 = 0$

$$h(0,x) = 1 + \frac{1}{5}sin(8\pi x), \quad hv(0,x) = \frac{1}{2}h(0,x)^2, \quad \text{with } x \in [0,1].$$

Shallow water model

$$\begin{cases} \frac{\partial}{\partial t}h + \frac{\partial}{\partial x}(hv) = 0, \\ \frac{\partial}{\partial t}(hv) + \frac{\partial}{\partial x}\left(h + \frac{1}{2}h^2\right) = \frac{1}{\varepsilon}\left(\frac{h^2}{2} - hv\right), \end{cases}$$

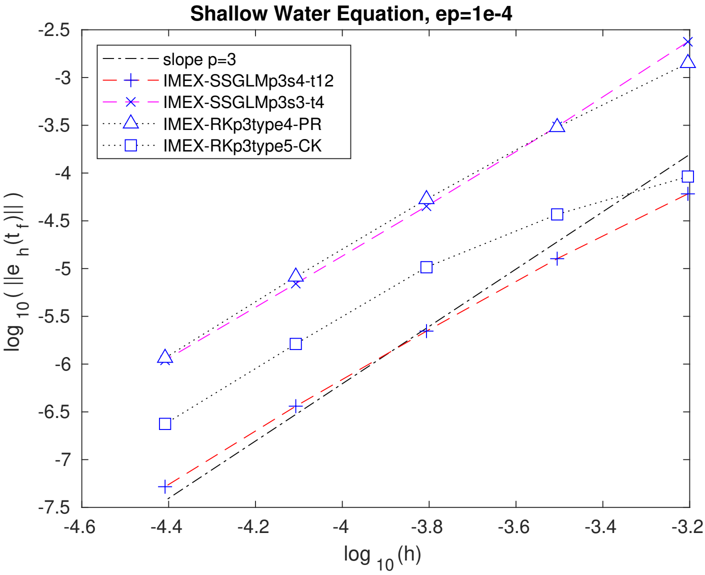
where h is the water height with respect to the bottom and hv is the flux.

We use periodic boundary conditions and initial conditions at $t_0 = 0$

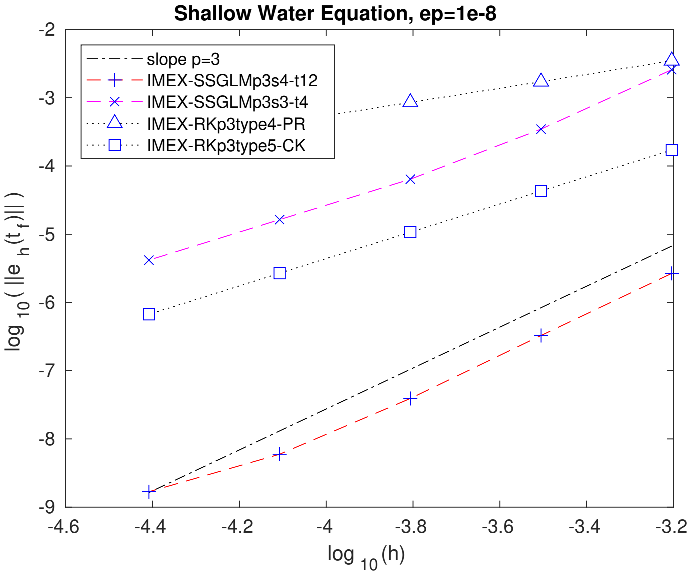
$$h(0,x) = 1 + \frac{1}{5}sin(8\pi x), \quad hv(0,x) = \frac{1}{2}h(0,x)^2, \quad \text{with } x \in [0,1].$$

The space derivative was discretized by a fifth order finite difference weighted essentially non-oscillatory (WENO5)

Shallow water model , $\varepsilon = 10^{-4}$



Shallow water model , $\varepsilon = 10^{-8}$



Work in progress

- Higher order explicit and implicit SSGLMs.
- Construction of higher order IMEX SSGLMs.
- Construction of *Asimptotically Accurate (AP)* IMEX methods for hyperbolic systems with relaxation.

Work in progress

- Higher order explicit and implicit SSGLMs.
- Construction of higher order IMEX SSGLMs.
- Construction of *Asimptotically Accurate (AP)* IMEX methods for hyperbolic systems with relaxation.

Future work

- Embedded SSGLM for error estimation.

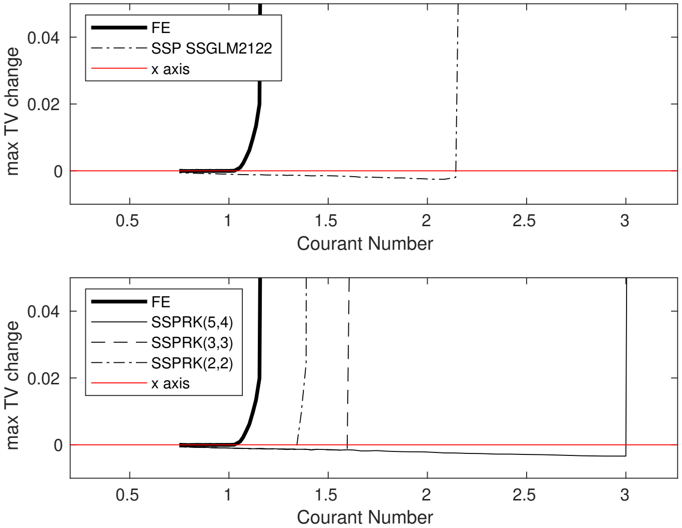
Work in progress

- Higher order explicit and implicit SSGLMs.
- Construction of higher order IMEX SSGLMs.
- Construction of *Asimptotically Accurate (AP)* IMEX methods for hyperbolic systems with relaxation.

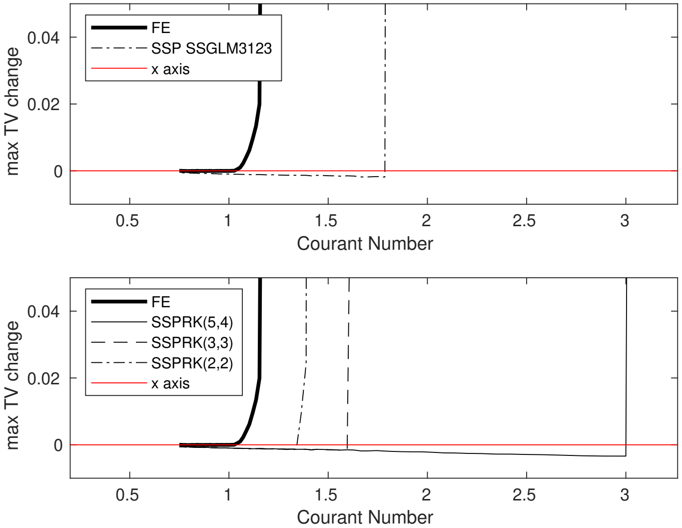
Future work

- Embedded SSGLM for error estimation.
- Strong Stability Preserving SSGLMs.

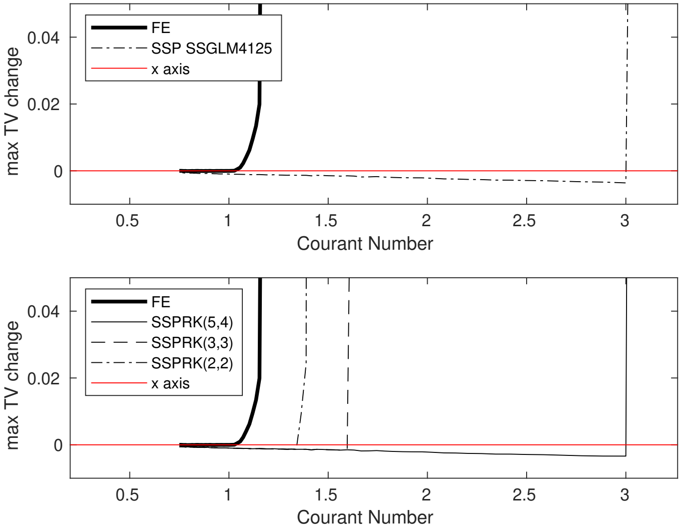
SSP SSGLMs, $p = 2$ - Inviscid Burgers' equation



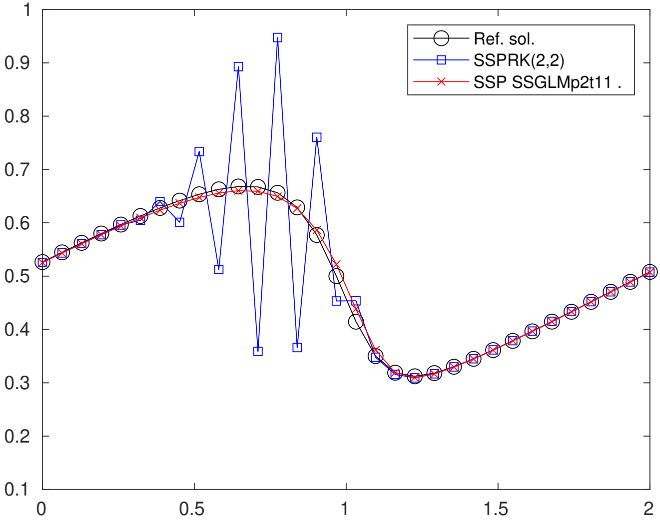
SSP SSGLMs, $p = 3$ - Inviscid Burgers' equation



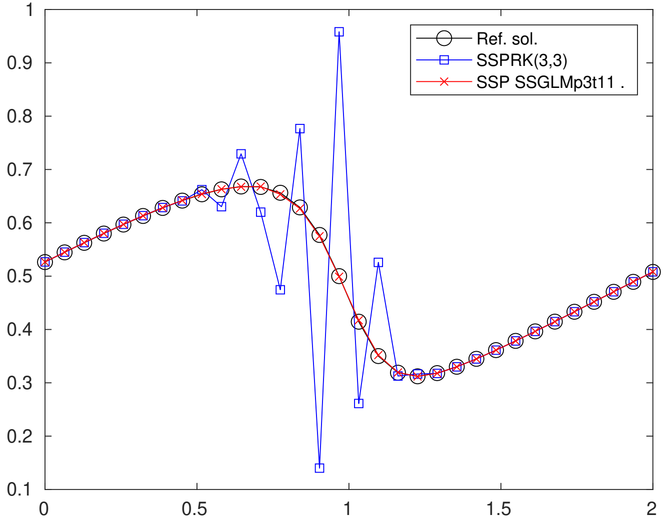
SSP SSGLMs, $p = 4$ - Inviscid Burgers' equation



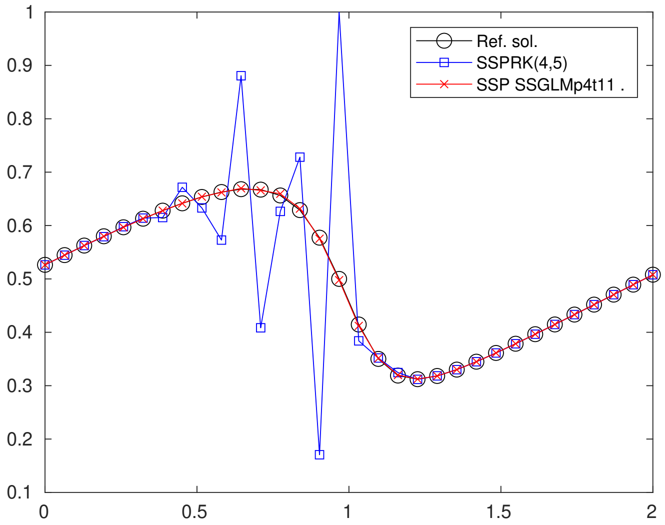
SSP SSGLMs, $p = 2$ - Inviscid Burgers' equation



SSP SSGLMs, $p = 3$ - Inviscid Burgers' equation



SSP SSGLMs, $p = 4$ - Inviscid Burgers' equation



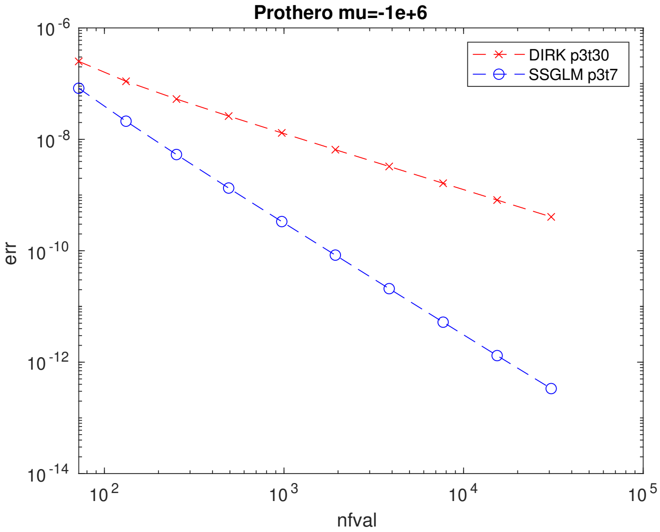
Work in progress

- Higher order explicit and implicit SSGLMs.
- Construction of higher order IMEX SSGLMs.
- Construction of *Asimptotically Accurate (AP)* IMEX methods for hyperbolic systems with relaxation.

Future work

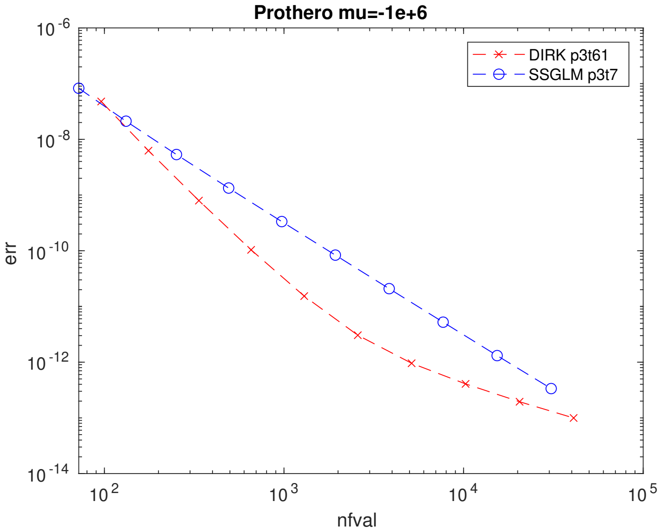
- Embedded SSGLM for error estimation.
- Strong Stability Preserving SSGLMs.
- Weak stage order for SSGLM.

WSO - Prothero with $\mu = -10^6$, $T = 10$, $p = 3$



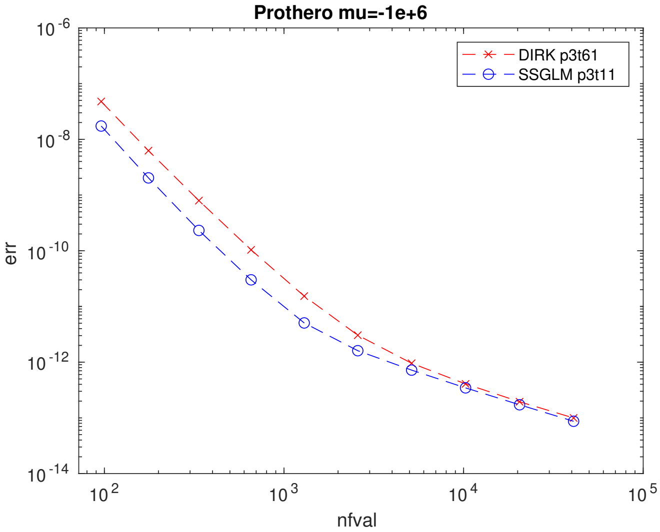
DIRK $p3s3$: red line, SSGLM $p3s3$: blue line.

WSO - Prothero with $\mu = -10^6$, $T = 10$, $p = 3$



DIRK $p3s4$ with WSO 3: red line, SSGLM $p3s3$: blue line.

WSO - Prothero with $\mu = -10^6$, $T = 10$, $p = 3$



DIRK $p3s4$, WSO 3: red line, SSGLM $p3s4$, WSO 3: blue line

How to get stage order $q = 3$?

$$\left[\begin{array}{c|c} \mathbf{A} & \mathbf{U} \\ \hline \mathbf{B} & \mathbf{V} \end{array}\right] = \left[\begin{array}{cccc|cc} \lambda & 0 & \cdots & 0 & u_{11} & u_{12} \\ a_{21} & \lambda & \cdots & 0 & u_{21} & u_{22} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ a_{s1} & a_{s2} & \cdots & \lambda & u_{s1} & u_{s2} \\ \hline a_{s1} & a_{s2} & \cdots & \lambda & u_{s1} & u_{s2} \\ 0 & 0 & \cdots & 1 & 0 & 0 \end{array}\right]$$

FSAL + *Special Structure* ensure the method to have the so-called *Runge-Kutta stability*, that is

$$p(w,z) = \det\left(w\mathbf{I} - \mathbf{M}(z)\right) = w(w - R(z)),$$

where $\mathbf{M}(z) = \mathbf{V} + z\mathbf{B}(\mathbf{I} - z\mathbf{A})^{-1}\mathbf{U}$.

How to get stage order $q = 3$?

$$\left[\begin{array}{c|c} \mathbf{A} & \mathbf{U} \\ \hline \mathbf{B} & \mathbf{V} \end{array}\right] = \left[\begin{array}{cccc|ccc} \lambda & 0 & \cdots & 0 & u_{11} & u_{12} & u_{13} \\ a_{21} & \lambda & \cdots & 0 & u_{21} & u_{22} & u_{23} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ a_{s1} & a_{s2} & \cdots & \lambda & u_{s1} & u_{s2} & u_{s3} \\ \hline a_{s1} & a_{s2} & \cdots & \lambda & u_{s1} & u_{s2} & u_{s3} \\ 0 & 0 & \cdots & 1 & 0 & 0 & 0 \\ b_{31} & b_{32} & \cdots & b_{3s} & v_{31} & v_{32} & v_{33} \end{array}\right]$$

where

$$\boxed{y^{[n]} = (\mathbf{W} \otimes \mathbf{I}) z(t_n, h) + O(h^{p+1}),}$$

and

$$z(t,h) = \left[y(t),\ hy'(t),\ h^2 y''(t),\ \ldots,\ h^p y^{(p)}(t) \right]^T .$$

Consider the case $\mathbf{W} = \left[\widetilde{W}, \mathbf{0} \right]$, where $\widetilde{W} = I_3$.

Thank you for your attention!!