DELFT UNIVERSITY OF TECHNOLOGY

REPORT 06-07

Numerical Methods for CVD Simulation

S. van Veldhuizen, C. Vuik, C.R. Kleijn

ISSN 1389-6520

Reports of the Department of Applied Mathematical Analysis

Delft 2006

Numerical Methods for CVD Simulation

S. van Veldhuizen* C. Vuik* C.R. Kleijn†

May 8, 2006

Abstract

In this study various numerical schemes for simulating 2D laminar reacting gas flows, as typically found in Chemical Vapor Deposition (CVD) reactors, are proposed and compared. These systems are generally modeled by means of many stiffly coupled elementary gas phase reactions between a large number of reactants and intermediate species. The purpose of this study is to develop robust and efficient solvers for the stiff heat-reaction system, whereby the velocities are assumed to be given. For non-stationary CVD simulation, an optimal combination in terms of efficiency and robustness between time integration, nonlinear solvers and linear solvers has to be found. Besides stability, which is important due to the stiffness of the problem, the preservation of non-negativity of the species is crucial. It appears that this extra condition on time integration methods is much more restrictive towards the time-step than stability. For a set of suitable time integration methods necessary conditions are represented. We conclude with a comparison of the workload between the selected time integration methods. This comparison has been done for a 2D test problem. The test problem does not represent a practical process, but represents only the computational problems.

1 Introduction

Chemical Vapor Deposition (CVD) is a process that synthesizes a thin solid film from the gaseous phase by a chemical reaction on a solid material. Applications of thin solid films as, for instance, insulating and (semi) conducting layers, can be found in many technological areas such as micro-electronics, optical devices and so on.

A CVD system is a chemical reactor, wherein the material to be deposited is injected as a gas and which contains the substrates on which deposition takes place. The energy to drive the chemical reaction is (usually) thermal energy. On the substrates surface reactions will take place resulting in deposition of a thin film.

*Delft University of Technology, Delft Institute of Applied Mathematics, Mekelweg 4, 2628 CD Delft, The Netherlands (s.vanveldhuizen@tudelft.nl, c.vuik@tudelft.nl)

†Delft University of Technology, Department of Multi Scale Physics, Prins Bernardlaan 6, 2628 BW Delft, The Netherlands (c.r.kleijn@tudelft.nl)

In CVD literature, and also other reactive flow literature, one is usually looking for the steady state solution of the so-called species equations (6). The species equations describe the transport of mass due to advective and diffusive transport, and due to the chemical reactions in the reactor. The usual procedure to find this steady state solution is to perform a (damped/relaxed) Newton iteration with an (arbitrary) initial solution. Hopefully, the Newton iteration then converges to the steady state. If this is not the case one performs some (artificial) time stepping in order to find a better initial solution for the next Newton iteration. In this paper we present suitable time integration methods for stiff problems. Furthermore, we compare these integration methods by their performance, in terms of efficiency.

In our research we are not looking for the steady state solution only, but we also want to approximate the transient solution. Since the time scales of advection and diffusion differ orders of magnitude from the time scales of the chemical reactions the system of species equations becomes stiff. Thus, in order to integrate the species equations in time in a stable manner, time integration methods have to be found that can handle stiff systems. Besides the stability issue for time integration, also the preservation of non-negativity of the species during time integration is important. It appears that this last condition on time integration methods is much more restrictive towards the time step than stability.

In this report issues on stability of time integration methods are discussed, as well as the positivity properties, see Section 3 and 4. The next step is to make a selection of suitable time integration methods, which will be presented in Section 5. This report will be concluded with numerical results, Section 6, obtained with the representative test problem proposed in Section 2.

2 Model for CVD Simulation

The mathematical model describing the CVD process consists of a set of partial differential equations with appropriate boundary conditions, which describe the gas flow, the transport of energy, the transport of species and the chemical reactions in the reactor.

The gas mixture in the reactor is assumed to behave as a continuum. This assumption is only valid when the mean free path of the molecules is much smaller than a characteristic dimension of the reactor. For Knudsen numbers $\mathrm{Kn} < 0.01$, where

$$\mathrm{Kn} = \frac{\xi}{L}, \tag{1}$$

the gas mixture behaves as a continuum. In (1) ξ is the mean free path length of the molecules and L a typical characteristic dimension of the reactor. For pressures larger than 100 Pa and typical reactor dimensions larger than 0.01 m the continuum approach can be used safely. See, for example, [9].

Furthermore, the gas mixture is assumed to behave as an ideal and transparent gas[1] behaving in accordance with Newton's law of viscosity. The gas flow in the reactor is

[1]By transparent we mean that the adsorption of heat radiation by the gas(es) will be small.

assumed to be laminar (low Reynolds number flow). Since no large velocity gradients appear in CVD gas flows, viscous heating due to dissipation will be neglected. We also neglect the effects of pressure variations in the energy equation.

The composition of the N component gas mixture is described in terms of the dimensionless mass fractions $\omega_i = \frac{\rho_i}{\rho}$, $i = 1, \ldots, N$, having the property

$$\sum_{i=1}^{N} \omega_i = 1. \tag{2}$$

The transport of mass, momentum and heat are described respectively by the continuity equation (3), the Navier-Stokes equations (4) and the transport equation for thermal energy (5) expressed in terms of temperature T:

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho \mathbf{v}), \tag{3}$$

$$\frac{\partial (\rho \mathbf{v})}{\partial t} = -(\nabla \rho \mathbf{v}) \cdot \mathbf{v} + \nabla \cdot \left[\mu \left(\nabla \mathbf{v} + (\nabla \mathbf{v})^{\mathbf{T}} \right) - \frac{\mathbf{2}}{\mathbf{3}} \mu (\nabla \cdot \mathbf{v}) \mathbf{I} \right] - \nabla \mathbf{P} + \rho \mathbf{g}, \tag{4}$$

$$\begin{aligned} c_p \frac{\partial (\rho T)}{\partial t} &= -c_p \nabla \cdot (\rho \mathbf{v} T) + \nabla \cdot (\lambda \nabla T) + \\ &\quad + \nabla \cdot \left(RT \sum_{i=1}^{N} \frac{\mathbb{D}_i^T}{M_i} \frac{\nabla f_i}{f_i} \right) + \sum_{i=1}^{N} \frac{H_i}{m_i} \nabla \cdot \mathbf{j}_i \\ &\quad - \sum_{i=1}^{N} \sum_{k=1}^{K} H_i \nu_{ik} R_k^g, \end{aligned} \tag{5}$$

with ρ gas mixture density, $\mathbf{v}$ mass averaged velocity vector, μ the viscosity, $\mathbf{I}$ the unit tensor, $\mathbf{g}$ gravity acceleration,c_p specific heat ($\frac{J}{\mathrm{mol} \cdot K}$), λ the thermal conductivity ($\frac{W}{m \cdot K}$) and R the gas constant. Gas species i has a mole fraction f_i, a molar mass m_i, a thermal diffusion coefficient $\mathbb{D}_i^T$, a molar enthalpy H_i and a diffusive mass flux $\mathbf{j}_i$. The stoichiometric coefficient of the i^{th} species in the k^{th} gas-phase reaction with net molar reaction rate R_k^g is ν_{ik}.

We assume that in the gas-phase K reversible reactions take place. For the k^{th} reaction the *net* molar reaction rate is denoted as R_k^g $\left(\frac{\text{mole}}{m^3 \cdot s}\right)$. For an explicit description of the net molar reaction rate, we refer to [9, 16]. The balance equation for the i^{th} gas species, $i = 1, \ldots, N$, in terms of mass fractions and diffusive mass fluxes is then given as

$$\frac{\partial (\rho \omega_i)}{\partial t} = -\nabla \cdot (\rho \mathbf{v} \omega_i) - \nabla \cdot \mathbf{j}_i + m_i \sum_{k=1}^{K} \nu_{ik} R_k^g, \tag{6}$$

where $\mathbf{j}_i$ is the diffusive flux. This mass diffusion flux is decomposed into concentration diffusion and thermal diffusion, e.g.,

$$\mathbf{j}_i = \mathbf{j}_i^C + \mathbf{j}_i^T. \tag{7}$$

The first type of diffusion, $\mathbf{j}_i^C$, occurs as a result of a concentration gradient in the system. Thermal diffusion is the kind of diffusion resulting from a temperature gradient. For a multicomponent gas mixture there are two approaches for the treatment of the ordinary diffusion, namely the full Stefan-Maxwell equations and an alternative approximation derived by Wilke. In this case, we have chosen for Wilke's approach. Then, the species concentration equations become

$$\begin{aligned}\frac{\partial(\rho\omega_i)}{\partial t} &= -\nabla\cdot(\rho\mathbf{v}\omega_i) + \nabla\cdot(\rho\mathbb{D}_i'\nabla\omega_i) + \nabla\cdot(\mathbb{D}_i^T\nabla(\ln T)) + \\ &\quad + m_i\sum_{k=1}^{K}\nu_{ik}R_k^g, \end{aligned} \tag{8}$$

where $\mathbb{D}_i'$ is an effective diffusion coefficient for species i and $\mathbb{D}_i^T$ the multi-component thermal diffusion coefficient for species i.

The main focus of our research is on efficient solvers for the above species equation(s) (8). Typically the time scales of the slow and fast reaction terms differ orders of magnitude from each other and from the time scales of the diffusion and advection terms, leading to extremely stiff systems.

2.1 Simplified CVD System

Since our research focuses on solving the species equations (8), we will only solve the coupled system of N species equations, where N denotes the number of gas-species in the reactor. Note that it suffices to solve the $N-1$ coupled species equations for all species except the carrier gas, where its mass fraction ω_{He} will be computed via the property

$$\sum_{i=1}^{N}\omega_i = 1. \tag{9}$$

For the moment we only focus on the development of efficient solvers for species equations. Therefore, we will omit the surface reactions in our system, because these boundary conditions will need some extra attention. Another simplification is that we assume that both the velocity field, temperature field, pressure field and density field are given. To be more precise, they are computed via another simulation package which was developed by Kleijn [10]. Furthermore, we omit thermal diffusion.

2.2 Reactor Geometry

The studied reactor configuration is illustrated in Figure 1. As computational domain we take, because of axisymmetry, one half of the $r-z$ plane.

The pressure in the reactor is 1 atm. From the top a gas-mixture, consisting of silane and helium, enters the reactor with a uniform temperature $T_{\mathrm{in}} = 300$ K and a uniform velocity u_{in}. The inlet silane mole fraction is $f_{\mathrm{in,SiH_4}} = 0.001$, whereas the rest is helium.

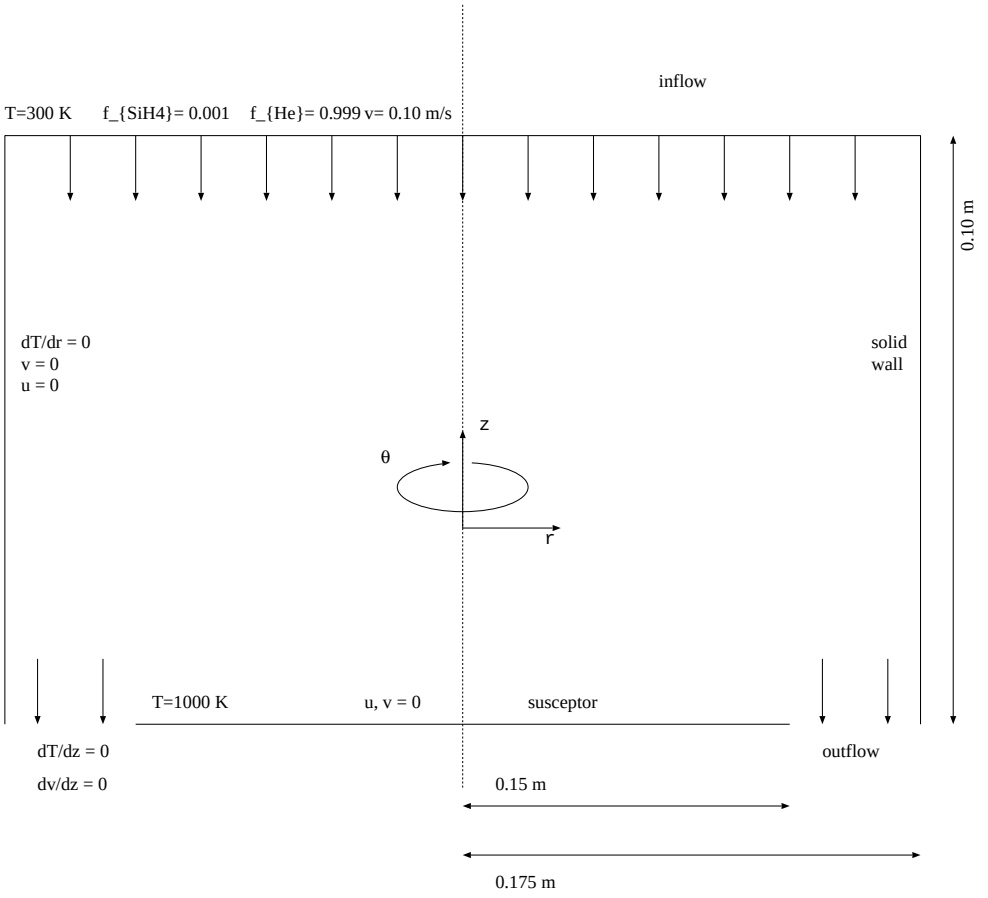


Figure 1: Reactor geometry

At a distance of 10 cm. below the inlet a susceptor with temperature $T = 1000$ K and a diameter 30 cm. is placed. As mentioned before, on this surface no reactions (will) take place. Unlike the problem considered in [10] the susceptor does not rotate.

2.3 Gas-Phase Reaction Model

We consider a CVD process that deposits solid silicon Si from gaseous silane SiH_4. In CVD literature one usually considers gas phase chemistry models consisting of 16 - 50 species. In our simplified CVD system we consider a gas mixture consisting of 7 species. Besides the carrier gas helium and the reactive specie silane, the mixture contains

- silane SiH_4,
- helium He,
- silylane SiH_2,
- H_2SiSiH_2,
- disilane Si_2H_6,
- trisilane Si_3H_8, and
- hydrogen H_2.

The reactive species in the gas mixture satisfy the reaction mechanism

$$\begin{array}{lccc} \text{G1}: & SiH_4 & \rightleftarrows & SiH_2 + H_4 \\ \text{G2}: & Si_2H_6 & \rightleftarrows & SiH_4 + SiH_2 \\ \text{G3}: & Si_2H_6 & \rightleftarrows & H_2SiSiH_2 + H_2 \\ \text{G4}: & SiH_2{+}Si_2H_6 & \rightleftarrows & Si_3H_8 \\ \text{G5}: & 2Si_2 & \rightleftarrows & H_2SiSiH_2 \end{array}$$

As can be found in [16], the forward reaction rate k_{forward} is fitted according to the modified Law of Arrhenius as

$$k^g_{\text{forward}}(T) = A_k T^{\beta_k} e^{\frac{-E_k}{RT}}. \tag{10}$$

The backward reaction rates are self consistent with

$$k^g_{\text{backward}}(T) = \frac{k^g_{\text{forward}}(T)}{K^g} \left(\frac{RT}{P^0}\right)^{\sum_{i=1}^N \nu_{ik}}, \tag{11}$$

where the reaction equilibrium K^g is approximated by

$$K^g(T) = A_{\text{eq}} T^{\beta_{\text{eq}}} e^{\frac{-E_{\text{eq}}}{RT}}. \tag{12}$$

In Table 1 the forward rate constants A_k, β_k and E_k are given. The fit parameters for the gas phase equilibrium are given in Table 2. In (10) - (12) R is the universal gas constant, i.e., $R = 8.314 \, \frac{\text{J}}{\text{mole·K}}$.

Reaction	A_k	β_k	E_k
G1	$1.09 \cdot 10^{25}$	-3.37	256000
G2	$3.24 \cdot 10^{29}$	-4.24	243000
G3	$7.94 \cdot 10^{15}$	0.00	236000
G4	$1.81 \cdot 10^{8}$	0.00	0
G5	$1.81 \cdot 10^{8}$	0.00	0

Table 1: Fit parameters for the forward reaction rates (10)

Reaction	A_{eq}	β_{eq}	E_{eq}
G1	$6.85 \cdot 10^{5}$	0.48	235000
G2	$1.96 \cdot 10^{12}$	-1.68	229000
G3	$3.70 \cdot 10^{7}$	0.00	187000
G4	$1.36 \cdot 10^{-12}$	1.64	-233000
G5	$2.00 \cdot 10^{-7}$	0.00	-272000

Table 2: Fit parameters for the equilibrium constants (12)

Besides the chemical model of the reacting gases, also some other properties of the gas mixture are needed. As mentioned before, the inlet temperature of the mixture is 300 K, and the temperature at the susceptor is 1000 K. The pressure in the reactor is 1 atm., which is equal to $1.013 \cdot 10^5$ Pa. Other properties are given in Table 3. The diffusion coefficients, according to Fick's Law, are given in Table 4.

Density $\rho(T)$	$1.637 \cdot 10^{-1} \cdot \frac{300}{T}$	[kg/m^3]
Specific heat c_p	$5.163 \cdot 10^3$	[J/kg/K]
Viscosity μ	$1.990 \cdot 10^{-5} \left(\frac{T}{300}\right)^{0.7}$	[kg/m/s]
Thermal conductivity λ	$1.547 \cdot 10^{-1} \left(\frac{T}{300}\right)^{0.8}$	[W/m/K]

Table 3: Gas mixture properties

SiH_4	$4.77 \cdot 10^{-6} \left(\frac{T}{300}\right)^{1.7}$
SiH_2	$5.38 \cdot 10^{-6} \left(\frac{T}{300}\right)^{1.7}$
H_2SiSiH_2	$3.94 \cdot 10^{-6} \left(\frac{T}{300}\right)^{1.7}$
Si_2H_6	$3.72 \cdot 10^{-6} \left(\frac{T}{300}\right)^{1.7}$
Si_3H_8	$3.05 \cdot 10^{-6} \left(\frac{T}{300}\right)^{1.7}$
H_2	$8.02 \cdot 10^{-6} \left(\frac{T}{300}\right)^{1.7}$

Table 4: Diffusion coefficients $\mathbb{D}_i'$, according to Fick's Law

3 Solution Methods for the Species Equations

The species equations (8) are PDEs of the advection-diffusion-reaction type. In order to have a unique solution appropriate boundary conditions and initial values have to be chosen. For a complete description we refer to [9, 16].

In general, it is impossible to find the exact solution of the system of $N-1$ coupled species equation. Therefore, we have to approximate its solution by numerical methods. To do this, we use the Method of Lines, shorter MOL, where the PDE (or system of PDEs) is first discretized in space on a certain grid Ω_h with mesh width $h > 0$ to yield a semi discrete system

$$w'(t) = F(t, w(t)), \quad 0 < t \leq T, \quad w(0) \text{ given}, \tag{13}$$

where $w(t) = (w_j(t))_{j=1}^m \in \mathbb{R}^m$, with m proportional to the number of grid points in spatial direction. The next step is to integrate the ODE system (13) with an appropriate time integration method.

We remark that the stiff reaction terms in CVD motivates to integrate parts of $F(t, w(t))$ implicitly. In general, due to the nonlinearities in the reaction term, (huge) nonlinear systems have to be solved. Most nonlinear solvers will need solutions of linear systems.

In order to approximate the solution of a system of species equations, choices have to be made with respect to

1. Spatial Discretization,
2. Time Integration,
3. Nonlinear Solver, and
4. Linear Solver.

The topic of this research is to find the best combination in terms of efficiency. Of course, efficiency becomes a hot topic in the case of transient 3D simulations, where the simulation times become huge. In the case of 2D simulations the simulation times are, with the nowadays PCs,s, acceptable. Note that if the computational cost of one time step is (very) expensive, then a time integration method that needs more, but computational cheaper, time steps is better in terms of efficiency.

Besides the efficiency criteria, also some other properties are desired, for example positivity, see Section 3.2. In order to make choices for spatial discretization, time integration, nonlinear and linear solvers, we formulate criteria in order of importance.

3.1 Stability

As already mentioned in Section 2 the system of species equations is stiff. First we pay some attention to this notion.

While the intuitive meaning of stiff is clear to all specialists, much controversy is going on about it's correct 'mathematical definition'. We cite Hundsdorfer & Verwer [7]:

> "Stiffness has no mathematical definition. Instead it is an operational concept, indicating the class of problems for which implicit methods perform (much) better than explicit methods."

The eigenvalues of the Jacobian $\frac{\delta f}{\delta y}$ play certainly a role in this decision, but quantities such as the dimension of the system, the smoothness of the solution or the integration interval are also important.

3.2 Positivity

An important property of chemical systems is *positivity*. By positivity we mean preservation of non-negativity for all components, which is a natural property for chemical species concentrations. As a consequence, it should also hold for the mathematical model of the process. Whenever numerical simulations of reactive flows will be done, the positivity property has to be fulfilled on four levels, i.e.

1. Mathematical Model,
2. Spatial Discretization,
3. Time Integration, and,
4. Iterative Solvers (Newton-Raphson, Krylov Subspace Methods)

The first one is obvious. When a chemically reacting flow is modeled in a way that preservation of non-negativity of the solution is not guaranteed, then this property cannot hold for the approximate solution.

Discretizing the PDE in space should not introduce 'wiggles', or (small) negative components, in the approximate solution, because then we (can) get blow up of the solution in finite time.

When the semi-discretization preserves non-negativity, then also the time integration method should preserve it. It appears that this extra condition on time integration methods, besides stability, is much more restrictive towards the time step than stability. We come back to this in the next section.

Finally, when positivity is preserved on the previous levels, then this property also has to be fulfilled within the iterative solvers needed in the time integration. Recall that (non)linear systems will appear in this application because of stiff reaction terms. Especially for the 3D simulations, and for 2D systems with a large number of species, iterative linear solvers will be needed. This topic will become important for future research.

3.3 Efficiency

Having stiff reaction terms in the species equations implies doing parts of the time integration implicitly. As remarked before, choosing the 'best' time integration method in terms of number of time steps, does not imply that this is the most efficient way to solve the system of $(N-1)$ species equations.

Nonlinear Solvers

Integrating nonlinear ODEs implicitly gives rise to nonlinear equations $F(x) = 0, x \in \mathbb{R}^n$. In order to solve these equations one has a limited choice of methods. Roughly speaking, we have the following methods:

1. (Pseudo) Newton Iteration,

2. Fixed Point Iteration, and

3. Broyden's Method.

To solve n dimensional systems $F(x) = 0$, with $F : \mathbb{R}^n \to \mathbb{R}^n$ and $x \in \mathbb{R}^n$, the Newton iteration is defined as

$$x^{k+1} = x^k - F'(x^k)^{-1}F(x^k). \tag{14}$$

Importance of the above method rests on the fact that in a neighborhood of x the error $\|x^{k+1} - x\|$ (remark that x is a solution of $F(x) = 0$) can be estimated by the inequality

$$\|x^{k+1} - x\| \leq c\|x^k - x\|^2. \tag{15}$$

Inequality (15) yields for a $c \in \mathbb{R}$ and a certain norm defined on $\mathbb{R}^n$. The above inequality states that the $(k+1)^{\text{th}}$ error is proportional to the k^{th} error squared.

Difficulties with the Newton iteration are that in each step a linear system has to be solved. In practical applications where the dimensions of the linear systems can be up to one million, or even larger, an appropriate solver has to be found. It appears that the computational costs of one time step depends mainly on the linear solver as well as the evaluation of $F(x)$ and $F'(x)$. Another difficulty is that convergence of the iteration is assured in a neighborhood of the solution. To get global convergence one can extend the Newton algorithm with, for example, line-search. Adding the line-search will guarantee that the succeeding iterates are norm reducing in the sense that

$$\|F(x^{k+1})\| \leq \|F(x^k)\| \quad k = 0, 1, 2, \ldots, \tag{16}$$

for some norm defined on $\mathbb{R}^n$. More on the topic of global convergence can be found in [8, 16].

The fixed point iteration has the advantage that it converges globally, as long as the spectral radius of the Jacobian of $F(x)$ is less than one. A disadvantage is that we have only first order convergence. This method is only attractive when the function evaluations are cheap.

Finally, the Broyden method is an extension of the secant method to n dimensions. In order to solve the scalar equation $f(x) = 0$ the *secant method* is given as

$$x^{k+1} = x^k - \frac{f(x^k)(x^k - x^{k-1})}{f(x^k) - f(x^{k-1})}, \tag{17}$$

where x^k is the current iteration and x^{k-1} the iteration before. As can be found in [16], the secant method converges with order equal to the golden ratio. Broyden's method is useful in the case that the Jacobian is not explicitly available, or computable.

Linear Solvers

As mentioned above, the computational costs of one time step depends mainly on the computational costs of solving linear systems that appear in the nonlinear solver. Iterative

linear solvers will come into play for three dimensional systems. In the case that two dimensional systems are considered, one can suffice with direct solvers. Via reordering of the unknowns and the discrete species equations, one can obtain band matrices with a 'small' bandwidth, such that the LU-factorization is still attractive.

4 Properties of Time Integration Methods

In this section we will pay more attention to the properties that are desired for the time integration methods. The main focus will be on positivity. First, we start with some (mathematical) definitions on stability.

4.1 Stability Continued

Since we are dealing with stiff problems, stability is an important issue. It is preferred to use methods that are unconditionally stable, by which we mean that there is no step-size restriction with respect to stability. A formal definition is given below. Before giving this definition we first have to define the so-called stability function.

Definition 4.1. *Consider the scalar, complex Dahlquist test equation*

$$y' = \lambda y, \quad \textit{with} \quad \lambda \in \mathbb{C}. \tag{18}$$

Application of an one-step method, like for example Euler Forward or Runge-Kutta method, gives approximations

$$w_{n+1} = R(\tau\lambda)w_n. \tag{19}$$

The function $R(z)$ $(z = \tau\lambda)$ is called the stability function of the method. The set

$$\mathcal{S} = \{z \in \mathbb{C} : |R(z)| \leq 1\}, \tag{20}$$

is called the stability region of the method.

Definition 4.2. *A method is called A-stable if the stability region $\mathcal{S} = \{z \in \mathbb{C} : |R(z)| \leq 1\}$ contains the left half plane $\mathbb{C}^-$.*
A method is called L-stable if the method is A-stable and $|R(\infty)| = 0$.

In the case that $|R(z)|$ are close to one for z with very large negative real part, the stiff parts of the approximate solution are damped out very slowly. To avoid difficulties with species with a short life span, which can appear in CVD, having L-stability is a desired property for CVD simulation.

An other form of stability, to be defined in the definition below, is A(α)-stability. This form of stability comes from the stability analysis of multi-step methods. By a multi-step method we mean that this method uses, unlike one-step methods like Euler Backward or Runge-Kutta methods, a fixed number of previous approximations. For more on multi-step methods we refer to [16, 5, 7].

Definition 4.3. *A method is called A(α)-stable (for $\alpha < \frac{\pi}{2}$) if the stability region $\mathcal{S} = \{z \in \mathbb{C} : |R(z)| \leq 1\}$ is a subset of*

$$\mathcal{S} \supset \{z \in \mathbb{C}^- : z = 0, \infty \text{ or } |\arg(-z)| \leq \alpha\} \tag{21}$$

From the above definition we see that A(α)-stability is a weaker form of stability than A-stability. For certain values of α, also these methods are interesting for time integration of species equations.

Example 4.1. *The BDF methods, see Section 5.1, are for $k = 1, 2$ A-stable. For $3 \leq k \leq 6$ the BDF methods is A(α)-stable, with α depending on k. See Table 5.*

k	1	2	3	4	5	6
α in rad	$\frac{\pi}{2}$	$\frac{\pi}{2}$	1.501	1.2741	0.8901	0.2967
α in degrees	90°	90°	86°	73°	51°	17°

Table 5: Values of α, in both radials as degrees, for given k.

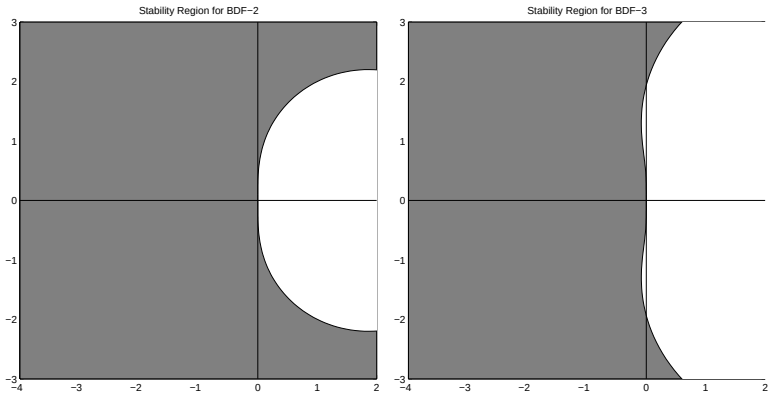


Figure 2: Stability regions of the BDF-2 method (left) and the BDF-3 method (right).

In Figure 2 the stability regions for the BDF-2 method and the BDF-3 method[2] *are given. Notice that the BDF-3 method is indeed not A-stable, as can be seen in Figure 2.*

IMEX

Instead of using a fully implicit time integration, one can also integrate only the extremely stiff part in an implicit way. Suppose we have the nonlinear system or semi-discretization

$$w'(t) = F(t, w(t)),$$

[2]BDF methods are extensively described in, for example, [7, 5, 16]

where $F(t, w(t))$ has the natural splitting

$$F(t, w(t)) = F_0(t, w(t)) + F_1(t, w(t)),$$

with F_0 is a non-stiff and F_1 stiff. In advection-diffusion-reaction systems the non-stiff term is for instance advection and the stiff terms the discretized diffusion and reactions. The non-stiff term is suitable for explicit time integration while the stiff terms are more suitable for implicit integration methods.

An example of a method that combines explicit as well as implicit treatment of respectively the non-stiff term $F_0(t, w(t))$ and stiff term $F_1(t, w(t))$ is the following one:

$$w_{n+1} = w_n + \tau \left(F_0(t_n, w_n) + (1-\theta) F_1(t_n, w_n) + \theta F_1(t_{n+1}, w_{n+1}) \right), \tag{22}$$

where the parameter $\theta \geq \frac{1}{2}$. This method is a combination of the Euler Forward method, which is explicit, and the implicit θ -method. Methods that are mixtures of *IM*plicit and *EX*plicit methods are called *IMEX* methods. The method given in (22) is called the IMEX-θ Method.

Consider the test equation

$$w'(t) = \lambda_0 w(t) + \lambda_1 w(t),$$

and let $z_j = \tau \lambda_j$ for $j = 0, 1$. Applying the IMEX- θ method to this test equation gives

$$R(z_0, z_1) = \frac{1 + z_0 + (1-\theta) z_1}{1 - \theta z_1}. \tag{23}$$

Stability of the IMEX- θ method thus requires

$$|R(z_0, z_1)| \leq 1. \tag{24}$$

To analyze the stability region (24) we have two starting points:

1. Assume the implicit part of the method to be stable, in fact A-stable, and investigate the stability region of the explicit part,

2. Assume the explicit part of the method to be stable and investigate the stability region of the implicit part.

Starting with the first point, we assume the implicit part of the IMEX- θ method to be A-stable. Define the set

$$\mathcal{D}_0 = \{ z_0 \in \mathbb{C} : \text{the IMEX scheme is stable } \forall z_1 \in \mathbb{C}^- \},$$

where $\mathbb{C}^-$ is the set

$$\mathbb{C}^- = \{ z \in \mathbb{C} : \text{Re } z \leq 0 \}.$$

The question is whether the set $\mathcal{D}_0$ is smaller, larger or equally shaped in comparison with the stability region of Euler Forward. The boundary of the region $\mathcal{D}_0$ is given in Figure

3. For a detailed derivation of this boundary, we refer to [16]. A similar way of reasoning can be done to derive the boundary of $\mathcal{D}_1$, whereby

$$\mathcal{D}_1 = \{z_1 \in \mathbb{C} \ : \text{the IMEX scheme is stable } \forall z_0 \in \mathcal{S}_0\}, \tag{25}$$

and $\mathcal{S}_0$ the stability region of Euler Forward. The boundary of $\mathcal{D}_1$ is also given in Figure 3.

We remark that for $\theta = 1$ the IMEX method has favorable stability properties. It could be seen as a form of time splitting where we first solve the explicit part with Euler Forward and the implicit part with Euler Backward. However, using this IMEX method we do not have errors as consequence of

- intermediate results that are inconsistent with the full equation,
- intermediate boundary conditions to solve these intermediate results.

If one uses this IMEX- θ method with $\theta = \frac{1}{2}$, then one has to pay a little more attention to stability. If, for instance, one has a system with complex eigenvalues, then the IMEX method will not be unconditionally stable for $\theta = \frac{1}{2}$.

The general idea of IMEX methods has been explained. We conclude with the remark that IMEX extensions can also be applied to multi-step methods, Runge-Kutta methods and so on. The IMEX - θ method is the simplest example of an IMEX-Runge-Kutta method.

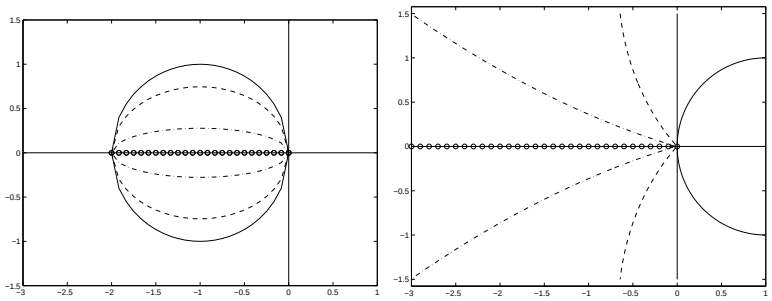


Figure 3: Boundary of regions $\mathcal{D}_0$ (left) and $\mathcal{D}_1$ (right) for $\theta = 0.5$ (circles), 0.51 (--·-), 0.6 (- -) and 1 (solid)

4.2 Positivity

Definition 4.4. *An ODE system*

$$w'(t) = F(t, w(t)) \quad t \geq 0, \tag{26}$$

is called positive, or non-negativity preserving, if

$$w(0) \geq 0 \Longrightarrow w(t) \geq 0, \quad \text{for all } t > 0. \tag{27}$$

It is easy to prove, for instance by using Theorem 4.5, which is given below, that linear systems

$$w'(t) = Aw(t), \quad w(t) \in \mathbb{R}^n, A \in \mathbb{R}^{n\times n}, A = (a_{ij}), \tag{28}$$

are positive if and only if $a_{ij} \geq 0$ for $i \neq j$. The next theorem provides a simple criterion on $F(t, w(t))$ to tell us whether the system

$$w'(t) = F(t, w(t)) \quad t \geq 0, \tag{29}$$

is positive.

Theorem 4.5. *Suppose that $F(t, w)$ is continuous and satisfies a Lipschitz condition with respect to w [3]. Then the system (29) is positive if and only if for any vector $w \in \mathbb{R}^m$ and all $i = 1, \ldots, m$ and $t \geq 0$,*

$$w \geq 0, \quad w_i = 0 \quad \Longrightarrow \quad F_i(t, w) \geq 0. \tag{30}$$

Proof. For a proof we refer to [7, Theorem 7.1, p.116] □

It is interesting to investigate the positivity for semi-discrete systems. Consider, for instance, the linear advection-diffusion equation in one dimension, i.e.,

$$u_t + (a(x,t)u)_x = (d(x,t)u_x)_x, \tag{31}$$

whereby $a(x,t)$ is the space and time dependent advection coefficient, and $d(x,t)$ the space and time dependent diffusion coefficient. Furthermore, $d(x,t) > 0$ and spatial periodic boundary conditions are assumed.

Discretizing (31) by means of central differences gives

$$\begin{aligned} w_j' &= \frac{1}{2h}\left(a_{j-\frac{1}{2}}(w_{j-1} + w_j) - a_{j+\frac{1}{2}}(w_j + w_{j+1})\right) + \\ &\quad \frac{1}{h^2}\left(d_{j-\frac{1}{2}}(w_{j-1} - w_j) - d_{j+\frac{1}{2}}(w_j - w_{j+1})\right), \end{aligned} \tag{32}$$

for $j = 1, \ldots, m$, where $w_j = w_j(t)$, $w_0 = w_m$, $w_{n+1} = w_1$ and

$$a_{j\pm\frac{1}{2}} = a(x_{j\pm\frac{1}{2}}, t), \quad d_{j\pm\frac{1}{2}} = a(x_{j\pm\frac{1}{2}}, t). \tag{33}$$

Using Theorem 4.5 it follows, after an elementary calculation, that the central discretization (32) is positive if and only if the cell Péclet numbers, defined as ah/d, satisfy

$$\max_{x,t} \frac{|a(x,t)|h}{d(x,t)} \leq 2. \tag{34}$$

[3] The condition

$$\|F(t, \tilde{v}) - F(t, v)\| \leq L\|\tilde{v} - v\|,$$

with $\tilde{v}, v \in \mathbb{R}^m$ and $F : \mathbb{R} \times \mathbb{R}^m \to \mathbb{R}^m$, is called a Lipschitz condition with Lipschitz constant L.

Discretizing (31) by means of first order upwind for the advection part, and second order central differences for the diffusive part, gives

$$\begin{aligned} w_j' &= \frac{1}{h}\left(a^+_{j-\frac{1}{2}}w_{j-1} + (a^+_{j-\frac{1}{2}} - a^-_{j-\frac{1}{2}})w_j + a^-_{j-\frac{1}{2}}w_{j+1})\right) \\ &\quad \frac{1}{h^2}\left(d_{j-\frac{1}{2}}(w_{j-1} - w_j) - d_{j-\frac{1}{2}}(w_j - w_{j+1})\right), \end{aligned} \tag{35}$$

where $a^+ = \max(a, 0)$ and $a^- = \min(a, 0)$. Hence, the semi-discrete system (35) is unconditionally positive.

Positivity of Semi-Discrete Species Equations

Adding reaction terms to (31) does not influence the positivity. Since the reaction terms that are being add are in the production-loss form

$$F(t, v) = p(t, v) - L(t, v)v, \tag{36}$$

where $p(t, v)$ is a vector and $L(t, v)$ a diagonal matrix. It can be found in [7] that source terms in this form are positive as long as $p(t, v)$ is positive.

Furthermore, we mention that the above results with respect to the one-dimensional advection-diffusion equation are easily generalized to higher dimensions. Therefore, discretizing the species concentrations equations in space by means of a hybrid scheme[4] ensures positivity of the semi-discretization.

A last remark with respect to upwind discretization. For higher order upwinding, for example third order upwinding, positivity is not ensured for all step-sizes.

Positivity on the Level of Time Integration

Positivity also restricts the use of time integration methods. It appears that unconditional positivity is a very restrictive requirement. In this section we will present results for non-linear systems $w'(t) = F(t, w(t))$. First we begin with exploring the positivity property for Euler Forward and Backward.

Euler Forward and Backward

We consider the non-linear semi-discretization $w'(t) = F(t, w(t))$. Suppose that $F(t, w(t))$ satisfies the condition:

Condition 4.6. *There is an $\alpha > 0$, with α as large as possible, such that $\alpha\tau \leq 1$ and*

$$v + \tau F(t, v) \geq 0 \quad \textit{for all} \quad t \geq 0, v \geq 0. \tag{37}$$

[4]The hybrid scheme uses central differences, except in the regions where $|Pe| > 2$. In the regions with $|Pe| > 2$ first order upwind is used.

Application of Euler Forward to the nonlinear system $w'(t) = F(t, w(t))$ gives

$$w_{n+1} = w_n + \tau F(t_n, w_n). \tag{38}$$

Provided that $w_n \geq 0$, Condition 4.6 guarantees positivity for w_{n+1} computed via Euler Forward (38).

Furthermore, assume that $F(t, w(t))$ also satisfies the following condition:

Condition 4.7. *For any $v \geq 0, t \geq 0$ and $\tau > 0$ the equation*

$$u = v + \tau F(t, u), \tag{39}$$

has a unique solution that depends continuously on τ and v.

According to the following theorem we have unconditional positivity for Euler Backward.

Theorem 4.8. *Conditions 4.6 and 4.7 imply positivity for Euler Backward for any step size τ.*

The following proof is taken from [7].

Proof of Theorem 4.8. For given t, v and with τ variable we consider the equation $u = v + \tau F(t, u)$ and we call its solution $u(\tau)$. We have to show that $v \geq 0$ implies $u(\tau) \geq 0$ for all positive τ. By continuity it is sufficient to show that $v > 0$ implies $u(\tau) \geq 0$. This is true because if we assume that $u(\tau) > 0$ for $\tau \leq \tau_0$, except for the i^{th} component $u_i(\tau_0) = 0$, then

$$0 = u_i = v_i + \tau_0 F_i(t, u(\tau_0)). \tag{40}$$

According to Condition 4.6 we have $F_i(t, u(\tau_0)) \geq 0$ and thus $v_i + \tau_0 F_i(t, u(\tau_0)) > 0$, which is a contradiction. □

Remark 4.9. *Unlike Condition 4.6, Condition 4.7 is not easy to be verified. As remarked in [7], it is sufficient to hold that $F(t, v)$ is continuously differentiable and*

$$\|I - \tau J_F(t, v)\| \leq C, \quad \text{for any } v \in \mathbb{R}^n,\ t \geq 0 \text{ and } \tau > 0, \tag{41}$$

whereby C is a positive constant and $J_F(t, v)$ the Jacobian of $F(t, v)$ with respect to v. Existence and uniqueness of the solution of

$$u = v + \tau F(t, u), \tag{42}$$

then follows from Hadamard's Theorem and the Implicit Function Theorem. The fact that the solution u of (42) depends continuously of τ, v and t also follows from the Implicit Function Theorem. Remark that condition (41) is easier to verify than Condition 4.7. Both Hadamard's Theorem and the Implicit Function Theorem can be found, for example, in [11].

Remark 4.10. *Application of Euler Backward to the nonlinear system $w'(t) = F(t, w(t))$ gives*

$$w_{n+1} - \tau F(t_n, w_{n+1}) = w_n. \tag{43}$$

As proven in Theorem 4.8, for every time step τ positivity of w_{n+1} is ensured as long as w_n is positive and $F(t, w)$ satisfies conditions (4.6) and (4.7). In practice it is impossible to compute the solution w_{n+1} of the nonlinear equation (43) exactly, thus one will use an iterative nonlinear solver. For example, the Newton Raphson iteration is in most applications a good choice. Since iterative nonlinear solvers will only give an approximation of the solution, it might be possible that it contains (small) negative components. Therefore, in practice even Euler Backward can produce negative concentrations.

As mentioned before, negative components of the concentration solution vector can cause a blow up in finite time. In the case the negative components of the solution are the result of rounding errors, then it is justified to set them to zero. We remark that in the case of rounding errors the negative components are very small (compared with the relative error, condition number and the machine precision). In the case one has negative components in the solution as consequence of the nonlinear (Newton) solver, then the most common method to avoid negative concentrations is *clipping*. A disadvantage of applying clipping is that mass is added, i.e., the integration method does not preserve mass any longer. A possible solution is given in [12], where for the positive integration of chemical kinetic systems a projection method is proposed. This projection method works as follows. First, a numerical approximation is computed via a traditional (read Runge Kutta or Rosenbrock method) scheme, which preserves mass. If there are negative components in the solution, then the nearest vector in the reaction simplex is found using a primal-dual optimization routine. The resulting vector, is a better approximation of the true solution and preserves mass. In the case of advection-diffusion-reaction systems, no references are available to do this.

Some General Remarks on Positivity

In the previous section it has been proven that the Euler Backward method is unconditionally positive, when Conditions 4.6 and 4.7 are satisfied. We remark that these conditions are actually conditions on $F(t, v)$. Unconditional positivity can only be true for implicit methods. One might hope to find more accurate methods with this unconditional positivity property. However, this hope is dashed by the following result, which holds for both linear and nonlinear systems, due to Bolley and Crouzeix [2].

Theorem 4.11. *Any method that is unconditionally positive, has order $p \leq 1$.*

For a proof we refer to [2]. The consequence of this theorem is that the only well-known method having unconditionally positivity is Euler Backward.

Finally, we remark that for higher order methods the time step can be restricted to impractically small values. Furthermore, we mention that for Diagonally Implicit Runge Kutta (DIRK) methods step size restrictions are known, such that positivity is guaranteed. We refer to [7, 16].

4.3 Relation Positivity and TVD

If the system of ODEs $w'(t) = F(t, w(t))$, with a appropriate initial condition $w(0) = w_0$, stands for a semi-discrete version of a conservation law, it is important that the fully discrete process is monotonic in the sense that

$$\|w_n\| \leq \|w_{n-1}\|. \tag{44}$$

The above monotonicity property reduces to the Total Variation Diminishing (TVD) property when we use the seminorm

$$|y|_{TV} = \sum_{j=1}^{n} |y_j - y_{j-1}|, \quad \text{with } y_0 = y_n, \text{ for } y \in \mathbb{R}^n, \tag{45}$$

in (44). When a numerical scheme satisfies this TVD property, then localized over- and undershoots are prevented.

The TVD property is developed for studying the properties of numerical schemes for solving hyperbolic conservation laws. In particular TVD and monoticity are important in the study of shocks in fluid flows.

In this section we will discuss some general results for TVD. It turns out that the conditions giving the TVD property are the same as giving the positivity property.

TVD Results for ODE Methods

Considering the ODE system $w'(t) = F(t, w(t))$ with a function F such that

$$|v + \tau F(t, v)| \geq |v| \quad \text{for all} \quad t \geq 0, \text{for all} v \in \mathbb{R}^n \quad \text{and } 0 \leq \tau \leq \tau^*. \tag{46}$$

Then, the above condition can be seen as a condition on the time step τ such that Euler Forward is TVD.

We remark that this condition is equivalent with the condition for positivity for Euler Forward (4.6). This can be seen as follows. Considering a positive nonlinear system $w'(t) = F(t, w(t))$, we have to find an α such that (4.6) holds. Taking

$$\alpha = \frac{1}{\tau^*}, \tag{47}$$

and considering $v \geq 0$, then (46) implies $v + \tau F(t, v) \geq 0$.

Applying the Backward Euler method to $w'(t) = F(t, w(t))$ gives

$$w_{n+1} = w_n + \tau F(t_{n+1}, w_{n+1}). \tag{48}$$

This method is TVD under Assumption (46) without any step size restriction. This can easily be seen from

$$\left(1 + \frac{\tau}{\tau^*}\right) w_{n+1} = w_n + \frac{\tau}{\tau^*} \left(w_{n+1} + \tau^* F(t_{n+1}, w_{n+1}).\right) \tag{49}$$

Using the TV seminorm we obtain

$$\left(1+\frac{\tau}{\tau^*}\right)|w_{n+1}|_{TV} \leq |w_n|_{TV} + \frac{\tau}{\tau^*}|w_{n+1}|_{TV}, \tag{50}$$

showing that $|w_{n+1}|_{TV} \leq |w_n|_{TV}$ for any $\tau > 0$.

For other one-step methods like Diagonally Implicit Runge Kutta methods step size restrictions can be derived to have the TVD property. These conditions are identical to those for positivity of DIRK methods. For TVD results for multistep, or other general results for TVD we refer to [3, 7, 20]. In [3] the rather disappointing result concerning the nonexistence of implicit TVD Runge Kutta or multistep methods of order higher than 1 has been proven. By adding some explicit stages to an explicit higher order RK or multistep scheme the TVD property can be recovered. In our case of stiff systems, this last artifact to save the TVD property is useless, because then the stiff reaction parts have to be integrated explicitly.

To conclude this section we would like to remark the following. With respect to positivity and TVD, implicitness of the method seems to have little added value, in clear contrast with stability. For example, the implicit trapezoidal rule

$$w_{n+1} = w_n + \frac{\tau}{2}\left(F(t_n, w_n) + F(t_{n+1}, w_{n+1})\right), \tag{51}$$

has a time step restriction in order to be TVD, when applied to time integration of $u_t + a \cdot u_x = 0$, whereby the advection term is discretized using limiters. Then, as can be found in [7, 20], the TVD property holds if the Courant number $\nu = \tau a/h \leq 1$. Comparing this with the Euler Backward method, which is unconditionally TVD, we see that implicitness has little added value in this case.

4.4 Concluding Remarks on Stability, Positivity and TVD

In order to integrate the species equations we need a method that can handle stiff problems and that is unconditionally positive or has the TVD property. In this section the following notions passed the revue:

- Explicit,
- Implicit,
- Stiff,
- Positivity,
- At Most First order Accurate,
- Higher Order Integration Methods,
- TVD,

- IMEX.

A time integration method that would be suitable to integrate the species equations needs to have a few of the above mentioned properties. Combinations of the properties that are possible are :

1. Positivity, Implicit, At Most First order Accurate and Stiff,
2. TVD, Implicit, At Most First order Accurate and Stiff,
3. IMEX, Higher Order Integration Methods, Positivity or TVD, Stiff,
4. Explicit, Higher Order Integration Methods, TVD,
5. Implicit, TVD, Higher Order Integration Methods, *Adding explicit stages.*

The last two combinations of properties, i.e., 4. and 5., are not suitable for solving the stiff species equations, because they contain explicit stages. Integrating the reactions terms explicitly will give a strong restriction on the time step. The latter is not preferred in practice.

5 Suitable Methods to Integrate

In this section we present integration methods that are suitable, from a theoretical point of view, for the time integration of the species equations. At the end of this section we will also mention nonlinear - and linear solvers.

From the previous sections it became clear that the Euler Backward method is a suitable method to perform time integration. It has the advantages of being unconditionally stable and positive. Disadvantages are the first order consistency and the probably high computational costs for one time step. The latter is due to the fact that the succeeding approximations are computed in a fully implicit manner.

5.1 Time Integration Methods

We will discuss a selection of time integration methods that have good properties in both stability and positivity, or TVD.

Rosenbrock Methods

Definition 5.1. *An s-stage Rosenbrock method is defined as*

$$\begin{aligned} k_i &= \tau F(w_n + \sum_{j=1}^{i-1} \alpha_{ij} k_j) + \tau \mathbf{A} \sum_{j=1}^{i} \gamma_{ij} k_j, \\ w_{n+1} &= w_n + \sum_{i=1}^{s} b_i k_i \end{aligned} \tag{52}$$

where $\mathbf{A} = \mathbf{A}_n$ *is the Jacobian* $F'(w(t))$.

Definition 5.1 is taken from [7]. The coefficients b_{ij}, α_{ij} and γ_{ij} define a particular method and are selected to obtain a desired level of consistency and stability.

Remark that computing an approximation w_{n+1} from w_n, in each stage i a linear system of algebraic equations with the matrix $(\mathbf{I} - \gamma_{ii}\tau\mathbf{A})$ has to be solved. To save computing time for large dimension systems $w'(t) = F(w(t))$ the coefficients γ_{ii} are taken constant, e.g., $\gamma_{ii} = \gamma$. Then, in every time-step the matrix $(\mathbf{I} - \gamma_{ii}\mathbf{A})$ is the same. To solve these large systems LU decomposition or (preconditioned) iterative methods could be used.

Define

$$\beta_{ij} = \alpha_{ij} + \gamma_{ij}, \quad c_i = \sum_{j=1}^{i-1} \alpha_{ij} \quad \text{and} \quad d_i = \sum_{j=1}^{i-1} \beta_{ij}.$$

In Table 5.1, taken from [7], the order conditions for $s \leq 4$ and $p \leq 3$ and $\gamma_{ii} = \gamma =$ constant can be found. In particular the two stage method

order p	order conditions
1	$b_1 + b_2 + b_3 + b_4 = 1$
2	$b_1 d_2 + b_3 d_3 + b_4 d_4 = \frac{1}{2} - \gamma$
3	$b_2 c_2^2 + b_3 c_3^2 + b_4 d_4^2 = \frac{1}{3}$
	$b_3 \beta_{32} d_2 + b_4(\beta_{42} d_2 + \beta_{43} d_3) = \frac{1}{6} - \gamma + \gamma^2$

Table 6: Order conditions of Rosenbrock methods with $\gamma_{ii} = \gamma$ for $s \leq 4$ and $p \leq 3$.

$$\begin{aligned} w_{n+1} &= w_n + b_1 k_1 + b_2 k_2 \\ k_1 &= \tau F(w_n) + \gamma\tau\mathbf{A}k_1 \\ k_2 &= \tau F(w_n + \alpha_{21} k_1) + \gamma_{21}\tau\mathbf{A}k_1 + \gamma\tau\mathbf{A}k_2, \end{aligned} \tag{53}$$

with coefficients

$$b_1 = 1 - b_2, \quad \alpha_{21} = \frac{1}{2b_2} \quad \text{and} \quad \gamma_{21} = -\frac{\gamma}{b_2},$$

is interesting. This method is of order two for arbitrary γ as long as $b_2 \neq 0$. The stability function is given as

$$R(z) = \frac{1 + (1 - 2\gamma)z + (\gamma^2 - 2\gamma + \frac{1}{2})z^2}{(1 - \gamma z)^2}. \tag{54}$$

The method is A-stable for $\gamma \geq \frac{1}{4}$ and L-stable if $\gamma = 1 \pm \frac{1}{2}\sqrt{2}$. By selecting for γ the larger value $\gamma_+ = 1 + \frac{1}{2}\sqrt{2}$, we have the property that $R(z) \geq 0$, for all negative real z. For diffusion-reaction problems, which have negative real eigenvalues, this property ensures positivity of the solution. In the case that advection is added to the system, the matrix has eigenvalues with negative real parts and relatively small imaginary parts. Then, the

positivity property is no longer guaranteed. It appears that the second order Rosenbrock method performs quite well with respect to the positivity property, as has been experienced in [17]. In [17] it is conjectured that the property that $R(z) \geq 0$ for all negative real z plays a role. We conclude this section with a remark on the implementation of the second order Rosenbrock scheme (56). In our code it is implemented with the parameters $b_1 = b_2 = \frac{1}{2}$ and $\gamma = 1 + \frac{1}{2}\sqrt{2}$. The matrix-vector multiplication in the second stage of (56) can be avoided by introducing

$$\tilde{k}_1 = k_1 \quad \text{and} \quad \tilde{k}_2 = k_2 - k_1, \tag{55}$$

and implementing the scheme

$$\begin{aligned} w_{n+1} &= w_n + \frac{3}{2}\tilde{k}_1 + \frac{1}{2}\tilde{k}_2, \\ \tilde{k}_1 &= \tau F(w_n) + \gamma\tau \mathbf{A}\tilde{k}_1, \\ \tilde{k}_2 &= \tau F(w_n + \tilde{k}_1) - 2\tilde{k}_1 + \gamma\tau \mathbf{A}\tilde{k}_2. \end{aligned} \tag{56}$$

Backward Differentiation Formulas (BDF)

The Backward Differentiation Formulas, shorter BDF, belong to the class of linear multistep methods, as defined in Definition 5.2

Definition 5.2. *The linear k-step method is defined by the formula*

$$\sum_{j=0}^{k} \alpha_j w_{n+j} = \tau \sum_{j=0}^{k} \beta_j F(t_{n+j}, w_{n+j}), \quad n = 0, 1, \ldots, \tag{57}$$

which uses the k past values $w_n, \ldots, w_{n+k-1}$ to compute w_{n+k}. Remark that the most advanced level is t_{n+k} instead of t_{n+1}.

The method is explicit when $\beta_k = 0$ and implicit otherwise. Furthermore, we will assume that $\alpha_k > 0$.

The BDF are implicit and defined as

$$\beta_k \neq 0 \quad \text{and} \quad \beta_j = 0 \quad \text{for} \quad j = 0, \ldots, k-1.$$

The coefficients α_i, see Definition 5.2, are chosen such that the order is optimal, namely k. [5] The 1-step BDF method is Backward Euler. The 2-step method is

$$\frac{3}{2}w_{n+2} - 2w_{n+1} + \frac{1}{2}w_n = \tau F(t_{n+2}, w_{n+2}), \tag{58}$$

[5] A linear k-step method is of order p if and only if

$$\sum_{i=0}^{k} \alpha_i = 0 \quad \sum_{i=0}^{k} \alpha_i i^j = 0 = j \sum_{i=0}^{k} \beta_i i^{j-1} \quad \text{for } j = 1, 2, \ldots, p.$$

For a proof we refer to [16].

and the three step BDF is given by

$$\frac{11}{6}w_{n+3} - 3w_{n+2} + \frac{3}{2}w_{n+1} - \frac{1}{3}w_n = \tau F(t_{n+3}, w_{n+3}). \tag{59}$$

In chemistry applications the BDF methods belong to the most widely used methods to solve stiff chemical reaction equations, due to their favorable stability properties. The BDF-1 and BDF-2 methods are both A-stable. The 3,4,5 and 6 step BDF methods are A(α)-stable, with α given in Example 4.1. For $k > 6$ the BDF-methods are unstable, see [4, Theorem 3.4, page 329].

Remark 5.3. *A disadvantage of linear multi-step methods is that the first $k-1$ approximations cannot be computed with the linear k-step scheme. To compute the first $(k-1)$ approximations, one could use a*

1. *Runge-Kutta scheme, for example, Euler Backward, or,*
2. *use for the first step a linear 1-step method, for the second approximation a linear 2-step method, ... and for the $(k-1)^{\text{st}}$ approximation a linear $(k-1)$-step scheme.*

As for Runge-Kutta methods the requirement of positivity does place a severe step size restriction on (implicit) BDF methods (and also on implicit multistep methods). Under Condition 4.6 and 4.7 we obtain positivity of $w'(t) = F(t, w(t))$ whenever $\alpha\tau \leq \gamma_R$. The parameter γ_R is the largest γ such that the stability function $R(z)$ is absolutely monotonic on $[-\gamma, 0]$. It is easy to check that for the BDF-2 method we have $\gamma_R = \frac{1}{2}$. Thus, positivity for BDF-2 is ensured whenever

$$\alpha\tau \leq \frac{1}{2}, \tag{60}$$

provided that w_1 is computed from w_0 by a suitable starting procedure. By suitable we mean that w_1 has to be computed from a method that ensures w_1 to be positive, such as for example BDF-1 (which is Euler Backward). Implementation of the methods is straightforward.

IMEX Runge-Kutta Chebyshev Methods

The IMEX extension of the class of Runge-Kutta Chebyshev methods is developed by Verwer et. al. [18, 19]. The Runge-Kutta Chebyshev methods belong to the class of explicit Runge-Kutta Chebyshev methods. They posses an extended real stability interval with a length proportional to s^2, with s the number of stages.

Definition 5.4. *The stability boundary $\beta(s)$ is the number $\beta(s)$ such that $[-\beta(s), 0]$ is the largest segment of the negative real axis contained in the stability region*

$$\mathcal{S} = \{z \in \mathbb{C} : |R(z)| \leq 1\}.$$

To construct the family of RKC methods we start with the explicit Runge-Kutta methods which have the stability polynomial

$$R(z) = \gamma_0 + \gamma_1 z + \cdots + \gamma_s z^s. \tag{61}$$

In order to have first order consistency we take $\gamma_0 = \gamma_1 = 1$.[6]

It can be proven that every explicit Runge-Kutta method has as optimal stability boundary $\beta(s) = 2s^2$. Thus the maximum value of $\beta(s)$ for explicit Runge-Kutta methods is $2s^2$. The upper boundary of $\beta(s)$ is achieved if we take the shifted Chebyshev polynomials of the first kind as stability polynomial. For a proof of both results, we refer to [7, 16].

In this paper we will not go into further details on the construction of the second order scheme. We fulfill with presenting it. The second order RKC is given as

$$\begin{aligned}
w_{n0} &= w_n, \\
w_{n1} &= w_n + \tilde{\mu}_1 \tau F(t_n + c_0\tau, w_{n0}), \\
w_{nj} &= (1 - \mu_j - \nu_j) w_n + \mu_j w_{n,j-1} + \nu_j w_{n,j-2} + \quad j = 1, \ldots, s \\
&\quad + \tilde{\mu}_1 \tau F(t_n + c_{j-1}\tau, w_{n,j-1}) + \tilde{\gamma}_j \tau F(t_n + c_0\tau, w_{n0}), \\
w_{n+1} &= w_{ns},
\end{aligned} \tag{63}$$

with coefficients

$$\omega_0 = 1 + \frac{\varepsilon}{s^2}, \quad \omega_1 = \frac{T_s'(\omega_0)}{T_s''(\omega_0)}, \tag{64}$$

$$b_j = \frac{T_j''(\omega_0)}{(T_j'(\omega_0))^2}, \quad c_j = \frac{T_s'(\omega_0)}{T_s''(\omega_0)} \frac{T_j''(\omega_0)}{T_j'(\omega_0)} \approx \frac{j^2 - 1}{s^2 - 1}, \tag{65}$$

$$\tilde{\mu}_1 = b_1 \omega_1, \quad \mu_j = \frac{2 b_j \omega_0}{b_{j-1}}, \quad \nu_j = -\frac{b_j}{b_{j-2}}, \tag{66}$$

$$\tilde{\mu}_j = \frac{2 b_j \omega_1}{b_{j-1}}, \quad \tilde{\gamma}_j = -a_{j-1} \tilde{\mu}_j. \tag{67}$$

The stability function of this method is given as

$$B_s(z) = 1 + \frac{T_s''(\omega_0)}{(T_s'(\omega_0))^2} \left(T_s(\omega_0 + \omega_1 z) - T_s(\omega_0) \right) \tag{68}$$

with

$$\omega_0 = 1 + \frac{\varepsilon}{s^2} \quad \omega_1 = \frac{T_s'(\omega_0)}{T_s''(\omega_0)}.$$

[6] This can be verified by considering the test equation $y' = \lambda y$. The local error of the test equation satisfies

$$\frac{e^z - R(z)}{\tau} = \mathcal{O}(\tau^p). \tag{62}$$

To achieve p^{th} order consistency the coefficients γ_i has to be chosen in such a way that (62) satisfies for p.

The parameter ε is a damping parameter to create a wider stability region. Compare in Figure 4 the undamped and damped case. For practical applications one would always use the damped stability function. The stability boundary is in the damped case equal to

$$\beta(s) = \frac{2}{3}(s^2 - 1)(1 - \frac{2}{15}\varepsilon),$$

which is about 80% of the optimal value.

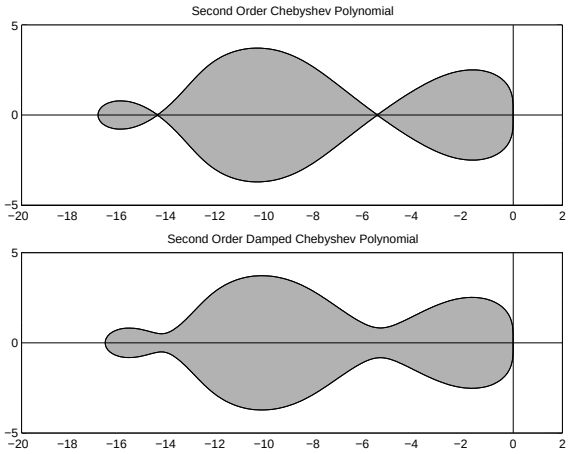


Figure 4: Stability region of $B_5(z)$ without damping (upper) and with damping (lower).

The IMEX extension of the above scheme is as follows. Suppose we have an ODE system $w'(t) = F(t, w(t))$, where $F(t, w(t))$ can be split as

$$F(t, w(t)) = F_E(t, w(t)) + F_I(t, w(t)). \tag{69}$$

In Eqn. (69) the term $F_I(t, w(t))$ is the part of F which is (supposed to be) too stiff to be integrated by an explicit Runge-Kutta Chebyshev method. Obviously, the term $F_E(t, w(t))$ is the moderate stiff part of F that can be integrated in an explicit manner using RKC methods.

The first stage of (63) becomes in the IMEX-RKC scheme

$$w_{n1} = w_n + \tilde{\mu}_1 \tau F_E(t_n + c_0\tau, w_{n0}) + \tilde{\mu}_1 \tau F_I(t_n + c_1\tau, w_{n1}), \tag{70}$$

with $\tilde{\mu}_1$ as defined before. Note that the highly stiff part of F is treated implicitly. The other $(s-1)$ subsequent stages of (63) will be modified in a similar way.

With respect to stability of this IMEX extension of (63) we remark the following. The implicit part is unconditionally stable, whereas the stability condition for the explicit part remains unchanged. Another pleasant property is that steady states are returned exactly. It takes an elementary calculation to prove this. Note that with operator splitting, or in this case time splitting, where the subsystems $w'(t) = F_E(t, w(t))$ and $w'(t) = F_I(t, w(t))$ are integrated completely decoupled within the time-steps, steady states are not returned exactly.

We conclude with some remarks on the implementation and use of the IMEX-RKC schemes in practice. We use a variable time step controller as is normal for Method of Lines solvers. Then, it would be desirable that for relatively small τ_n the code automatically switches to a lower number of stages s. For relatively large τ_n the same is desired, of course. This is only important for the integration of advection diffusion part of the right hand side of (13).

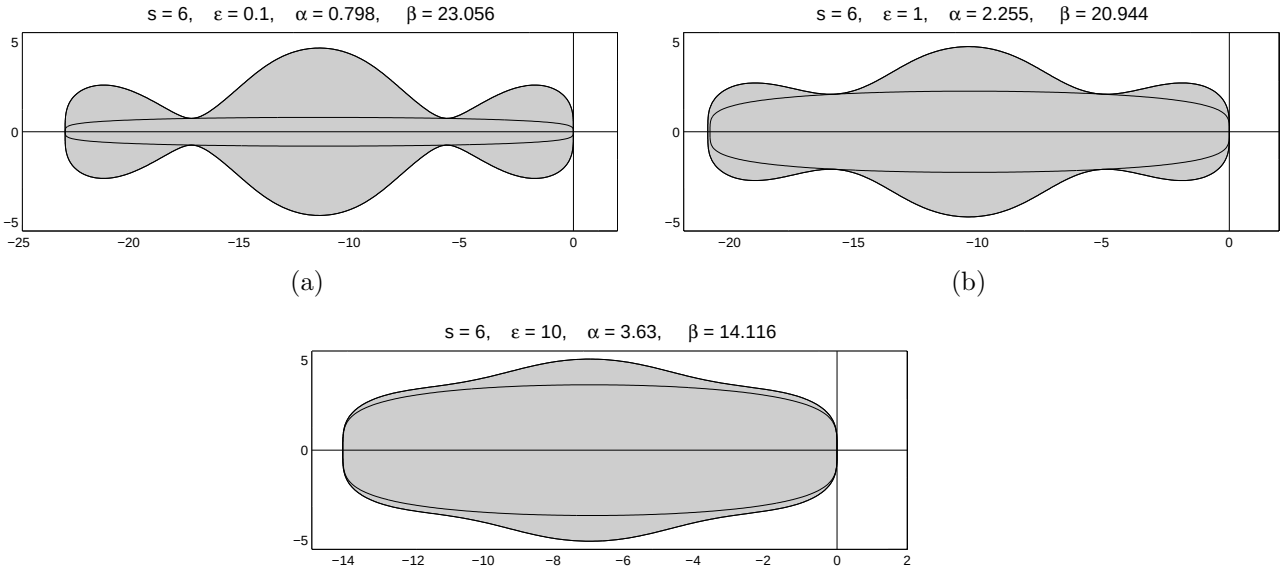
We will briefly describe the idea behind this variable number of stages procedure, which is taken from [19]. In [20] time step size conditions are given for the standard spatial discretizations of the m-dimensional scalar model

$$u_t + \sum_{k=1}^{m} a_k u_{x_k} = d \sum_{k=1}^{m} u_{x_k x_k}, \tag{71}$$

guaranteeing eigenvalues emerging from von Neumann stability analysis to lie inside geometric figures like squares, ellipses, half ellipses and ovals. For the explicit integration of advection and diffusion via the RKC method one has to fit an appropriate figure inside the stability region $\mathcal{S}$. In [19] ovals are selected. In Figure 12 stability regions with inscribed ovals

$$\left(\frac{x}{\beta/2} + 1\right)^2 + \left(\frac{y}{\alpha}\right)^4 = 1, \tag{72}$$

for $s = 6$ and $\varepsilon = 0.1, 1, 10$. Recall that epsilon is a damping factor, see (64). For β the value $\beta(s)$ has been taken, while α has been derived numerically. Assume that for diffusion



(a) (b) (c)

Figure 5: Stability regions $\mathcal{S}$ and inscribed ovals.

second order central discretization, while for advection the κ-scheme

$$\frac{dw}{dt} = \frac{a}{4h}((1-\kappa)w_{j-2} + (3\kappa-5)w_{j-1} + (3-3\kappa)w_j + (1+\kappa)w_{j+1}), \tag{73}$$

is used. The κ-scheme is the second order central scheme for $\kappa = 1$, the second order upwind scheme for $\kappa = -1$ and third order upwind-biased scheme for $\kappa = \frac{1}{3}$. The diffusion step size condition is then the familiar condition

$$\tau \leq \frac{\beta(s)}{2d\sum_{k=1}^{m} h_k^{-2}(2+(1-\kappa)P_k)}, \tag{74}$$

where $P_k = \frac{|a_k|h_k}{d}$ is a mesh Péclet number. Violation of (74) will give instability. Also taken from [20] is the advection time step restriction

$$\tau \leq q_1 \left(\frac{4d\alpha^4(s)}{\beta(s)}\right)^{\frac{1}{3}} / \sum_{k=1}^{m} \left(\frac{a_k^4}{h_k^2}\right)^{\frac{1}{3}}, \tag{75}$$

where the parameter q_1 depends on the choice of κ, see Table 5.1. For the actual imple-

κ	q_1
$\frac{1}{3}$	0.635
1	1
-1	0.323

Table 7: Values for q_1 for popular κ values

mentation the following lower bound for the ratio $\frac{\alpha^4(s)}{\beta(s)}$ has been used

$$\frac{\alpha^4(s)}{\beta(s)} = \begin{cases} 2 & s = 2, \\ 4(6-s) + 6.15(s-4) & s = 4, 6, \\ 6.15\frac{(10-s)}{2} + 15.5\frac{(s-6)}{4} & s = 8, \\ 15.5 & s = 10, 12, \ldots \end{cases} \tag{76}$$

As maximum of the ratio $\frac{\alpha^4(s)}{\beta(s)}$ the value 15.5 has been taken, which corresponds with $s = 10$. For larger values of s the slope in the curve, by which we mean the upper half of the oval (72), becomes too small.

Next, we will describe how to choose the number of stages s. Suppose a step size τ^* has been obtained by a local error estimator. Then, it can be checked whether τ^* satisfies the inequalities (74) and (75). If necessary, τ^* can be adjusted such that these inequalities hold. Simultaneously, the number of stages can be adjusted such that the number of stages needed to satisfy (74) is larger or equal than the number of stages needed to satisfy (75).

We remark that it makes no sense to spend more stages on advection than required for diffusion. Introduce the parameters

$$\Psi_1 = \frac{1}{2d\sum_{k=1}^{m} h_k^{-2}(2+(1-\kappa)P_k)}, \quad \text{and,} \quad \Psi_2 = \frac{4dq_1^3}{\left(\sum_{k=1}^{m}\left(\frac{a_k^4}{h_k^2}\right)^{\frac{1}{3}}\right)^3}. \tag{77}$$

The actual algorithm that is used in the code is given in Algorithm 1. For more background

Algorithm 1 Step size and Number of Stages Selection

1: If $\tau^* \leq 2\Psi_1$, then $s = 2, \tau = \min\{\tau^*, (2\Psi)^{1/3}\}$ and stop,
2: Put $\tau = \min\{\tau^*, (15.5\Psi)^{1/3}\}$. If $\tau^* \leq 2\Psi_1$, then $s = 2$ and stop,
3: Determine $s_d \geq 4$ such that $\tau \leq \beta(s_d)\Psi_1$ to satisfy (74),
4: Determine $s_a \geq 4$ such that $\tau \leq ((\alpha^4(s_a)/\beta(s_a))\Psi_2)^{1/3}$ to satisfy (75),
5: If $s_a \leq s_d$, then $s = s_d$ and stop. Otherwise $\tau = 0.8\tau$ and go to 3.

information we refer to [16, 18, 19].

5.2 Nonlinear Solvers

It is obvious that with the second order convergence the Newton iteration is a suitable choice as nonlinear solver. The disadvantage of having local convergence will disappear if one uses a line-search algorithm within the Newton solver. In order to have a decreasing sequence $\|F(x_n)\|$, we will adjust the Newton step $d = -F'(x_n)^{-1}F(x_n)$as follows. Find the smallest integer m such that

$$\|F(x_n + 2^{-m}d)\| \leq (1 - \varpi 2^{-m})\|F(x_n)\|, \tag{78}$$

and let the Newton-step be $s = 2^{-m}d$. Condition (78) is called the sufficient decrease of $\|F\|$. The parameter ϖ in (78) is a small number, chosen such that the sufficient decrease condition is satisfied as easy as possible. In [8, 16] ϖ is taken equal to 10^{-4}.

5.3 Linear Solvers

Because the nonlinear solver is of the Newton type, in each Newton iteration linear systems have to be solved. In CVD literature one uses iterative linear solvers for both 2D and 3D problems. We share the idea that for 3D problems iterative linear solvers are definitely needed. For 2D problems we think that direct solvers are still applicable.

2D Problems

In most 2D applications direct solvers like LU factorization are still applicable to solve linear systems. To reduce the amount of work one usually reorders the unknowns (and the

equations), in order to reduce the bandwidth of the matrix. Also in our case it is possible to reduce the bandwidth of the Jacobian considerably.

Using a finite volume discretization of the right hand side of (6), where the unknowns are arranged per species and lexicographic in the grid, we get a Jacobian matrix with a structure as in Figure 6. We change the arrangement of unknowns into: unknown

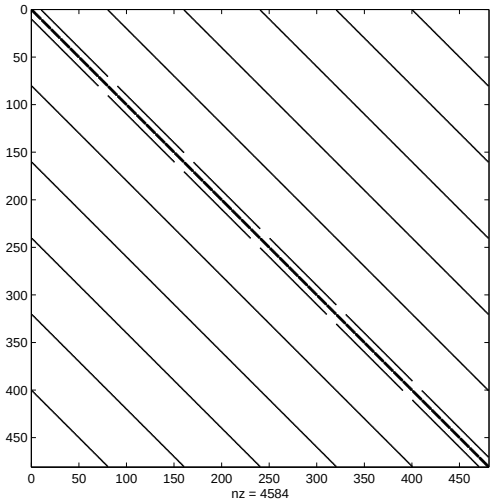


Figure 6: Jacobian matrix belonging to the test-problem, where 6 chemical species are considered. Number of grid-points in r direction is 10 and in z direction 8. This Jacobian matrix is with lexicographic arrangement per species. We see that the upper and lower bandwidth are both $(N-1)\cdot G$, where N is the number of species and G the number of grid points.

concentration species 1 in grid-point 1, unknown concentration species 2 in grid-point 1, ..., unknown concentration species N in grid-point 1, and so on. The second step of a successful re-arrangement is then to re-arrange the discrete ODEs as follows. Firstly, the discrete balance for species 1 in grid-point 1 is taken, then the discrete balance for species 2 in grid-point 1 is taken, and so on. This reordering results in a Jacobian matrix with a structure as given in Figure 7.

3D Problems

For 3D problems direct solvers like the LU factorization are no longer applicable. To approximate the solution linear systems one has roughly speaking two options, e.g. Krylov subspace methods and multigrid, or a combination.. For this application, simulating reacting gas flow simulations, the latter choice is the more obvious one.

In the 2D case with a gas phase chemistry model with a large number of species the complexity of solving resulting linear systems will become equal to the complexity of solving linear systems of 3D discrete systems. In that case iterative linear solvers will become more and more attractive.

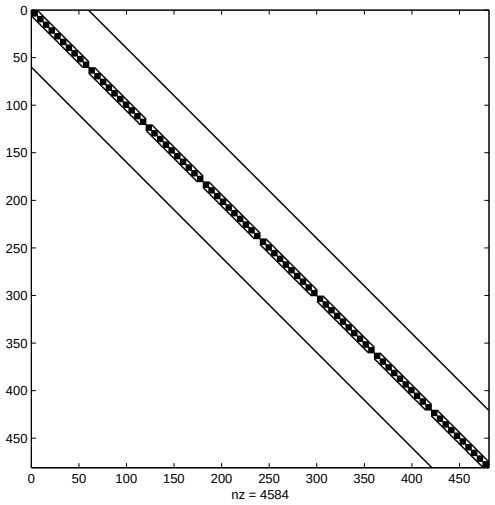


Figure 7: Jacobian matrix belonging to the test-problem, where 6 chemical species are considered. We have the same number of grid-points in r and z direction. This Jacobian matrix is constructed with the re-arrangement of the unknowns as described in Section 5.3. In this re-arranged case the bandwidth is equal to $N \cdot G_r$, where G_r is the number of grid points in the r direction.

6 Numerical Simulation

Before giving the results of the numerical simulations, where we mainly considered the efficiency of the time integration methods, we give details on the simulation itself. In this simulation a variable time step algorithm is used, which will also be explained in this section.

6.1 General Outline of the Simulation

We perform a transient simulation on the problem described in Section 2. The simulation stops when 'numerical' steady state is reached. By 'numerical' steady state we mean that for a certain index n yields,

$$\frac{\|y_{n+1} - y_n\|_2}{\|y_n\|_2} \leq \vartheta, \tag{79}$$

where y_n is the numerical solution of the semi-discretization

$$\frac{dw}{dt} = Aw + b + F(t, w(t)), \tag{80}$$

on time $t = t_n$, and ϑ a small parameter. In (80) A represents the discretized advection diffusion operator, b a vector of boundary conditions and $F(t, w(t))$ a nonlinear vector function representing the gas phase reactions. In our simulations for ϑ the value $\vartheta = \mathcal{O}(10^{-6})$ has been taken.

In order to describe the algorithm that is used for CVD simulation we introduce the function 'time_integration' as follows. As input we have y_n, the approximated solution of w on time $t = t_n$, $\tau = t_{n+1} - t_n$, m the molar mass field and f the vector of molar fractions. The output is y_{n+1}, i.e., the approximated solution of w on time t_{n+1}. Summarized

$$y_{n+1} = \text{time\_integration}(y_n, \tau, m, f_i). \tag{81}$$

The exact description of the function 'time_integration' depends on the choice of time integration method. See Section 5.

Based on the inlet concentrations of silane SiH_4 an initial mass fraction profile can be constructed. The outline of the algorithm to perform the CVD simulation is given as Algorithm 2. As mentioned earlier, for ϑ the value $\vartheta = 10^{-6}$ has been taken.

Algorithm 2 Simulation

1: Initial values for y_0 and f_i
2: **while** $\frac{\|y_1 - y_0\|}{\|y_0\|} \geq \vartheta$ **do**
3: Compute average molar mass of gas mixture m
4: $y_1 = \text{time\_integration}(y_0, \tau, m, f)$
5: Check whether $y_1 \geq 0$, if not then $\tau = \frac{1}{2}\tau$ and go to 4
6: Estimate the local (time integration) error D_n
7: **if** $D_n > Tol$ **then**
8: $\tau = r \cdot \tau$, r estimated such that $D_n \leq Tol$, and go to 4
9: **end if**
10: Compute mole fractions f
11: $y_0 = y_1$
12: **end while**

6.2 Variable Time Stepping

We briefly explain the variable time stepping algorithm as it is implemented in our code. Consider an attempted step from t_n to $t_{n+1} = t_n + \tau_n$ with time step size τ_n that is performed with an p^{th} order time integration method. Suppose an estimate D_n of order $\hat{p}$ of the norm of the local truncation error is available. Then, if $D_n < Tol$ this step τ_n is accepted, whereas if $D_n > Tol$ the step is rejected and redone with time step size $\frac{1}{2}\tau_n$. If $D_n < Tol$, then the new step size is computed as

$$\tau_{\text{new}} = r \cdot \tau, \quad r = \left(\frac{Tol}{D_n}\right)^{\frac{1}{\hat{p}+1}}. \tag{82}$$

It is also possible to put bounds on the growth factor r of the new step size. This is simply done by giving bounds on r.

6.2.1 Local Error Estimation for Euler Backward

The local error of the Euler Backward method

$$w_{n+1} = w_n + \tau F(t_{n+1,} w_{n+1}) \tag{83}$$

satisfies

$$\delta_n = -\frac{1}{2}\tau^2 w''(t_n) + \mathcal{O}(\tau^3). \tag{84}$$

The local truncation error δ_n can be estimated as

$$d_n = -\frac{1}{2}\left(w_{n+1} - w_n - \tau F(t_n, w_n)\right). \tag{85}$$

See, for example, [7].

6.2.2 Local Error Estimation for BDF2

For the BDF-2 scheme the local error estimation is as follows. Introduce the ratio $r = \frac{\tau_n}{\tau_{n-1}}$, where τ_n is defined as above, e.g., $\tau_n = t_{n+1} - t_n$. The second order BDF-2 scheme can be rewritten in the form with the ratio r as

$$w_{n+2} - \frac{(1+r^2)}{1+2r} w_{n+1} + \frac{r^2}{1+2r} w_n = \frac{1+r}{1+2r}\tau F(t_{n+2}, w_{n+2}). \tag{86}$$

In a similar way as in Section 6.2.1 we obtain the first order estimator

$$d_n = \frac{r}{1+r}\left(w_{n+1} - (1+r)w_n + r w_{n-1}\right). \tag{87}$$

A second order estimator, see [7], is

$$d_n = \frac{1+r}{1+2r}\left(w_{n+1} + (r^2-1)w_n - r^2 w_{n-1} - (1+r)\tau_n F(t_n, w_n)\right). \tag{88}$$

We remark that in the first time step, where BDF-1 is used, the local error is estimated by

$$d_0 = \frac{1}{2}\left(w_1 - w_0 - \tau_0 F(t_0, w_0)\right). \tag{89}$$

6.2.3 Local Error Estimation for IMEX-Runge-Kutta-Chebyshev

The local error estimation for the IMEX -Runge-Kutta-Chebyshev methods is the same as for the explicit Runge-Kutta-Chebyshev schemes, see [19]. The asymptotically correct estimate of the local error is

$$d_n = \frac{1}{15}\left[12(w_n - w_{n+1}) + 6\tau_n(F(t_n, w_n) + F(t_{n+1}, w_{n+1}))\right], \tag{90}$$

which is taken from [14].

6.3 Numerical Results for the Simplified CVD System

In this part we present the numerical results of the simplified test problem as defined in Section 2.1 to 2.3. In this section we distinguish two different simulations, namely, the simulation with physical initial conditions, see Section 6.3.1, and the simulation with a constant initial profile, see Section 6.3.2.

The experiments are done in FORTRAN. The computations are done on a serial Pentium 4 (2.8 GHz) computer with 1Gb memory capacity. Moreover, the code is compiled with FORTRAN g77 on LINUX.

6.3.1 Physical Initial Conditions

At $t = 0$ we start with the zero concentration profile for all species, except the carrier gas, and let the reactive specie silane SiH_4 enter the reactor at the inflow boundary. Then, we stop the simulation at steady state, which is reached when the relative change of the solution vector is less than 10^{-6}, i.e.,

$$\frac{\|u_{n+1} - u_n\|_2}{\|u_n\|_2} \leq 10^{-6}. \tag{91}$$

For a comparison between the workloads of the various TIM, we look to the amount of CPU time, the number of time steps and the total number of Newton iterations (if needed) it takes to reach steady state.

The solutions computed by different TIM also have been compared with a reference solution u_{ref}, which has been computed with high accuracy. It appeared that the solutions of the different TIM, denoted by u_{TIM}, all had the same quality, by which we mean that

$$\frac{\|u_{TIM} - u_{ref}\|_2}{\|u_{ref}\|_2} = \mathcal{O}(10^{-7}). \tag{92}$$

In Figure 8 the residual $\|F(w(t))\|_2$ versus the time step, for different TIM, is given. Recall that $F(w(t))$ is the semi-discretization resulting from the Method of Lines approach, see (13). In Figure 9 the time step size versus the time step are given. Recall that the time step sizes are computed using the time step controller of Section 6.2.

With respect to the IMEX Runge-Kutta-Chebyshev method we have the following remarks. Because the advection and diffusion part are integrated explicitly, there is always a stability condition on the time step. This stability condition makes the IMEX RKC method not suitable for computing a steady state solution. We implemented the IMEX RKC scheme with a variable step size controller and a variable stage controller as described in Section 5.1. The number of stages per time step is given in Figure 10, whereas the time step size and $\|F(w(t))\|_2$ are given in Figure 11.

We conclude with the contour plots of the steady state solution in Figure 12.

TIM	CPU time	# time steps	# Newton iterations
Euler Backward	1061 CPU sec	120	236
ROS2	579 CPU sec	190	0
BDF-2	689 CPU sec	99	182
IMEX-RKC	13141 CPU sec	1127	9075

Table 8: Workloads of various TIM for simulation with physical initial conditions.

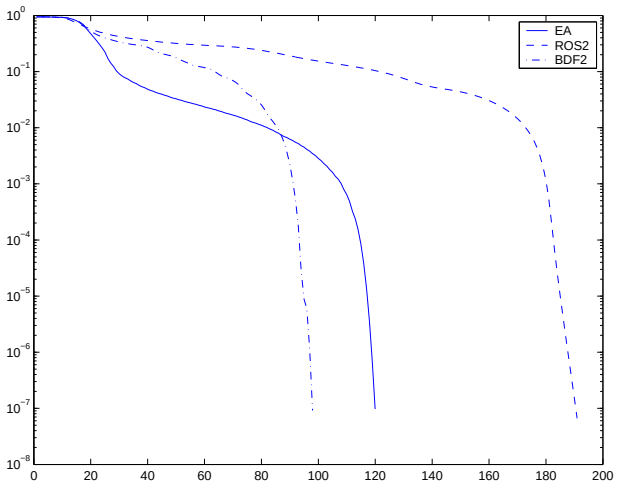


Figure 8: Residual $\|F(w(t))\|_2$ versus time step for simulation with physical initial conditions

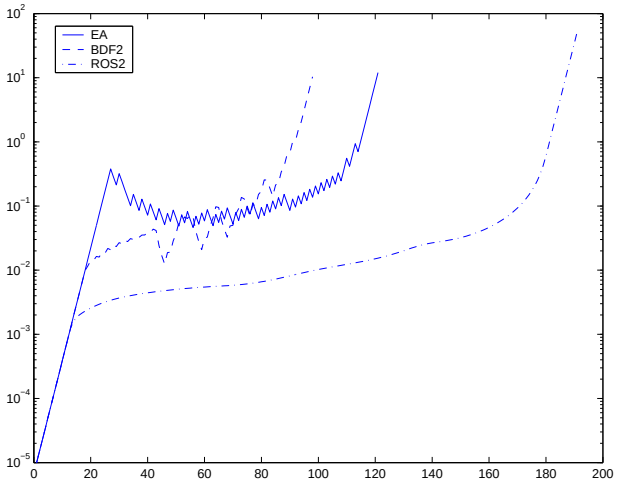


Figure 9: Time step size versus time step for simulation with physical initial conditions

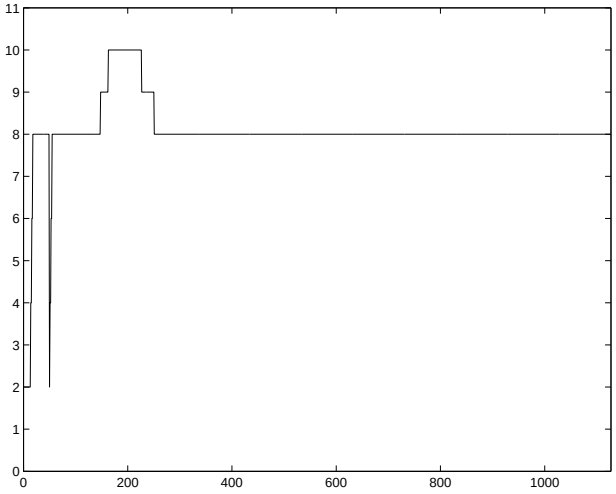


Figure 10: Number of stages of IMEX Runge Kutta Chebyshev scheme versus time step for simulation with physical initial conditions

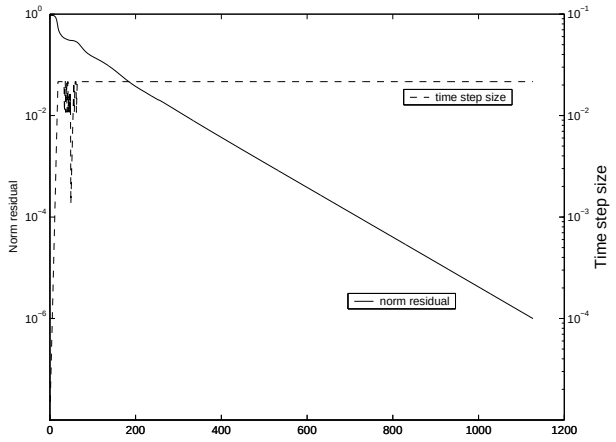


Figure 11: Residual $\|F(w(t))\|_2$ and time step size versus time step for the IMEX Runge-Kutta-Chebyshev scheme for simulation with physical initial conditions

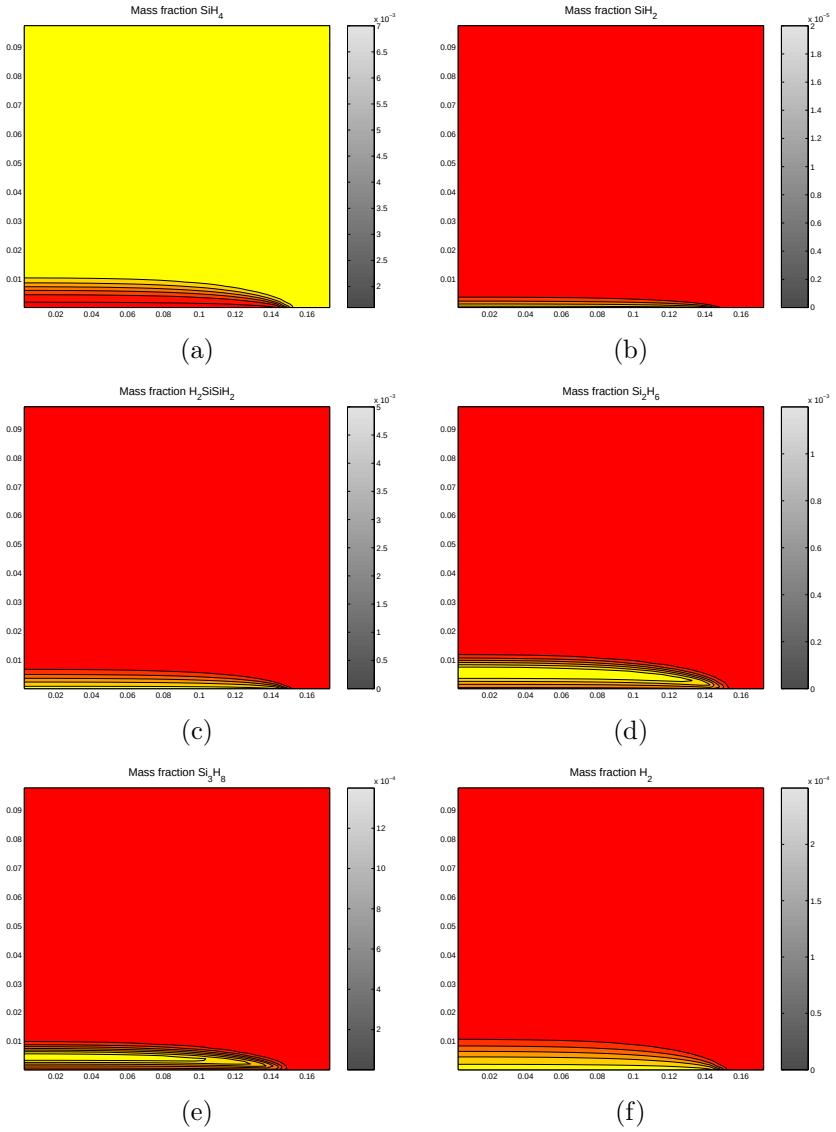


Figure 12: Contour plots of the steady state mass fractions of SiH_4 (a), SiH_2 (b), H_2SiSiH_2 (c), Si_2H_6 (d), Si_3H_8 (e) and H_2 (f). The outflow boundary is situated on $r = 15.0$ cm. to $r = 17.5$ cm.

6.3.2 Constant Initial Profile

In order to test the robustness of the code we perform a test proposed by Kleijn. This so called 'Constant Initial Profile' test starts at $t = 0$ with a constant initial profile consisting of silane SiH_4 and helium He, such that

$$f_{SiH_4} = 0.001 \quad \text{and} \quad f_{He} = 0.999, \tag{93}$$

on the whole computational domain. In Figures 13 and 14 and Table 9 results for the different time integration methods, except IMEX Runge-Kutta-Chebyshev, are given. The result for the IMEX RKC scheme are presented in Figure 15 and 16. The steady state solution that has been found is identical as the one found in the simulations of Section 6.3.1.

From Table 9 we see that with the constant initial profile as initial condition a smaller number of time steps is needed to converge to steady state. This observation is easily explained by the fact that for the constant initial profile no silane has to flow into the reactive zone. In this case the gas phase reactions will start immediately, such that towards a chemical equilibrium can be computed.

TIM	CPU time	# time steps	# Newton iterations
Euler Backward	502 CPU sec	53	119
ROS2	266 CPU sec	78	0
BDF-2	401 CPU sec	54	116
IMEX-RKC	8573 CPU sec	718	5797

Table 9: Workloads of various TIM for constant initial profile simulation.

7 Conclusions

Based on Table 8 and 9 can be concluded that for the simplified CVD system as introduced in Section 2, the second order Rosenbrock method is the most efficient time integration method in terms of CPU time. The difference between Rosenbrock, Euler Backward and BDF is minimal. We expect that for 2D systems with a larger number of species and reactions, these methods perform equally well, to compute a steady state solution.

The opposite is true for the IMEX RKC scheme. As mentioned before, this method is not suitable for simulation until steady state. We expect that in the case of 3D and a large number of reactive species this scheme will perform much better, under the assumption that for future work transient simulation is considered. This expectation depends highly on the fact that the nonlinear systems in the IMEX RKC scheme will become (relatively) cheaper to solve if the number of dimensions (and species) increases.

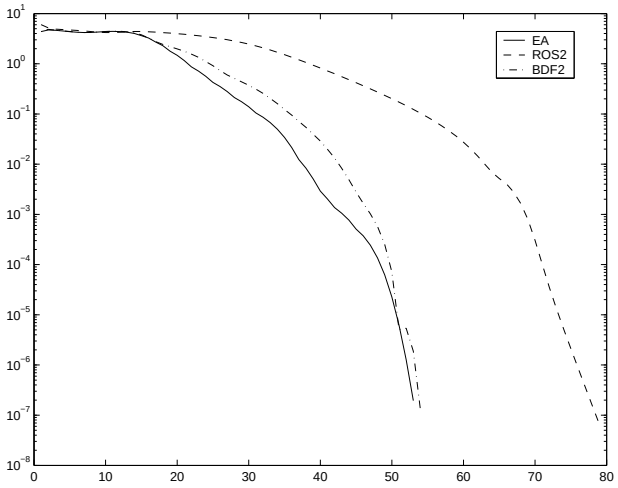


Figure 13: Residual $\|F(w(t))\|_2$ versus time step for constant initial profile simulation

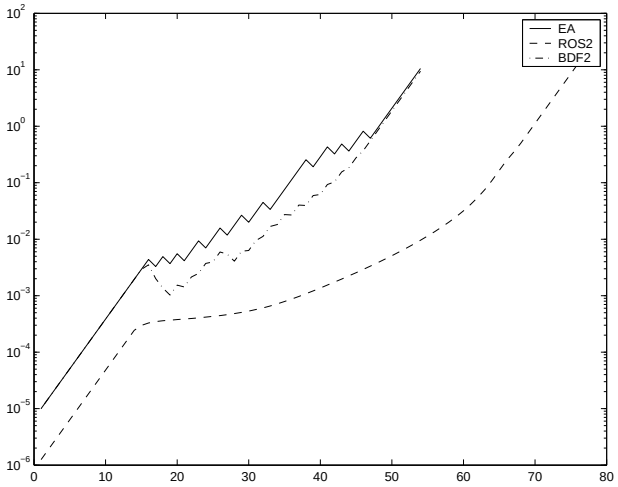


Figure 14: Time step size versus time step for constant initial profile simulation

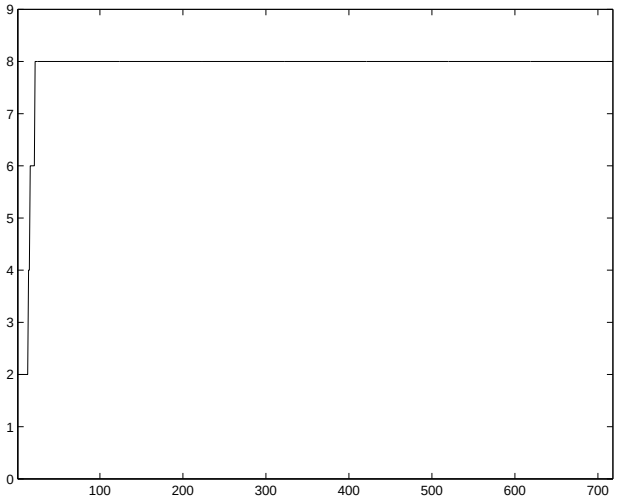


Figure 15: Number of stages of IMEX Runge Kutta Chebyshev scheme versus time step for constant initial profile simulation

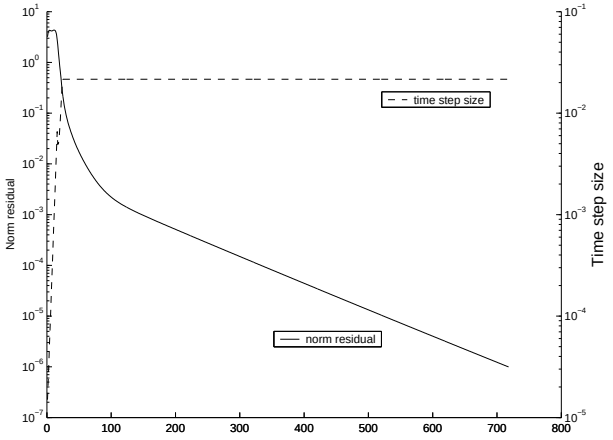


Figure 16: Residual $\|F(w(t))\|_2$ and time step size versus time step for the IMEX Runge-Kutta-Chebyshev scheme for constant initial profile simulation

References

[1] A. Berman and R.J. Plemmons, *Nonnegative Matrices in the Mathematical Sciences*, SIAM, Philadelphia, (1994)

[2] C. Bolley and M. Crouzeix, *Conservation de la Positivité Lors de la Discrétisation des Problèmes d'Évolution Paraboliques*, RAIRO Anal. Numer. 12, pp. 237-245, (1973)

[3] S. Gottlieb, C.-W. Shu and E. Tadmor, *Strong Stability-Preserving High-Order Time Discretization Methods*, SIAM Review 43, pp. 89-112, (2001)

[4] E. Hairer, S.P. Nørsett and G. Wanner, *Solving Ordinary Differential Equations I: Nonstiff Problems*, Springer Series in Computational Mathematics, 8, Springer, Berlin, (1987)

[5] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*, Second Edition, Springer Series in Computational Mathematics, 14, Springer, Berlin, (1996)

[6] R.A. Horn and C.R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, (1999)

[7] W. Hundsdorfer and J.G. Verwer, *Numerical Solution of Time-Dependent Advection-Diffusion-Reaction Equations*, Springer Series in Computational Mathematics, 33, Springer, Berlin, (2003)

[8] C.T. Kelley, *Solving Nonlinear Equations with Newton's Method*, Fundamentals of Algorithms, SIAM, Philadelphia, (2003)

[9] C.R. Kleijn, *Transport Phenomena in Chemical Vapor Deposition Reactors*, PhD thesis, Delft University of Technology, Delft, (1991)

[10] C.R. Kleijn, *Computational Modeling of Transport Phenomena and Detailed Chemistry in Chemical Vapor Deposition- A Benchmark Solution*, Thin Solid Films, 365, pp. 294-306, (2000)

[11] J.M. Ortega and W.C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Reprint of the 1970 original, Classics in Applied Mathematics 30, SIAM, Philadelphia, (2000)

[12] A. Sandu, *Positive Numerical Integration Methods for Chemical Kinetic Systems*, Technical Report at Michigan Technological University, CSTR-9905, (1999)

[13] C.-W. Shu and S. Osher, *Efficient Implementation of Essentially Non-Oscillatory Shock Capturing Schemes*, J. Comput. Phys. 77, pp. 439-471, (1988)

[14] B.P. Sommeijer, L.F. Shampine and J.G. Verwer, *RKC: An Explicit Solver for Parabolic PDEs*, J. Comput. Appl. Math. 88, pp. 315-326, (1997)

[15] M.N. Spijker, *Contractivity in the Numerical Solution of Initial Value Problems*, Numer. Math. 42, pp. 271-290, (1983)

[16] S. van Veldhuizen, *Efficient Solution Methods for Stiff Systems of Advection-Diffusion-Reaction Equations*, Literature Study, Technical Report at the Delft University of Technology, TWA-05-05, Delft, (2005)

[17] J.G. Verwer, E.J. Spee, J.G. Blom and W. Hundsdorfer, *A Second-Order Rosenbrock Method Applied to Photochemical Dipersion Problems*, SIAM Journal on Sci. Comp., 20, pp.1456-1480, (1999)

[18] J.G. Verwer and B.P. Sommeijer, *An Implicit-Explicit Runge-Kutta-Chebyshev Scheme for Diffusion-Reaction Equations*, SIAM Journal on Sci. Comp., 25, pp.1824-1835, (2004)

[19] J.G. Verwer, B.P. Sommeijer and W. Hundsdorfer, *RKC Time-Stepping for Advection-Diffusion-Reaction Problems*, Journal of Comp. Physics, 201, pp. 61-79, (2004)

[20] P. Wesseling, *Principles of Computational Fluid Dynamics*, Springer Series in Computational Mathematics, 29, Springer, Berlin, (2001)