Comparison of ODE Methods for Laminar Reacting Gas Flow Simulations

S. van Veldhuizen,[1] **C. Vuik,**[1] **C. R. Kleijn**[2]
[1] *Delft Institute of Applied Mathematics, Delft University of Technology, Mekelweg 4, 2628 CD Delft, The Netherlands*

[2] *Department of Multi Scale Physics, Delft University of Technology, Prins Bernardlaan 6, 2628 BW Delft, The Netherlands*

Received 5 June 2007; accepted 29 August 2007
Published online 20 November 2007 in Wiley InterScience (www.interscience.wiley.com).
DOI 10.1002/num.20305

Two-dimensional transient simulations are presented of the transport phenomena and multispecies, multireaction chemistry in chemical vapor deposition (CVD). The transient simulations are run until steady state, such that the steady state can be validated against the steady state solutions from literature. We compare various time integration methods in terms of efficiency and robustness. Besides stability, which is important due to the stiffness of the problem, preservation of non-negativity is crucial. It appears that this latter condition on a time integration method is much more restrictive toward the time step size than stability. © 2007 Wiley Periodicals, Inc. Numer Methods Partial Differential Eq 24: 1037–1054, 2008

Keywords: advection–diffusion–reaction equations; chemical vapor deposition; positivity; time integration

I. INTRODUCTION

Applications of thin solid films can be found in many technological areas, such as microelectronics, solar cells, optical, mechanical, and decorative coatings. Various technologies are used to produce these thin layers, such as, for instance sputtering and evaporation. Another technology that distinguishes itself by involving chemical reactions is chemical vapor deposition (CVD), in which the material to be deposited is introduced as a gas into a reactor chamber and is deposited on a solid surface via several chemical reactions.

Numerical flow simulations are widely used for the design of CVD reactor chambers and for the optimization of CVD processes [1]. Mostly, the simulations are performed in steady state. However, there is an interest in studying transients, e.g. during start up and shut down, and in inherently transient processes such as atomic layer deposition (ALD) [2].

In this article, nonstationary simulations of the CVD process are presented, where we mainly focus on solving the species equations, which describe species mass transport due to advection

Correspondence to: S. van Veldhuizen, Delft Institute of Applied Mathematics, Delft University of Technology, Mekelweg 4, 2628 CD Delft, The Netherlands (e-mail: s.vanveldhuizen@tudelft.nl)

and diffusion, and their conversion due to chemical reactions. Since the time scales of advection, diffusion, and chemical reactions often differ orders of magnitude, the system of species equations can be extremely stiff. Therefore, to stably integrate the species equations in time, a suitable time integration method has to be found. Furthermore, we want that numerical integration of the species equations conserves the non-negativity of the species concentrations. It appears that this so-called positivity property puts a severe restriction on time integration methods, and consequently also on the nonlinear and linear solvers.

In [3], we already discussed some results on the transient numerical solution of a CVD problem, which, in steady state form, was proposed as a benchmark problem by Kleijn [4]. In this article, we add the Soret effect (thermal diffusion), discuss other strategies to solve the stiff system of species equations, and compare the quality of the solutions obtained by the different strategies. As in [3], it is assumed that the velocities, temperatures, pressures, and densities are known in the computational domain. Our work in [3] and this article, differ from work in literature on (laminar) reacting flows, in which, as a result of computational costs and stiffness, the simulations are limited to steady state and mostly 2D (see [1]). However, in [5], transient simulations are presented for a CVD model consisting of six species and four reactions. In [5], the authors started their transient simulation from a pseudo steady state solution until it is completely stabilized. In [6], transient results for the temperature distribution in a CVD reactor are presented.

The purpose of this article is to examine the transient behavior of the 2D Benchmark problem proposed by Kleijn [4] and to develop robust and efficient numerical methods to solve the stiff system of species equations. In particular, we focus on properties of time integration methods in combination with (non)linear solvers. In the 2D Benchmark problem, a reactive gas, diluted in an inert carrier gas, reacts through a chain of 26 gas phase reactions to create a mixture of 16 different gas species. These may react at the surface to form a solid film. Unlike the transient simulations in [5], the transient simulations in this article start from the moment the reactive gases start entering the reactor and run until steady state has been reached. In Section V, we give details of this CVD process.

II. MATHEMATICAL MODEL FOR CVD

To mathematically model a CVD process, the gas flow, the transport of energy, the transport of species, and the chemical reactions in the reactor have to be described. We assume that the gas mixture in the reactor behaves as a continuum, as an ideal gas and in accordance with Newton's law of viscosity. The gas flow in the reactor is assumed to be laminar.

The composition of the N component gas mixture is described in terms of the dimensionless mass fractions ω_i, which sum up to one. Transport of total mass, momentum, and heat are described, respectively, by the continuity equation, the Navier–Stokes equations, and the transport equation for thermal energy. Note that the consumption and production of heat due to the chemical reactions is also included in the energy equation. For most CVD systems, especially when the reactant are highly diluted, the heat of reactions has a negligible influence on the gas temperature distribution. For such systems, the computation of the laminar flow and the temperature field is a relatively trivial task. The difficulty lies in solving the set of highly nonlinear and strongly coupled species equations, which is the topic of this article.

Transport of mass fraction ω_i is described by the species equation

$$\frac{\partial(\rho\omega_i)}{\partial t} = -\nabla\cdot(\rho\mathbf{v}\omega_i) + \nabla\cdot\left[(\rho\mathbb{D}_i\nabla\omega_i) + \left(\mathbb{D}_i^T\nabla(\ln T)\right)\right] + m_i\sum_{k=1}^{K}\nu_{ik}R_k^g, \qquad (2.1)$$

where diffusive mass fluxes are due to concentration diffusion and thermal diffusion. In (2.1), $\mathbb{D}_i$ is the effective diffusion coefficient, $\mathbb{D}_i^T$ is the multicomponent thermal diffusion coefficient for species i [4], $\mathbf{v}$ the mass-averaged velocity obtained from the Navier–Stokes equations, and ρ the density of the gas mixture. Under the assumption that K reversible gas-phase reactions of the form

$$\sum_{i=1}^{N} \nu'_{ik} \mathcal{A}_i \underset{k^g_{k,\text{backward}}}{\overset{k^g_{k,\text{forward}}}{\rightleftarrows}} \sum_{i=1}^{N} \nu''_{ik} \mathcal{A}_i \tag{2.2}$$

take place, the net molar reaction rate R_k^g for the kth reaction, see the last term on the right hand side of (2.1), is defined as

$$R_k^g = k^g_{k,\text{forward}} \prod_{i=1}^{N} \left(\frac{P \omega_i m}{RT m_i} \right)^{\nu'_{ik}} - k^g_{k,\text{backward}} \prod_{i=1}^{N} \left(\frac{P \omega_i m}{RT m_i} \right)^{\nu''_{ik}}. \tag{2.3}$$

In Eq. (2.2), $\mathcal{A}_i$ are the species in the gas mixture, ν'_{ik} the forward stoichiometric coefficient for species i in reaction k, and ν''_{ik} the backward stoichiometric coefficient for species i in reaction k. The net stoichiometric coefficient ν_{ik} is then defined as $\nu_{ik} = \nu''_{ik} - \nu'_{ik}$. In Eq. (2.3), P is the pressure in Pa, T the temperature, R the universal gas constant, m_i the molar mass of species i, and m the average molar mass computed as

$$m = \left(\sum_{i=1}^{N} \frac{\omega_i}{m_i} \right)^{-1}. \tag{2.4}$$

Usually, the forward reaction rate constant $k^g_{k,\text{forward}}$ is fitted according to a modified Arrhenius expression:

$$k^g_{k,\text{forward}}(T) = A_k T^{\beta_k} e^{\frac{-E_k}{RT}}, \tag{2.5}$$

where A_k, β_k, and E_k are fit parameters. For the CVD process considered in this article, these fit parameters are available through the references presented in Section V. The backward reaction rate constants $k^g_{k,\text{backward}}$ are computed self-consistently from the forward reaction rate constants and reaction thermochemistry (see [7]). The forward and backward rate constants of the fastest and slowest reactions can differ many (e.g. 25) orders of magnitude, introducing stiffness into the species Eq. (2.1). For a detailed description of the mathematical model for CVD and the corresponding boundary conditions, we refer to [4, 7].

III. POSITIVITY OF THE SPECIES EQUATIONS

In this article, we focus on properties of time integration of semidiscrete systems $w(t)' = F(t, w(t))$ arising from the spatial discretization of Eq. (2.1), where $w(t)$ is a vector containing all species mass fractions in all grid points. The stiff, nonlinear reaction terms in $F(t, w(t))$ have to be integrated implicitly due to stability requirements. As a result, the implicit integration of stiff reaction terms spends most of its computing time for the evaluation of the Jacobian and in the solution of linear equations. For 3D simulations, the linear algebra should be accelerated by, for instance, Krylov methods. For the 2D simulations in this article, direct solvers are still applicable (see Section IV).

An important property of the concentrations of chemical species in reacting systems is *positivity*. By positivity we mean preservation of non-negativity for the concentrations of all components. This property should also hold for the mathematical model of the process. It can be shown that the mathematical model of CVD as presented in Eqs. (2.1)–(2.3) preserves positivity. The next step is that neither spatial discretization nor time integration should introduce wiggles or negative components into the solution vector. We use the hybrid finite volume (FV) scheme from [7] to discretize in space, which conserves non-negativity. It appears that this extra condition on time integration methods, besides stability, is much more restrictive toward the time step than stability.

A. Positive Time Integration

Definition. *An ODE system $w'(t) = F(t, w(t))$, $t \geq 0$, is called positive, or non-negativity preserving, if $w(0) \geq 0$ (component-wise) $\Longrightarrow w(t) \geq 0$, for all $t > 0$.*

The next theorem provides a simple criterion on $F(t, w(t))$ to test whether the system $w'(t) = F(t, w(t))$, $t \geq 0$, is positive. For a proof, we refer to [8].

Theorem 3.1. *Suppose that $F(t, w)$ is continuous and satisfies a Lipschitz condition with respect to w. Then the system $w'(t) = F(t, w(t))$, $t \geq 0$, is positive if and only if for any vector $w \in \mathbb{R}^m$ and all $i = 1, \ldots, m$, and $t \geq 0$ yields*

$$w \geq 0 \quad \text{(componentwise)}, \quad w_i = 0 \Longrightarrow F_i(t, w) \geq 0 \tag{3.1}$$

It is interesting to investigate positivity for semidiscrete systems. Consider, for instance, the one-dimensional linear advection–diffusion equation

$$\frac{\partial}{\partial t}u(x,t) + \frac{\partial}{\partial x}(a(x,t)u(x,t)) = \frac{\partial}{\partial x}\left(d(x,t)\frac{\partial}{\partial x}u(x,t)\right), \tag{3.2}$$

with periodic boundary conditions, and where $a(x,t)$ is the space and time-dependent advection coefficient, and $d(x,t) > 0$ the space and time-dependent diffusion coefficient. Application of Theorem 3.1 shows that finite difference discretization by means of central differences gives a positive semidiscretization if and only if the cell Péclet numbers, defined as ah/d, satisfy

$$\max_{x,t} \frac{|a(x,t)|h}{d(x,t)} \leq 2. \tag{3.3}$$

Discretizing the advection part by means of first-order upwind, and second-order central differences for the diffusive part, gives an unconditionally positive semidiscretization. The reaction terms (2.3) can be written in the production-loss form

$$\bar{R}_k^g(t, w) = p(t, w) - L(t, w)w, \tag{3.4}$$

where $p(t, w) \geq 0$ (componentwise) is a vector and $L(t, w) \geq 0$ (componentwise) a diagonal matrix, whose components $p_i(t, w)$ and $L_i(t, w)$ are of polynomial type with non-negative coefficients and can easily be found. Addition of reaction terms (2.3), which can be written in the production-loss form (3.4), to the advection–diffusion Eq. (3.2) and applying Theorem 3.1 gives a positive semidiscretization for the one-dimensional advection–diffusion–reaction equation if and only if $p(t, w) \geq 0$ (see also [8, Section I.7]).

The one-dimensional results above are easily generalized to higher dimensions and to FV schemes. Therefore, discretizing the species equations in space by means of a hybrid FV scheme as introduced in [7], which uses the central difference scheme if possible and the first-order upwind scheme if necessary, maintains positivity. We remark that for higher order upwinding, such as, for example, third-order upwinding, positivity is not ensured for all step-sizes (see [8, Section I.7]).

Definition. *A time integration method $w_{n+1} = \varphi(w_n)$ is called positive if for all $n \geq 0$ holds, $w_n \geq 0 \Longrightarrow w_{n+1} \geq 0$*

Positivity restricts the use of time integration methods. In this section, we will present results for nonlinear systems $w'(t) = F(t, w(t))$. First, we start exploring the positivity property for Euler Forward and Backward time integration.

Positivity for Euler Forward and Euler Backward. Suppose that the right hand side of the nonlinear semidiscretization $w'(t) = F(t, w(t))$ satisfies

Condition 3.2. There is an $\alpha > 0$, depending on $F(t, w)$, such that for a time step τ holds: if $\alpha\tau \leq 1$, then $w + \tau F(t, w) \geq 0$ for all $t \geq 0$ and $w \geq 0$

Provided that $w_n \geq 0$, Condition 3.2 guarantees positivity for w_{n+1} computed via Euler Forward (EF). For linear semidiscrete systems $w'(t) = Aw(t)$ with entries $A_{ij} \geq 0$ for $i \neq j$, $A_{ii} \geq -\zeta$ for all i and $\zeta > 0$ fixed, Condition 3.2 is easily illustrated. Application of EF to these systems gives a positive solution if $1 + \tau A_{ii} \geq 0$ for all i. This will hold if $\alpha\tau \leq 1$. To write down such an expression for α for Eq. (2.1) is almost undoable, because of the complicated structure of the chemical source terms.

Furthermore, assume that $F(t, w(t))$ also satisfies

Condition 3.3. For any $v \geq 0, t \geq 0$ and $\tau > 0$ the equation $w = v + \tau F(t, w)$, has a unique solution w that depends continuously on τ and v.

According to the following theorem we have unconditional positivity for Euler Backward (EB). The proof is taken from [8].

Theorem 3.4. *Condition 3.2 and 3.3 imply positivity for EB for any step-size τ*

Proof. For given t, v, and with a chosen τ, we consider the equation $w = v + \tau F(t, w)$ and we call its solution $w(\tau)$. We have to show that $v \geq 0$ implies $w(\tau) \geq 0$ for all positive τ. By continuity, it is sufficient to show that $v > 0$ implies $w(\tau) \geq 0$. This is true because if we assume that $w(\tau) > 0$ for $\tau \leq \tau_0$, except for the ith component $w_i(\tau_0) = 0$, then $0 = w_i = v_i + \tau_0 F_i(t, w(\tau_0))$. According to Condition 3.2, we have $F_i(t, w(\tau_0)) \geq 0$ and thus $v_i + \tau_0 F_i(t, w(\tau_0)) > 0$, which is a contradiction. ∎

Remark. Application of EB to the nonlinear semidiscretization $w'(t) = F(t, w(t))$ needs the solution of the nonlinear vector equation

$$w_{n+1} - \tau F(t_n, w_{n+1}) = w_n. \tag{3.5}$$

Theorem 3.4 ensures for every time step-size τ positivity of the exact solution of (3.5). In practice, however, the solution of (3.5) can be approximated by an iterative solver, and thus, it is not guaranteed to be positive. See also Section B

General Remarks on Positive Time Integration. We remark that unconditionally positive schemes can be implicit schemes only, like EB. One might hope to find more accurate methods with this unconditional positivity property. However, this hope is dashed by the following result, due to Bolley and Crouzeix [9].

Theorem 3.5. *Any unconditionally positive time integration method has order* $p \leq 1$.

For a proof, we refer to [9]. The consequence is that the only well-known method having unconditionally positivity is EB. Finally, we remark that for higher order methods the need to preserve positivity may necessitate the use of impractically small time steps.

B. Time Step Strategy

A popular strategy to avoid negative mass fractions as consequence of the nonlinear (Newton) solver is *clipping*, i.e., putting a negative mass fraction to zero. A disadvantage of clipping is that mass is added. In this article, we avoid clipping by adjusting the time step-size if necessary. We briefly explain our strategy.

To compute a Newton update, a linear system has to be solved. For the 2D simulations in this article, direct solvers are still feasible. In the Newton update vector, the components that can be classified as round off errors are removed. After convergence of the Newton iteration, the solution is tested for negative components. If their magnitude is larger than a possible round-off error, the time step-size is halved and the Newton iteration is redone.

C. Relation Between Positivity and Total Variation Diminishing, and Concluding Remarks

Like positivity, total variation diminishing (TVD) is a form of super stability. The TVD property is developed for studying the properties of numerical schemes to solve hyperbolic conservation laws (see, for instance, [8, 10–12]).

In [10], it has been proven that higher order TVD schemes can be explicit only. Addition of explicit stages to an higher order implicit scheme could retrieve the TVD property. For our application where stiff chemistry is involved, the latter is not preferred. The same will hold for positivity of these schemes, since results for positivity are derived in the same way. The last remark can be found in, for instance, [11].

We conclude by noting that, for higher order time integration of the stiff species equations, one has to fulfill a tight restriction on the time step to maintain positivity, or to have a look at, for example, IMplicit EXplicit (IMEX) methods [8, Section IV.4].

IV. SUITABLE METHODS TO INTEGRATE

In this section, we give a brief overview of the ODE methods and the nonlinear and linear solvers we use to solve the PDEs that model the CVD process (2.1).

A. Time Integration Methods

From the previous sections, it became clear that EB is an ideal time integration method. It has the advantages of being unconditionally stable and positive. A disadvantage is the first-order

consistency. Next, we will discuss a selection of higher order time integration methods that are suitable to integrate (2.1) from a theoretical point of view.

All ODE schemes we tested are equipped with a variable time step selector, as is usual in the ODE field. Details on variable time stepping can be found in [3, 8, 13, 14].

Rosenbrock Methods. Rosenbrock methods are linearly implicit Runge–Kutta type methods for stiff ODEs, which have proven to be effective for low to moderate accuracy for various stiff problems (see [8, 13]). In our experiments, we used the second-order scheme, which depends on the choice of the parameters b_2 and γ, ROS2

$$\begin{aligned} w_{n+1} &= w_n + b_1 k_1 + b_2 k_2 \\ k_1 &= \tau F(t_n, w_n) + \gamma \tau J_F k_1 \\ k_2 &= \tau F(t_n + \alpha_{21}\tau, w_n + \alpha_{21} k_1) + \gamma_{21} \tau J_F k_1 + \gamma \tau J_F k_2, \end{aligned} \tag{4.1}$$

with coefficients $b_1 = 1 - b_2, \alpha_{21} = \frac{1}{2b_2}$, and $\gamma_{21} = -\frac{\gamma}{b_2}$. In (4.1), J_F is the Jacobian of $F(t_n, w_n)$. ROS2 is second-order consistent for arbitrary γ and $b_2 \neq 0$, 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 $\mathcal{R}(z) \geq 0$, for $z \in \mathbb{R}^-$, where $\mathcal{R}(z)$ is the stability function of ROS2. For diffusion–reaction problems, which have a Jacobian with negative real eigenvalues, this property ensures a positive solution. Adding advection introduces imaginary parts to the eigenvalues, such that positivity is no longer guaranteed. However, as has been experienced in [15], the ROS2 scheme performs quite well with respect to positivity for advection–diffusion–reaction problems. Although there is no explanation for this unexpected behavior, it is conjectured that the property that $\mathcal{R}(z) \geq 0$ for all $z \in \mathbb{R}^-$ plays a role [15].

Backward Differentiation Formulas. In chemistry applications, the backward differentiation formula (BDF) methods belong to the most widely used methods to solve stiff chemical reaction equations, due to their favorable stability properties. The k-step BDF methods are implicit of order k and defined as

$$\sum_{j=0}^{k} \alpha_j w_{n+j} = \tau F(t_{n+k}, w_{n+k}), \quad n = 0, 1, \ldots, \tag{4.2}$$

which uses the k past values $w_n, \ldots, w_{n+k-1}$ to compute w_{n+k}. Note that the most advanced level is t_{n+k} instead of t_{n+1}. The one-step BDF method is EB, whereas the two-step method is

$$\frac{3}{2} w_{n+2} - 2 w_{n+1} + \frac{1}{2} w_n = \tau F(t_{n+2}, w_{n+2}). \tag{4.3}$$

The BDF-1 and BDF-2 methods are A-stable, but for $k > 2$ they are $A(\alpha)$-stable and for $k > 6$ even unstable [16].

Remark. A disadvantage is that the first $(k - 1)$ approximations cannot be computed with the k-step BDF scheme. Instead, they should be obtained by another scheme

As for Runge–Kutta methods, the requirement of positivity does place a severe time step-size restriction on BDF methods. Under Conditions 3.2 and 3.3, we obtain positivity for BDF2 whenever $\alpha\tau \leq \frac{1}{2}$, provided that w_1 is computed positively from w_0. Remark that this positivity condition is a factor 2 tighter than the condition for EF.

IMEX Runge–Kutta Chebyshev Methods. The IMEX extension of the class of Runge–Kutta Chebyshev (RKC) methods, developed by Verwer et al. [17], is designed to solve stiff systems of ODEs. The RKC methods belong to the class of explicit Runge–Kutta methods. They posses an extended real stability interval with a length proportional to s^2, with s the number of intermediate stages.

Definition. 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

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

Construction of the second-order RKC scheme is based on choosing shifted Chebyshev polynomials (of the first kind) as stability functions such that the optimal stability bound for explicit RK schemes, $\beta(s) = 2s^2$, is best as possible achieved [8]. The scheme is given as

$$\begin{aligned}
w_{n0} &= w_n, \\
w_{n1} &= w_n + \tilde{\mu}_1 \tau F(t_n, w_{n0}), \\
w_{nj} &= (1 - \mu_j - \nu_j) w_n + \mu_j w_{n,j-1} + \nu_j w_{n,j-2} \\
&\quad + \tilde{\mu}_1 \tau F(t_n + c_{j-1}\tau, w_{n,j-1}) + \tilde{\gamma}_j \tau F(t_n, w_{n0}), \quad j = 2, \ldots, s \\
w_{n+1} &= w_{ns},
\end{aligned} \tag{4.4}$$

with coefficients

$$\omega_0 = 1 + \frac{\varepsilon}{s^2}, \quad \omega_1 = \frac{T_s'(\omega_0)}{T_s''(\omega_0)}, \tag{4.5}$$

$$b_j = \frac{T_j''(\omega_0)}{(T_j'(\omega_0))^2}, \quad c_j = \frac{T_s'(\omega_0)}{T_s''(\omega_0)} \frac{T_j''(\omega_0)}{T_j'(\omega_0)} \approx \frac{j^2 - 1}{s^2 - 1}, \tag{4.6}$$

$$\tilde{\mu}_1 = b_1 \omega_1, \quad \mu_j = \frac{2 b_j \omega_0}{b_{j-1}}, \quad \nu_j = -\frac{b_j}{b_{j-2}}, \tag{4.7}$$

$$\tilde{\mu}_j = \frac{2 b_j \omega_1}{b_{j-1}}, \quad \tilde{\gamma}_j = -a_{j-1} \tilde{\mu}_j, \quad \text{and} \quad a_j = 1 - b_j T - j(\omega_0). \tag{4.8}$$

In (4.5) and (4.6), $T_i(z)$ is the ith order Chebyshev polynomial of the first kind. For $z \in \mathbb{C}$, it is recursively defined as

$$T_j(z) = 2z T_{j-1}(z) - T_{j-2}(z), \tag{4.9}$$

where $T_0(z) = 1$ and $T_1(z) = z$. In this case $\beta(s) = 2/3(s^2 - 1)$; the stability function is $\mathcal{R}(z) = a_s + b_s T_s(\omega_0 + \omega_1 z)$ and the stability region for $s = 5$ is illustrated in Fig. 1(a). For explicit integration of practical problems, the small stability region around the real axis is not favorable. Therefore, it is possible to introduce a damping parameter ε into the coefficient ω_0, such that stability regions as in Fig. 1(b) are obtained. The stability boundary is in the damped case $2/3(s^2 - 1)(1 - 2/15\varepsilon)$ instead of the optimal value $2s^2$ (see [17]).

The IMEX extension of the above scheme is as follows. Suppose we have an ODE system $w'(t) = F(t, w(t))$, where $F(t, w)$ can be split as $F(t, w) = F_E(t, w) + F_I(t, w)$ with $F_I(t, w)$ the part of $F(t, w)$, which is too stiff to be integrated explicitly. The term $F_E(t, w)$ is the moderate

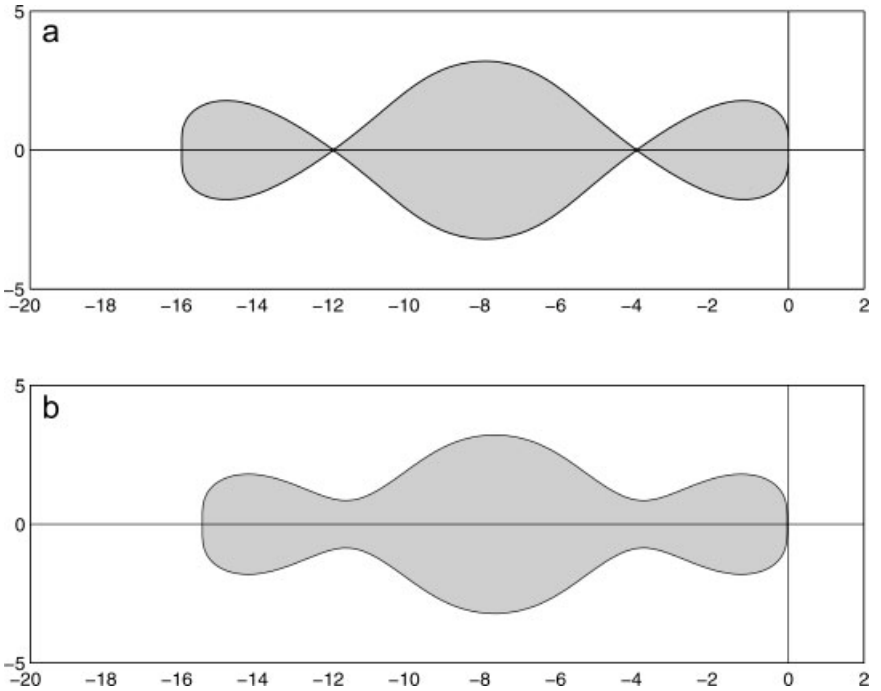

FIG. 1. Stability region of (4.4) for $s = 5$ and $\varepsilon = 0.1$ (10% damping) without damping (a) and with damping (b).

stiff part of F that can be integrated explicitly by the RKC method. Then, the IMEX extension of (4.4) from [17] reads

$$
\begin{aligned}
w_{n0} &= w_n,\\
w_{n1} &= w_n + \tilde{\mu}_1 \tau F_E(t_n + c_0\tau, w_{n0}) + \tilde{\mu}_1 \tau F_I(t_n + c_1\tau, w_{n1}),\\
w_{nj} &= (1 - \mu_j - \nu_j) w_n + \mu_j w_{n,j-1} + \nu_j w_{n,j-2} + \tilde{\mu}_j \tau F_E(t_n + c_{j-1}\tau, w_{n,j-1})\\
&\quad + \tilde{\gamma}_j \tau F_E(t_n + c_0\tau, w_{n0}) + [\tilde{\gamma}_j - (1 - \mu_j - \nu_j)\tilde{\mu}_1]\tau F_I(t_n + c_0\tau, w_{n0})\\
&\quad - \nu_j \tilde{\mu}_1 \tau F_I(t_n + c_{j-2}\tau, w_{n,j-2}) + \tilde{\mu}_1 F_I(t_n + c_j\tau, w_{nj}) \qquad (4.10)\\
w_{n+1} &= w_{ns}.
\end{aligned}
$$

Note that the highly stiff part of F is treated implicitly.

The IRKC scheme holds that the implicit part is unconditionally stable as long as the eigenvalues of the Jacobian of $F_I(t, w)$ are real, whereas the stability condition for the explicit part remains unchanged. Another property is that steady states are returned exactly, which is not true for other operator splittings (see [8, Section IV.I]). Unconditional positivity is not guaranteed; the exact condition is not known to the authors.

We conclude with some remarks on the implementation of the IRKC solver. For efficiency reasons, a modified Newton iteration, in which the Jacobian of $F(t, w)$ is evaluated only once, is used to solve the s nonlinear systems of equations arising in the s stages in the IRKC scheme. Then, per time step, only one LU factorization of $[I - \tilde{\mu}_1 \tau F_I'(t, w(t))]$, where $F_I'(t, w)$ is the Jacobian

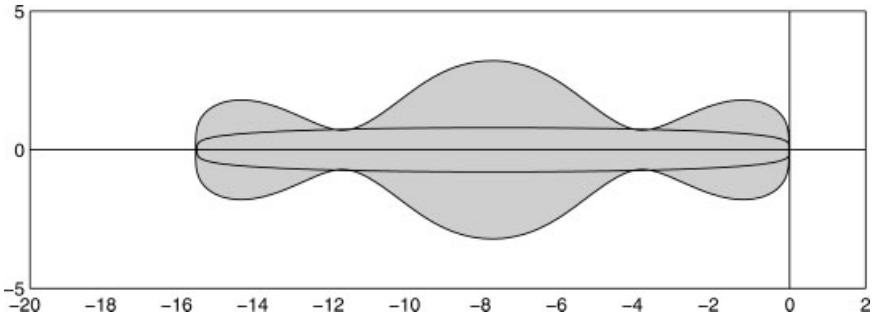

FIG. 2. Stability region of (4.4) with inscribed oval.

of $F_I(t, w)$ with respect to w, has to be computed. Details of the variable time step controller, which tests the current solution for accuracy and the explicitly integrated part for stability, can be found in [17]. This controller also adjusts the number of stages s, depending on the time step-size and the conditions for stable integration of advection and diffusion called IRKC(full) or for stable integration of diffusion only [called IRKC(fly)]. If the eigenvalues emerging from von Neumann stability analysis are lying in ovals like in Fig. 2, then stability for the advection–diffusion part is ensured. How this can be done is described in [12, 17].

B. Nonlinear Solvers

An obvious choice to approximate nonlinear solutions of $G(x) = 0, x \in \mathbb{R}^k, k \in \mathbb{N}$, is the second-order convergent Newton iteration. It is generally known that if the initial guess of the solution is sufficiently near the exact solution, the Newton iteration converges, and otherwise it diverges.

The disadvantage of local convergence can be discarded by extending the Newton algorithm with an Armijo-based rule [18]: To have a decreasing sequence $\|G(x_n)\|_2$, the Newton step $d_n = -G'(x_n)^{-1}G(x_n)$ is adjusted by the smallest integer m and a small parameter ϖ such that

$$\|G(x_n + 2^{-m}d_n)\|_2 \leq (1 - \varpi 2^{-m})\|G(x_n)\|_2, \tag{4.11}$$

and let the Newton-step be $2^{-m}d_n$. Condition (4.11) is called the sufficient decrease of $\|G\|$. The parameter ϖ is equal to α in the Armijo condition (also known as the α-condition) [19]:

$$f(x_n + d_n) \leq f(x_n) + \alpha \nabla f(x_n)^T d_n. \tag{4.12}$$

Condition 4.12 is a sufficient condition on f to let d_n be a sufficient decrease direction. It is an easy exercise to show that $\varpi = \alpha$ by substituting $f = 1/2\|G\|_2^2$ into (4.12). A typical value for α, and thus also for ϖ, is 10^{-4}. In the case that after two reductions by halving the Newton step does not lead to sufficient decrease, we build a quadratic polynomial model of

$$\phi(\varsigma) = \|G(x_n + \varsigma d)\|^2, \tag{4.13}$$

based on interpolation of ϕ at the three most recent values of ς. The next ς is the minimizer of (4.13), subject to a safeguard that the reduction of ς is at least one half and at most a factor of ten, (see [18]).

As is custom in Method of Lines solvers for PDEs (with direct linear solvers), we also have the possibility to update the Jacobian occasionally, instead of performing a full Newton iteration in every time step. We will briefly explain the strategy, which can be found in [13]. Define the convergence rate of the Newton iteration as

$$\Theta_n = \|d_n\|/\|d_{n-1}\|, \quad n \geq 1, \tag{4.14}$$

where d_n is the Newton update as defined before. Then, we do not recompute the Jacobian in the next time step when (i) the Newton process converges in one iteration, or, (ii) the convergence rate in the last Newton iteration was very small, e.g., $\Theta_n \leq 10^{-3}$, which means that the last Newton iteration gives fast convergence (see [13, Chapter IV.8]).

C. Linear Solvers

For the 2D problem in this article, the linear systems in the Newton iteration can be solved directly by means of an LU factorization of the Jacobian. Traditionally, the unknowns are arranged per species and lexicographic in the grid (see [7]). To reduce the amount of work to factorize the Jacobian, we reorder the unknowns and equations, which reduces the bandwidth of the Jacobian considerably (see [3, 14]).

For the IRKC scheme, this reordering of unknowns is required for efficient integration of (2.1) (see [17]). Define NGRID as the number of grid points in the computational grid and recall that N is the number of species. If $F_E(t, w)$ contains the discretized advection and diffusion terms and $F_I(t, w)$ the chemical source terms, then for this scheme NGRID uncoupled linear systems of dimension $N \times N$ have to be solved per Newton iteration. See [17].

Because of the stiff reaction terms, the condition number of the Jacobian appearing in EB, ROS2, and BDF2 can be of $\mathcal{O}(10^{10})$. Application of diagonal scaling will decrease the condition number by approximately two orders of magnitude. As a result, the direct linear solver is two orders of magnitude more accurate. The estimates of the condition number of the Jacobian in our code are obtained with routines from the LAPACK package (see [14, 20]). We remark that in our tests the LAPACK routine(s) for estimating the condition number occasionally overestimates the condition number several orders of magnitude. If an estimated condition number grows several orders in comparison with the condition number of the previous time step, then in most cases its magnitude was larger than machine precision. For these exceptions, we manually checked with MATLAB experiments that it was indeed overestimated. In that case, we have used the previous estimation, i.e., either the one from the previous Newton iteration or from the previous time step.

V. NUMERICAL SIMULATION

In this section, we present results on the simulations of the benchmark problem of Kleijn [4]. Before discussing the numerical results, we first present relevant information on the chemical model used. Another reference containing valuable information to reproduce our simulations is [3]. Note that some tables in [4] contain typographical and printing errors, which have been corrected in [3].

A. Chemistry Model in the Benchmark [4]

In this article, we are interested in the transient solution of a CVD model consisting of 17 species and 26 reactions (see Tables 2 and 4 in [4] and Table 1 in [3]). The reactor configuration is

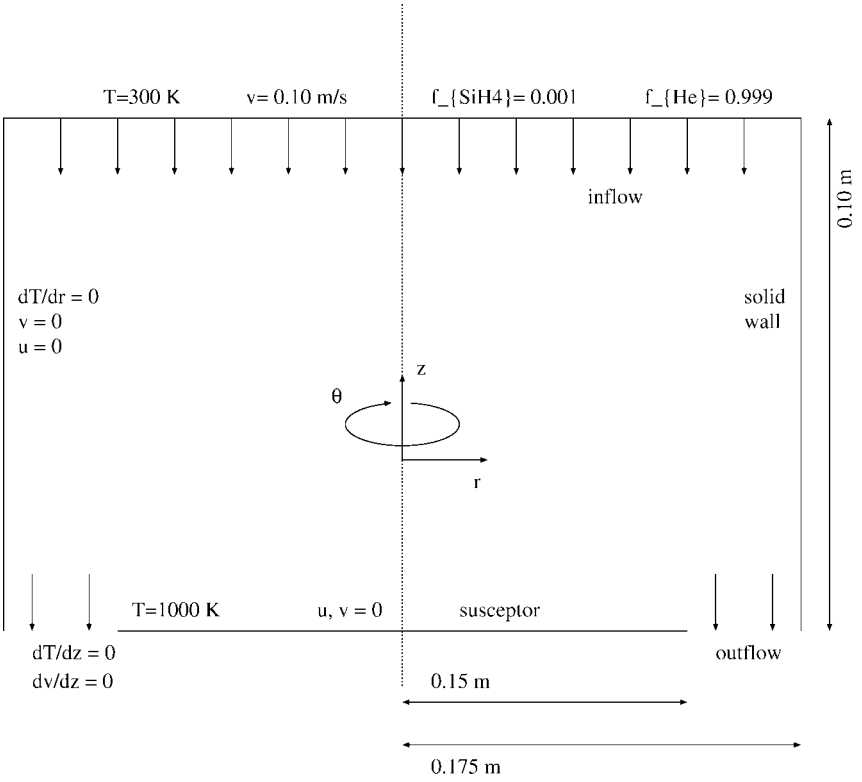

FIG. 3. Reactor geometry.

as in Fig. 3. In the hot gas region above the susceptor, see Fig. 3, the reactive gas silane SiH_4 decomposes into silene SiH_2 and hydrogen H_2. This gas phase reaction initiates a chain of 25 homogeneous reactions leading to the formation of 14 different silicon containing species, which each may diffuse to the susceptor to produce a thin solid silicon film.

As computational domain we take, because of axisymmetry, one half of the $(r–z)$ plane. The pressure in the reactor is 1 atm $= 1.01325 \times 10^5$ Pa. From the top a gas-mixture enters the reactor with a uniform temperature $T_{\text{in}} = 300$ K and a uniform velocity $u_{\text{in}} = -0.10$ m/s. The inlet silane mole fraction is $f_{\text{in,SiH}_4} = 0.001$ and the rest is helium. At a distance of 0.1 m below the inlet a susceptor with temperature $T_s = 1000$ K and a diameter 0.3 m is placed. Unlike the problem considered in [4] the susceptor does not rotate. Furthermore, the outer walls of the reactor are adiabatic and do not rotate.

Note that it suffices to solve 16 nonlinearly coupled species equations (2.1), since the mass fraction of the carrier gas helium will be computed via the property that they sum up to one. The reactions terms in (2.1) are constructed as in (2.3) and (2.5), where the fit parameters can be looked up in [3, 4]. The backward reaction rates are selfconsistently computed from

$$k^g_{\text{backward}}(T) = \frac{k^g_{\text{forward}}(T)}{K^g} \left(\frac{RT}{P^0} \right)^{\sum_{i=1}^{N} \nu_{ik}}, \tag{5.1}$$

where the reaction equilibrium K^g is approximated by $K^g(T) = A_{\text{eq}} T^{\beta_{\text{eq}}} e^{\frac{-E_{\text{eq}}}{RT}}$, with R the universal gas constant and the fit parameters A_{eq}, β_{eq}, and E_{eq} as in [3, 4].

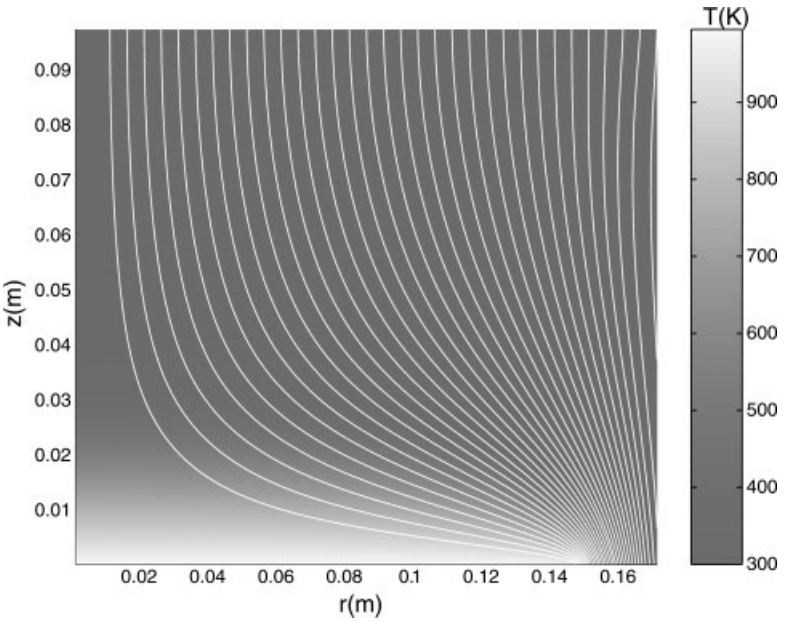

FIG. 4. Streamlines and temperature field in Kelvin for the right half part of the reactor configuration given in Fig. 3.

Since the reactants in the gas mixture are highly diluted in the carrier gas, we use the steady state velocity fields, temperature field, pressure field, and density field computed by Kleijn [4]. In these computations, buoyancy has not been accounted for. In Fig. 4, the streamlines and temperature field are shown.

At the susceptor S irreversible reactions take place, such that gaseous reactants will be transformed into solid and gaseous reaction products. Deposition of solid (s) silicon Si, and desorption of gaseous (g) hydrogen, is due to one of the surface reactions:

$$\mathrm{Si}_n\mathrm{H}_{2m} \xrightarrow{R^S_{\mathrm{Si}_n\mathrm{H}_{2m}}} n\mathrm{Si(s)} + m\mathrm{H}_2\mathrm{(g)}, \tag{5.2}$$

$$\mathrm{Si}_n\mathrm{H}_{2m+1} \xrightarrow{R^S_{\mathrm{Si}_n\mathrm{H}_{2m+1}}} n\mathrm{Si(s)} + m\mathrm{H}_2\mathrm{(g)} + \mathrm{H(g)}, \tag{5.3}$$

where $n = 1, 2$, or 3, and $m = 0, 1, 2, 3$, or 4. The molar reaction rate R_i^S for the decomposition of gas species i is given as

$$R_i^S = \frac{\gamma_i}{1 - \frac{\gamma_i}{2}} \frac{P f_i}{(2\pi m_i R T_s)^{\frac{1}{2}}}, \tag{5.4}$$

where T_s the temperature of the wafer surface. The sticking coefficient γ_i is equal to one for all silicon containing species, except for $\gamma_{\mathrm{Si}_3\mathrm{H}_8} = 0$, $\gamma_{\mathrm{Si}_2\mathrm{H}_6} = 0.537 \exp(\frac{-9400}{T_s})$, and $\gamma_{\mathrm{SiH}_4} = \frac{1}{10}\gamma_{\mathrm{Si}_2\mathrm{H}_6}$. The deposition rate $\mathcal{G}_{\mathrm{Si}}$ of solid silicon Si is defined as

$$\mathcal{G}_{\mathrm{Si}} = \frac{m_{\mathrm{Si}}}{\rho_{\mathrm{Si}}} \sum_{s=1}^{S} R_s^S \chi_{\mathrm{Si}}, s, \tag{5.5}$$

where $\chi_{Si,s}$ is the stoichiometric coefficient as given in (5.2) and (5.3), and m_{Si} and ρ_{Si} are the molar mass and the density of (solid) silicon.

The species mass fractions are computed in the cell centers. Therefore, computing the mass flux due to surface chemistry is not straightforward when it needs to be positive. In [3], we already described how this could be taken care off. To be self contained we shortly describe the idea. According to Eq. (5.4), the molar reactive surface flux F_{wall} is linearly proportional to the species molar concentration at the reacting surface, and thus also linearly proportional to the species mass fraction at the wafer. Thus, we can write $F_{wall} = K\omega_{wall}$, where ω_{wall} is the unknown species mass fraction at the wafer. For the class of CVD problems considered in this article advective transport is negligible near the wafer. Thus, only diffusive transport occurs near the wafer, and thus the diffusive mass flux near the wafer should be equal to F_{wall}. From this equality, an expression for ω_{wall} can be found, and thus the reactive surface flux can be computed. It appears that ω_{wall} is always positive [3].

B. Numerical Results on the Benchmark Problem

We compare the different ODE methods of the previous section in terms of efficiency for solving the transient CVD problem as presented in [3]. The simulation runs from the the instant that the reactor is completely filled with helium carrier gas and a mixture of helium and silane starts to enter the reactor, until steady state. The spatial computational grid consists of 35 equidistant grid points in radial direction and 32 nonequidistant grid points in axial direction. The grid spacing in axial direction gradually increases towards the wafer surface. In our experiments, steady state is obtained when for a certain time step t_n the inequality

$$\frac{\|w_{n+1} - w_n\|_2}{\|w_n\|_2} \leq 10^{-6} \tag{5.6}$$

holds, where w_n is the numerical solution of the semidiscretization $w(t) = F(t, w)$ on time $t = t_n$. Thereafter, (i) our steady state solution is benchmarked against the steady state solution obtained by Kleijn [4], (ii) the transient solutions of the different schemes are compared with a time accurate ODE solution, and (iii) we compare the integration statistics.

In Fig. 5, the steady state axial concentration profiles at $r = 0$ of some selected species are given, for both our simulations and the ones done by Kleijn [4]. The total deposition rate of solid silicon, by which we mean the instantaneous deposition rate due to the contributed effect of all depositing silicon species, found by Kleijn [4] is 1.92 nm/s at $r = 0$. In our simulations, we found a deposition rate of 1.93 nm/s. With respect to the quality of our steady state solution(s), we conclude that they agree well with the steady state solution found in [4]. In addition, in Fig. 6, the transient deposition rates due to some selected species are presented. Similar transient results on this problem, or a problem of similar complexity, are not known to the authors.

In Tables I–III, numerical results for the various time integration methods, with either the full or modified Newton iteration to solve the nonlinear systems, and the relative errors with respect to a time accurate ODE solution on some fixed times in the L_2 norm, are given. We used relative errors, because the solution contains relatively small components. The user-specified quantity TOL to monitor the local truncation error is taken equal to 10^{-3}. We observe that for the global errors as presented in Tables I–III, the behavior is as expected.

For the unconditional positive EB time integration scheme can be remarked that Newton iterations with an approximated Jacobian as in Section IV see Table II, have influence on the positivity of the solution, i.e., the number of rejected time steps due to negative species increases, and is in this case equal to 22. Rejected time steps due to negative entries in the solution vector should

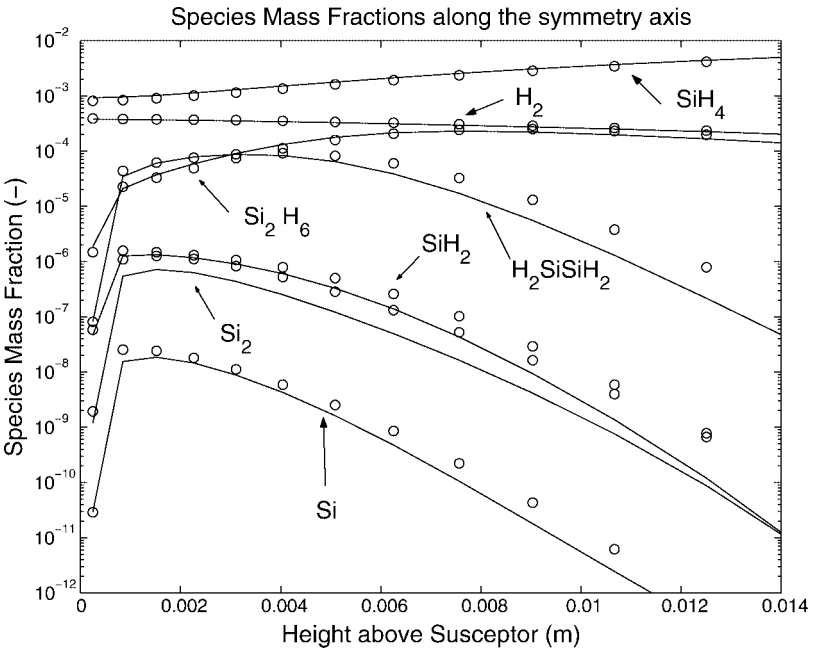

FIG. 5. Axial steady state concentration profiles at $r = 0$ due to some selected species. Solid lines are Kleijn's solutions [4], circles are obtained with the Euler Backward scheme with full Newton solver.

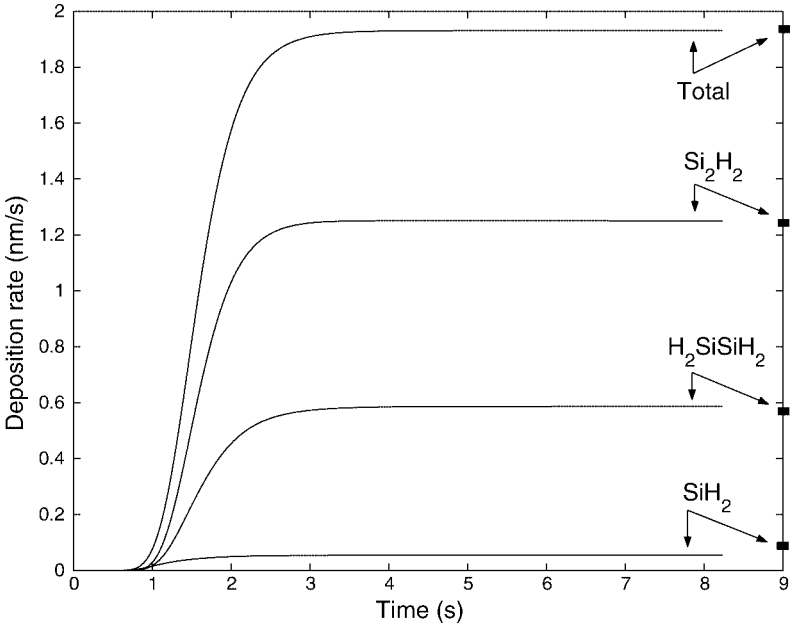

FIG. 6. Transient deposition rates at $r = 0$ due to some selected species. On the right vertical axis: steady state deposition rates obtained with Kleijn's steady state code [4].

TABLE I. Integration statistics for EB and BDF-2, with full Newton solver.

Number of	EB	BDF-2
F	190	757
F'	94	417
Linesearch	11	0
Newton iters	94	417
Rej. time steps	1	10
Acc. time steps	38	138
CPU Time	6500	30,500
Relative error ($t = 1.6/t = 3.2$)	$6.8 \times 10^{-3}/7.9 \times 10^{-4}$	$2.2 \times 10^{-3}/1.4 \times 10^{-4}$

be redone with smaller time steps, resulting in a larger number of F evaluations (the number of Jacobian evaluations is approximately equal). Thus, as a result of an increasing number of Newton iterations, the total computational costs increase.

For the BDF2 scheme, application of modified Newton strategy of Section IV gives more satisfying results. From Table II, it can be concluded that for BDF2 an increasing number of cheaper Newton iterations is computationally cheaper than factorizing the Jacobian in every Newton iteration.

With respect to the other higher order time integration schemes, we note the following. ROS2 is the cheapest higher order time integrator for this CVD process. For the IRKC scheme, we see that both versions perform equally well. Since there is no gain in efficiency by using "on the fly" stability conditions for the explicit part, the more robust fully CFL-protected IRKC(full) is preferred.

With respect to positivity of the solution during transient simulation, we note the following. Omission of the reacting surface and thermal diffusion in the reaction Jacobian gives very poor Newton convergence. We also observed that in this case the solution conserves positivity for very small time steps only, even for EB. We conclude that for this CVD problem it is required to use the exact Jacobian, in which case also the derivatives of the reacting surface and thermal diffusion are included.

VI. CONCLUSIONS

In this article, we presented two-dimensional transient simulations of a benchmark CVD problem [4], where the accent has been put on the efficient solution of the system of stiff advection–diffusion–reaction equations. We tested, in terms of efficiency, a collection of time integration

TABLE II. Integration statistics for EB and BDF2, with modified Newton, as explained in Section IV(B).

Number of	EB	BDF-2
F	720	1786
F'	84	163
Linesearch	39	33
Newton iters	463	1441
Rej. time steps	31	33
Acc. time steps	88	121
CPU Time	10,800	17,000
Relative error ($t = 1.6/t = 3.2$)	$6.8 \times 10^{-3}/7.9 \times 10^{-4}$	$2.2 \times 10^{-3}/1.4 \times 10^{-4}$

TABLE III. Integration statistics for ROS2, IRKC(fly), where stability for the explicitly integrated part is tested for diffusion only, and IRKC(full), where stability conditions are forced for both advection and diffusion, schemes.

Number of	ROS2	IRKC(fly)	IRKC(full)
F	424	429,662	427,911
F'	142	2005	2008
Linesearch	0	50	30
Newton iters	0	17,425	17,331
Rej. time steps	2	729	728
Acc. time steps	140	1276	1284
CPU Time	8000	20,000	19,500
Relative error($t = 1.6/t = 3.2$)	$1.1 \times 10^{-3}/2.5 \times 10^{-4}$	$1.8 \times 10^{-3}/8.3 \times 10^{-5}$	

methods, which have to be suitable for integration of stiff equations, as well as to be positive. For the resulting (non)linear systems standard techniques are used. It appeared that a clever way of using standard techniques still can give satisfying results. However, for 3D geometries, iterative linear solvers will be needed. For problems from chemistry, like the one in this article, this is still a challenging task for future research.

Based on the observations in our experiments, we conclude that in terms of efficiency Euler Backward is the best choice to do a transient simulation (running into steady state). The difference in computational costs has to do with the unconditionally positivity of EB, where the other integration methods are conditionally positive. For transient simulations of CVD models with more complicated chemistry involved, and 3D simulations, we expect that IRKC is also an excellent candidate, because the dimension of the linear systems appearing in this method do not change when going from two to three spatial dimensions.

The work of S. van Veldhuizen was supported by the Delft Center for Computational Science and Engineering. Furthermore, the authors thank both the referees for their suggestions that improved the manuscript considerably.

References

1. C. R. Kleijn, R. Dorsman, K. J. Kuijlaars, M. Okkerse, and H. van Santen, Multi-scale modeling of chemical vapor deposition processes for thin film technology, J Cryst Growth 303 (2007), 362–380.
2. M. A. Alam and M. L. Green, Mathematical description of atomic layer deposition and its application to the nucleation and growth of HfO_2 gate dielectric layers, J Appl Phys 94 (2003), 3403–3413.
3. S. van Veldhuizen, C. Vuik, and C. R. Kleijn, Numerical Methods for Reacting Gas Flow Simulations, Int J Multiscale Comput Eng 5 (2007), 1–10.
4. C. R. Kleijn, Computational modeling of transport phenomena and detailed chemistry in chemical vapor deposition—a benchmark solution, Thin Solid Films, Vol. 365, Elsevier, 2000, pp. 294–306.
5. T. P. Merchant, M. K. Gobbert, T. S. Cale, and L. J. Borucki, Multiple scale integrated modeling of deposition processes, Thin Solid Films, Vol. 365, Elsevier, 2000, pp. 368–375.
6. A. Kersch and T. Schafbaur, Thermal modelling of RTP and RTCVD processes, Thin Solid Films, Vol. 365, Elsevier, 2000, pp. 307–321.
7. C. R. Kleijn, Transport phenomena in chemical vapor deposition reactors, PhD Thesis, Delft University of Technology, Delft, 1991.
8. W. Hundsdorfer and J. G. Verwer, Numerical solution of time-dependent advection–diffusion–reaction Equations, Springer series in computational mathematics, Vol. 33, Springer, Berlin, 2003.

9. C. Bolley and M. Crouzeix, Conservation de la Positivité Lors de la Discrétisation des Problèmes d'Évolution Paraboliques, RAIRO Anal Numer 12 (1973), 237–245.

10. S. Gottlieb, C.-W. Shu, and E. Tadmor, Strong stability-preserving high-order time discretization methods, SIAM Rev 43 (2001), 89–112.

11. W. Hundsdorfer, S. J. Ruuth, and R. J. Spiteri, Monoticity-preserving linear multistep methods, SIAM J Numer Anal 41 (2003), 605–623.

12. P. Wesseling, Principles of computational fluid dynamics, Springer series in computational mathematics, Vol. 29, Springer, Berlin, 2001.

13. E. Hairer and G. Wanner, Solving ordinary differential equations II: stiff and differential-algebraic problems, 2nd Ed., Springer series in computational mathematics, Vol. 14, Springer, Berlin, 1996.

14. S. van Veldhuizen, C. Vuik, and C. R. Kleijn, A note on the numerical simulation of Kleijn's benchmark problem, Technical report at the Delft University of Technology, Report 06-15, Delft University of Technology, Delft, 2006.

15. J. G. Verwer, E. J. Spee, J. G. Blom, and W. Hundsdorfer, A second-order rosenbrock method applied to photochemical dispersion problems, SIAM J Sci Comp 20 (1999), 1456–1480.

16. E. Hairer, S. P. Nørsett, and G. Wanner, Solving ordinary differential equations I: nonstiff problems, Springer series in computational mathematics, Vol. 8, Springer, Berlin, 1987.

17. J. G. Verwer, B. P. Sommeijer, and W. Hundsdorfer, RKC time-stepping for advection-diffusion-reaction problems, J Comp Phys 201 (2004), 61–79.

18. C. T. Kelley, Solving nonlinear equations with Newton's method, fundamentals of algorithms, SIAM, Philadelphia, 2003.

19. J. E. Dennis and R. B. Schnabel, Numerical method for unconstrained optimization and nonlinear equations, series in automatic computing, Prentice-Hall, Englewood Cliffs, NJ, 1983.

20. E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. Sorensen, LAPACK users' guide, 2nd Ed., SIAM, Philadelphia, 1995.