Simulation of a Multi-Component Transport Model for the Chemical Reaction of a CVD-Process

J. Geiser, R. Röhle

Abstract—In this paper we present discretization and decomposition methods for a multi-component transport model of a chemical vapor deposition (CVD) process. CVD processes are used to manufacture deposition layers or bulk materials. In our transport model we simulate the deposition of thin layers. The microscopic model is based on the heavy particles, which are derived by approximately solving a linearized multicomponent Boltzmann equation. For the drift-process of the particles we propose diffusion-reaction equations as well as for the effects of heat conduction. We concentrate on solving the diffusion-reaction equation with analytical and numerical methods. For the chemical processes, modelled with reaction equations, we propose decomposition methods and decouple the multi-component models to simpler systems of differential equations. In the numerical experiments we present the computational results of our proposed models.

Keywords—Chemical reactions, chemical vapor deposition, convection-diffusion-reaction equations, decomposition methods, multi-component transport.

AMS subject classifications: 35K57, 76R50, 74S20, 65M12.

I. INTRODUCTION

We are motivated to study the deposition of a CVD process with low-temperature and low-pressure plasma. In the last years, due to the research in producing high temperature films by depositing of low pressure processes the interest on such apparatuses has increased. We present such a model for low temperature and low pressure plasma, that can be used to implant or deposit thin layers of important materials. The applications are used in the production of so called metallic bipolar plates, which are used in polymer electrolyte fuel cells (PEFC), see [2].

Our apparatus is based on a plasma, which is modeled by a Boltzmann equation for the heavy particles and for their drift we use a classical diffusion equation, see [22]. For the simplification we neglect the electron modeling, because of their sufficient small influence.

Further we concentrate on modelling the reaction-processes in the CVD apparatus, which can be done with coupled reaction equations.

The numerical approximation is done by applying semianalytical methods or finite difference methods of second order in space and time. For the effective computations we propose the Waveform-relaxation method as a decomposition method for the multi-species reaction equations.

The paper is outlined as follows.

In section 2 we present our mathematical model and a possible reduced model for the further approximations. In section 3 we discuss the time and spatial discretization methods. Further in section 4, we propose the Waveform-relaxation method for decoupling the complicate equations to reduce the amount of computational work. The numerical experiments are given in Section 5. In the contents, that are given in Section 6, we summarize our results.

II. MATHEMATICAL MODEL

In the following a model is presented due to the motivation in [22] and [14].

The modeling is considered for the chemical reactions in the plasma reactor.

The simulation of the thin-film deposition can be done with a reaction equation combined with a diffusion equation, see [20].

We will concentrate on the reaction equations, which are given as

\[ \begin{aligned}
\partial_t u_1 + \nu_1 \nabla u_1 &= -\lambda_1 u_1 + \lambda_2 u_2 + \ldots + \lambda_m u_m \quad \text{in } (0,T), \quad (1) \\
\partial_t u_m + \nu_m \nabla u_m &= \lambda_1 u_1 + \lambda_2 u_2 + \ldots + \lambda_m u_m \quad \text{in } (0,T)
\end{aligned} \]

where the unknowns are given as \( u_i = u_i(x,t) \) till \( u_m = u_m(x,t) \text{ in } (0,T) \subset \mathbb{R}^+ \).

m is the number of species, e.g. reactants, heavy particles, etc. The interactions are chemical reactions between the species and given as \( \lambda_i \text{ till } \lambda_m \).

We assume the same densities \( c_0 \text{ till } c_m \) of the species as initial condition. For the boundary conditions we assume Neumann- or Dirichlet boundaries.

A. Large Knudsen Numbers

The model assumes that the heavy particles can be described with a fluid dynamical model, where the elastic collision define the dynamics and few inelastic collisions are, among other reasons, responsible for the chemical reactions.

To describe the individual mass densities, as well as the global momentum and the global energy as the dynamical conservation quantities of the system, corresponding conservation equations are derived from Boltzmann equations.
The individual character of each species is considered by mass-conservation equations and the so called difference equations.

The Boltzmann equations for the heavy particles (Ions and Neutral elements) are given as

\[
\frac{\partial}{\partial t} n_{i} + \frac{\partial}{\partial r} (\rho u + nT) = Q_{i}^{(s)},
\]

where \( \rho \) is the density, \( u \) is the velocity and \( T \) is the temperature of the plasma. The equations are strongly coupled and a decomposition can be done in the discretized form.

The drift diffusion for the heavy particles are in the following fluxes. The dissipative fluxes of the impulse and energy balance are linear combinations of generalized forces

\[
q = \lambda_{E} \langle E \rangle - \lambda_{T} T - \sum_{\alpha} n_{\alpha} \frac{1}{n_{\alpha}} \frac{\partial}{\partial r} n_{\alpha},
\]

\[
\bar{\xi} = -\eta \left( \frac{\partial}{\partial r} u + \frac{\partial}{\partial r} u + \frac{2}{3} \frac{\partial}{\partial r} u \bar{\xi} \right),
\]

\[
\varepsilon_{s} = \sum_{\alpha=1}^{N} \rho_{\alpha} c_{\alpha}^{s} + \frac{1}{2} \rho u^{s} + \frac{3}{2} nT.
\]

The diffusion of the species is underlying to the given plasma and described by the following equations

\[
\frac{\partial}{\partial t} n_{i} + \frac{\partial}{\partial r} (n_{i} u + n_{i} c_{i}) = Q_{i}^{(s)},
\]

\[
c_{i} = \mu_{i} \langle E \rangle - d^{(s)}_{i} \frac{\partial}{\partial r} T - \sum_{\alpha=1}^{N} D_{i}^{(s)} \frac{1}{n_{\alpha}} \frac{\partial}{\partial r} n_{\alpha}.
\]

The densities of the species are dynamical values and the species transport and mass transport are underlying to the following constraint conditions

\[
\sum_{i} m_{i} n_{i} = \rho,
\]

\[
\sum_{i} n_{i} m_{i} c_{i} = 0.
\]

Field model

The plasma transport equations are Maxwell equations and are coupled with a field. They are given as

\[
\frac{1}{\mu_{0}} \nabla \times B_{dyne} = -en_{u} + j_{\omega} c_{i},
\]

\[
\nabla \cdot B_{dyne} = 0,
\]

\[
\nabla \times E = \frac{\partial}{\partial t} B_{dyne}.
\]

B. Simplified model for large Knudsen Numbers

For the numerical analysis and for the computational results, we reduce the complex model and derive a system of coupled Boltzmann and Diffusion equations.

We do the following assumptions:

\[
q^{*} = -\lambda^{*} \frac{\partial}{\partial r} T,
\]

\[
\varepsilon^{*} = 0,
\]

\[
Q_{s,\text{inel}}^{(s)} = \text{const}
\]

and we obtain the system of equations:

\[
\frac{\partial}{\partial t} \rho + \frac{\partial}{\partial r} (\rho u + nT) = 0,
\]

\[
\frac{\partial}{\partial t} \rho u + \frac{\partial}{\partial r} (\rho u^{*} + nT) = \sum_{\alpha=1}^{N} n_{\alpha} \langle E \rangle - Q_{s,\text{inel}}^{(s)}.
\]

Remark 1. We obtain three coupled equations for the density, velocity and the temperature of the plasma. The equations are strong coupled and a decomposition can be done in the discretized form.

C. Chemical Reactions

For the application of the material balance to model the CVD apparatus, the chemical reactions are important to understand the chemical kinetics of the system, see [20].

We discuss the influence of concentration on kinetics and especially to gross reaction types.

To describe our chemical reactions, the following types are necessary:

- Autocatalytic Reactions
- Consecutive Reactions
- Parallel Reactions
- Complex Reactions

In the models we achieve the following terms with the reaction parts:

1.) Autocatalytic Reactions: Here we have the following reaction: \( A \rightarrow k_{A} P \), where \( A \) and \( P \) are the reactants.

For the reaction rates, we obtain the term: \( R_{A} = -k_{A} c_{A} \), so that our differential equation is given as

\[
\frac{\partial c_{A}}{\partial t} = -k_{A} c_{p}, \quad (3)
\]

where \( c_{A} \), \( c_{p} \) are the concentration of each species.

2.) Consecutive Reactions: Here we have the following reactions: \( A \rightarrow k_{B} B \) and \( B \rightarrow k_{C} C \), where \( A \), \( B \) and \( C \) are the reactants.

For the reaction rates, we obtain the terms: \( R_{A} = -k_{A} c_{A} \), \( R_{B} = k_{B} c_{A} - k_{B} c_{B} \) and \( R_{C} = k_{C} c_{B} \). The differential equation is given as
\[
\begin{align*}
\frac{\partial c_A}{\partial t} &= -k_AC_A, \\
\frac{\partial c_P}{\partial t} &= k_PC_A - k_EC_B, \\
\frac{\partial c_E}{\partial t} &= k_EC_B,
\end{align*}
\]

where \(c_A\), \(c_P\) and \(c_E\) are the concentration of each species.

3.) Parallel Reactions: Here we have the following types: \(A \xrightarrow{k_1} P\), \(A \xrightarrow{k_2} X\), where \(A\), \(P\) and \(X\) are the reactants. For the reaction rates, we obtain the terms: \(R_A = -(k_1 + k_2)c_A\), \(R_P = k_e c_e\) and \(R_X = k_e c_e\). The derived differential equations are given as:

\[
\begin{align*}
\frac{\partial c_A}{\partial t} &= -(k_1 + k_2)c_A, \\
\frac{\partial c_E}{\partial t} &= k_e c_e, \\
\frac{\partial c_X}{\partial t} &= k_e c_e,
\end{align*}
\]

where \(c_A\), \(c_E\) and \(c_X\) are the concentrations of each species.

4.) Complex Reactions: We deal with the following examples: \(A \xrightarrow{k_1} P \xrightarrow{k_2} Q\) with \(A \xrightarrow{k_3} X\), \(P \xrightarrow{k_4} Y\) and \(A + B \xrightarrow{k_5} P\) with \(A + P \xrightarrow{k_6} X\). Here we have combinations of parallel and consecutive reactions.

The differential equations can be solved analytically or numerically. We propose the Waveform-relaxation method, see [17], [23], because of its decomposition idea to simpler parts. We have also taken into account that we solve the operator-splitting and decouple the transport and reaction equations, see [9].

III. TIME- AND SPATIAL DISCRETIZATION METHODS

In the following we describe the time and spatial discretization methods. We consider the following ordinary linear differential equation:

\[\frac{\partial c}{\partial t} = Ac(t) + Bc(t),\]

where the initial-conditions are given as \(c^0 = c(t^0)\). The operators \(A\) and \(B\) are assumed to be spatial discretised, e.g. convection or reaction operators.

A. Time Discretization methods

For the multiple species reaction models we need to investigate numerical methods for solving the equations. Such time-dependent equations we treat with Runge-Kutta and Backward Differential Formula (BDF) and Implicit-Explicit (IMEX) methods.

Runge-Kutta, BDF and IMEX methods

For the time-discretization of our diffusion-reaction equations we take into account the stiff and non-stiff parts of the equation-parts. Because of the nonstiff diffusion part and the stiff reaction part, we propose the combination of explicit and implicit methods, see [3] and [1].

Therefore we propose the Runge-Kutta and BDF methods as adapted time-discretization methods to reach higher-order results.

For the time-discretization we use the following higher-order discretization methods.

**Runge-Kutta method**

We use the implicit trapezoidal rule:

\[
\begin{array}{c|c|c|c}
0 & 1 & 1 & 1 \\
1 & 2 & 6 & 4 \\
\end{array}
\]

Furthermore we use the following Gauss-Runge-Kutta method:

\[
\begin{array}{c|c|c|c|c}
1 & \sqrt{3} & 1 & \sqrt{3} & 1 \\
2 & 6 & 4 & 6 & 4 \\
\end{array}
\]

To use these Runge-Kutta methods with the Waveform-Relaxation method, we have to take into account that we solve equations of the form \(\partial u_t = Au + b\) in each iteration step, where \(b = Bu_{i+1}\) is the right hand side and given at the solution \(u_{i+1}\).

For the implicit trapezoidal rule this is no problem, because we do not need the values at any sub-points. However, for the Gauss method we need to know the values of \(b\) at the sub-points \(t_0 + c_i h\) and \(t_0 + c_i h\) with \(c = \left(1 - \frac{1}{6} - \frac{1}{6} \cdot \frac{1}{6}\right)^{\frac{1}{3}}\).

Therefore we must interpolate \(b\). On that account we choose the cubic spline functions.

Numerical experiments show that this works properly with non-stiff problems, but not very well with stiff problems.

B. BDF method

Because the higher-order Gauss-Runge-Kutta method combined with cubic spline interpolation does not work properly with stiff problems, we use the following BDF method of order three, which does not need any sub-points and therefore no interpolation.

The BDF3 method is defined by

\[
\frac{1}{6}u^{n+3} - 3u^{n+2} + 3u_{n+1} - 1u^n = A(u^n) + B(u^{n+3})
\]

For the pre-stepping, i.e. to obtain \(u^n\), \(u_{n+1}\), we use the implicit trapezoidal rule (10).

**Implicit-explicit methods**
The IMEX schemes have been widely used for time integration of spatial discretized partial differential equations of the diffusion-convection type. These methods are applied to decouple the implicit and explicit terms. Treating the convection-diffusion equation for example, one can use the explicit part for the convection and the implicit part for the diffusion term. In our application we divide between the stiff and non-stiff term, so we apply the implicit part for the stiff operators and the explicit part for the non-stiff operators.

**FSRK method**
We propose the A-stable fractional-stepping Runge-Kutta (FSRK) scheme, see [3], of first and second order for our applications. The tableau in the Butcher form is given as

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>88/9</td>
<td>12/5</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>45/4</td>
<td>144/50</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>75/4</td>
<td>25/15</td>
<td>0</td>
</tr>
</tbody>
</table>

To obtain second-order convergence in numerical examples it is important to split the operator in the right way as we will show later.

**SBDF Method**
We use the following stiff backward differential formula (SBDF) method, which is a modification of the third-order backward differential formula (BDF3) method. As pre-stepping method we again use the implicit trapezoidal rule.

\[
1 \left( \frac{11}{6} u^{n+1} - 3u^n + \frac{3}{2} u^{n-1} - \frac{1}{3} u^{n-2} \right) = 3A(u^n) - 3A(u^{n-1}) + A(u^{n-2}) + B(u^{n+1}).
\]

Again it is important to split the operator in the right way.

**C. Spatial Discretization methods**
We discretize the diffusion-dispersion-equation with implicit time-discretization and finite-volume method for the following equation

\[
\partial_t c + \nabla \cdot (D \nabla c) = 0
\]

The boundary-values are denoted by \( n \cdot \nabla c(c,t) = 0 \), where \( x \in \Gamma = \partial \Omega \) is the boundary, cf. [10]. The initial conditions are given by \( c(x,0) = c_0(x) \).

We integrate the equation (17) over space and time and derive

\[
\int_{\Omega} \int_t^{t+\tau} \nabla \cdot (D \nabla c) \, dx \, dt = \int_{\Omega} \int_t^{t+\tau} \nabla \cdot \left( (D \nabla c) \right) \, dx.
\]

The time-integration is done by the backward-Euler method and the diffusiondispersion term is lumped, cf. [12]

\[
\int_{\Omega}(R(c^{n+1}) - R(c^n)) \, dx = \tau \int_{\Omega} \nabla \cdot \left( (D \nabla c^{n+1}) \right) \, dx,
\]

The equation (17) is discretized over the space with respect of using the Greensformula.

\[
\int_{\Omega}(R(c^{n+1}) - R(c^n)) \, dx = \tau \int_{\Omega} \nabla \cdot \left( (D \nabla c^{n+1}) \right) \, dy,
\]

where \( \Gamma_j \) is the boundary of the finite-volume cell \( \Omega_j \). We use the approximation in space, cf. [12].

The spatial-integration for (18) is done by the mid-point rule over the finite boundaries and given as

\[
VR(c^{n+1}) - VR(c^n) = \tau \sum_{\text{area } j} \left( \sum_{\text{neighbors } k} \Gamma_{jk} \right) \nabla \cdot \left( D_{jk} \nabla c^{n+1}_{jk} \right).
\]

The gradients are calculated with the piecewise finite-element-function \( \phi_j \) and we obtain

\[
\nabla c^{n+1}_{jk} = \sum_{\text{neighbors } k} c^{n+1}_{jk} \nabla \phi_j(x_{jk}^n).
\]

We get with the difference-notation for the neighbor-point \( j \) in the following equation, cf. [10]

\[
VR(c^{n+1}) - VR(c^n) = \tau \sum_{\text{area } j} \left( \sum_{\text{neighbors } k} \Gamma_{jk} \nabla \cdot \left( D_{jk} \nabla \phi_j(x_{jk}^n) \right) \left( c^{n+1}_{jk} - n_{jk}^{n+1} \right) \right).
\]

where \( j = 1, \ldots, m \).

**IV. DECOMPOSITION METHODS**
In the following we present the different decomposition methods, with respect to our application.

**A. Operator Splitting method**
The following splitting methods we propose to decomposition in different operators. Our methods are based on sequential splitting method, see [8].

We consider the following ordinary linear differential equation:

\[
\partial_t c(t) = A c(t) + B c(t),
\]

where the initial-conditions are given as \( c^* = c(t) \). The operators \( A \) and \( B \) are assumed to be bounded linear operators in the Banach-space \( X \) with \( A, B : X \to X \). In applications the operators correspond to the physical operators, e.g. the convection- and the diffusion-operator.

The operator-splitting method is introduced as a method which solves two equation-parts sequentially, with respect to initial conditions. The method is given as following

\[
\partial_t c(t) = A c(t) + B c(t),
\]

where \( c^* = c(t) \). The operators \( A \) and \( B \) are assumed to be bounded linear operators in the Banach-space \( X \) with \( A, B : X \to X \). In applications the operators correspond to the physical operators, e.g. the convection- and the diffusion-operator.

The operator-splitting method is introduced as a method which solves two equation-parts sequentially, with respect to initial conditions. The method is given as following

\[
\partial_t c(t) = A c(t) + B c(t),
\]

where \( c^* = c(t) \). The operators \( A \) and \( B \) are assumed to be bounded linear operators in the Banach-space \( X \) with \( A, B : X \to X \). In applications the operators correspond to the physical operators, e.g. the convection- and the diffusion-operator.
\[
\frac{\partial c^*}{\partial t} = Ac^* (t), \text{ with } c^* (t^*) = c^*, \quad (23)
\]
\[
\frac{\partial c^{**}}{\partial t} = Ac^{**} (t), \text{ with } c^{**} (t^*) = c (t^{**}).
\]
where the time-step is given as \( t^* = t^{**} - t^* \). The solution of the equation (22) is \( c^{**} = c^* (t^{**}) \).

The splitting-error of the method is derived with Taylor-expansion, cf. [12]. We obtain the global error as
\[
\rho = \frac{1}{2} \tau^2 \left[ (\partial^2 c^{*2} / \partial t^2) - \exp (\tau^* (A + B)) - \exp (\tau^* (B + A)) \right] c(t^*) = \rho_1(t^*),
\]
where \( [A, B] = AB - BA \) is the commutator of \( A \) and \( B \). We get an error \( O(\tau^*) \) if the operators \( A \) and \( B \) do not commute, otherwise the method is exact.

**B. The Waveform-Relaxation Method**

A method to solve large coupled differential equations is the Waveform-Relaxation scheme. The iterative method was discussed in [23], [17] and [15]. For the method, there exist Gauss- or Jacobian schemes to decouple at least the schemes more or less effective.

We deal with the following ordinary differential equation or assume a semidiscretised partial differential equation:
\[
u_i = f (u_i, t), \text{ in } (0, T),
\]
\[
u(0) = c_0,
\]
where \( u = (u_1, u_2, ..., u_m) \) is the initial condition.

We apply the Waveform-Relaxation method for \( i = 1, ..., m \) and have:
\[
\frac{\partial u_i}{\partial t} = f_i (u_i, u_{i+1}, t), \text{ with } u_i (t^*) = u_i (t^*) \quad (25)
\]
\[
\frac{\partial u_{i+1}}{\partial t} = f_{i+1} (u_{i+1}, u_{i+2}, t), \text{ with } u_{i+1} (t^*) = u_{i+1} (t^*) \quad (26)
\]
\[
\frac{\partial u_m}{\partial t} = f_m (u_m, u_{m-1}, t), \text{ with } u_m (t^*) = u_m (t^*) \quad (27)
\]
where we have \( u_{i-1} (t) = u_i (t^*), ... u_{m-1} (t) = u_m (t^*) \) for the initialisation of the first step.

We reduce to two equations and reformulate the method to our iterative splitting methods. So we deal with:
\[
\frac{\partial u_i}{\partial t} = f_i (u_i, t) + f_{i+1} (u_{i+1}, t), \text{ in } (0, T),
\]
\[
\frac{\partial u_{i+1}}{\partial t} = f_{i+1} (u_i, t) + f_{i+2} (u_{i+2}, t), \text{ in } (0, T),
\]
\[
u(0) = c_0,
\]
where \( u = (u_1, u_2) \).

**C. Application of the Waveform-Relaxation Method**

For an effective application, we modify the Waveform-Relaxation method. We deal with the following equations:
\[
\frac{\partial u_1}{\partial t} = -\lambda u_1 + \lambda u_2 + ... + \lambda u_n \text{ in } \Omega \times (0, T),
\]
\[
\frac{\partial u_m}{\partial t} = -\lambda u_1 + \lambda u_2 + ... + \lambda u_n \text{ in } \Omega \times (0, T),
\]
\[
u(0) = (c_1, c_n) \in \Omega.
\]

The algorithm is given with semi-analytical solutions of the reaction equations
\[
u^* = \exp (-\Lambda t) \nu^0 = \Lambda^{-1} (1 - \exp (-\Lambda t)) \nu^0 \quad (39)
\]
where the matrices are given as
\[
\Lambda = \begin{pmatrix}
\lambda_1 & 0 & \cdots & 0 \\
0 & \lambda_2 & \cdots & 0 \\
\vdots & \ddots & \ddots & \vdots \\
0 & \cdots & \lambda_n & 0
\end{pmatrix}
\]
\[
\Lambda^{-1} = \begin{pmatrix}
\lambda_1^{-1} & 0 & \cdots & 0 \\
0 & \lambda_2^{-1} & \cdots & 0 \\
\vdots & \ddots & \ddots & \vdots \\
0 & \cdots & \lambda_n^{-1} & 0
\end{pmatrix}
\]
The iterative algorithm is given as
Step 1: \( \nu = (c_1 (x), ..., c_n (x)) \)
Step 2: Solve $\tilde{u}^{i+1}$ with
$$\tilde{u}^{i+1} = \exp(-\Lambda t)\tilde{u}^0 + \Lambda^{-1}(1 - \exp(-\Lambda t))\Delta u^i,$$ (44)

\[
\vdots
\]
Step k:
$$\tilde{u}^i = \exp(-\Lambda t)\tilde{u}^0 + \Lambda^{-1}(1 - \exp(-\Lambda t))\Delta u^{i-1},$$ (45)

The stop-criterion is controlled after each step:
We have the absolute error of the solution vector $|\tilde{u}^i - \tilde{u}^i+1| \leq \text{err}$
and we finish the algorithm and obtain the results:
$$u(t) = \tilde{u}^i(t).$$

Remark 2. For modification to the Waveform-Relaxation method, we have the following contributions:
1.) Gauss-Seidel Waveform-Relaxation Method:
Here we apply the lower matrix for the iteration method.
2.) Block Jacobian Waveform Relaxation Method:
Here we have $2 \times 2$ blocks for the diagonals.
3.) Block Gauss-Seidel Waveform Relaxation Method:
Here we have $2 \times 2$ blocks for the diagonals and the lower
matrix.

V. EXPERIMENT FOR THE PLASMA REACTOR
In the following experiments, we step by step introduce the model-equations for the plasma reactor.

A. Simplified Model
We deal with two species of the reaction system. Based on this small system we apply the numerical and analytical methods to get some experiences.
The decay-factors are given as $\lambda_1$ and $\lambda_2$.
We have the following equations
$$\frac{\partial}{\partial t}u_i = -\lambda_1 u_i + \lambda_2 u_i \text{ in } \Omega \times (0,T),$$ (46)
$$\frac{\partial}{\partial t}u_i = \lambda_1 u_i - \lambda_2 u_i \text{ in } \Omega \times (0,T),$$ (47)
$$u_i(0) = u_{i0}, \quad u_i(0) = u_{i20} \in \Omega,$$ (48)
where the particle densities are given as $u_i = u_i(t)$ and $u_i = u_i(t)$ in $(0, T) \in \mathbb{R}^+$. The interactions are given with $\lambda_1 = 0.1$ and $\lambda_2 = 0.01$. The initial conditions are given as $u_{i0} = 1.0$ and $c_{20} = 0.1$.
The analytical solutions are given as
$$u_i = \lambda_2 \exp\left(\frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2}\right)(-c_i),$$
$$u_2 = \exp\left(\frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2}\right)\left(-c_i\right),$$
where the values $c_0$ and $c_1$ are given as
$$c_0 = \frac{u_{i0} + u_{i20}}{1 + \frac{\lambda_2}{\lambda_1}}, \quad c_1 = u_{i20} - c_0,$$
The numerical results are given in Fig. 1.

B. Gas adsorption
In a next example we simulate the gas adsorption with a diffusion reaction equation, see [24].
equation system with three equations. We have taken into account 1000 time steps. The analytical method is based on a transformation to an eigenvalue problem and this method can be used to solve the results with the Waveform-Relaxation method.

The reaction is given as

\[
\begin{align*}
\dot{c}_1 &= -c_{k_{12}} a_1 + c_{k_{13}} a_1 + c_{k_{14}} a_1, \\
\dot{c}_2 &= -c_{k_{21}} a_2 + c_{k_{23}} a_2 + c_{k_{24}} a_2, \\
\dot{c}_3 &= -c_{k_{31}} a_3 + c_{k_{32}} a_3 + c_{k_{34}} a_3.
\end{align*}
\]

The equation is given as, cf. [16]:

\[
\begin{align*}
\dot{c}_1 &= -(k_{12} + k_{13} + k_{14}) c_1 + k_{12} c_2 + k_{13} c_3, \\
\dot{c}_2 &= -(k_{21} + k_{23} + k_{24}) c_2 + k_{21} c_1 + k_{23} c_3, \\
\dot{c}_3 &= -(k_{31} + k_{32} + k_{34}) c_3 + k_{31} c_1 + k_{32} c_2.
\end{align*}
\]

We solve the reaction with analytical methods, cf. [16] with three species.

We have the following parameters:

\[
c_1 = c_2 = c_3 = 1/3,
\]

\[
k_{12} = 0.5, k_{13} = 0.5, k_{14} = 0.25,
\]

\[
k_{21} = 0.05, k_{23} = 0.3, k_{24} = 0.15.
\]

The end-time is given as \( T = 10.0 \).

In Fig. 3 we have simulated the three species. The end concentration of the species results in a loss of concentration of \( A_1 \) (first species), a nearly balance of concentration of \( A_2 \) (second species) and a tremendous increase of concentration of \( A_3 \) (third species). So at least the reaction on this end situation has taken place from \( A_1 \) to \( A_3 \).

D. Real-life example: Sputtering Reactions

In the following experiment, we deal with the sputtering reactions which is discussed in [4].

The CVD-process has different chemical reactants, which react and decay in different time periods. We concentrate on a simple sputtering reaction, which has at least seven reactants.

The reactions are given as

\[
\begin{align*}
\lambda_\beta &\rightarrow A_B \rightarrow \lambda_{(1-\beta)} \rightarrow A_C \rightarrow \lambda_{(1-\sigma)} \rightarrow A_F \\
\lambda_{(1-\beta)} &\rightarrow \lambda_{\sigma} \rightarrow \lambda_{(1-\eta)} \\
A_D &\rightarrow A_E \rightarrow \lambda_{\eta} \rightarrow \lambda_{\epsilon} \\
\end{align*}
\]

We have the following reaction equations:

\[
\begin{align*}
\dot{c}_1(c_{\text{tot}}) &= -\lambda_{1} c_{\text{tot}}, \\
\dot{c}_2(c_{B}) &= \lambda_{1} (1-\beta) c_{\text{tot}}, \\
\dot{c}_3(c_{B}) &= \lambda_{2} \beta c_{\text{tot}} - \lambda_{2} c_{B}, \\
\dot{c}_4(c_{B}) &= \lambda_{2} (1-\sigma) c_{B} - \lambda_{2} c_{C}, \\
\dot{c}_5(c_{C}) &= \lambda_{3} \sigma c_{B}, \\
\dot{c}_6(c_{C}) &= \lambda_{3} (1-\eta) c_{C}, \\
\dot{c}_7(c_{F}) &= \lambda_{4} \eta c_{C}.
\end{align*}
\]

where \( c_{\text{tot}} \) is the total mass of the particles coming from the substrate. \( c_A, c_B, c_C, c_D, c_E \), and \( c_F \) are intermediate masses of the particles, only \( c_A \) and \( c_F \) arrive at the targets.

We have the following reaction-matrix:

\[
A = \begin{pmatrix}
-\lambda_{1} & 0 & 0 & 0 & 0 & 0 & 0 \\
\lambda_{1} (1-\beta) & 0 & 0 & 0 & 0 & 0 & 0 \\
\lambda_{2} \beta & 0 & -\lambda_{2} & 0 & 0 & 0 & 0 \\
0 & 0 & \lambda_{2} (1-\sigma) & -\lambda_{3} & 0 & 0 & 0 \\
0 & 0 & 0 & \lambda_{3} \sigma & 0 & 0 & 0 \\
0 & 0 & 0 & \lambda_{3} (1-\eta) & -\lambda_{4} & 0 & 0 \\
0 & 0 & 0 & \lambda_{4} \eta & 0 & 0 & 0 \\
\end{pmatrix}
\]

We apply an analytical method, cf. [16].

We have the following parameters:

\[
c_{A,B} = \ldots = c_{F,0} = 0.1, \\
c_{A,0} = 1.0, \\
\lambda_{1} = \lambda_{2} = 0.1, \lambda_{3} = 0.05, \\
\beta = 0.1, \sigma = 0.5, \eta = 0.9
\]

The end-time is given as \( T = 10.0 \).

In Fig. 4 we have simulated the sputtering reactions. In the results, we obtain the arrival of the \( c_A \) and \( c_F \) mass particles.

The first deposition of \( c_{\text{tot}} \) results in \( c_A \). The second deposition of \( c_{\text{tot}} \) results, due to the reaction chain, in \( c_F \) and is tremendous decreasing. In our final time situation we have approximately 0.8 amount of the initial concentration that rests on the targets. As a result we lose nearly 20 % of the substrate and should additionally increase our source substrate to 1.2 amounts.

VI. CONCLUSIONS AND DISCUSSIONS

We present a plasma model for thin layers. The models are given as diffusion-reaction equations. The efficient Waveform-Relaxation method is used for the chemical reaction equations. We discuss various applications of chemical reactions. The results verify the theoretical investigations and we simulate a real life apparatus with sputtering reactions.

REFERENCES


Fig.1 Experiment of a two species reaction, \( \lambda_1 = 0.1 \), \( \lambda_2 = 0.01 \), upper figure: \( c_1 \), middle figure: \( c_2 \), lower figure: \( c_1 \) (red), \( c_2 \) (green).
Fig. 2 Experiment with the gas-sorption of two species, upper figure: $C_d$, lower figure: $C_{P'}$.

Fig. 3 Experiment of a three species reaction: $\alpha_1$ (green), $\alpha_2$ (red), $\alpha_3$ (blue).

Fig. 4 Experiment of a seven species reaction: upper figure: $c_d$ (green), $c_f$ (red); lower figure: $c_{tot}$ (green), $c_a$ (red), $c_b$ (blue), $c_c$ (yellow), $c_D$ (magenta), $c_E$ (black), $c_F$ (cyan).