UNIVERSITY OF LATVIA

Summary of Doctoral Thesis

Maksims Marinaki

PARAMETER OPTIMIZATION AND PATTERN RECOGNITION FOR COMBUSTION AND REACTION KINETICS APPLICATIONS

Riga 2021

UNIVERSITY OF LATVIA

FACULTY OF PHYSICS, MATHEMATICS AND OPTOMETRY

Maksims Marinaki

PARAMETER OPTIMIZATION AND PATTERN RECOGNITION FOR COMBUSTION AND REACTION KINETICS APPLICATIONS

SUMMARY OF DOCTORAL THESIS

Submitted for the degree of Doctor of Mathematics
Subfield of Mathematical Modelling

Riga 2021

The doctoral thesis was carried out at the Department of Mathematics, Faculty of Physics, Mathematics and Optometry, University of Latvia in time period from 2016 to 2021.

The thesis contains the introduction, 5 chapters, conclusions and the list of references.

Form of the thesis: Dissertation in Mathematics, Subdiscipline of Mathematical Modelling.

Scientific supervisor: Dr. math. prof. Uldis Strautiņš.

Dissertation Reviewers:

1. Raimondas Čiegis, Dr. habil. prof., Vilnius Tech / VGTU, Lithuania
2. Andrejs Koliškins, Dr. math. prof., Riga Technical University
3. Oleg Iliev, Ph. D. in mathematics and physics, prof., Fraunhofer ITWM, Germany

The thesis will be defended at the public session of the Doctoral Committee of Mathematics, University of Latvia, House of Science, Jelgavas Street 3, Riga.

The thesis is available in the Library of the University of Latvia, Raiņa Blvd. 19, Room 203.

Chairman of the Doctoral Committee: Dr. math. prof. Andrejs Cibulis.
Secretary of the Doctoral Committee: Dr. math. doc. Sergejs Smirnovs.

ISBN 978-9934-18-757-5
ISBN 978-9934-18-758-2 (PDF)

UNIVERSITY OF LATVIA

Abstract

Department of Mathematics
Faculty of Physics, Mathematics and Optometry

Doctor of Philosophy

Parameter optimization and pattern recognition for combustion and reaction kinetics applications

by Maksims MARINAKI

In this thesis we consider the simplified combustion models and describe the chemical kinetics via different reaction mechanisms. We solve the resulting problems by the finite element techniques. Then we consider some model reduction techniques and verify the results with the experimental ones by doing the parameter optimization. The leftover patterns of the optimization method are stored as fixed points of the discrete dynamical system in order to be reproduced when the experimental data is modified.

Keywords: combustion, chemical kinetics, mathematical modelling, finite element method, PSO optimization, model reduction, pattern storage.

Acknowledgements

Author would like to acknowledge the financial support from the Latvian Research Cooperation Project of the Latvian Council of Science Nr. 623/2014, the ERAF project Nr. 1.1.1.1/16/A/004 and SAM project Nr. 8.2.2.0/18/A/010.

Author would like to express his sincere gratitude to his thesis supervisor Uldis Strautiņš, project supervisor Harijs Kalis, colleagues Andrejs Reinfelds, Andrejs Cibulis and all the others who supported the accomplishment of the thesis.

Contents

List of Symbols

q	scalar quantity (unless stated vectorial)
$\mathbf{v}$	vectorial quantity/column-vector
N	number of species/PDEs/vector-function's components
M	number of chemical reactions
d	number of spatial variables
n	finite dimensional function space's dimension/number of experimental quantities/any natural number
D	mixture molecular diffusivity/ the dimension of the pattern
S	number of particles in a swarm
D_p	number of parameters
L	number of snapshots
K	number of elements in the reduced basis
P	number of patterns
N_E	number of experiments
N_S	number of recalculation stages
N	number of iterations
T	number of triangles
t	time variable/triangle matrix
p	node matrix/any natural number
$j = 1..M$	for j from 1 till M
$\frac{\mathrm{d}u}{\mathrm{d}t}$, $u'(t)$, u'	first derivative

$\frac{\partial u}{\partial t}$, u_t, $\partial^i u$	partial derivative
$\frac{\mathrm{D}}{\mathrm{D}t}$	total derivative
∇	nabla operator
$\nabla\cdot$	divergence operator
$\mathbf{u}\cdot\mathbf{v}$, $(\mathbf{u},\mathbf{v})$	dot product
$\lVert\cdot\rVert$	element norm (type specified)
$\lVert\cdot\rVert_2$	$\mathbb{R}^n$ element Euclidean norm
$L(\cdot)$	differential operator
$\mathbf{v}^T$	transpose vector/matrix
det	determinant of a matrix
E	unit matrix
$\mathbb{N}$	set of natural numbers
$\mathbb{N}_0$	set of natural numbers and zero
$\mathbb{R}$	set of real numbers
$\mathbb{R}^+$	set of positive real numbers
$\partial\Omega$	set boundary
$\Omega_1\cup\Omega_2$	set union
$\Omega_1\cap\Omega_2$	set intersection
$\overline{\Omega}$	set closure
$\Omega_1\times\Omega_2$	set Cartesian product
$\mathrm{int}\Omega$	set interior
$\emptyset$	empty set
$\inf\Omega$	set infimum
H, $\mathcal{H}$, G, $\mathcal{G}$	function space/Hilbert space
C^∞	space of infinitely many times differentiable functions
C_0^∞	space of infinitely many times differentiable functions with compact support
$H^{-1}(\Omega)$	the dual space
I, I_h	interpolation operator

$\dim H$	space dimension
Π_m	space of m-th order polynomials
$\sim U(a, b)$	uniformly distributed random numbers
$\mathrm{tr} A$	trace operator
$\mathbf{e_i}$	i-th unit vector
$\mathrm{diag}[A_1, \ldots, A_n]$	block diagonal matrix
$\vee$	logical OR
$:=$	is defined as
$\Rightarrow$	logical consequence (implication)
$\widehat{x}$	solution representative (snapshot)
x^p	p-th vector (pattern)

Abbreviations:

ODE	Ordinary Differential Equation
PDE	Partial Differential Equation
BCs	Boundary Conditions
FEM	Finite Element Method
PSO	Particle Swarm Optimization
SVD	Singular Value Decomposition
PCA	Principal Component Analysis
DDS	Discrete Dynamical System
PC	Pattern Creation
Prop.	proposition (lemma or theorem)
Def.	definition
Alg.	algorithm
cgs	centimetre–gram–second system of units

Chapter 1

Introduction

The idea to write thesis on this particular topic has been a direct consequence of the collaboration between two research institutes: Institute of Mathematics and Computer Science of University of Latvia (the one where the author's employed) and the Institute of Physics of University of Latvia. In the latter institution the researchers have been widely interested in topics such as an efficient combustion of biomass for a very long time and been facilitating the experiments involving the combustion and gasification of different fuels.

In [Mar18b] and [Mar19] the modelling and experimental results have been published for experiments such as straw co-firing with peat or propane.

In [Mar18a] and [Mar16] for similar experiments the impact of electric and magnetic fields have been discussed and corresponding experiments have been facilitated.

In [Vey05] the modern theory and applications of combustion and gasification processes is introduced. More classical theory is found in [Wil85] and [Bor01].

These processes are mostly modelled by systems of partial differential equations (PDEs) and the choice of suitable numerical methods along with the theory of PDEs is to be found in [Hac17], [Ben13], [Ang03].

The optimization of parameters, mainly the technical part of the process itself for any optimization problem found in [BAE05], [Cle06].

When the problem becomes a high dimensional one, the principal component analysis (PCA) comes to rescue and the results compatible with the models to be proposed in this thesis are found in [Wil10], [Gre08]. Another aspect is the pattern recognition out of high dimensional problem in order to minimize number of calling out the costly procedure of obtaining the numerical solution thereof and general theory and applications of recurrent networks can be found in [Cha01].

The main idea of this thesis is to develop the mathematical model that governs the main basic physical processes, occurring in the experiments, facilitated by the team, and at the same time uses the experimental data in order to improve itself by using several modern techniques.

If one wants the concise formulation of thesis' aims, these would be:

1) Find a way how to model chemical reactions with PDEs. The models should be as simple as possible, but in two or three dimensions and the underlying chemistry should be described thoroughly and the transition between the design of chemical reaction and the design of PDE should be visible (Chapter 2).

2) Provide the necessary mathematics for the chosen set of models - the existence and uniqueness results, briefly discuss approximation and choose the numerical methods - universal for the obtained models (Chapter 3).

3) Develop concepts of measure between experimental and modelling data and formulate all the necessary problems (Chapter 4).

4) Perform several numerical experiments with either solution obtaining process as it is or the parameter optimization with the pattern storage procedures - as proposed in the abstract (Chapters 5-6).

In the end we formulate the conclusions on what has been done, provide the list of literature and list of publications and conferences the author has been involved in.

Chapter 2

Combustion models

Whilst participating in projects, author had to analyse several books containing mathematical models of combustion process and reacting flows. [Vey05] along with its author's lecture notes gave the best insight on topic and is the most recent and decent one containing all the modern theory and applications. More classical ones such as [Wil85] and [Bor01] were of a great value for achieving goals of our researches as well.

2.1 Equations solved for mass fractions

We use the notation Y_k for the mass fraction of k-th specie in the mixture and define it as a partial density and mixture of gases density ratio $Y_k := \dfrac{\rho_k}{\rho}$.

The species equation for the specie $Y_k,\ k = 1..N$, has its simplified form:

$$\rho \frac{\mathrm{D}Y_k}{\mathrm{D}t} = \nabla \cdot (\rho D \nabla Y_k) + \dot{\omega}_k. \tag{2.1}$$

The derivation of this form can be found in the main thesis variant. We have used the notation

$$\frac{\mathrm{D}Y_k}{\mathrm{D}t} := \frac{\partial Y_k}{\partial t} + \mathbf{u} \cdot \nabla Y_k \tag{2.2}$$

and D is the molecular diffusivity in the mixture, ρ is the density as an amount of kilograms per cubic meter and $\mathbf{u}$ represents the velocity field of the reacting flow.

In the next section we pin down the reaction term $\dot{\omega}_k$.

2.2 The reaction term

We wish to write down the reaction term $\dot{\omega}_k$ as a sort of a template which holds for each reaction mechanism and to be fulfilled with the parameters that are different for each mechanism. We split it into different parts.

The term $\dot{\omega}_k$ is the sum over all reactions:

$$\sum_{j=1}^{M} \dot{\omega}_{kj}, \tag{2.3}$$

where $\dot{\omega}_{kj}$ is expressed as $\dot{\omega}_{kj} = q_j w_k \nu_{kj}$.

The explanation of this form starts with the ν_{kj} multiplier. This is constructed as following: we consider M reactions in our model, through which N species are reacting. We consider forward reactions here and each j-th reaction, $j = 1..M$, usually has the form

$$\sum_{k=1}^{N} \nu'_{kj} M_k \to \sum_{k=1}^{N} \nu''_{kj} M_k, \ j = 1..M, \tag{2.4}$$

where ν_{kj} is the number of moles of the k-th specie in the j-th reaction, and M_k is the nomenclature of the corresponding specie.

Then we denote $\nu_{kj} := \nu''_{kj} - \nu'_{kj}$. The w_k is the molecular weight of specie k, while the q_j is the multiplier that covers the *Arrhenius law.*

For the forward reaction it has the following form:

$$q_j = K_j \prod_{k=1}^{N} N_k^{\nu'_{kj}}, \tag{2.5}$$

$$K_j = A_j e^{\frac{-E_j}{RT}} T^{\beta_j}. \tag{2.6}$$

Here N_k is the concentration of the specie k as the amount of moles per cubic meter. The reactivity parameters for the reaction j are the pre-exponential factor A_j, the activation energy E_j and the temperature exponent β_j and are either to be determined from the existing tables or their values would be estimated and afterwards optimized which is the course of chapter 4. R is the ideal gas constant.

There's an explicit temperature dependence and an implicit mass fraction dependence in (2.5),(2.6). The concentration - mass fraction relation to use is

$$N_k = \frac{\rho Y_k}{w_k}. \tag{2.7}$$

Thus we have a framework where one takes the reaction mechanism as a list of formulas (2.4), constructs the matrix ν by doing the arithmetics to a stoichiometry coefficients and in a very convenient manner plugs everything into formula for the reaction term along with the parameters to be discussed in chapter 4.

2.3 Temperature equation

The temperature equation has its simplified form. The derivation of this form can be found in main thesis variant.

$$\rho c_p \frac{\mathrm{D}T}{\mathrm{D}t} = \nabla \cdot (\lambda \nabla T) + \omega'_T, \tag{2.8}$$

where ρ, c_p and λ are respectively mass density, heat capacity and the thermal conductivity.

The source term for this equation has the form [Vey05]

$$\omega'_T = -\sum_{k=1}^{N} \triangle H_{f,k} \dot{\omega}_k, \tag{2.9}$$

where $\triangle H_{f,k}$ stands for the enthalpy of formation.

The enthalpy of formation is to be found in the table for each specie in the reacting mixture, $\dot{\omega}_k$ multiplier is in form (2.3).

At the end of the day we are going to work with the reaction-diffusion partial differential equation system (2.1), (2.8).

In the next chapter we are going to address some results of mathematical physics and some numerics for problems of mathematical physics.

Chapter 3

Partial differential equations and the FEM

The description of physical laws in their differential forms in the previous chapter 2 led to parametrized system of partial differential equations (2.1), (2.8). The present chapter provides the definitions and the common classification of the partial differential equations (PDEs). Then we address the aspects of the solution space discretization which leads to an equivalence of the discretized problem to a problem of linear algebra.

Author mostly inspired by books [Fou03], [Hac17], [Ben13], [Ang03].

3.1 The classification of PDEs

We shall start with the classification of PDEs which is required for choosing the right tool to solve it either analytically or numerically. Classification provided here built in correspondence with the classification of second order curves in the analytical geometry.

We consider the second order linear partial differential equation

$$Lu = -\nabla \cdot (A\nabla u) + \mathbf{b}^T \nabla u + cu = f, \tag{3.1}$$

where the unknown function $u : \ \Omega \to \mathbb{R}$, $A \in \mathbb{R}^{d \times d}$ and elements of vector $\mathbf{b}$ along with c and f - are bounded functions $\Omega \to \mathbb{R}$, $\Omega \subset \mathbb{R}^d$, $d \in \mathbb{N}$.

The classification of (3.1) depends only on the *principal part* of the equation, i.e $-\nabla \cdot (A\nabla u)$.

Def. 3.1.1. *We call an equation (3.1) elliptic at point* $\mathbf{x}$ *if all the eigenvalues of* A $\lambda_i > 0, \ i = 1..d$ *or* $\lambda_i < 0, \ i = 1..d$ *at* $\mathbf{x}$.
We call an equation (3.1) parabolic at point $\mathbf{x}$ *if all but one eigenvalues of* A $\lambda_i > 0, \ i = 1..d-1$ *or* $\lambda_i < 0, \ i = 1..d-1$ *whilst* $\lambda_d = 0$ *at* $\mathbf{x}$.

Def. 3.1.2. *We call an equation (3.1) elliptic/parabolic in Ω if it's elliptic/parabolic $\forall \mathbf{x} \in \Omega$.*

Def. 3.1.3. *We call a differential operator in (3.1) uniformly elliptic if $\xi^T A \xi \geq \gamma \|\xi\|^2$, $\gamma > 0$, $\xi \in \mathbb{R}^d$. The matrix A in this case is called uniformly positive definite.*

Remark 3.1.1. *The equations (2.1) and (2.8) in two or three spatial variables are elliptic if made stationary, linearised and decoupled since A is in form cE, $c \in \mathbb{R}^+$ - a positive function of positive physical constants.*

Remark 3.1.2. *The equations (2.1) and (2.8) in two or three spatial and one time variable are parabolic when linearised and decoupled since there's no second order derivatives with respect to time involved, thus A has a zero-valued row and column.*

3.2 Boundary and initial value problems

Well-posedeness of a problem containing equation (3.1) depends on an appropriate choice of boundary conditions. Several types of boundary conditions are going to be considered. Further definitions will deal with a boundary $\partial\Omega$ of a certain degree of smoothness - a *Lipshitz boundary*.

Def. 3.2.1. *We say that a bounded domain $\Omega \subset \mathbb{R}^d$ has a Lipshitz boundary, if*

$$\forall x \in \partial\Omega \;\; \exists B(x) \subset O_i :$$

$$O_i \cap \Omega = O_i \cap \Omega_i, \tag{3.2}$$

where $B(x)$ is a sphere, O_i, $i = 1..m$ - open sets,

$$\Omega_i = \{(x_1, x_2) \in \mathbb{R}^d : x_1 \in \mathbb{R}^{d-1}, x_2 \in \mathbb{R}, x_2 < \phi_i(x_1)\}, \tag{3.3}$$

ϕ_i, $i = 1..m$ - Lipshitz functions:

$$\exists L > 0 \;\; \forall x, y \in \partial\Omega \;\; |\phi_i(x) - \phi_i(y)| \leq L|x - y|. \tag{3.4}$$

Now that we have a boundary, the procedure is to define two types of boundary conditions separately and afterwards mix them together in order to obtain the model-type problem suitable for applications to be discussed further on.

Def. 3.2.2. *We call a problem*

$$\begin{cases} Lu = -\nabla \cdot (A\nabla u) + \mathbf{b}^T \nabla u + cu = f, \Omega, \\ u = 0, \;\; \partial\Omega \end{cases} \tag{3.5}$$

a Dirichlet boundary value problem (BVP) for an elliptic PDE.

Def. 3.2.3. *We call a problem*

$$\begin{cases} Lu = -\nabla \cdot (A\nabla u) + \mathbf{b}^T \nabla u + cu = f, \Omega, \\ A\nabla u \cdot \widehat{n} = h, \quad \partial\Omega \end{cases} \tag{3.6}$$

a Neumann boundary value problem (BVP) for an elliptic PDE.

Here $\widehat{n}$ is an outer normal of Ω, h is a scalar function of spatial variables.

The Dirichlet boundary conditions here and in further claims are considered homogeneous. Any inhomogeneity can be eliminated by redefining the right hand side of the problem by considering the auxiliary function.

For one and the same problem one can prescribe Dirichlet and Neumann boundary conditions on different portions of the boundary Γ_D and Γ_N respectively such that $\Gamma_D \cap \Gamma_N = \emptyset$ and $\bar{\Gamma}_D \cup \bar{\Gamma}_N = \partial\Omega$.

Def. 3.2.4. *We call a problem*

$$\begin{cases} Lu = -\nabla \cdot (A\nabla u) + \mathbf{b}^T \nabla u + cu = f, \Omega, \\ u = 0, \quad \Gamma_D, \\ A\nabla u \cdot \widehat{n} = h, \quad \Gamma_N \end{cases} \tag{3.7}$$

a mixed boundary value problem (BVP) for an elliptic PDE.

Instead of considering zero row and column in matrix A due to nature of applications of parabolic equations such as heat and diffusion equations the first order derivative with respect to time is written down as a separate term and not a part of the gradient term.

In similar fashion we define the problem of finding the solution $u(t, \mathbf{x})$ of parabolic equation with mixed boundary conditions.

Def. 3.2.5. *We call a problem*

$$\begin{cases} \dfrac{\partial u}{\partial t} + Lu = f, (0, T) \times \Omega, \\ u(t, \mathbf{x}) = 0, \quad (0, T) \times \Gamma_D, \\ A\nabla u \cdot \widehat{n} = h, \quad (0, T) \times \Gamma_N, \\ u(0, \mathbf{x}) = u_0, \quad \Omega \end{cases} \tag{3.8}$$

an initial boundary value problem for parabolic PDE with mixed boundary conditions.

Such approach lets us build the theory in such fashion that we start with the elliptic part and regarding the time-dependence the problem gets split into sequence of such problems.

In theory of classical solutions the smoothness required for the solution is such as second order derivatives exist in the interior or the domain Ω. Numerics we are going to apply usually deal with solutions of less smoothness requirements. In the next section we address the weak formulations of boundary value problems (3.7), (3.8) that seek the solutions with weakened requirements and discuss the solvability afterwards.

3.3 Weak formulations

For elliptic PDEs the aim is to reformulate the BVP (3.7) in operator equation form

$$a(u,v) = f(v) \tag{3.9}$$

in appropriate function spaces for u and v elements.

For parabolic PDEs the aim is to reformulate the IBVP (3.8) in operator equation form

$$\frac{\mathrm{d}}{\mathrm{d}t}(u,v) + a(u,v) = f(v) \tag{3.10}$$

in appropriate function spaces for u and v elements.

In this section we set and examine the nature of right and left hand sides of these equations.

Def. 3.3.1. *We call a mapping*
$a(u,v): \mathcal{H} \times \mathcal{H} \to \mathbb{R}$ *a bilinear form, if the following linearity axioms hold:*

$\forall \alpha \in \mathbb{R}\ \forall u, v, u_1, v_1 \in \mathcal{H}:$

1. $a(\alpha u, v) = a(u, \alpha v) = \alpha a(u,v)$,
2. $a(u + u_1, v) = a(u,v) + a(u_1, v)$,
3. $a(u, v + v_1) = a(u,v) + a(u, v_1)$.

Def. 3.3.2. *We call a mapping* $f(v): \mathcal{H} \to \mathbb{R}$ *a linear functional, if the following linearity axioms hold:*

$\forall \alpha \in \mathbb{R}\ \forall v, v_1 \in \mathcal{H}$:

1. $f(\alpha v) = \alpha f(v)$,
2. $f(v + v_1) = f(v) + f(v_1)$.

First we obtain the weak formulation for BVP (3.7).

Def. 3.3.3. *We call a problem:*

Find the function $u \in \mathcal{H}$, that satisfies

$$a(u, v) = f(v) \;\; \forall v \in \mathcal{G}, \tag{3.11}$$

where $a(u, v)$ - bilinear form, $f(v)$ - linear functional,
a weak formulation of a problem (3.7).

There are several existence requirements on $a(u, v)$ and $f(v)$. We ask u and v along with their partial derivatives to be square integrable.

Def. 3.3.4. *For $1 \leq p < \infty$, the space $L^p(\Omega)$ is defined as follows:*

$$L^p(\Omega) = \{u : \int\limits_{\Omega} |u(x)|^p \mathrm{d}\mathbf{x} < \infty\}. \tag{3.12}$$

We define a norm in space $L^p(\Omega)$:

$$\|u\|_{L^p} := \|u\|_{0,p} := (\int\limits_{\Omega} |u(x)|^p \mathrm{d}x)^{\frac{1}{p}}. \tag{3.13}$$

The space $L^2(\Omega)$ is a Hilbert space when equipped with a scalar product

$$(u, v)_{L^2} := (u, v)_0 := \int\limits_{\Omega} uv \mathrm{d}\mathbf{x}. \tag{3.14}$$

Def. 3.3.5. *We call a function $w \in L^2(\Omega)$ a weak derivative of a function $u \in L^2(\Omega)$ if*

$$(u, \partial^\alpha v)_0 = (-1)^{|\alpha|}(w, v)_0 \;\; \textit{for each test function} \;\; v \in C_0^\infty(\Omega), \tag{3.15}$$

where $\alpha = (\alpha_1, \ldots, \alpha_n)$ - multi-index, $|\alpha| := \sum\limits_{i=1}^{n} \alpha_i$, $\partial^\alpha v := \dfrac{\partial^{\alpha_1}}{\partial_{x_1}^{\alpha_1}} \cdots \dfrac{\partial^{\alpha_n}}{\partial_{x_n}^{\alpha_n}}$.

The defined derivatives of u exist inside of the integration operator with respect to test functions. The other concise notation to use is $D^\alpha u$. Piecewise linear functions we are about to consider when dealing with finite element procedures are weakly but not strongly differentiable.

Def. 3.3.6. *For $m > 0$ and $p \geq 1$, Sobolev space $W^{m,p}(\Omega)$ is defined as follows:*

$$W^{m,p}(\Omega) = \{u : u \in L^p(\Omega),\; D^\alpha u \in L^p(\Omega),\; |\alpha| \leq m\}. \tag{3.16}$$

The space $W^{m,p}(\Omega)$ is a normed spaced when equipped with a norm:

$$\|u\|_{m,p} := (\sum_{|\alpha|\leq m} \|D^\alpha u\|_{0,p}^p)^{\frac{1}{p}}.$$

The space $W^{m,2}(\Omega)$ is Hilbert space, when equipped with a scalar product

$$(u,v)_m := \sum_{|\alpha|\leq m} (D^\alpha u, D^\alpha v)_0, \tag{3.17}$$

and the notation often used is $H^m(\Omega) := W^{m,2}(\Omega)$.

Def. 3.3.7. *The space $H^1(\Omega)$ defined as follows:*

$$H^1(\Omega) = \{u : u \in W^{1,2}(\Omega)\}. \tag{3.18}$$

The space H^1 contains the square integrable functions along with their square integrable weak derivatives.

Now we are ready for the technical part of obtaining the weak formulation of (3.7). Both sides of its equation

$$-\nabla \cdot (A\nabla u) + \mathbf{b}^T \nabla u + cu = f$$

get multiplied by the sufficiently smooth test function v. The necessary order of smoothness of u and v to be determined later.

The resulting expression

$$-\nabla \cdot (A\nabla u)v + \mathbf{b}^T \nabla uv + cuv = vf \tag{3.19}$$

we integrate over Ω:

$$-\int_\Omega \nabla \cdot (A\nabla u)v\mathrm{d}\mathbf{x} + \int_\Omega \mathbf{b}^T \nabla uv\mathrm{d}\mathbf{x} + \int_\Omega cuv\mathrm{d}\mathbf{x} = \int_\Omega fv\mathrm{d}\mathbf{x}. \tag{3.20}$$

We apply partial integration for the principal part $-\int_\Omega \nabla \cdot (A\nabla u)v\mathrm{d}\mathbf{x}$ such that there's no longer second order smoothness requirement.

For partial integration the divergence theorem is required in order reduce the smoothness order.

Prop. 3.3.1. *(Divergence theorem. For proof see [Pfe12].) For the domain $\Omega \subset \mathbb{R}^d$ with a boundary $\partial\Omega$, fulfilling requirements (3.2.1) and the continuously differentiable vector* *field* $\mathbf{u}$ *the following holds:*

$$\int_{\Omega} \nabla \cdot \mathbf{u}\, \mathrm{d}\mathbf{x} = \int_{\partial\Omega} (\mathbf{u}^T \widehat{n}) \mathrm{d}\sigma,$$

where $\widehat{n}$ *is the outer normal of* Ω*.*

In order to extend the notion of function's restriction to the boundary for Sobolev functions, we need the following result.

Prop. 3.3.2. *(Trace mapping theorem. For proof see [Fou03].) For the domain* $\Omega \subset \mathbb{R}^d$ *with a boundary* $\partial\Omega$*, fulfilling requirements (3.2.1) there exists a unique continuous linear mapping*
$\gamma_0 : H^1(\Omega) \to L^2(\partial\Omega)$ *such that for all* $u \in C^1(\overline{\Omega})$ *the following holds:*

$$\gamma_0(u) = u|_{\partial\Omega}.$$

Prop. 3.3.3. *(Partial integration of the principal part)*
Under the conditions (3.18) for ∇u *and* v *and a boundary* $\partial\Omega$*, fulfilling requirements (3.2.1), we can integrate the principal part of (3.1) as follows:*

$$-\int_{\Omega} \nabla \cdot (A\nabla u) v \mathrm{d}\mathbf{x} = -\int_{\partial\Omega} v(A\nabla u) n \mathrm{d}\sigma + \int_{\Omega} \nabla v^T A \nabla u \mathrm{d}\mathbf{x}. \tag{3.21}$$

Proof. Presented in the main paper variant. □

The conditions for weak formulation (3.7) thus can be smoothened to $u,\ v \in H^1$.

The operator equation for the weak formulation in problems (3.5), (3.6), (3.7) now reads:

$$-\int_{\partial\Omega} v(A\nabla u) n \mathrm{d}\sigma + \int_{\Omega} \nabla v^T A \nabla u \mathrm{d}\mathbf{x} + \int_{\Omega} b^T \nabla u v \mathrm{d}\mathbf{x} + \int_{\Omega} c u v \mathrm{d}\mathbf{x} = \int_{\Omega} f v \mathrm{d}\mathbf{x}. \tag{3.22}$$

For Dirichlet problem (3.5)

$$\begin{aligned} a(u,v) &= \int_{\Omega} \nabla v^T A \nabla u \mathrm{d}\mathbf{x} + \int_{\Omega} b^T \nabla u v \mathrm{d}\mathbf{x} + \int_{\Omega} c u v \mathrm{d}\mathbf{x}, \\ f(v) &= \int_{\Omega} f v \mathrm{d}\mathbf{x}. \end{aligned} \tag{3.23}$$

For Dirichlet problems we want boundary values to be incorporated into space definition. Therefore we define the space for functions u and v for the homogeneous Dirichlet problem.

Def. 3.3.8. *The space $H_0^1(\Omega)$ is defined as follows:*

$$H_0^1(\Omega) = \{u : u \in H^1(\Omega);\ \ \gamma_0(u) = 0, \partial\Omega\}.$$

We are all set for the weak formulation for Dirichlet problem:

Find function $u \in H_0^1(\Omega)$ that satisfies the equation

$$a(u,v) = f(v)\ \ \forall v \in H_0^1, \tag{3.24}$$

where $a(u,v)$, $f(v)$ are in form (3.23).

Boundary conditions for Neumann problems (3.6) should be incorporated in the right-hand functional:

$$\begin{aligned} a(u,v) &= \int_\Omega \nabla v^T A \nabla u \mathrm{d}\mathbf{x} + \int_\Omega b^T \nabla u v \mathrm{d}\mathbf{x} + \int_\Omega c u v \mathrm{d}\mathbf{x}, \\ f(v) &= \int_\Omega f v \mathrm{d}\mathbf{x} + \int_{\partial\Omega} h \gamma_0(v) \mathrm{d}\sigma. \end{aligned} \tag{3.25}$$

Thus the weak formulation for Neumann problem (3.6) reads:

Find function $u \in H^1(\Omega)$, that satisfies the equation

$$a(u,v) = f(v)\ \ \forall v \in H^1, \tag{3.26}$$

where $a(u,v)$, $f(v)$ are in form (3.25).

Now we are ready to mix two together and in similar fashion obtain the weak formulation for the mixed problem (3.7).

The boundary conditions for mixed problems should be incorporated in the right-hand functional and into function space:

$$\begin{aligned} a(u,v) &= \int_\Omega \nabla v^T A \nabla u \mathrm{d}\mathbf{x} + \int_\Omega b^T \nabla u v \mathrm{d}\mathbf{x} + \int_\Omega c u v \mathrm{d}\mathbf{x}, \\ f(v) &= \int_\Omega f v \mathrm{d}\mathbf{x} + \int_{\Gamma_N} h \gamma_0(v) \mathrm{d}\sigma. \end{aligned} \tag{3.27}$$

The space H for the solution and the test functions is defined as follows:

$$H = \{u \in H^1(\Omega),\ \gamma_0(u) = 0, \Gamma_D\}. \tag{3.28}$$

Thus the weak formulation for the mixed boundary value problem (3.7) reads:

Find function $u \in H$, that satisfies the equation

$$a(u,v) = f(v) \ \ \forall v \in H, \tag{3.29}$$

where $a(u,v)$, $f(v)$ are in form (3.27).

The reason for considering this type of problem is that exactly this type of boundary conditions is going to be considered mostly in the applications section of this thesis.

For the initial boundary value problems (3.8) the solution space should be redefined first.

Def. 3.3.9. *For Banach space H we define the space $L^2(0,T;H)$ as the space of functions $u: \ (0,T) \to H$ equipped with the norm $\|u\|_{L^2(0,T;H)} := \left(\int\limits_0^T \|u(t)\|_H^2 \mathrm{d}t\right)^{\frac{1}{2}}$.*

In order to interpret the weak time derivative we need an appropriate definition as well. This one is similar to (3.3.5) but with respect to $L^2(0,T;H)$. We present the following weak formulations in the corresponding function spaces. More on choice of these spaces e.g in [Ang03].

Def. 3.3.10. *We call a function $w \in L^2(0,T;H^{-1})$ a weak derivative of a function $u \in L^2(0,T;H)$ if*

$$\int\limits_0^T u(t)v'(t)\mathrm{d}t = -\int\limits_0^T w(t)v(t)\mathrm{d}t \ \forall v \in C_0^\infty(0,T). \tag{3.30}$$

Def. 3.3.11. *The space $W(0,T;H)$ is defined as follows:*

$$\{u: \ u \in L^2(0,T;H), \ u' \in L^2(0,T;H^{-1})\}. \tag{3.31}$$

As for the operator equation for the weak formulation of (3.8) the procedure is similar to (3.20) with the new term corresponding to the time derivative:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int\limits_\Omega uv\mathrm{d}\mathbf{x} - \int\limits_\Omega \nabla\cdot(A\nabla u)v\mathrm{d}\mathbf{x} + \int\limits_\Omega \mathbf{b}^T\nabla uv\mathrm{d}\mathbf{x} + \int\limits_\Omega cuv\mathrm{d}\mathbf{x} = \int\limits_\Omega fv\mathrm{d}\mathbf{x}, \tag{3.32}$$

and after the partial integration procedure due to (3.21) one gets the weak formulation for the initial boundary value problem with mixed boundary conditions (3.8).

Find function $u \in W(0, T; H)$, that satisfies the equation

$$\frac{\mathrm{d}}{\mathrm{d}t}(u, v)_0 + a(u, v) = f(v) \;\; \forall v \in H, \tag{3.33}$$

along with the initial condition

$$u(0) = u_0 \in L^2(\Omega), \tag{3.34}$$

where $a(u, v)$, $f(v)$ are in form (3.27).

With the reduced smoothness requirements the solution existence results from classical theory don't apply anymore, thus we have to consider different results that propose the existence and uniqueness of solution for weakly formulated problems.

3.4 Existence and uniqueness results

There are several lemmas that guarantee the solution existence and uniqueness for problems (3.29) and (3.33), (3.34) . The aim of this section is to present the appropriate results found in literature that state that several properties of objects used in (3.29) hold and afterwards to check these properties.

Def. 3.4.1. *Given H a Hilbert space. Defined in $H \times H$ bilinear form $a(u, v) : H \times H \to \mathbb{R}$ is called continuous, if*

$$\exists \lambda > 0, \; \forall u, v \in H \; |a(u, v)| \leq \lambda \|u\|_H \|v\|_H. \tag{3.35}$$

Def. 3.4.2. *Given H a Hilbert space. Defined in $H \times H$ bilinear form $a(u, v) : H \times H \to \mathbb{R}$ is called coercive or H-elliptic, if*

$$\exists \gamma > 0, \; \forall u \in H \; \gamma \|u\|_H^2 \leq a(u, u). \tag{3.36}$$

Def. 3.4.3. *Defined in a normed space H linear functional $f(v) : H \to \mathbb{R}$ is called continuous, if*

$$\exists M > 0, \; \forall v \in H \; |f(v)| \leq M \|v\|_H. \tag{3.37}$$

The next results guarantee the existence and uniqueness for the solutions of weakly-formulated problems (3.29) and (3.33),(3.34). The conditions to fulfill rely on properties of the 'principal' part of the bilinear form.

Prop. 3.4.1. *(Lax-Milgram theorem. For proof see [Hac17], [Ben13], [Ang03].)*
Given Hilbert space H. If defined in $H \times H$ bilinear form
$a(u, v) : H \times H \to \mathbb{R}$ is continuous and H-elliptic,
and a linear functional $f(v) : H \to \mathbb{R}$ - continuous, then the weakly-formulated problem (3.29) has a unique solution.

Prop. 3.4.2. *Problem (3.29) fulfills the conditions of the Lax-Milgram theorem 3.4.1 assuming $L(\cdot)$ is uniformly elliptic.*

Proof. Presented in the main paper variant. □

There is a result with the conditions similar to ones that the Lax-Milgram theorem uses and it serves as an extension to parabolic case. It shall serve us as an existence-uniqueness result thereof.

Prop. 3.4.3. *Given Hilbert space H. If defined in $H \times H$ bilinear form*
$a(u, v) : H \times H \to \mathbb{R}$ is continuous and H-elliptic,
$f \in L^2(0, T; L^2(\Omega))$ and $u_0 \in L^2(\Omega)$, then the weakly-formulated problem (3.33), (3.34) has a unique solution.

Proof. Presented in the main paper variant. □

3.5 Finite dimensional problems and the FEM

The approach behind all the techniques we are about to mention is the function space discretization. If we consider the finite amount of basis elements, our weak formulations (3.29) and (3.33),(3.34) would become equivalent to problems of linear algebra. This approach is closely related to *Galerkin method* and would involve full matrices in case if our basis functions each is defined and mainly non-zero in the whole interior of the geometrical space. In order to obtain sparse matrices for convenience in numerics, one can combine the solution and test function space-discretization approach with the geometrical space discretization such that the basis functions each have a local support i.e each is zero in most of the interior of the geometrical space and locally non-zero. This approach is closely related to the *Finite element method*.

The discrete analogue to our weak formulation (3.29) shall look like this:

Find the function $u_h \in H_h(\Omega)$, that satisfies the equation

$$a(u_h, v_h) = f(v_h) \quad \forall v_h \in H_h(\Omega). \tag{3.38}$$

The discrete analogue to our weak formulation (3.33), (3.34) shall look like this:

Find the function $u_h(t) \in W(0,T;H_h(\Omega))$ such that $u_h(0) = I_h u_0 \in H_h$, that satisfies the equation

$$\frac{\mathrm{d}}{\mathrm{d}t}(u_h(t), v_h)_0 + a(u_h(t), v_h) = f(v_h) \ \ \forall v_h \in H_h(\Omega). \tag{3.39}$$

We need the representation and the existence-uniqueness result for the discrete case.

Prop. 3.5.1. *The formulation (3.38) is equivalent to linear system $A\xi = f;\, \xi, f \in \mathbb{R}^N$.*
The formulation (3.39) is equivalent to initial-value problem for the ODE system
$\dot{\xi}(t) + B\xi(t) = f(t),\ \xi(0) = \xi_0;\ \xi, f :\ \mathbb{R} \to \mathbb{R}^N,\ \xi_0 \in \mathbb{R}^N$.
If the bilinear form is continuous and H-elliptic then each problem has a unique solution.

Proof. Presented in the main paper variant. □

We are left with a choice of our basis functions. If we want our matrices to be sparse, this choice is to be closely related to a partition of domains. Here we have arrived to definitions of the *triangulation*and the *finite element*.

Def. 3.5.1. *The bounded domain's $\Omega \subset \mathbb{R}^n$ subdivision $\mathcal{T}_h$ onto elements $T \subset \mathbb{R}^n$, $T \in \mathcal{T}_h$ is called the triangulation of set Ω if the following holds:*

1. $\overline{\Omega} = \bigcup\limits_{T \in \mathcal{T}_h} T$.
2. $\forall T \in \mathcal{T}_h$: *$T$ - closed set, $int(T) \neq \emptyset$, ∂T - Lipshitz boundary according to definition (3.2.1).*
3. $\forall T_i, T_j \in \mathcal{T}_h,\ i \neq j:\ int(T_i) \cap int(T_j) = \emptyset$.

   For polyhedral subdivisions we require an admissibility condition:
4. $\forall T_i \in \mathcal{T}_h$ *: each T_i face is either other element's T_j face or it belongs to set's Ω boundary $\partial\Omega$.*

We shall use the polynomial finite elements hence the definition.

Def. 3.5.2. *The space*

$$S_h^{(m)} := \{u \in C^0(\bar{\Omega}) :\ u|_{T_i} \in \Pi_m \ \forall T_i \in \mathcal{T}_h\} \tag{3.40}$$

is called the m-th order polynomial finite element space with respect to triangulation $\mathcal{T}_h$.

One of the most popular special cases is the *Linear Lagrangian elements*:

$$S_h^{(1)} := \{u \in C^0(\bar{\Omega}) : \; u|_{T_i} \in \Pi_1 \; \forall T_i \in \mathcal{T}_h\}. \tag{3.41}$$

This one is to be used in our applications chapter 6.

One way to refer to a finite element is to refer to a $\mathcal{T}_h$ element, but there exists more general definition, which takes into account several method's aspects.

Def. 3.5.3. *We call a finite element in $\mathbb{R}^n$ a triplet (T, P, Σ), where:*

1. *T - closed set, $int(T) \neq \emptyset$, ∂T - Lipshitz boundary.*
2. *P - defined on T finite dimensional function $T \to \mathbb{R}$ space.*
3. *Σ - finite set of linearly independent functionals φ_i with the unisolvency property: $\forall \alpha_i \in \mathbb{R} \; \exists p \in P : \quad \varphi_i(p) = \alpha_i, \; i = 1..\dim P$.*

Thus, the finite element is characterized by the basis functions $p \in P$, defined on this element, and the degrees of freedom φ_i.

The property of the set Σ gives us a chance to unambiguously set the linear or polynomial functions, defined on element T.

In case when these degrees of freedom are in form $p \to p(a_i)$, where a_i are the mesh points, the task of finding the basis functions is a simple interpolation task.

There's a result, which would say that the Galerkin solution u_h itself, without saying anything about the nature of shape functions, is as close to an analytical solution as any other function from H_h.

Prop. 3.5.2. *(Céa's lemma [Hac17], [Ben13], [Ang03].)*

Under the conditions that the bilinear form a is continuous and H^1-elliptic, the solutions of problems (3.29) and (3.38) u and u_h do satisfy the following:

$$\|u - u_h\|_{H^1} \leq c \inf_{v_h \in H_h} \|u - v_h\|_{H^1}, \tag{3.42}$$

where $c > 0$.

Proof. Presented in the main paper variant. □

The result is the key to the approximation and convergence results in different norms which we do not present here as they are beyond the scope of this thesis but one can find them in literature such as [Hac17], [Ben13], [Ang03].

3.6 Treatment of the non-linear time-dependent system

We do an extension to several dimensions for our functions to become the vector-valued ones. The capital N notation for the number of components for our vector functions is going to be used and is in consistence with the number of species from chapter 2 and thus the number of the equations and unknown functions in the system. We define the vector functions $\mathbb{R}^d \to \mathbb{R}^N$ as $\mathbf{u} := (u_1, \dots, u_N)^T$, $\mathbf{v} := (v_1, \dots, v_N)^T$. The components are either elements of $L^2(\Omega)$ or $H^1(\Omega)$ thus $[L^2(\Omega)]^N$ and $[H^1(\Omega)]^N$ are still Hilbert spaces equipped with scalar product $(\mathbf{u}, \mathbf{v})_k = \sum_{i=1}^{N} (u_i, v_i)_k$ and the induced norm $[(\mathbf{u}, \mathbf{u})_k]^{\frac{1}{2}}$ for $k = 0, 1$.
The construction of our elliptic or parabolic PDE system is in accordance with models (2.1), (2.8) from chapter 2:

$$
\begin{cases}
L_1 u_1 = f_1(u_1, \dots, u_N), \Omega, \\
\dots \\
L_n u_n = f_n(u_1, \dots, u_N), \Omega, \\
u_i(\mathbf{x}) = 0, \quad \Gamma_{D_i}, \\
\nabla u_i \cdot \widehat{n} = 0, \quad \Gamma_{N_i},
\end{cases} \tag{3.43}
$$

$i = 1..N$.
Similar construction is obtained for parabolic case.

$$
\begin{cases}
\dfrac{\partial u_i}{\partial t} + L_{ii} u_i = f_i(u_1, \dots, u_N), \quad (0, T) \times \Omega, \\
u_i(t, \mathbf{x}) = 0, \quad (0, T) \times \Gamma_{D_i}, \\
\nabla u \cdot \widehat{n} = 0, \quad (0, T) \times \Gamma_{N_i}, \\
u_i(0, \mathbf{x}) = u_{0,i}, \quad \Omega,
\end{cases} \tag{3.44}
$$

$i = 1..N$.
We arrive to weak formulations similar to (3.27):

$$
a(\mathbf{u}, \mathbf{v}) = \int_{\Omega} [L_1 u_1 v_1 + \dots + L_n u_n v_n] \mathrm{d}\mathbf{x}, \ f(\mathbf{v}) = \int_{\Omega} [f_1 v_1 + \dots + f_n v_n] \mathrm{d}\mathbf{x}. \tag{3.45}
$$

The coerciveness and continuity would again guarantee the solvability and the uniqueness of solution for weak formulations of (3.43) and (3.44) (if the vector-valued initial condition is also in $[L^2(\Omega)]^N$) due to the results (3.4.1) - (3.4.3). Thus we formulate the following proposition.

Prop. 3.6.1. *The bilinear form in (3.45) is $[H^1(\Omega)]^N$ continuous and coercive, the linear form in (3.45) is $[H^1(\Omega)]^N$ continuous.*

Proof. Presented in the main paper variant. □

In the similar fashion, since the uniqueness is proven, we can jump to the discretization aspects. These are again a slight generalization of what has already been done.

After discretizing both problems from chapter 2, we have to solve the non-linear system of algebraic equations $\mathbf{f}(\xi) = 0$, where $\mathbf{f} : \mathbb{R}^N \to \mathbb{R}^N$ - nonlinear, assumed differentiable vector function.

This is due to the fact that by using a suitable time-stepping scheme the linear first term of $M\dot{\xi} + A\xi = g(\xi)$ with respect to $\xi(t)$ now becomes linear with respect to all introduced quantities in discrete time.

One of the choices is the *backward Euler* scheme [Hac17], [Ang03].

Def. 3.6.1. *We call the scheme $\dot{\xi} \approx \dfrac{\xi_k - \xi_{k-1}}{\tau_k}$ a backward Euler scheme*

with the time steps τ_k, $k = 1..\text{N}$.
Then, by substituting that in the ODE system, we obtain

$$M\frac{\xi_k - \xi_{k-1}}{\tau_k} + A\xi_k = g(\xi_k), \tag{3.46}$$

or in more compact way $\mathbf{f}(\xi) = 0$ with

$$\mathbf{f}(\xi_k) = (M + \tau_k A)\xi_k - M\xi_{k-1} - \tau_k g(\xi_k). \tag{3.47}$$

The scheme is selected as a more or less universal one, but other implicit schemes would yield the same model - a system of non-linear algebraic equations.

Thus in the following we omit the vectorial notation and present some results on linearisation of general non-linear systems $f(x) = 0$.

Def. 3.6.2. *We call the method*

$$x_{k+1} = x_k + h_k f(x_k),\ x_0 = \tilde{x_0},\ h_k \in \mathbb{R} \setminus \{0\}, \tag{3.48}$$

a simple iteration method,

$k = 0..\text{N}$.

One can observe that if the convergence has been obtained, i.e
$x^* = x^* + h_k f(x^*)$, then $f(x^*) = 0$, which means that the fixed point is one of the

solutions of the given non-linear system.

But this fact is not enough since one need to exclude the divergent iterations, therefore one requires the notion of the *contraction map*.

We define our function f in $\Omega \subset \mathbb{R}^n$ - convex, bounded set, so $f : \Omega \to \Omega$. We consider the results quite abstractly - in metric spaces.

Let Ω be a metric space (Ω, ρ) with standard metric space axioms:
$\rho : \Omega \times \Omega \to [0, \infty)$:

1) $\rho(x, y) \geq 0$,

2) $\rho(x, y) = 0 \Leftrightarrow x = y$,

3) $\rho(x, y) = \rho(y, x)$,

4) $\rho(x, y) \leq \rho(x, z) + \rho(z, y) \ \forall z \in \Omega$.

We can choose $\rho(x, y) = \|x - y\|$ if Ω - linear normed space.

Def. 3.6.3. *(Convergence) A sequence $\{x_n\}$ of metric space's (Ω, ρ) elements does converge to an element $x^* \in \Omega$ iff $\rho(x_n, x^*) \to 0, \ n \to \infty$.*

Def. 3.6.4. *(Contraction) We call a mapping $f : \Omega \to \Omega$ a contraction when*
$\rho(f(x), f(y)) \leq q \cdot \rho(x, y)$,
$\forall x, y, f(x), f(y) \in \Omega, \ 0 \leq q < 1$.

Prop. 3.6.2. *Assume that $\Omega \subset \mathbb{R}^n$ - convex, bounded set.*
Then for each contraction $f : \ \Omega \to \Omega \ \exists! \ x^* \in \Omega :$
$f(x^*) = x^*$
- a fixed point.

Proof. Presented in the main paper variant. □

Prop. 3.6.3. $g : \Omega \to \Omega, \ \Omega \subset \mathbb{R}^n, \ g \in C^1(\Omega)$.
The iteration method in form $x_{k+1} = g(x_k)$ does converge (g is contraction) if
$\|J(g(x))\| < 1, \forall \ x \in \Omega$.

Proof. Presented in the main paper variant. □

The special case of this condition is the *Newton method* - the one that guarantees that the condition (3.6.3) holds by neglecting the growth factor $J(g(x)) \equiv \theta_{n \times n}$

since $\|J(g(x))\| = 0 \Leftrightarrow J(g(x)) \equiv \theta_{n\times n}$.

The method has the form

$$x_{k+1} = x_k - J^{-1}(f(x_k)) \cdot f(x_k),\ k = 0..\mathrm{N}. \tag{3.49}$$

The next chapter deals with special techniques on solution post-processing.

Chapter 4

Parameter optimization and model reduction strategies

The main motivation to consider an optimization technique is that the modelling results are not necessarily consistent with the experimental results. Thus we need a concept to minimize the distance between the chosen quantities.

The reason to choose the stochastic numerical method is that the data might be not smooth enough and the functional to optimize appears as a sort of black box in complicated cases even though in simple cases the structure can be visible.

In general the expression to minimize shall look like this:

$$f(\mathbf{p}) = \|\mathbf{Q}_{exp} - \mathbf{Q}_{mod}(\mathbf{p})\|^2, \tag{4.1}$$

where $\mathbf{Q}_{exp} \in \mathbb{R}^n$ is the given vector of experimental values and $\mathbf{Q}_{mod} \in \mathbb{R}^n$ is the result of post-processing of the PDE system (2.1), (2.8) solution. The minimimization problem then should look like this

$$\min_{\mathbf{p}} \|\mathbf{Q}_{exp} - \mathbf{Q}_{mod}(\mathbf{p})\|^2. \tag{4.2}$$

Also $\mathbf{p}$ is the vector of parameters for our parametrized model - dimension to be denoted as D_p depends on the nature of the task.
In our models (2.1), (2.8) it might be maximal and outlet temperatures comparison to name the least:

$$\min_{\mathbf{p}} \|T_{max_{exp}} - T_{max_{mod}}\|^2,$$

$$\min_{\mathbf{p}} \left\| \begin{pmatrix} T_{max_{exp}} \\ T_{out_{exp}} \end{pmatrix} - \begin{pmatrix} T_{max_{mod}} \\ T_{out_{mod}} \end{pmatrix} \right\|^2.$$

These and other choices of cost functional to be discussed in the applications chapter 6.

In the first section we discuss the choice of the optimization technique, present its algorithm, convergence criterion and prove the convergence criterion.

4.1 Particle swarm optimization

The choice of this numerical non-gradient optimization method is due to the fact that in general we don't have an explicit expression of the functional. This is either due to the iterative nature of finding the solution of the PDE system (2.1), (2.8) when we code the solver by ourselves or to the complete black-box nature of our solving process when we use an industrial package for solving it.

Thus we look onto the set of gradient-free methods.

For several other applications throughout the PHD studies author had made his choice on the PSO due to simplicity, universality and since it has been showing good result in terms of leading to optimal directions.

In this section we present the method (4.3), (4.4) and the convergence criterion 4.1.1. The results are adopted out of several books and publications [BAE05], [Cle06].

We present the algorithm in its classical form. It uses a modification of discrete velocities and positions of particles and for each particle determines winning states $\mathbf{p}^i \in \mathbb{R}^{D_p}$ and for the entire swarm - the winning state $\mathbf{g} \in \mathbb{R}^{D_p}$ and depends on method's parameters $w,\ c_1,\ c_2 \in \mathbb{R}$ and $r_1,\ r_2 \in \mathbb{R}$:

$$v_d^i(k+1) = wv^i(k) + c_1 r_1 (p_d^i - x_d^i(k)) + c_2 r_2 (g_d - x_d^i(k)), \tag{4.3}$$

$$x_d^i(k+1) = x_d^i(k) + v_d^i(k+1), \tag{4.4}$$

$k = 1..\mathrm{N},\ d = 1..D_p,\ i = 1..S$, where N - number of iterations, D_p - number of parameters, S - number of particles.

$c_1 r_1$ and $c_2 r_2$ are usually combined and the notations $\phi_1 := c_1 r_1$ and $\phi_2 := c_2 r_2$ are widely used and is due to the fact that $r_{1,2}$ are usually picked as two uniformly distributed random numbers $\sim U(0,1)$.

The choice of parameters to be discussed below.

The process (4.3), (4.4) iterates the quantity $x_d^i(k)$ in upper and lower bounds $\mathbf{b}_{lo},\ \mathbf{b}_{up}$.

In order to program this method as we are going to in chapter 5 (see full text), we need to construct the algorithm out of our formulas (4.3), (4.4) along with the initialization of data.

We come up with the simplest way how to do it and the algorithm looks as follows:

Alg. 4.1.1.

1) The initial cycle.
Randomize the swarm's best known position $\mathbf{g}$ and the position of each particle

$$\mathbf{x}^i \sim U(\mathbf{b}_{lo}, \mathbf{b}_{up}),\ \mathbf{g} \sim U(\mathbf{b}_{lo}, \mathbf{b}_{up}),\ i = 1..S, \tag{4.5}$$

and the particle's best known position set to its initial position $\mathbf{p}^i := \mathbf{x}^i$.
If $f(\mathbf{p}^i) < f(\mathbf{g})$ then set $\mathbf{g} := \mathbf{p}^i$.
Initialize the particle's velocity:

$$\mathbf{v}^i \sim U(-|\mathbf{b}_{up} - \mathbf{b}_{lo}|,\ |\mathbf{b}_{up} - \mathbf{b}_{lo}|). \tag{4.6}$$

The randomization and absolute value taking is happening component-wise.

2) The main cycle.
For $k = 1..\mathrm{N},\ d = 1..D_p,\ i = 1..S$:
implement the formulas (4.3), (4.4).
Update the winning states:

$$\begin{aligned} f(\mathbf{x}^i) < f(\mathbf{p}^i) \ &\Rightarrow\ \mathbf{p}^i := \mathbf{x}^i, \\ f(\mathbf{p}^i) < f(\mathbf{g}) \ &\Rightarrow\ \mathbf{g} := \mathbf{p}^i. \end{aligned} \tag{4.7}$$

The implementation of this algorithm in Matlab is going to be pinned down in the next chapter (see full thesis variant).

Since upper and lower bounds $\mathbf{b}_{up}$, $\mathbf{b}_{lo}$ are present, we need to apply some boundary controls. The linear operations in (4.3), (4.4) can easily compute the new state $x^i_d(k)$ out of bounds. This thing should be taken care of a-priori.

The author came up with the following naive set of actions, which so far have shown to be consistent with all the author's experiments and in particular the ones from the applications chapter 6.

In order to perform a constrained optimization between $\mathbf{b}_{lo}$, $\mathbf{b}_{up}$, we apply the following actions.

For each k (round brackets omitted):

$$x_d^i + v_d^i < b_{lo_d} \ \vee \ x_d^i + v_d^i > b_{up_d} \ \Rightarrow v_d^i := -v_d^i, \tag{4.8}$$

$$\begin{aligned} x_d^i + v_d^i < b_{lo_d} \ &\Rightarrow v_d^i := b_{lo_d} - x_d^i, \\ x_d^i + v_d^i > b_{up_d} \ &\Rightarrow v_d^i := b_{up_d} - x_d^i. \end{aligned} \tag{4.9}$$

Prop. 4.1.1. *The choice of parameters such as* $\max\{|\lambda_1|, |\lambda_2|\} < 1$, *where* $\lambda_1 = \dfrac{1 + w - \phi_1 - \phi_2 + \gamma}{2}$ *and* $\lambda_2 = \dfrac{1 + w - \phi_1 - \phi_2 - \gamma}{2}$, $\gamma = \sqrt{(1 + w - \phi_1 - \phi_2)^2 - 4w}$ *guarantees the convergence of (4.3), (4.4) to the best state of their population* $\mathbf{g}$*:*

$$\lim_{k\to\infty} x_d^i(k) = g_d, \ d = 1..D_p, \ i = 1..S. \tag{4.10}$$

Proof. Presented in the main paper variant. □

4.2 Reduction due to principal components

The main technique we are going to use here is the *Principal Component Analysis*.

The author is mainly inspired by the published results [Wil10], [Gre08].

A principal component analysis is required in order to speed up the evaluation of the functional whilst performing the optimization.

Since we are going to mostly deal with vectorial quantities, the bold notation for vectors is going to be omitted till the end of the chapter.

The procedure requires making of the matrix of snapshots first:

$$X = [\hat{x}_1, \ldots, \hat{x}_L]. \tag{4.11}$$

This matrix should contain the representative solutions $\hat{x}_k$, $k = 1..L$ at different time steps. This should be calculated once with full dimension. Afterwards this matrix is going to be used to form a suitable basis.

If we seek the representation of the snapshots in the reduced basis with the introduced measure of quality of the representation, we wish to minimize the quantity

$$\sum_{k=1}^{L} \|\hat{x}_k - \hat{V}C\|_2^2 \tag{4.12}$$

subject to orthogonal bases of K elements, stored as the columns of matrix $\hat{V} \in \mathbb{R}^{n \times K}$, $K < L$. Small n here is in correspondence with the number of basis elements for Galerkin or FEM solution from chapter 3.

C represents the vector of coefficients in this basis - for each $\hat{V}$ to be uniquely determined.

Matrix of snapshots is rectangular and there's a necessity to introduce the generalization of matrices eigenrepresentation: the *Singular Value Decomposition*.

Def. 4.2.1. *The SVD - a (compact) singular value decomposition is a factorization of a matrix* $X \in \mathbb{R}^{n \times L}$ *in form* $X = V\Sigma U^T$, *where* $V \in \mathbb{R}^{n \times L}$ *is an orthogonal matrix,* $\Sigma \in \mathbb{R}^{L \times L} = diag([\sigma_1, \ \sigma_2, \ldots, \sigma_L])$ *with positive diagonal elements,* $U \in \mathbb{R}^{L \times L}$ *- an orthogonal matrix.*

If we consider the left singular vector matrix V in the SVD $X = V\Sigma U^T$, where $X = [\hat{x_k}]$ - matrix of the solution representatives (4.11), we need to know the representation of the basis vectors as well as the quality of approximation of snapshots.

Prop. 4.2.1. *For a matrix of snapshots* $X \in \mathbb{R}^{n \times L}$ *the left singular vector matrix* V *is the orthogonal eigenvectors matrix of* XX^T.
The right singular vectors in U *are the orthogonal eigenvectors of* $X^T X$.
The singular values are the square roots of eigenvalues of $X^T X$.
If we choose the set of $K < L$ *left singular vectors to form a reduced basis in (4.12), the value of (4.12) shall be* $\|\sigma\|_2^2$, *where* $\sigma \in \mathbb{R}^{L-K}$ *- the vector of leftover singular values, corresponding to* $L - K$ *left singular vectors.*

Proof. Presented in the main paper variant. □

Thus, in order to form a suitable reduced basis, one should take the most 'energetic' singular vectors - the ones corresponding to largest singular values.

Whenever we execute the PDE solution process, described in Chapter 3 and arrive to a system of ODEs after spatially discretizing the system of parabolic PDEs with FEMs or to a linear system in steady case, we introduce the vector of reduced states x_r, $Vx_r \approx x$ and $x_{0,r}$, $Vx_{0,r} \approx x_0$. The spatially discretized system then becomes:

$$\dot{x_r} = V^T BVx_r + V^T g(Vx_r), \ x_{0,r} = V^T x_0. \tag{4.13}$$

Overall, as we see in the applications' chapter 6, leaving behind the vectors corresponding to small singular values and taking into basis the energetic ones shall show very good results and the dimension shall be reduced and the quality of approximation is given by Prop. 4.2.1.

After time discretization and applying the back substitution $x = Vx_r$ one does the regular solution post-processing in order to compute the value of the functional (4.1) in the optimization procedure.

Thus we end up with the approximated zero of the functional. To tackle this approximation, the author came up with the algorithm for the applications section:

Alg. 4.2.1.

1) Before running but after initializing the PSO (4.3), (4.4), compute the snapshot matrix for initial vector of parameters.

2) Run the optimization procedure with the reduced states.

3) Converge to an approximated zero of the functional.

4) Compute the value of the functional with the p=argmin for the solution in full basis. At the same time get the new snapshot matrix.

5) If the tolerance not reached, repeat the procedure.

4.3 Information storage and pattern recognition

The goal is to reduce the number of evaluations of the functional (4.1).

Consider the sequence of experiments E_i, $i = 1..N_E$. If the new experimental data is considered for some i, one can use the gathered knowledge from the previous optimization stages for previous experiments since the optimization technique processes a lot of solutions not feasible for the present experiment, but that might be feasible for any subsequent experiment.

The idea is to store some of the information from previous experiments in a form of network - a discrete dynamical system with a certain convergence-guaranteed properties, which can be addressed as a first step of the optimization strategy.

Similar approach is to be found in theory of recurrent systems and finite automation [Cha01]. Then the author adopts it for the considered applications and couples it with the optimization procedure (4.3), (4.4).

The idea is to introduce the discrete dynamical system

$$x(k+1) = f(x(k)),\ x(0) = x_0, \tag{4.14}$$

where

$$f(x) = \mathbf{sign}(Ax),\ k = 0..\mathrm{N},\ x_0, x(k) \in \{-1,1\}^D,\ D \in \mathbb{N} \tag{4.15}$$

and the bold-case sign stands for the sign function component-wise with zero mapped to one. Each pattern x^i, $i = 1..P$ to be stored can be represented as a vector of binary values thus the number D can be large.

One of the ways to define the matrix in (4.14),(4.15) is the following:

$$A = \sum_{i=1}^{P} x^i [x^i]^T. \tag{4.16}$$

The latter representation allows to dynamically add or remove patterns.

We need a system to converge to a fixed point and the best scenario is that this fixed point is one of the patterns.

Prop. 4.3.1. *The system (4.14),(4.15) with A constructed as in (4.16) with* $[Ax]_d \neq 0,\ d = 1..D,\ x \in \{-1,1\}^D$ *always converges to a fixed point* $x^* :\ f(x^*) = x^*$.
Moreover if the condition $\sum\limits_{\substack{i,j=1 \\ i\neq j}}^{P} |(x^i, x^j)| < D$ *holds, then it converges to a fixed point, which is one of the patterns:* $x^* \in \{x^1, \ldots, x^P\}$.

Proof. Presented in the main paper variant. □

The system matrix A is constructed out of the patterns arising from the data left over of the optimization procedure for (4.3), (4.4). Each time one has to find the parameters for the system (2.1), (2.8) given some experimental data regarding the solution, the vector x has to be constructed. One of the ways is to put the given data inside and regarding the unknown values put the deterministic function such as the extreme or average value.

Then running the procedure (4.14) several times will find the closest pattern in terms of the predefined measure. The value of the measure is to be chosen separately for each application in 6.

One can come up with the following algorithm for this matter.

Alg. 4.3.1.

1) Analyse the dimensions of the given solution and corresponding parameter data and create the patterns such that they fulfil or are close to condition in (4.3.1).

2) Convert the given initial value in pattern form. In case if some parameter values are left to be unknown, use the deterministic or random values in the given intervals.

3) Call out the DDS (4.14) with the reasonable amount of iterations.

4) Decode the obtained values.

Then the PDE system (2.1), (2.8) is to be solved with the determined parameters and if the result is close enough in terms of the measure (4.1) from the first section of this chapter, one stops, or alternatively the result can be used as a good starting point for further optimization.

Chapter 5. Test problem treatment with MATLAB

The implementation of the algorithms in Matlab is presented in the main paper variant.

Chapter 6

Applications

In the last chapter the author wants to show the numerical results out of frameworks presented in the previous chapters in form of pictures and tables.

In all the following experiments the geometry of the combustion chamber is going to be represented either by a cylinder of radius 1 meter and height 4 meters or the axial symmetry may be considered depending on situation and then it gets solved in two dimensions on the rectangle. The inhomogeneous Dirichlet boundary conditions are to be set at inlet and the homogeneous Neumann boundary conditions to be set at all the remaining portions of the boundary. The plots of PDE solutions are going to be in 3 dimensions.

First section deals with one single-step reaction in a three-dimensional setting.

6.1 Solving PDEs for the single reaction model

For the beginning the model with one standard combustion reaction

$$2H_2 + O_2 \to 2H_2O \tag{6.1}$$

is considered.

The same framework can be applied for single reaction with different species (the determined parameter values would change).

This leads to 4 PDEs (3 species equations and one temperature equation). The species equation for the specie Y_k has it's form (2.1), as in chapter 2:

$$\rho \frac{DY_k}{Dt} = \nabla \cdot (\rho D \nabla Y_k) + \dot{\omega}_k. \tag{6.2}$$

The term $\dot{\omega}_k$ has its form as in (2.3)

$$\sum_{j=1}^{M} \dot{\omega}_{kj}. \tag{6.3}$$

The numeration for the species is as follows:

$$Y_1 := Y(H_2),\ Y_2 := Y(O_2),$$
$$Y_3 := Y(H_2O). \tag{6.4}$$

Then the matrix ν in the reaction mechanism (2.4) is $\begin{pmatrix} -2 \\ -1 \\ 2 \end{pmatrix}$, and the matrix $\dot{\omega}$ is $\begin{pmatrix} -2q_1 w_1 \\ -q_1 w_2 \\ 2q_1 w_3 \end{pmatrix}$.

According to the Arrhenius law (2.5), (2.6),

$$q_1 = A_1 e^{\frac{-E_1}{RT}} \left(\frac{\rho Y_1}{w_1}\right)^2 \left(\frac{\rho Y_2}{w_2}\right). \tag{6.5}$$

Also we consider the axial velocity component w only. The diffusivity D is considered as a constant and equal for every specie in the system.

In this case, the species equations are (terms are divided by ρ):

$$\begin{cases} \dfrac{\partial Y_1}{\partial t} + w\dfrac{\partial Y_1}{\partial x} = D\triangle Y_1 - 2A_1 e^{\frac{-E_1}{RT}} \dfrac{\rho^2}{w_1 w_2} Y_1^2 Y_2, \\ \dfrac{\partial Y_2}{\partial t} + w\dfrac{\partial Y_2}{\partial x} = D\triangle Y_2 - A_1 e^{\frac{-E_1}{RT}} \dfrac{\rho^2}{w_1^2} Y_1^2 Y_2, \\ \dfrac{\partial Y_3}{\partial t} + w\dfrac{\partial Y_3}{\partial x} = D\triangle Y_3 + 2A_1 e^{\frac{-E_1}{RT}} \dfrac{\rho^2 w_3}{w_1^2 w_2} Y_1^2 Y_2. \end{cases} \tag{6.6}$$

The temperature equation is subject to one more simplification: the heat capacity we shall consider independent of the specie or the temperature (described in the main paper variant). The right hand side of the temperature equation is

$$-\sum_{k=1}^{N} (\triangle H_{f,k} + h_{s,k})\dot{\omega}_k, \tag{6.7}$$

which we divide into two sums:

$$-\sum_{k=1}^{N} \triangle H_{f,k} - \sum_{k=1}^{N} h_{s,k}\dot{\omega}_k. \tag{6.8}$$

Under the assumption that the heat capacity doesn't depend on the temperature

$$h_{s,k} = \int_{T_0}^{T} c_{pk} dT = (T - T_0) c_{pk} \tag{6.9}$$

or the species

$$\sum_{k=1}^{N} h_{s,k} \dot{\omega}_k = (T - T_0) c_p \sum_{k=1}^{N} \dot{\omega}_k = 0, \tag{6.10}$$

the only enthalpies we need are the enthalpies of formation to be determined from the tables [Mil00].

The temperature equation is as in (2.8):

$$\rho c_p \frac{DT}{Dt} = \nabla \cdot (\lambda \nabla T) - \sum_{k=1}^{N} \triangle H_{f,k} \dot{\omega}_k, \tag{6.11}$$

and now we rewrite it explicitly by using the right hand sides of the species equations (the diffusivity λ is considered as a constant):

$$\rho c_p \frac{\partial T}{\partial t} + \rho c_p w \frac{\partial T}{\partial x} = \lambda \triangle T + A_1 e^{\frac{-E_1}{RT}} \frac{\rho^2}{w_1^2 w_2} Y_1^2 Y_2 \Big[2w_1 \triangle H_{f,1} + w_2 \triangle H_{f,2} - 2w_3 \triangle H_{f,3} \Big]. \tag{6.12}$$

The resulting system of partial differential equations in its dimensional form now is:

$$\begin{cases} \dfrac{\partial Y_1}{\partial t} + w \dfrac{\partial Y_1}{\partial x} = D \triangle Y_1 - 2A_1 e^{\frac{-E_1}{RT}} \dfrac{\rho^2}{w_1 w_2} Y_1^2 Y_2, \\ \dfrac{\partial Y_2}{\partial t} + w \dfrac{\partial Y_2}{\partial x} = D \triangle Y_2 - A_1 e^{\frac{-E_1}{RT}} \dfrac{\rho^2}{w_1^2} Y_1^2 Y_2, \\ \dfrac{\partial Y_3}{\partial t} + w \dfrac{\partial Y_3}{\partial x} = D \triangle Y_3 + 2A_1 e^{\frac{-E_1}{RT}} \dfrac{\rho^2 w_3}{w_1^2 w_2} Y_1^2 Y_2, \\ \rho c_p \dfrac{\partial T}{\partial t} + \rho c_p w \dfrac{\partial T}{\partial x} = \lambda \triangle T + A_1 e^{\frac{-E_1}{RT}} \dfrac{\rho^3}{w_1^2 w_2} Y_1^2 Y_2 \cdot \\ \cdot \Big[2w_1 \triangle H_{f,1} + w_2 \triangle H_{f,2} - 2w_3 \triangle H_{f,3} \Big]. \end{cases} \tag{6.13}$$

We subject the vector of unknown functions $\mathbf{X} = (Y_1, Y_2, Y_3, T)^T$ to the boundary and the initial conditions:

$$\mathbf{X}|_{\Gamma_D} = \mathbf{X}_D,$$

$$\mathbf{X}|_{\Gamma_N} = 0,$$
$$\mathbf{X}(0) = \mathbf{X}_D e^{-x}, \tag{6.14}$$

where Γ_D is the Dirichlet portion of the boundary (the inlet), $\mathbf{X}_D$ are the prescribed values at Γ_D and Γ_N is the Neumann portion of the boundary. The smooth exponential decay initial condition is considered.

In order to start the simulation one has to specify all the parameters values, e.g. to answer the question, what values are inside this vector of unknowns:

$$\left(\rho, c_p, D, \lambda, w_1, w_2, w_3, w, A_1, E_1, R, \triangle H_{f,1}, \triangle H_{f,2}, \triangle H_{f,3}, Y_{1,D}, Y_{2,D}, Y_{3,D}\right)^T. \tag{6.15}$$

The parameters are to be taken from different tables [Mil00], [Woo02], [20 04], [Tab20].

For the simulation we consider the density of mixture ρ to be $1[\frac{kg}{m^3}]$. The heat capacity for the mixture to be $1000[\frac{J}{kg \cdot K}]$.

The molecular diffusivity D is $5 \cdot 10^{-5}[\frac{m^2}{s}]$ and the thermal conductivity λ is $5 \cdot 10^{-2}[\frac{J}{m \cdot s \cdot K}]$.

The atomic weights $w_{1,2,3}$ are straightforwardly to be taken from the periodic table: $w_1 = 0.002[\frac{kg}{mol}]$, $w_2 = 0.032[\frac{kg}{mol}]$, $w_3 = 0.018[\frac{kg}{mol}]$.
The constant axial velocity is set to $1[\frac{m}{s}]$.

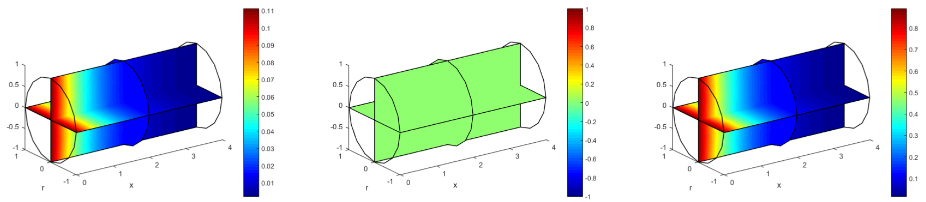
The pre-exponential factor and the activation energy for the reaction (6.1) are something to be discussed in the optimization section. If we are lucky, the parameters can be found in Chemkin format table or found in publications [Woo02], [20 04]. These are the empirical results and not necessarily suit our experiments. As for now we choose $A_1 = 1.7 \cdot 10^{13}[cgs]$ and $E_1 = 47780[\frac{cal}{mol}] = 47780 \cdot 4.18[\frac{J}{mol}]$ as we've seen some estimates in [Wil10] or values for similar but not exact same reactions in [Woo02], [20 04].

In order to convert the pre-exponential factor A_1 accurately into the SI units, one has to see what the dimension of this factor is for the corresponding reaction. Each term of the dimensional species equation has a dimension of $[s^{-1}]$. By considering the reaction term of, for example, first equation, e.g $-2A_1 e^{\frac{-E_1}{RT}} \dfrac{\rho^2}{w_1 w_2} Y_1^2 Y_2$, the only dimensional multiplier left is $\dfrac{\rho^2}{w_1 w_2}$, which is in $\left[\dfrac{\cancel{kg^2} mol^2}{m^6 \cancel{kg^2}}\right]$. Thus the dimension of the pre-exponential factor is $\left[\dfrac{m^6}{s \cdot mol^2}\right]$. If it is given in the cgs units, 12 digits are to be taken off, which for the factor of 10^{13} is good news from the numerical point of view.

The ideal gas constant $\mathcal{R} = 8.314[\frac{J}{mol \cdot K}]$ should be divided by the mean atomic weight $\bar{w}$ in order to use it in the right hand sides of our equations: $R = \frac{\mathcal{R}}{\bar{w}}$.
The enthalpies of formation for the species H_2 and O_2 are by convention zero at the reference temperature. Then the enthalpy of formation for the H_2O is set to $-285800[\frac{J}{mol}]$.
Finally, to prescribe the constant values on the Dirichlet portion of the boundary, we use the stoichiometric ratio $s = \dfrac{0.032}{2 \cdot 0.002} = 8$, and then solve the system for the mass fractions at stoichiometry:

$$\begin{cases} \frac{Y_1}{Y_2} = \frac{1}{8}, \\ Y_1 + Y_2 = 1, \end{cases} \tag{6.16}$$

which gives us $Y_{1,D} = \frac{1}{9}$ and $Y_{2,D} = \frac{8}{9}$.
The product $Y_{3,D}$ is set to zero on this portion of the boundary.
By means of the spatial linear finite element discretization (3.39), (3.41) and the treatment of non-linear time depending system (3.45), (3.47), one gets the 4-second process graphical description with the time step of $\tau = 0.01$.
In figures 6.1-6.4 we present the snapshots for the distributions of $Y(H_2)$, $Y(O_2)$, $Y(H_2O)$ and T at times $t = 0, 4$ s. More distributions are presented in the main paper variant.



(A) $Y(H_2)$ (B) $Y(H_2O)$ (C) $Y(O_2)$

FIGURE 6.1: Mass fractions at $t = 0$ s

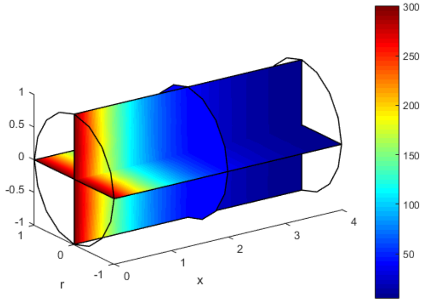
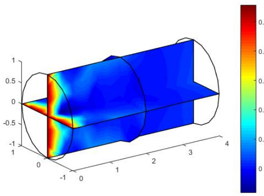
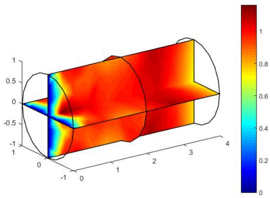


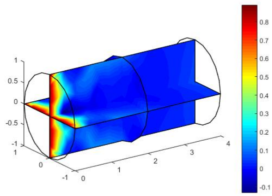
FIGURE 6.2: T at 0 s



(A) $Y(H_2)$



(B) $Y(H_2O)$



(C) $Y(O_2)$

FIGURE 6.3: Mass fractions at $t = 4$ s

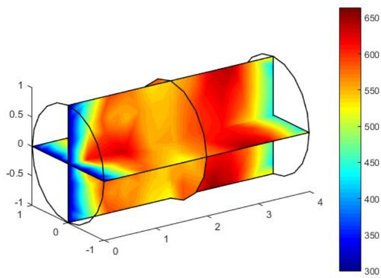


FIGURE 6.4: T at 4 s

According to the rates of corresponding reactions in (6.6) we obtained the decay of the first and the third mass fraction, growth (tendency to unity) of the second and the distribution of temperatures according to (6.12).

For the chosen parameters (fully determined in this example), at 4 seconds, the maximal temperature is close to 600 K and the outlet temperature is close to 450 K.

The numerical simulation takes several minutes to perform. More on computational speeds in the last section of this chapter.

This application is good start for considering the next one 6.2, where we extend the number of reacting species. Also it motivates us for considering the application in section 6.3, whereas we have to optimize the parameters since the parameters in this one are chosen to be determined only for sake of actually being able to perform the computations and aren't at all commonly approved. The parameter region usually is given. More on that in sections 6.3, 6.4.

6.2 Solving PDEs for the several reaction model

The model with three combustion reactions

$$O_2 + CO \to O + CO_2,$$
$$CO_2 + H \to CO + OH,$$
$$H_2 + OH \to H_2O + H \tag{6.17}$$

is now considered.

The applied framework is presented in the main thesis variant.

6.3 Parameter optimization for 3D model with one or several criteria

We would like to start with the first optimization application. This would be the three dimensional setting with a single reaction as in section 6.1, but the parameters are not explicitly given this time and the interval thereof is estimated.

For each experiment we consider the reduced order solutions as in 4.2 and perform it in a way that is is easy to store the data as in 4.3.

The region is deducted from the fact that there are similar reactions (with similar list of species, but not exactly the same), that can be found in Chemkin format table [20 04] or other published works, which are empirically obtained. The author then gathers data from these tables of works and estimates the interval. In figures 6.5a - 6.6b the one is represented by the red square. Following the notation from Chapter 4, our vectors of boundary values $\mathbf{b}_{lo}$ and $\mathbf{b}_{up}$ are respectively $\begin{pmatrix} 1.7 \cdot 10^{-3} \\ 47780/100 \end{pmatrix}$ and $\begin{pmatrix} 1.7 \cdot 10^{1} \\ 47780 \cdot 100 \end{pmatrix}$.

The division, multiplication by a hundred as well as the power of one have intentionally been left as they are so the nature of obtaining the parameter interval is visible.

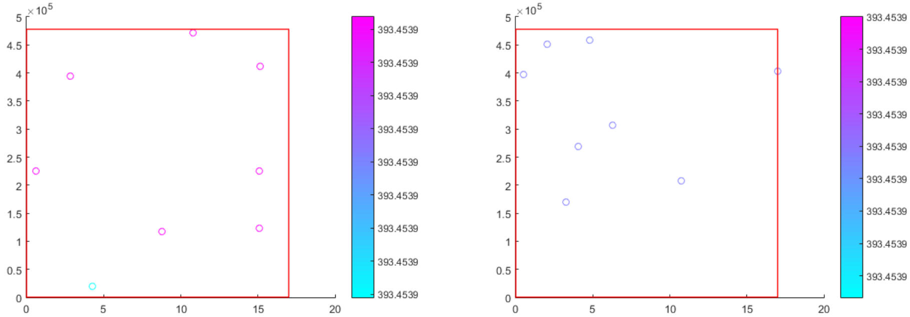
This estimates are obtained by gathering as much as possible data for similar reactions since this particular reaction hasn't been found there in its very form. For different reaction mechanisms this approach has given good results.

For this particular application we compare the maximal temperature $\max_{\mathbf{x}\in\Omega} T(\mathbf{x})$ with the experimental value of 740 K.

The functional choices had been described in Chapter 4 and is due to the formula (4.1).

One can see in figures 6.5a - 6.6b the motion of two dimensional particles. The blue dot represents the winning state. The winning parameter values are approximately $\begin{pmatrix} 10.44 \\ 5235.25 \end{pmatrix}$.

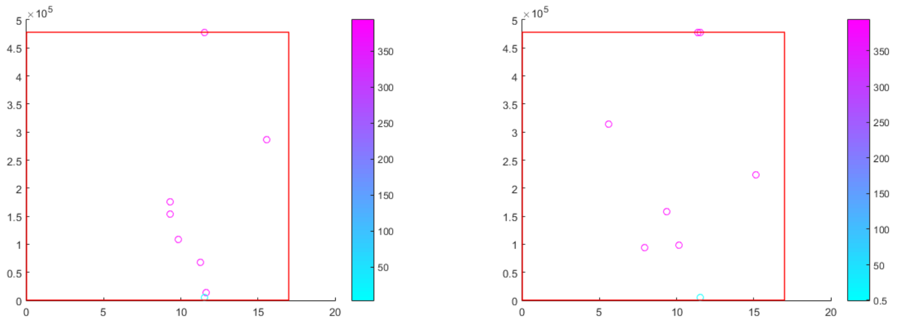
The measure in this example has been obtained and kept small (less than 1 K).



(A) 1st iteration

(B) 3rd iteration

FIGURE 6.5: Particle positions at $k = 1, 3$



(A) 15th iteration

(B) 30th iteration

FIGURE 6.6: Particle positions at $k = 15, 30$

As discussed previously, for the proposed goals we are totally fine with only one argmin that brings the modelling data close to experimental data. The series of further numerical experiments (see main paper variant) can show that the close to zero distance can be obtained at different points.

The process of obtaining this measure took around 6-10 hours when the initial guess wasn't close enough. When the initial guess was close enough, the time could be substantially reduced, however for this application the random positions are chosen at the beginning. This motivates us to use the model reduction techniques when necessary from chapter 4 section 4.2 and the pattern storage application from 4 section 4.3. The former would reduce the amount of hours whilst keeping the resulting quantities close and the latter shall define the auxiliary problem to be called out beforehand.

The algorithm 4.2.1 introduced in chapter 4 for the optimization with the reduced states can be applied when necessary. The motivation is to compute the PDE solutions for the optimization purposes in the reduced basis most of the times and come up with several stages of the functional value rechecking. This should be a time saver to a certain extent. The figure 6.7 illustrates its behaviour.

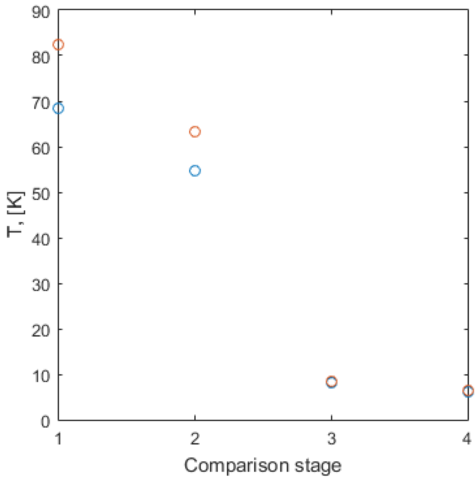


FIGURE 6.7: Four stages of comparison

This is a simple illustration on how the optimization procedure can solve the PDE system mostly with reduced states. The number of PSO iterations (experimentally, usually N is around 30 is enough) is split into several parts (N_S - number of stages - four on 6.7) in order to apply the algorithm 4.2.1. The matrix of snapshots and its SVD is recalculated at each stage. According to the algorithm these are the only times when the solution gets calculated in full basis and thus for N $\gg N_S$ it gets calculated in reduced bases only.

For the considered types of models, experimentally, it is possible to reduce the dimension by $10\% - 20\%$ and number of stages N_S around $4 - 6$ is enough. This is a time saver, but the whole procedure still takes hours to perform and motivates us to consider the pattern storage concept in 6.4. However when the experiments are not connected, we consider completely new model, or it gets substantially modified, etc, the PCA is always a good way to start.

The problem can become more complex when we introduce several criteria. Theoretically this concept can be used to solve the whole closest curve or surface finding task by using the proposed techniques. This would be the topic of further researches and to show that this works in the present paper we extend the number of criteria by one. Same task will be solved in storage pattern section 6.4 by using present results.

So we consider the PSO iterations with 2 criteria:
$T_{max} = 740$ K and $T_{out} = 450$ K.

30 iterations of minimizing the functional (4.1) takes the same time as in case with one criterion but since the dimension has been increased the value of the functional can be not as close to zero as in case with one criterion. Hence additional iterations are required if one wishes to keep the measure between experimental and modelling data small and this motivates us even more to consider pattern storage in 6.4 in case if there are more parameter optimization tasks with similar model to go.

The convergence obtained obviously mean that we approximately reach the desired temperature values. We'll show the one dimensional projections of these 3D graphics after the pattern storage procedure in the next section.

6.4 Pattern storage results

The nature of figures 6.5a - 6.6b for section 6.3 is the processing of the optimization intermediate states. In this simple example we store the data of the evolution of the winning (blue) particle from 6.5a - 6.6b. Form a discrete dynamical system as in (4.14). Then call it out with the new experimental data and unknown (but guessed as an initial state) parameter values. Then see how close is the result obtained by the system (4.14) in terms of measure (4.1). In case of necessity it can be used as a starting guess in the PSO (4.3), (4.4) and not apply the randomization in the beginning. We show how the computational time can be substantially reduced.

The first experiment is as follows: we store the winning trajectory in (4.3), (4.4) along with the maximal temperatures at last time step.

We create the discrete dynamical system by applying (4.16).

Then assume then the next experiment has been held with the new value of 640 K. Remember that 30 iterations for 740 K took around $6 - 10$ machine computation hours depending on the initial randomization.

The computational time of creating the patterns depends on condition in the result 4.3.1 and randomization but usually takes several minutes for chosen dimension.

The iterations of the discrete dynamical system takes several seconds.

We don't know the parameters - all we know is the new maximal temperature value at last time step, which is 640 K. We choose the values in the allowed intervals for two parameters and feed the system the matrix $\begin{pmatrix} 6 & 4 & 2 \\ 1 & 0 & 1 \\ 4 & 1 & 4 \end{pmatrix}$. Since $\mathbf{b}_{lo}$ and $\mathbf{b}_{up}$ are taken from section 6.3 and respectively are $\begin{pmatrix} 1.7 \cdot 10^{-3} \\ 47780/100 \end{pmatrix}$ and $\begin{pmatrix} 1.7 \cdot 10^{1} \\ 47780 \cdot 100 \end{pmatrix}$,

this is a good place to start.

After several iterations of the processes the machinery has produced the vector $\begin{pmatrix} 640 \\ 10 \\ 8200 \end{pmatrix}$, which is too good to be true so solving the system of PDEs (2.1), (2.8) with these parameters was necessary which gave the close enough value between 635 and 645 Kelvins. The figure (6.8) shows the closeness to the temperature profile.

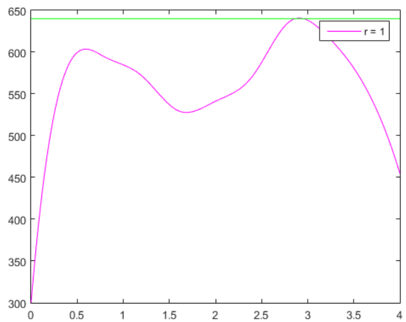


FIGURE 6.8: T at 4 s - max T around 640 K

In table 6.4 we see how we have substantially reduced the computational time for the new experiment. The times are in seconds as results of tic and toc in Matlab.

T value/method	PSO	DDS with PC	DDS	PDE
740 K	37412	-	-	-
640 K	-	160	-	119.4
700 K	-	-	32.2	119.3

It all started (first row in table 6.4) with the full PSO for 740 K which took around 10 hours to perform on a powerful but not super computer.

The sequence of experiments afterwards is considered whereas we do not wish to wait that long and use the leftover information of the optimization trajectories.

For illustration we considered two subsequent experiments with finding the parameters for solution with maximal temperatures of 640 and 700 K (second and third rows of table 6.4).

DDS stands for iterating the discrete dynamical system and the one with PC (stands for pattern creation) is when we create the patterns first. This took between 2 and 3 minutes and gave an approximation since the standard forms of numbers had been considered and by feeding to the system the trajectory of only one particle there is no guarantee the chosen maximal values are close to the values on the trajectory.

Therefore the one last solution of PDE (the last column) were called out with the wait time of around about 2 minutes.

For the experiment in the 3rd row and all the hypothetical subsequent experiments there is no need for pattern creation so it takes around 30 seconds to iterate the DDS several times and find the approximated parameter values.

In comparison with lots of hours if we had to optimize the parameters every time this is a significant time saver.

In the next application we consider the two criteria for optimization - maximal and outlet temperature with the model as in the last part of section 6.3. This would demonstrate that for more complicated problem the pattern storage is a life saver for the sequence of subsequent experiments and if there are even more sophisticated potential applications the framework can be applied as well. Details presented in the main paper variant.

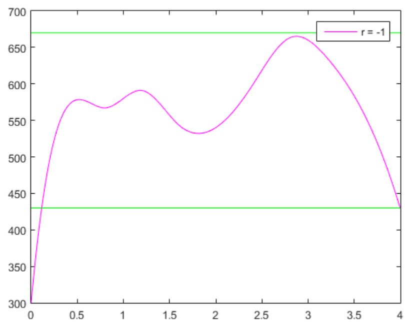


FIGURE 6.9: T at 4 s - max T around 670 K, out T around 430 K

T values/method	PSO	DDS with PC	DDS	PDE
740, 450 K	25816.8	-	-	-
670, 430 K	-	868.9	-	105.9
700, 440 K	-	-	277.7	107.5

One can conclude that for two criteria the timings haven't been significantly increased (even decreased at some point but this is as usual a matter of an initial guess) and in case of necessity one can increase the number of criteria and the codes have been created such as we can easily implement it.

Formulation of the problem whereas the number of criteria is greater than the number of parameters is also possible after examining the behaviour of the solutions for different values of parameters. This can lead to values of measure not as close to zero. That is also the case for the 'improper' choice of criteria values when the numbers coincide or the number of criteria is smaller than the number of parameters. The nature of our considered problems however (comparison of modelling and experimental data and modification of models therefore) makes us analyse the experimental data first and then formulate the optimization problem. In case when the measure is obtained inadmissible one may consider reformulating the optimization problem. The figure 6.10 shows how the measure can be obtained in case when the number of criteria is larger (we've added the third one - the temperature value in the middle) and the formulation of the corresponding optimization problem and the execution of the algorithms is still possible.

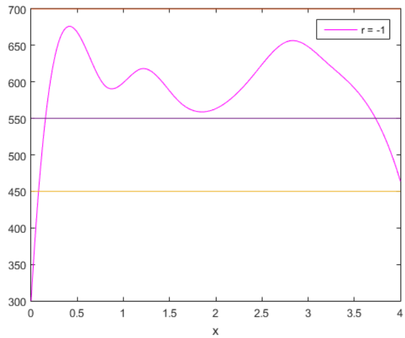


FIGURE 6.10: T at 4 s - desired max T - 700 K, out T - 450 K, mid T - 550 K.

In similar fashion we can continue the experiments for the chosen types of models, which shall be a matter of author's further researches, or can apply the obtained machinery for different types of models (not even necessarily described by PDEs).

This ends the applications section. We have chosen illustrative examples to show how the machinery works and given pictorial and tabular representations of our results.
The formulation of the conclusions of the whole thesis is the subject of the last pages as well as the list of author's publications and the list of literature.

Chapter 7

Conclusions

Sticking to the plan proposed in chapter 1 led to results applicable in the field of combustion and chemical kinetics experiments. The techniques used for that are obviously applicable elsewhere as well since the code had been created such that its separate parts can be used as a part of another code for the field or in other various fields.

In particular, the framework when we know the list of chemical reactions in the model and don't know some of their parameters but do know how to estimate their regions has been created. Very briefly at the end of the day:

In chapter 2 we've learned how to take the list of forward chemical reactions and construct the source terms in the equations of the PDE system. The other classical terms of the equations of the PDE system have been discussed as well.

In chapter 3 we've chosen universal methods to be safe to obtain the feasible solutions of the PDE system and have presented the necessary theory of weak solutions, function space approximation, finite element method and the linearisation.

In chapter 4 we went step further, considering the process of obtaining the numerical solution has been developed, we have presented the parameter optimization technique and at the same time concluded that the process is time demanding and thus considered the model reduction and pattern storage and recognition techniques.

In chapter 5 (see main paper variant) the simple test codes in Matlab have been presented to serve as building blocks for the general code governing all the techniques presented in the thesis.

In chapter 6 some graphical and numerical results along with the tables have been presented where one can see that proposed techniques in the thesis work very well in simple cases.

The main author's contribution is the application of different modern techniques not yet ever used in this particular combination for the combustion and reaction kinetics applications along with their modifications and adopting them for this particular matter.

The pattern storage results are something the author had come up within the last steps of gathering data for the thesis and the simple examples such as single and two criteria recognition has only been considered, but theoretically this concept can be used to solve the whole closest curve or surface finding task by using the proposed machinery. This would be the topics of author's further researches.

Appendix A

Author's participation in published works

Publications presented here are the ones author has been involved in for the last couple of years whilst researching the topic.

These have a lot of co-authors too since there have been a lot of colleagues in several projects the author has been involved in.

Amongst the wider list of publications and proceedings with author's contributions on similar but not exactly the same topic, we present the list of thematically connected publications on the topic of this particular thesis only and omit most of the conference proceedings.

- On the numerical simulation of the vortex breakdown in the combustion process with simple chemical reaction and axial magnetic field, International Journal of Differential Equations and Applications, Volume 14, No. 3, 2015, p. 235-250. Coauthors: H. Kalis, U. Strautiņš, O. Lietuvietis.

- Magnetic Field Control of Combustion Dynamics. Latvian Journal of Physics and Technical Sciences, Vol. 53, N4, p. 36-47, 2016. Coauthors: I. Barmina, R. Valdmanis, M. Zaķe, H. Kalis, U. Strautiņš.

- On numerical simulation of electromagnetic field effects in the combustion process. Mathematical Modelling and Analysis, 23(2), p. 327-343, 2018. Coauthors: H. Kalis, U. Strautiņš, M. Zaķe.

- Experimental Study and Mathematical Modelling of Straw Co-Firing with Peat. Chemical Engineering Transactions, Vol. 65, p. 91-96, 2018. Coauthors: I. Barmina, A. Kolmičkovs, R. Valdmanis, M. Zaķe, H. Kalis.

- Experimental study and mathematical modelling of straw co-firing with propane. Chemical Engineering Transactions, Vol.74, p. 19-24, 2019. Coauthors: I. Barmina,
  A. Kolmičkovs, R. Valdmanis, M. Zaķe, H. Kalis.

- Mathematical Modelling and Experimental Study of Straw co-Firing with Gas. Mathematical Modelling and Analysis, Vol. 24, N4, p. 505-529, late 2019. Coauthors: I. Barmina, H. Kalis, A. Kolmičkovs, L. Ozola, U. Strautiņš, R. Valdmanis, M. Zaķe.

- Mathematical Modelling and Experimental Study of Straw co-Firing with Gas Using Electric Field Control of Combustion Characteristics. Contents of Proceedings of 19th International Scientific Conference Enginering for Rural Development, p. 1059-1064, 20.-22.05.2020, Jelgava, Latvia. Coauthors: H. Kalis, A. Kolmičkovs, R. Valdmanis.

The present results are to be published in 2021.

Appendix B

Author's participation in international conferences

The list contains all the events during the last couple of years, whereas the author participated with the thematically connected presentations.

- On the vortex formation in the combustion process with simple chemical reaction and axial magnetic field // Mathematical Modelling and Analysis (MMA 2015) : 20th International Conference, Sigulda, Latvia, May 26-29, 2015 : abstracts Riga : University of Latvia, 2015. p. 43.
- On the numerical simulation of the combustion process with simple chemical reaction, 7th Baltic Heat Transfer Conference, August 24-26, 2015, Tallinn, Estonia.
- Mathematical modeling of MHD flow in the combustion process . Acta Societatis Mathematicae Latviensis, Abstr. of the 11-th Latvian Mathematical Conference,
  15.04.2016, Daugavpils, Latvia, p. 37.
- On mathematical modelling of the combustion process of biomass. Acta Societatis Mathematicae Latviensis, Abstr. of the 11-th Latvian Mathematical Conference,
  15.04.2016, Daugavpils, Latvia, p. 38.
- On numerical modelling of swirl flow in the combustion process // Mathematical Modelling and Analysis (MMA 2016) : 21st International Conference, Tartu, Estonia, June 01-04, 2016 : abstracts Tartu : University of Tartu, 2016. p. 36.
- Experimental and numerical study of the development of swirling flow and flame dynamics and combustion characteristics at biomass thermo-chemical

conversion. 16th International Scientific Conference Enginering for Rural Development, 24.-26.05.2017, Jelgava, Latvia.

- Effects of gradient magnetic field on swirling flame dynamics, 16th International Scientific Conference Enginering for Rural Development 24.-26.05.2017, Jelgava, Latvia.
- Electric field effects on gasification/combustion at thermo-chemical conversion of
  biomass mixtures. 16th International Scientific Conference Enginering for Rural Development, 24.-26.05.2017, Jelgava, Latvia.
- On numerical simulation of electromagnetic field effects in the combustion process. Abstr. of MMA2017, May 30-June 02, 2017, Druskininkai, Lithuania, p. 41.
- Mathematical Modelling on Electromagnetic Field Control of the Combustion Process, Modelling for Materials Processing, Riga, September 21-22, 2017.
- On mathematical modelling of the chemical reactions for two-dimensional diffusion flames. Acta Societatis Mathematicae Latviensis, Abstr. of the 12-th Latvian Mathematical Conference, 13.04.2018, Ventspils, Latvia, p. 40.
- Development of combustion dynamics at co-combustion of straw with wood. 17th International Scientific Conference Enginering for Rural Development, 23.-25.05.2018, Jelgava, Latvia.
- Influence of electric field on thermo-chemical conversion of mixtures of straw pellets with coal. 17th International Scientific Conference Enginering for Rural Development, 23.-25.05.2018, Jelgava, Latvia.
- Mathematical modelling and experimental study of co-firing straw with gas. Abstr. of MMA2018, May 29-June 1, 2018, Sigulda, Latvia.
- Numerical study of electrodynamic control of straw co-firing with propane. Engineering for Rural Development: 18th International Scientific Conference, May 22-24, 2019, Jelgava, Latvia.
- Mathematical Modelling and Experimental Study of Straw co-Firing with Gas Using Electric Field Control of Combustion Characteristics. Engineering for Rural Development: 19th International Scientific Conference, 20.-22.05.2020, Jelgava, Latvia.

Bibliography

[20 04] Kinetic Mechanism for 20 torr. In: (2004). URL: http://web.mit.edu/anish/www/pub20torr.mec.

[Ang03] P. Knabner, L. Angermann. "Numerical Methods for Elliptic and Parabolic Partial Differential Equations". In: *Springer* (2003).

[BAE05] F. van den Bergh, A.P. Engelbrecht. "A study of particle swarm optimization particle trajectories". In: *Information Sciences* (2005).

[Ben13] M.G. Larson, F. Bengzon. "The Finite Element Method Theory, Implementation, and Applications". In: *Springer* (2013).

[Bor01] E.S. Oran, J.P. Boris. "Numerical Simulation of Reactive Flow". In: *Cambridge University Press, NY USA* (2001).

[Cha01] D. Mandic, J. Chambers. "Recurrent neural networks for prediction: learning algorithms, architectures, and stability". In: *John Wiley* (2001).

[Cle06] M. Clerc. "Particle Swarm Optimization". In: *Iste* (2006).

[Fou03] R.A. Adams, J.J.F. Fournier. "Sobolev Spaces". In: *Elsevier* (2003).

[Gre08] M.A. Singer, W.H. Green. "Using adaptive proper orthogonal decomposition to solve the reaction–diffusion equation". In: *Applied Numerical Mathematics* (2008).

[Hac17] W. Hackbusch. "Elliptic Differential Equations Theory and Numerical Treatment". In: *Springer* (2017).

[Mar16] I. Barmina, R. Valdmanis, M. Zaķe, H. Kalis, U. Strautiņš, M. Marinaki. "Magnetic Field Control of Combustion Dynamics". In: *Latvian Journal of Physics and Technical Sciences, Vol. 53, N4, p. 36-47* (2016).

[Mar18a] H. Kalis, U. Strautiņš, M. Zaķe, M. Marinaki. "On numerical simulation of electromagnetic field effects in the combustion process". In: *Mathematical Modelling and Analysis, 23(2), p. 327-343* (2018).

[Mar18b] I. Barmina, A. Kolmičkovs, R. Valdmanis, M. Zaķe, H. Kalis, M. Marinaki. "Experimental Study and Mathematical Modelling of Straw Co-Firing with Peat". In: *Chemical Engineering Transactions, Vol. 65, p. 91-96* (2018).

[Mar19] I. Barmina, A. Kolmičkovs, R. Valdmanis, M. Zaķe, H. Kalis, M. Marinaki. “Experimental study and mathematical modelling of straw co-firing with propane”. In: *Chemical Engineering Transactions, Vol.74, p. 19-24* (2019).

[Mil00] R.J. Kee, F.M. Rupley, J.A. Miller. “The Chemkin Thermodynamic database”. In: *Reaction design, Sandia National Laboratories Report SAND87-8215B* (2000).

[Pfe12] W.F. Pfeffer. “The Divergence Theorem and Sets of Finite Perimeter”. In: *Taylor and Francis* (2012).

[Tab20] Dynamic Periodic Table. In: (2020). URL: https://ptable.com/.

[Vey05] T. Poinsot, D. Veynante. “Theoretical and Numerical Combustion”. In: *Edwards, PA USA* (2005).

[Wil10] M. Buffoni, K. Willcox. “Projection-based model reduction for reacting flows”. In: *40th Fluid Dynamics Conference and Exhibit* (2010).

[Wil85] F.A. Williams. “Combustion Theory. The Fundamental Theory of Chemically Reacting Flow Systems”. In: *Taylor and Francis, FI USA* (1985).

[Woo02] D.M.T. Hall, D.L. Torek, P.V. Schrock, C.R. Wooldridge. “H2/O2 reaction mechanism”. In: *M.S., Proc. Combust. Inst. 29 (2002), in press* (2002).