Popular Science Summary

The foundation of modeling physical phenomena is their description by mathematical models, most prominently *differential equations* (*DE*). *DE*s are used to model phenomena in all natural sciences, engineering disciplines and more: Classical mechanical engineering applications such as construction or vehicle design, electrical field and circuit modelling, simulation of chemical reactions, climate and vegetation simulation, infectious disease modelling and many more, are all based on *DE*s, albeit each with their own nuances and flavors.

*DE*s relate rate of change to a given state and known inputs. Solving a *DE* means to determine future states based on an initial state. An example is Newton's second law: *Force* = *mass* $\times$ *acceleration*. Acceleration is the rate of change of velocity, which is the rate of change of position. Solving this *DE* means to determine future position based on a known initial position, force and mass.

Analytical solutions, obtained via solving the *DE* by hand, are extremely rare. Instead, with the rise of computers and computing power, we solve them using *numerical methods*, algorithms designed to approximate their solutions. This enables the modelling of highly complex physical phenomena. In particular, product design and development based on numerical models is typically faster, cheaper and safer than traditional methods using physical prototypes.

While artificial intelligence and machine learning are very hot topics right now, these are unlikely to replace physical based modeling using *DE*s, since machine learning is mostly used for finding patterns in data. However, most physical phenomena are well described by *DE*s, the difficulty is to accurately, yet efficiently solve them. Thus, the development of *efficient numerical methods* is of key importance, since it enables e.g., more precise weather forecasts and climate change predictions. Additionally, current development on increased computing power involves parallel machines and making use of them requires suitable numerical methods.

This thesis consists of three parts centered around *numerical methods* for *DE*s. One solves

these in a step-wise manner, yielding a solution at fixed time-points, e.g. weather data every 15 minutes. Smaller steps give a more accurate result, but increase the cost of computing the whole solution. Another approach is to *adaptively* choose stepsizes. That is choosing them as large as possible and as small as necessary to reduce computational cost, while retaining a target accuracy. *Adaptive* approaches are based on estimating the accuracy and then choosing a suitable stepsize.

In the first part of the thesis we developed and analyzed *adaptive* methods in the context of *goal-oriented problems*. In practice, one solves *DE*s to answer questions such as "How much energy does this turbine produce?", "How fuel efficient is this vehicle design?", "Is component X sufficiently cooled?" or "How fast are the polar caps melting?". Here, the goal is the computation of a *quantity of interest (QoI)* (e.g., average energy produced, maximum temperature, total ice loss per year) based on the solution of the *DE*, to answer the relevant question.

We derived an *adaptive* method for the solution of *goal-oriented problems* that chooses stepsizes to control the accuracy in the *QoI*. Additionally, we established mathematical conditions for when such an approach is sensible, with guidelines to predict performance gains or losses. Our new method shows notable performance improvements in various test cases.

The remaining part of the thesis concerns *coupled problems*. These are physical phenomena described by a multitude of connected *DE*s. An example is thermal fluid-structure interaction in the cooling of rocket nozzles. This involves simulating the interaction between extremely hot combustion products, the solid nozzle structure, liquid coolant within the structure and the ambient air outside. Another example is climate simulation, featuring connected *DE*s for atmosphere/ocean dynamics + chemistry, vegetation, cloud and ice formation, and possibly more.

There are different approaches in designing simulation software for coupled problems. The *monolithic approach* is to view the coupled problem as a whole and to design and implement a tailor-made algorithm to solve it. However, changes or additions to the coupled problem require changes to the whole algorithm. Additionally, designing an efficient algorithm requires the combined expertise of doing so for all subproblems.

We consider the so-called *partitioned approach*, which aims to solve a coupled problem using solvers for the subproblems combined with a suitable coupling method. Partitioned coupling methods typically involve an iterative process in which the subproblems need to be solved multiple times each. The advantage of the partitioned approach is that it allows re-using existing specialized codes for the subsolvers. In particular, this does not necessarily require a lot of specialized knowledge on the subproblems.

The re-use of software is a pre-requisite for sustainable software development and being able to jointly work in larger groups. This approach makes it far easier to exchange subsolvers

or add new components to a coupled problems.

We studied *Waveform Relaxation (WR)* methods, an iterative partitioned coupling method that allows a large degree of independence in solving the subproblems. This is highly desirable when coupling different physical phenomena that require different numerical methods or stepsizes for solving them.

First we developed a *WR* method for solving dynamic heterogeneous coupled heat equations. That is, modeling the time-dynamics of heat exchange between different materials as e.g., present in the fluid-structure interaction when cooling rocket nozzles. The resulting method performs better than existing methods, by using independent and *adaptive* stepsizes when solving the subproblems, resolving the coupling reliably in very few iterations.

Secondly, we developed a novel parallel *WR* method, which requires fewer iterations than classical parallel *WR* methods, by utilizing asynchronous communication techniques. The resulting method shows promising performance results. Its analytical description and theoretical results generalize existing theory on *WR* methods.

Lastly, we developed software for solving coupled problems using *WR* methods. In particular, we are able to couple *DE*s implemented using certain free open-source packages that provide a vast range of tools for solving complex *DE*s at relative ease. With this, we are able to couple different *DE* subsolvers almost at a "plug-and-play" principle, to create solvers for multiphysics problems.