Chapter 1

Introduction

As powerful computers are widely available for reasonable prices, computer simulations of complex processes gain more and more importance. A wide range of different needs for computer simulations exist, of which most, however, rely on the solution of partial differential equations. Examples are computational solid mechanics, computational fluid mechanics, simulation of wave scattering etc. Especially the field of computational fluid dynamics requires sophisticated techniques to solve the equations, which describe the physical problems. At the *Institut für Verbrennung und Gasdynamik* (*Institute for Combustion and Gas-Dymanics*) people have been working an numerical methods, to solve fluid dynamical problems, for a decade, now. The staff fluctuations have been quite large during this period (PhD theses, guest researchers, students, etc.). At the start of this work, a number of different programs for different fluid dynamics applications have been in use at the institute. The work of R.Vilsmeier [1] shall be particularly mentioned here, since it forms the numerical and algorithmic basis for the new developments, made in the scope of this work. A closer look at some of the codes, however, showed that some aspects occurred in different forms in all of them (e.g. mesh-handling, file-handling, certain numerical methods, etc.). Especially someone working with unstructured grids needs to do lots of work, before getting started to solve the real problem.

This work has been funded as part of a project dedicated to the *Development of Navier-Stokes Solvers on Hybrid Grids*. It has been a cooperation with the *Projet SINUS* at the *INRIA* in Sophia Antipolis in France. In the beginning, it had to be decided if the existing unstructured codes, mainly written in Fortran, should be used. The alternative was to redesign the software in an object-oriented manner. Discussions within the workgroup at the institute made clear that maintenance and extensions of the existing codes got more and more difficult. It cannot be said, finally, if it would have been wise to keep developing the original codes. Possibly the achievements in solving aerodynamical problems would have been better in that case. After all a lot of code had to be rewritten in *C++*. Other projects at the institute, however, which now use unstructured grids, might not have done so, due to the enormous efforts in the beginning. The amount of coding work to write an unstructured solver is quite high. If then the code has to be parallelized, someone starting a new project might fall back to a simpler approach (e.g. Cartesian methods). This might happen, even if the advantages of an unstructured approach would be obvious,

simply due to the limited time a researcher has. If now an existing library could be used, the necessary development work can be drastically decreased and people can concentrate on the numerical task they have to solve.
Once a decision has been made in favor of a new and object-oriented development, it seems desirable to keep the developments as independent as possible from the specific problems to be solved. Thus the software, which arose from this project, is not even restricted to computational fluid dynamics. The goal has been to develop a numerical toolbox, which integrates all necessary steps of a simulation process into a single platform. This should include pre- and post-processing, as well as the numerical simulation itself. Especially the integration of the visualization process proved to be very efficient while debugging numerical programs. The new developments have been made available in form of a *C++* library.
This text describes the numerical aspects, as well as the software design. Since fluid dynamics have been the main field of application a more detailed description of the *Navier-Stokes Equations* can be found in chapter 2. Chapter 3 will give an insight in the numerical basis. Furthermore it tries to outline that flexibility has been a major guideline, not only in terms of software design, but also with respect to the numerical approaches used. One numerical aspect, namely the FAS multi-grid which has been used, will be mentioned in chapter 5. The necessary developments have been a very close co-operation with L.Fournier, who at the time has been working in Sophia Antipolis. An alternative algorithm for unstructured mesh generation will be presented in chapter 4. Chapters 6 and 7 will go into detail of the software-design. It has been tried to outline the benefits, as well as the pitfalls related to object-oriented programming, if used for numerical simulations. A few example simulations will be shown in chapter 8. Finally chapter 9 tries to summarize the achievements as well as the problems and will also try to give few perspectives for future work. Also a small introduction of some of the projects, making use of the numerical library, will be given in that chapter.
The mathematical notation, especially everything related to unstructured grid indices, is explained in appendix A.