Implementation and Application of the Multiblock NSFLEX Navier-Stokes Solver
K.M. Wanie, M.A. Schmatz
Messerschmitt-Bölkow-Blohm GmbH, FE211, Postfach 80 11 60, D-8000 München 80, Germany

ABSTRACT

The need of design aerodynamics to calculate viscous flow past complex configurations like complete aircraft with wings, fins, flaps and external stores represents one of the major challenges for computational fluid dynamics. If a structured grid topology is to be applied, which has proven to be very advantageous for viscous calculations at least near solid surfaces, a block decomposition of the space around the vehicle can simplify grid generation considerably. Furthermore storage requirements can be reduced by such a decomposition. For this purpose the present paper focuses on the implementation of a multiblock technique into the Navier–Stokes code NSFLEX. The governing equations and the solution method are shortly reviewed. The fundamental details of the multiblock incorporation like indirect addressing and use of boundary indicators are described thoroughly. The results achieved for hypersonic flow past several configurations demonstrate the capabilities of the approach.

INTRODUCTION

The design of new aerodynamic shapes requires a detailed investigation of the flow properties past entire configurations. With the advent of modern high speed computers computational fluid dynamics (CFD) is more and more applicable for these investigations and has become an important tool of design aerodynamics. It is commonly used today to support the experimental work.

Advantages of CFD as compared to experiments are lower cost and lower turn-around times. This holds especially for the design of hypersonic cruise aircraft or re-entry vehicles. Due to the high enthalpies of the free-stream flows windtunnel experiments are very expensive in the hypersonic flow region. A
tremendous technical effort is necessary to simulate effects like radiation, catalytic walls or chemical reactions correctly. CFD is in principle suited to overcome these problems, provided a suitable mathematical modelling for the particular effects is available. The development of these models can, of course, be a very difficult task.

In order to get valuable contributions to the design process it will always be necessary to compute the flow past entire complex configurations including wings, fins, flaps and external stores. For viscous calculations it is furthermore mandatory to use a very good spatial resolution in the boundary-layer region and it has proven to be advantageous to use a structured grid at least in this part of the flow field. Bearing these constraints in mind it is obvious that the generation of a single block grid can be very difficult or even impracticable.

The above consideration lead to a multiblock decomposition technique for the discretisation of the flowfield in a natural way. Advantages of such a multiblock technique are considerable simplifications of the grid generation procedure and a reduction of the storage requirements, which is very important especially with regard to large three-dimensional problems.

The present paper focuses on the implementation of a multiblock technique exploiting these advantages into the Navier-Stokes code NSFLEX. The single-block version of NSFLEX was widely used and verified in the last years for a large number of different applications ranging from low Mach number subsonic flow past cars to very high Mach number hypersonic flow past re-entry vehicles [1,2]. The code is able to solve both the full or thin-layer Navier-Stokes and the inviscid Euler equations.

The overall goal of the present work is to end up with a numerical method suitable especially for engineering applications. For this purpose the code must be flexible enough to allow changes in flow conditions, geometry, grid topology or boundary conditions without a large amount of additional programming. Furthermore storage requirements have to be minimized in order to allow either for a fine spatial resolution, if desired, or to keep cost for the calculations small. As a last point it is desired to incorporate cases like two-dimensional or axisymmetric flow into the three-dimensional code. In this way the effort for the program maintenance can be kept as small as possible. The multiblock technique incorporated fulfills all these requirements due to a very flexible treatment of the block boundaries, a one-dimensional storage technique and generalized approach for 3D-, 2D- and quasi-2D-flows.

In the following the governing equations and some details of the fundamental computational method are given. After this details of the multiblock implementation are discussed thoroughly. Features like indirect addressing and treatment