The Performance of a Euler Solver Using a Distributed System

RENATO S. SILVA REGINA C. ALMEIDA

Visiting Scholars, Texas Institute for Computational and Applied Mathematics
The University of Texas at Austin

In the last years, with the advent of successful discretization methods and powerful computer technologies, computer simulation has been seen as a valuable approach to be combined with experimental methods to solve industrial and scientific problems.

In this work we are interested in the numerical simulation of compressible Euler flows for which we have to deal with large, sparse and non-symmetric systems. For these types of problems the computational cost is a major concern and so is the accuracy. Obviously, the accuracy is related with the appropriate choice of the numerical method to solve the Euler equations. Ideally, it should be stable in the presence of shocks and should have good convergence properties. A good step to this direction was the development of the Petrov-Galerkin Finite Element models, which modify the Galerkin's weighting functions by adding a perturbation term but keeping the consistency property in the sense that the exact solution satisfies the approximate problem. Moreover, the possibility of using discontinuous weighting functions, as first proposed with the SUPG Method (Streamline Upwind Petrov Galerkin Method) to the scalar convection dominated problems, opened a fascinating way to control the instabilities often present in approximate solutions with steep gradients. For the systems of equations, the perturbation term in this method acts only in the generalized streamline direction, chosen as the upwind direction. As its scalar counterpart, the resulting method has good stability and accuracy properties if the exact solution is regular. However, for non-regular solutions, those properties only worth out of a neighborhood containing sharp gradients, where spurious oscillations remain.

Considerable success has been achieved using the concept of generalized approximate upwind direction, introduced by the CAU Method (Consistent Approximate Upwind Method) for systems of equations. This idea yields a method that keeps the perturbation term over the generalized streamline direction and adds, in a consistent way, another perturbation which provides the control over the derivatives in the direction of the generalized approximate gradient, avoiding completely spurious oscillations. Besides, the accuracy is kept since the latter perturbation vanishes in the regions where the solution is regular. Those are the reasons that lead us to choose this method to solve the compressible Euler equations. However, despite of its good properties, this does not preclude the necessity of using an efficient solver to reduce the computational costs when large problems are concerned.

It is well known that iterative methods perform better than direct methods for large problems, but to determine the best iterative algorithm to solve non-symmetric systems is still an open issue. However, many conjugate gradient like methods have been widely use with relative success. Included in this class, the GMRES method introduced by Saad and Schultz seems to be a good choice because it seldom breaks down and presents monotonic convergence. But the method possesses a drawback because, in its original version, the orthogonalization process dictates an increasing amount of computational work with increasing iterations. This drawback can be avoided using a restarted GMRES version (GMRES(k)) since it consumes less memory than the original one while keeping its good properties.

A good solver means not only a choice of the best method but a combination between the method and a fast implementation. Recently considerable attention has been focused in heterogeneous distributed systems as a solution for high performance computations. By definition, a heterogeneous distributed system is a collection of different computers loosely connected by a Local Area Network (LAN) or/and a Wide Area Network (WAN). The popularity of these systems can be explained by the increasing performance and lower prices of high speed networks and general-purpose workstations. This means that is much cheaper to form and to maintain these systems than a massively parallel machine (MPM) and, in certain cases, their performances can be similar.

Unlikely MPMs, heterogeneous distributed systems can be formed with different types of processors and with different network types. However, they are composed by general-purpose enviroments, which means that it is possible to have different users running their jobs in the machines of the system. As a consequence, the load of each machine and the network traffic can vary at each instant and the choice of the communication pattern has to be done carefully. Thus, in this work, we present a distributed

implementation of the GMRES(k), for which the computational cost depends only on the dimension of the Krilov Space (k), and we show how the performance is affected by using two different global communication schemes: the Recursive Doubling Global Collect and the Bucket Collect algorithms. The former can be affected by network conflicts while the latter is less sensitive to such conflicts. Finally, we show that this algorithm running on a collection of 16 IBM RISC 6000 workstations is scalable with a relative small number of degrees of freedom.