

Airfoil shape optimization by coupling computational fluid dynamics with evolutionary multiobjective optimization

Daniela Zaharie^{1,2}, Silviu Panica^{1,2}, Marius Stoia-Djeska³,
Mircea Dragan^{1,2}, Dana Petcu^{1,2}

¹ Department of Computer Science, West University of Timisoara, Romania
{silviu,dzaharie,dragan,petcu}@info.uvt.ro

² e-Austria Institute, Timisoara, Romania

³ Politehnica University of Bucharest,
marius.stoia@rosa.ro

Abstract. The problem of coupling in an efficient way computational fluid dynamics and evolutionary multiobjective optimization codes in order to solve problems of optimal design is discussed. Both the problem of providing an easy to use framework and that of the computational cost are addressed. Moreover, a user interface was designed to allow the execution of different instances, with respect to the parameters of the evolutionary algorithm, of the combined code on several machines from a Grid infrastructure.

1 Introduction

The optimal design problems in aerodynamics have two main characteristics: the existence of multiple criteria to be optimized and the high computational cost of objective functions evaluation. For instance, in the case of airfoil shape optimization the aim could be to identify profiles that minimize the drag of the airplane but in the same time ensure a sufficient value for the lift and minimizes the difference between the actual pressure on the wing surface and a target pressure. Dealing with the tradeoff between these criteria means to solve a multiobjective optimization problem and to obtain a set of reciprocally nondominated solutions. For multiobjective optimization problems, the traditional gradient-based methods are not appropriate, thus derivative free stochastic methods, like evolutionary algorithms, became popular [3-5]. When applying evolutionary algorithms for optimal shape design there are at least two difficulties that arise: (i) the high cost of evaluating the population elements; (ii) the stochastic character of the evolutionary algorithms and their sensitivity to the parameters choice. A useful tool for designers would allow the easy choice of an evolutionary multiobjective optimization algorithm (EMOA) without requiring advanced knowledge in the field of evolutionary algorithms. On the other hand, it would offer support for parallel and/or distributed computing in order to make the computational fluid dynamics (CFD) computations efficient.

2 The problem

We will illustrate the main ideas of our approach for the optimization problem of airfoil shape. In this case, the aim is to find wing profiles, which optimize the aerodynamic forces, i.e. lift and drag. The profile is described by two Bezier curves, one for the upper surface and one for the lower surface, as in [3-5]. To define these curves a few control points are used and the coordinates of these control points are to be adjusted by the optimization procedure. For each profile, i.e. set of control points, the flow around the wing is simulated and the corresponding aerodynamic forces are estimated by a CFD code.

3 The CFD component

The CFD solver that we use in the context of our problem is based on a cell-centered, second order Godunov type finite volume discretization of the partial differential equation describing the flow. An implicit marching in time procedure allows us to obtain the final steady-state solution starting from an arbitrary initial state. To avoid the solution of large linear algebraic systems at each time step we use a dual-time approach combined with a fourth-order Runge-Kutta scheme. The boundary conditions are fixed in time. The convergence criterion is the reduction of the residuals under a predefined threshold. The CFD solver is implemented and available as executable code.

4 The EMOA component

The evolutionary optimization is based on a population of profile descriptions. Each element of the population is thus a set of coordinates of control points describing the upper and the lower curves defining the profile. Since these coordinates are real values a real coded EMOA, e.g. NSGA-II [2], is appropriate. Starting from a random initial population of profiles, new profiles are generated by specific mutation and crossover operations and those which are acceptable with respect to all optimization criteria are selected to survive in the next generation. The main problem with such an evolutionary approach is the high cost of the iterative process of evolution since at each generation each element of the population has to be evaluated with respect to all criteria. This means to generate/update the computational mesh corresponding to the current profile and to apply the CFD code. It is obvious that in such a case the most computational expensive operation is that of profiles evaluation which implies that an adequate parallelization strategy for EMOA would be based on the master-slave model [1]. The EMOA component is implemented in Java and uses MPIJava as communication library for parallel processing. Moreover it allows specifying the use of external codes in the evaluation of population elements.

5 A coupling framework

Having the CFD and EMOA codes independently implemented, the problem that arises consists in coupling them in order to obtain a functional evolutionary design of optimal shapes. The variant we propose is based on a small module ensuring the interface between the EMOA code and the CFD code. The main role of this module is to prepare the input data for the CFD code starting from the population element to be evaluated. On the other hand, we also designed a user interface for the coupling framework allowing the execution of different instances, with respect to the parameters of the EMOA, of the combined code on several machines from a Grid infrastructure. In this way, the user can deal with the stochastic character and parameters sensitivity of EMOAs and to improve the effectiveness of the approach.

Acknowledgment. This work is supported by the projects RO-CEEX 95/ 03.10.2005 (GridMOSI - UVT) and RO-CEEX 65/31.07.2006 (SIAPOM - IeAT).

References

1. E. Alba, M. Tomassini, *Parallelism and evolutionary algorithms*, IEEE Trans. Evol. Comput. 5 (6), pp. 443-462, 2002.
2. Deb K. et al. *A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multi-objective Optimization: NSGA-II*, in M. Schonauer et al (Eds.), Proceedings of PPSN 2000, LNCS 1917, pp. 849-858, 2000.
3. R. A. E. Makinen, P. Neittaanmaki, J. Periaux and J. Toivanen, *A Genetic Algorithm for Multiobjective Design Optimization in Aerodynamics and Electromagnetics*, In K. D. Papailiou et al., editor, Proceedings of the ECCOMAS 98 Conference, vol. 2, pp. 418-422, 1998.
4. N. Marco, J-A. Desideri, S. Lanteri, *Multi-objective optimization in CFD by genetic algorithms*, Rapport de Recherche INRIA, no. 3686, 1999.
5. H-K. Ng, D. Lim, Y-S. Ong, B-S. Lee, L. Freund, S. Parvez, B. Sendhoff, *A multi-cluster Grid enabled evolution framework for aerodynamic airfoil design optimization*, in L. Wang, K. Chen and Y. S. Ong (Eds.): ICNC 2005, LNCS 3611, pp. 1112-1121, 2005.