Nowadays the art of computer simulation has reached some maturity; and even for yet unsolved problems engineers have learned to extract meaningful answers and trends for their design from rough simulations: numerical simulation is one of the tools on which intuition can rely! Yet for those who want to study trends and sensitivities more rationally the tools of automatic differentiation and optimization are there. This book deals with them and their application to the design of the systems of fluid mechanics. But brute force optimization is too often an inefficient approach and so our goal is not only to recall some of the tools but also to show how they can be used with some subtlety in an optimal design program.

On the application side, this book deals with shape optimization problems for fluids and it also includes some aspects of optimization under fluid and structure coupling. These problems are of great practical importance in computational fluid dynamics (CFD) for airplanes, cars, turbines, and many other industrial applications.

Optimal shape design (OSD) has received considerable attention already. It has become vast enough to branch into several disciplines: on the theoretical side many results deal with the existence of solutions to the problem or its relaxed form, on the practical side topological shape optimization which solves numerically the relaxed problem or by local shape variation as carried out in this book. Because this book deals with fluid flow optimization we will not present the tools of topological optimization. Let us warn our potential readers that in many other aspects also the book is more a collection of case studies than a synthesis; it builds on known materials but it does not present these materials in a synthetic form for the simple reason of size and time, as the task of doing a survey would be somewhat formidable. The main application we had in mind deals with full airplane shape optimization; so the fluid is modeled by the full Navier-Stokes equations with a turbulence model. It is nearly impossible to compute the exact derivatives of the finite dimensional approximations of these equations with respect to shape and mesh motion, therefore we needed to use automatic differentiation (AD) of programs.
So the book begins with a chapter on optimal shape design by local shape variations for simple linear problems, discretized by the finite element method. The goal is to provide tools to do the same with the complex partial differential equations of CFD. A general presentation of optimal shape design problems and of their solution by gradient algorithms is given. In particular, the existence of solutions, sensitivity analysis at the continuous and discrete levels are discussed, and the implementation problems for each cases are pointed out. This chapter is therefore an introduction to the rest of the book. It summarizes the current knowhow for OSD, except topological optimization as well as global optimization methods such as evolutionary algorithms.

In Chapter 2 the equations of fluid dynamics are recalled, together with the $k - \varepsilon$ turbulence model, which is used later on for high Reynolds number flows. The fundamental equations of fluid dynamics are recalled; this is because applied OSD for fluids requires a good understanding of the state equation: Euler and Navier-Stokes equations in our case, with and without turbulence models together with the inviscid and/or incompressible limits. The $k - \varepsilon$ model is presented briefly with wall-laws. The domain of validity of the model is also recalled. We explain why wall-laws are potentially better for OSD than low-Reynolds number models. The wall-laws presented include pressure correction for separated flows and extension for compressible flows for both adiabatic and isothermal walls. It is interesting to notice that the large eddy simulation approach is giving a new life to the wall-function approach, as there is no hope to simulate high-Reynolds flows up to the wall with LES. Of course, by wall-laws we understand domain decomposition with a reduced dimension model near the wall. In other word, there is no universal wall-laws and when using a wall-function, it needs to be compatible with the model used far from the wall.

Chapter 3 deals with the numerical methods that will be used for the flow solvers. As in most commercial and industrial packages, unstructured meshes with automatic mesh generation and adaptation are used together with finite volume or finite element discretization for these complex geometries. The iterative solvers and the flux functions for upwinding are also presented here.

Then in Chapter 4 automatic differentiation is presented. First the theory, then a home-made C++ class for AD by operator overloading, and finally our experience with Odyssee, another AD system using code generation operating in both direct and reverse modes. As stated earlier, sensitivity analysis and computation of derivatives are
major problems in design in general and in OSD in particular. Chapter 4 describes different methods for the evaluation of the gradients of cost function and constraints: finite-difference, complex variable method, direct linearization, and the adjoint method. Advantages and drawbacks of each approach are given. In our opinion, AD has given new life to gradient-based methods. We describe the different possibilities and through simple programs give a comprehensive survey of direct AD by operator overloading and for the reverse mode, the adjoint code method used in Adol-C, Adifor, and Odysséé.

Chapter 5 presents several acceleration procedures that will allow three dimensional optimization with turbulent flows to be carried out on moderate size computers and even on work-stations. The problem of choice of the parameters, the approximations of gradients and the remeshing are discussed there. Implementation issues are important if we want an efficient optimization "platform". We describe different strategies for shape deformation within and without (CAD-Free) computed aided design data structures during optimization. For time-dependent problems, we discuss the pros and the cons of injection/suction boundary conditions equivalent to moving geometries when the motion is small. We present some optimization algorithms used in our optimization loop and the dynamical system analogy. The presentation is not intended to be exhaustive but rather reflects our practical experience. We show that this formulation is suitable for multi-physics problems where a coupling between different models is necessary. In addition, we see that the dynamical system approach is natural with incomplete sensitivity and state evaluations. By incomplete sensitivity we mean that only the deformation of the geometry is accounted for and the change of the state variable due to the change of geometry is ignored. We show by experience that the accuracy is sufficient for quasi-Newton algorithms and also that the complexity of the method is drastically reduced. Some strategies to couple mesh adaptation and the shape optimization loop are presented. The aim is to obtain a multi-grid effect and improve convergence.

In Chapter 6 we put forward a general argument to support the use of approximate gradients within optimization loops integrated with mesh refinements. Although this does not justify all the procedures that are presented in Chapter 5, it throws some light on why they work. We prove also that smoothers are essential. This part was done in collaboration with E. Polak and N. Dicesare.

Finally, the book closes with the presentation of many applications for stationary flows in Chapter 7 and unsteady problems in Chapter
8. We gather in Chapter 7 examples of shape optimization in two and three space dimensions using the tools presented above for both inviscid and viscous turbulent cases, for flow Mach number ranging from 0 to 20. Chapter 8 presents applications of our shape optimization algorithms to cases where the flow is unsteady for rigid and elastic bodies and shows that control problems and OSD problems are particular cases of a general approach. Closed loop control algorithms are presented together with an analogy with dynamical systems. The aim here is to present realistic "active control" examples, with injection and suction devices for drag reduction.

The selection of this material corresponds to what we think to be a good compromise between complexity and accuracy for the numerical simulation of nonlinear industrial problems, keeping in mind practical aspects at each level of the development, illustrating our proposal, with many practical examples which we have gathered during several industrial cooperations. In particular, the concepts are explained more intuitively than with complete mathematical rigor. Thus this book should be important for whoever wishes to solve a practical OSD problem. In addition to the classical mathematical approach, the application of some modern techniques such as automatic differentiation and unstructured mesh adaptation to OSD are presented, and multi-model configurations and some time-dependent shape optimization problems are discussed.

The book has been influenced by the reactions of students who have been taught this material twice at the Master level at the University of Paris. We think that what follows will be particularly useful for engineers interested in the implementation and solution of optimization problems using commercial packages, or in-house solvers, graduate and Ph.D. students in applied mathematics, aerospace, or mechanical engineering needing, during their training and research, to understand and solve design problems and research scientists in applied mathematics, fluid dynamics and CFD looking for new exciting research and development areas involving realistic applications.
Acknowledgments

The authors are grateful to P. Aubert, A. Baron, N. Dicesare, G. Medic, M. Stanciu, N. Petruzzelli, G. Puigt, F. Hecht and E. Polak for their contributions in the form of scientific material published elsewhere in collaboration with us.

For their encouragement and sharing of ideas the authors would like to thank A. Dervieux, C. Farhat, M. Giles, R. Glowinski, M. Gunzburger, W. Habashi, M. Hafez, H. Kawarada, P. Le Tallec, J.L. Lions, P. Moin, M. Naon, P. Neittanmaaki, J. Periaux and B. Perthame.

We thank also our colleagues at the universities of Montpellier II and Paris VI and at INRIA, for their comments on different points related to this work, namely: H. Attouche, B. Koobus, P. Redont, M. Cner, M. O. Bristeau, J. F. Bourgat, A. Hassim, P.L. George, A. Marrocco, P. Frey, P. Laug, M. Vidrascu, E. Saltel, S. Larteri, A. Desideri, E. Laporte,

We are also very happy to acknowledge the contributions of our industrial partners personified by MM. Duffa, Pirotais, Galais, Canton-Desmeuzes, at CEA-CEST A. MM. Stoufflet, Mallet, Rostand, Roeg, Dinh at Dassault Aviation, MM. Cormery at Aerospatiale-Matra, M. D. Laurence at EDF. MM. J. Cousteix and P. Sagaut at ONERA and M. S. Moreau at Valeo.

Finally, considerable help was given to us by the automatic differentiation specialists and especially by C. Bishof, C. Faure, P. Hovland, N. Rostaing, A. Griewank and J.C. Gilbert.
Contents

1. Optimal shape design
   1.1 Introduction
   1.2 Examples
      1.2.1 Minimum weight of structures
      1.2.2 Wing drag optimization
      1.2.3 Synthetic jets and riblets
      1.2.4 Stealth wings
      1.2.5 Optimization of a stealth wing
      1.2.6 Optimal breakwater
      1.2.7 Two academic test cases: nozzle optimization
   1.3 Existence of solutions
      1.3.1 Topological optimization
      1.3.2 Sufficient conditions for existence
   1.4 Solution by optimization methods
      1.4.1 Gradient methods
      1.4.2 Newton methods
      1.4.3 Constraints
      1.4.4 A constrained optimization algorithm
   1.5 Sensitivity analysis
      1.5.1 Sensitivity analysis for the nozzle problem
   1.6 Discretization with triangular elements
      1.6.1 Sensitivity of the discrete problem
   1.7 Numerical issues
      1.7.1 Implementation problems
      1.7.2 Independence from the cost function
      1.7.3 Addition of geometrical constraints
      1.7.4 Automatic Differentiation
   1.8 Appendix: Optimal Design for Navier-Stokes flows
      1.8.1 Optimal shape design for Stokes flows
      1.8.2 Optimal shape design for Navier-Stokes flows
   Bibliography

2. Partial differential equations for fluids
   2.1 Introduction
   2.2 The Navier-Stokes equations
      2.2.1 Conservation of mass
      2.2.2 Conservation of momentum
      2.2.3 Conservation of energy and the law of state
   2.3 Inviscid flows
   2.4 Incompressible flows
   2.5 Potential flows
   2.6 Turbulence modeling
2.6.1 The Reynolds number
2.6.2 Reynolds Equations
2.6.3 The $k - \varepsilon$ model
2.7 Equations for compressible flows in conservation form
   2.7.1 Boundary and initial conditions
2.8 Wall-laws
   2.8.1 Generalized wall functions for $u$
   2.8.2 Wall function for the temperature-energy equation
   2.8.3 $k$ and $\varepsilon$
2.9 Appendix 1: Generalized wall functions
   2.9.1 Pressure correction
   2.9.2 Corrections on adiabatic walls for compressible flows
   2.9.3 Prescribing $\rho_w$
   2.9.4 Correction for Reicardt law
2.10 Appendix 2: Wall-functions for isothermal walls
Bibliography

3. Some numerical methods for fluids and examples
3.1 Introduction
3.2 Numerical methods for compressible flows
   3.2.1 A FEM-FVM discretization
   3.2.2 Approximation of the convection fluxes
   3.2.3 Accuracy improvement
   3.2.4 Positivity
   3.2.5 Time integration
   3.2.6 Local time stepping procedure
   3.2.7 Implementation of the boundary conditions
   3.2.8 Solid walls: transpiration boundary condition
   3.2.9 Solid walls: implementation of wall-laws
3.3 Incompressible flows
   3.3.1 Solution by a projection scheme
   3.3.2 Spatial discretization
   3.3.3 Local time stepping
   3.3.4 Numerical approximations for the $k - \varepsilon$ equations
3.4 Some simulations of the direct problem
   3.4.1 Flow over a backward step
   3.4.2 Transonic flow over a RAE2822
   3.4.3 Flow over a square cylinder
   3.4.4 Flow over a circular cylinder
   3.4.5 Flow over a circular-arc airfoil
   3.4.6 Stall prediction at low speed
   3.4.7 Flow over an adiabatic expansion ramp
   3.4.8 Flow over an isothermal compression ramp
3.5 Appendix: Wall-law subroutine
4. **Automatic differentiation**
   4.1 Introduction
   4.2 Computations of derivatives
      4.2.1 Finite differences
      4.2.2 Complex variables method
      4.2.3 State equation linearization
      4.2.4 Adjoint method
      4.2.5 Adjoint method and Lagrange multipliers
      4.2.6 Automatic differentiation
      4.2.7 A Class library for the direct mode
   4.3 Nonlinear PDE and AD
   4.4 A simple inverse problem
   4.5 A shock problem solved by AD
      4.5.1 A simple example $R \rightarrow R^2 \rightarrow R$
      4.5.2 DO - IF
      4.5.3 Nested loops
      4.5.4 Interprocedural differentiation
   4.6 Appendix 1: Odyssee automatic differentiator
   4.7 Appendix 2: Direct and reverse modes of AD
   4.8 Appendix 3: More on FAD classes

Bibliography

5. **Optimization platform and implementation issues**
   5.1 Introduction
   5.2 Shape parameterization and shape and mesh deformation tools
      5.2.1 CAD-based
      5.2.2 Basis of shape functions
      5.2.3 CAD-free
   5.3 Handling domain deformations
      5.3.1 Explicit deformation
      5.3.2 Adding an elliptic system
      5.3.3 Injection boundary condition
      5.3.4 Geometrical constraints
   5.4 Minimization algorithms
      5.4.1 State equations
      5.4.2 Pseudo-unsteady control and optimization algorithm
      5.4.3 More sophisticated pseudo-unsteady systems
      5.4.4 Interior point algorithms
   5.5 Efficiency with AD
      5.5.1 Limitations when using AD
      5.5.2 Storage strategies
      5.5.3 Keys points when using AD
   5.6 Incomplete sensitivities and boundary integrals

Bibliography
5.6.1 Incomplete sensitivities and equivalent boundary condition

5.6.2 Incomplete gradient: application to advection-diffusion equation

5.6.3 Incomplete gradient: application to channel flows

5.6.4 Incomplete sensitivities: time-dependent phenomena

5.6.5 Newton law: pressure distribution prediction for bluff bodies

5.6.6 General formulation and validity domain

5.6.7 Multi-level construction

5.6.8 Back to finite differences or complex variables method

5.6.9 Coupled configurations

5.6.10 Incomplete sensitivities and the Hessian

5.6.11 Redefinition of cost functions

5.7 Mesh adaptation and optimization

5.7.1 Delaunay mesh generator

5.7.2 Metric definition

5.7.3 Adaptive optimization algorithm

Bibliography

6. Consistent approximations and approximate gradients

6.1 Generalities

6.2 Consistent approximations

6.2.1 Consistent approximation

6.3 Application to a control problem

6.3.1 Verification of the hypothesis

6.3.2 Numerical example

6.4 Application to optimal shape design

6.4.1 Problem statement

6.4.2 Discretization

6.4.3 Optimality Conditions: the continuous case

6.4.4 Optimality Conditions: the discrete case

6.4.5 Definition of $\theta_h$

6.4.6 Implementation trick

6.4.7 Orientation

6.4.8 Numerical example

6.4.9 A nozzle optimization

6.4.10 Numerical results

6.4.11 Drag reduction for an airfoil

6.5 Approximate gradients

6.5.1 A control problem with domain decomposition

6.5.2 Numerical results

6.6 Conclusion
6.7 Appendix
   6.7.1 Verification of the hypothesis of Theorem 1
   6.7.2 Inclusion
   6.7.3 Continuity
   6.7.4 Consistency
   6.7.5 Continuity of \( \theta \)
   6.7.6 Continuity of \( \theta_h(\alpha_h) \)
   6.7.7 Convergence

Bibliography

7. Numerical results on shape optimization
   7.1 Introduction
   7.2 Drag reduction for a supersonic flow
   7.3 4-element airfoil optimization
   7.4 Transonic turbulent flow
   7.5 Shape optimization at high speed
   7.6 Heat transfer optimization
   7.7 Sonic boom reduction
   7.8 Blade cascade
   7.9 3D inviscid wave-drag reduction for a wing
   7.10 Drag reduction for a transonic business jet
   7.11 Aerodynamic stability improvement
   7.12 Supersonic business jet

Bibliography

8. Numerical results on shape optimization for unsteady flows
   8.1 Introduction
   8.2 Control of flows around rigid bodies
      8.2.1 Flow control behind a cylinder
      8.2.2 Buffeting control over a RA 16
   8.3 Control in multi-disciplinary context
   8.4 ODE based models for the Structure
   8.5 Structural model for the elastic CAD-Free parameterization
   8.6 Fluid models
   8.7 Coupling strategies
      8.7.1 First order explicit coupling
      8.7.2 First order implicit coupling
      8.7.3 First order semi-implicit coupling
      8.7.4 Second order implicit coupling
      8.7.5 Second order coupling with prediction
   8.8 Time dependent minimization problem
      8.8.1 Second order dynamic system
      8.8.2 Coupling the control and state equations
   8.9 Sensitivity analysis by CVM
8.9.1 CVM sensitivity to the increment
8.10 Control of aeroelastic instabilities
8.11 Using more sophisticated fluid models
   8.11.1 2D Aeroelastic control with stable asymptotic behavior
   8.11.2 2D Aeroelastic control with unstable asymptotic behavior
   8.11.3 3D Aeroelastic control
Bibliography
1 Optimal shape design

1.1 Introduction

In mathematical terms, an optimal shape design requires the optimization to one or several criteria \( \{E_i(x)\}_i \) which depend on design parameters \( x \in X \) which define the shape of the system within the admissible set of values \( X \).

Multi-criteria. Optimization is a difficult field in itself of which we shall retain only the min-max idea,

\[
\min_{x \in X} \{J(x) : E_i(x) \leq J(x), \quad i = 1, ..., I \}
\]

and the Pareto minimization

\[
\min_{x \in X} \sum_{i=1}^I \alpha_i E_i(x). 
\]

For some suitable values of \( \alpha_i \in (0, 1) \), both problems are equivalent and solve in some intuitive sense the multi-criteria problem. Standard optimization and control theory is applied to these derived problems.

Optimal control for distributed systems is a branch of optimization for problems which involve a parameter or control variable \( u \), a state variable \( y \) and a partial differential equation \( A \) (with boundary conditions \( b \)), to define \( y \) in a domain \( \Omega \):

\[
\min_u \{J(u, y) : A(x, y, u) = 0 \quad \forall x \in \Omega, \quad b(x, y, u) = 0 \quad \forall x \in \partial \Omega \}.
\]

For example,

\[
\min_u \int_{\Omega} (y - 1)^2 : -\Delta y = 0 \quad \forall x \in \Omega, \quad y|_{\partial \Omega} = u \quad (1.1)
\]

attempts to find a boundary condition \( u \) for which \( y \) would be as close to the value 1 as possible.

For (1.1) there is a trivial solution \( u = 1 \) because then \( y = 1 \) is the solution to the Laplace equation.

Optimal shape design is a special case of control theory where the control is the boundary \( \partial \Omega \) itself. For example, if \( D \) is given, consider

\[
\min_{\{\partial \Omega, \Omega \subset \Omega\}} \{ \int_D (y - 1)^2 : -\Delta y = g, \quad y|_{\partial \Omega} = 0 \} \quad (1.2)
\]

This problem has a physical meaning [39], it answers the following: is it possible to build a support for a membrane bent by its own weight
\((g = -9.81)\) which would bring its deflection as close to 1 as possible in a region of space \(D\).

The intuitive answer is no unless \(D\) is a singular set, and we see it mathematically by noting that \(y + g \frac{\omega}{2}\) is an analytical function and so if it is 1 in \(D\) it must be 1 everywhere; but then this fact is not compatible with the boundary conditions for non-singular set \(\Omega\).

Thus the existence, uniqueness, and regularity of solution can be a problem even in simple cases.

In smooth situations, the PDE can be viewed as an implicit map \(u \to \Omega(u) \to y(u)\) where \(u \to \Omega(u)\) is the parameterization of the domain by a (control) parameter \(u\) and the problem is to minimize the function \(J(u, y(u))\). If it is continuously differentiable in \(u\), then the algorithms of differentiable optimization can be used (see [38] for instance) and so it remains only to explain how to compute \(J_u'\).

Analytic computation of derivatives for OSD problems is possible both for the continuous and the discretized problems. It may be tedious but it is usually possible. When it is difficult one may turn to automatic differentiation (AD), but then other difficulties pop up and so it is a good idea to understand the theory even when using AD.

Therefore we begin this chapter by giving simple examples of OSD problems. Then we recall some theorems of existence of solutions and give for simple cases a method to derive optimality conditions. Finally, we show the same on OSD problems discretized by the finite element method of degree one on triangulations. More details can be found in [12, 37, 35, 24].

### 1.2 Examples

#### 1.2.1 Minimum weight of structures

In 2D linear elasticity, for a structure clamped on \(\Gamma = \partial \Omega\), and subject to volume forces \(F\), the vertical displacement \(u = (u_1, u_2) \in V\) is found by solving:

\[
\int_{\Omega} \tau_{ij}(u) \epsilon_{ij}(v) = \int_{\Omega} F \cdot v \quad \forall v \in V_0 = \{ u \in H^1(\Omega)^2 : u|_{\Gamma} = 0 \}
\]

where \(\epsilon_{ij}(u) = \frac{1}{2} (\partial_i u_j + \partial_j u_i)\), and

\[
\begin{pmatrix}
\tau_{11} \\
\tau_{22} \\
\tau_{12}
\end{pmatrix} = \begin{pmatrix}
2\mu + \lambda & \lambda & 0 \\
\lambda & 2\mu + \lambda & 0 \\
0 & 0 & 2\mu
\end{pmatrix} \begin{pmatrix}
\epsilon_{11} \\
\epsilon_{22} \\
\epsilon_{12}
\end{pmatrix}
\]

and where \(\lambda, \mu\) are the Lamé coefficients.

Many important problems of design arise when one wants to find the structure with minimum weight yet satisfying some inequality constraints for the stress such as in the design of light weight beams
for strengthening an airplane floors, or for crank shaft weight optimization.

For all these problems the criteria for optimization is the weight

\[ J(\Omega) = \int_\Omega \rho, \]

where \( \rho \) is the density of the material. But there are constraints of the type

\[ \tau(x) \cdot d(x) < \tau_{\text{max}} \]

at some points \( x \) and for some directions \( d(x) \).

Indeed, a wing, for instance, needs to have a different response to stress span-wise and chord-wise. Moreover, due to coupling between physical phenomena, the surface stresses come in part from fluid forces acting on the wing. This implies many additional constraints on the aerodynamic (drag, lift, moment) and structural (Lamé coefficients) characteristics of the wing. Therefore, the Lamé equations of the structure must be coupled with the equations for the fluid (fluid structure interactions). This is why most optimization problems nowadays require the solution of several state equations, fluid and structure in this example.

1.2.2 Wing drag optimization

An important industrial problem is the optimization of the shape of a wing to reduce the drag. The drag is the reaction of the flow on the wing; its component in the direction of flight is the drag proper and the rest is the lift. A few percent of drag optimization means a great saving on commercial airplanes. For viscous drag the Navier-Stokes equations must be used. For wave drag the Euler system is sufficient.

For a wing \( S \) moving at constant speed \( u_\infty \) the force acting on the wing is

\[ F = \int_S [\mu(\nabla u + \nabla u^T) - \frac{2\mu}{3} \nabla \cdot u]n - \int_S pn \]

where \( n \) is the normal to \( S \) pointing outside the domain occupied by the fluid.

The first integral is a viscous force, the so-called viscous drag/lift, and the second is called the wave drag/lift. In a frame attached to the wing, and with uniform flow at infinity, the drag is the component of \( F \) parallel to the velocity at infinity (i.e. \( F \cdot u_\infty \)). The viscosity of the fluid is \( \mu \) and \( p \) is its pressure.

The Navier-Stokes equations govern \( u \) the fluid velocity, \( \theta \) the temperature, \( \rho \) the density and \( E \) the energy:

\[ \partial_t \rho + \nabla \cdot (\rho u) = 0 \]
\[ \partial_t (\rho u) + \nabla.(\rho u \otimes u) + \nabla p - \mu \Delta u - \frac{1}{3} \mu \nabla(\nabla \cdot u) = 0, \]
\[ \partial_t [\rho E] + \nabla \cdot [u \rho E] + \nabla \cdot (pu) = \nabla \cdot \{ \kappa \nabla \theta + [\mu(\nabla u + \nabla u^T) - \frac{2}{3} \mu \mathbf{I}] \nabla \cdot u \} \]

where \( E = \frac{u^2}{2} + \theta \) \quad \rho = (\gamma - 1) \rho \theta.

The problem is to minimize
\[ J(S) = F, u_\infty \]
with respect to the shape of \( S \).

There are several constraints:
- A geometrical constraint such as the volume being greater than a given value, else the solution will be a point.
- An aerodynamic constraint: the lift must be greater than a given value or the wing will not fly.

The problem is difficult because it involves the Navier-Stokes equations at high Reynolds number. It can be simplified by considering only the wave drag, i.e. the pressure term only in the definition of \( F \) [27]. Then the viscous terms can be dropped in the Navier-Stokes equations (\( \mu = \kappa = 0 \)); Euler’s equations remain. However, there may be side effects to such simplifications. In transonic regimes, for instance, the shock position for a Navier-Stokes flow is upstream compared to an inviscid (Euler) simulation at the same Mach number. Figs. (1.1.-1.7) display the results of two optimizations using Euler equations and a Navier-Stokes equations with \( k - \varepsilon \) turbulence modeling for a NACA 0012 at Mach number of 0.75 and 2 degrees of incidence. The cost function involves the drag coefficient \( C_d \), the lift coefficient \( C_l \) and the area ”Vol” of the wing profile (see Chapter 6):
\[ J(S) = \frac{C_d}{C_d^0} + 0.05 \frac{|C_l - C_l^0|}{C_l^0} + 0.1 \frac{|\text{Vol} - \text{Vol}^0|}{\text{Vol}^0}. \]

**Simplifying the state equation**
Assuming irrotational flow an even greater simplification replaces the Euler equations by the compressible potential equation:
\[ u = \nabla \varphi, \quad \rho = (1 - |\nabla \varphi|^2)^{1/(\gamma - 1)}, \quad p = \rho^\gamma, \quad \nabla.(\rho u) = 0. \]

Or even, if at low Mach number, by the incompressible potential flow equation:
\[ u = \nabla \varphi, \quad -\Delta \varphi = 0. \quad (1.3) \]
The pressure is given by the Bernoulli law $p = p_{\text{ref}} - \frac{1}{2}u^2$ and so only an optimization of the lift would be:

$$\min_{S \in G} \left\{ - \int_S f\left( \frac{u^2}{2} \right) n \cdot u_\infty : \text{subject to (1.3) and} \right\}$$

$$\left. \frac{\partial \phi}{\partial n} \right|_{n} = u_\infty \cdot n,$$

$$\left. \frac{\partial \phi}{\partial n} \right|_{s} = 0$$
Fig. 1.3. Transonic drag reduction. Iso-Mach contours for the inviscid simulation after optimization.

Fig. 1.4. Transonic drag reduction. Iso-Mach contours for the viscous simulation after optimization.

for some admissible set of shapes $G$ and some local criteria $f$.

**Multi-point optimization**

Engineering constraints on admissible shapes are numerous:

- Minimal thickness, given length.
- Maximum admissible radius of curvature.
Fig. 1.5. Transonic drag reduction. Pressure coefficient.

Fig. 1.6. Transonic drag reduction. Initial and final shapes for the inviscid and viscous optimizations. The differences between the two shapes increase with the deviation between the shock positions.

- Minimal angle at the trailing edge.

Another problem arises due to instability of optimal shapes with respect to data. It has been seen that the leading edge at the optimum is a wedge. Thus if the incidence angle for $u_\infty$ is changed the solution becomes bad. A multi-point functional must be used in the
Fig. 1.7. Transonic drag reduction. Convergence histories for the gradients using the inviscid and viscous flows. The convergence seems to be more regular for the viscous flow: with a robust solver for the turbulence model, optimization is actually easier than with the Euler equations. Of course, the CPU time is larger because the viscous case requires a finer mesh.

\[
\min_S J(S) = \sum \alpha_i u_i^\infty \cdot F_i^i,
\]

or

\[
\min J : u_i^\infty \cdot F_i^i \leq J \quad \forall i
\]

at given lifts where the \( F_i^i \) are computed from the Navier-Stokes equations with boundary conditions \( u = u_i^\infty, \quad u|_S = 0 \).

1.2.3 Synthetic jets and riblets

The solution to a time-dependent optimization problem is time-dependent. But for wings this would give a deformable shape, with motion at the time scale of the turbulence in (1.3). As this is computationally unreachable, suboptimal solutions may be searched. One direction is to replace moving surfaces by mean surfaces which can “transpire”. For instance, consider a surface with tiny holes each connected to a rubber reservoir activated by an electronic device capable therefore of blowing and sucking air so long as the net flow is zero over a period. The reservoir may be ignored and the mean surface may be considered with a “transpiration condition”. This topic is currently being investigated [31, 18, 32].

In the class of time-independent shapes with time-dependent flows it is not even clear that the solution is smooth. In [31], the authors
showed that ribblets, little groves in the direction of the flow, actually reduce the drag by a few percent. The simulation was done with a large eddy simulation (LES) model for the turbulence and at the time of writing this book shape optimization with LES is beyond our computational power. But this is certainly an important research area for the future.

1.2.4 Stealth wings

Maxwell equations

The optimization of the far-field energy of a radar wave reflected by an airplane in flight requires the solution of Maxwell’s equations for the electric field $E$ and the magnetic field $H$:

$$\varepsilon \partial_t E + \nabla \times H = 0 \quad \nabla \cdot E = 0, \quad \mu \partial_t H - \nabla \times E = 0 \quad \nabla \cdot H = 0.$$ 

The electric and magnetic coefficients $\varepsilon, \mu$ are constant in air but not so in an absorbing medium. One variable, $H$ for instance, can be eliminated by differentiating in $t$ the first equation:

$$\varepsilon \partial_{tt} E + \nabla \times \left( \frac{1}{\mu} \nabla \times E \right) = 0.$$ 

It is easy to see that $\nabla \cdot E = 0$ if it is zero at initial time.

Helmholtz equation

Now if the geometry is cylindrical with axis $z$ and if $E = (0, 0, E_z)^T$ then the equation becomes a scalar wave equation for $E_z$. Furthermore, if the boundary conditions are periodic in time at infinity, $E_z = \mathcal{R} \varepsilon v e^{i\omega t}$ and compatible with the initial conditions then the solution has the form $E_z = \mathcal{R} \varepsilon v(x) e^{i\omega t}$ where $v$, the amplitude of the wave $E_z$ of frequency $\omega$, is solution of:

$$\nabla \left( \frac{1}{\mu} \nabla v \right) + \omega^2 \varepsilon v = 0. \quad (1.4)$$

Remark 1.1

- Notice the wrong sign for ellipticity in the “Helmholtz” equation (1.4).
- This equation arises in acoustics also.
- In vacuum $\mu \varepsilon = c^2$, $c$ the speed of light, so for numerical purposes it is a good idea to rescale the equation. The critical parameter is then the number of waves on the object, i.e. $\omega c/(2\pi L)$ where $L$ is the size of the object.

Boundary conditions

The reflected signal on solid boundaries $S$ satisfies

$$v = 0 \text{ or } \partial_n v = 0 \text{ on } S$$

depending on the type of waves (transverse magnetic polarization requires Dirichlet condition).
When there is no scattering object this Helmholtz equation has a simple sinusoidal set of solutions which we call \( v_\infty \):

\[
v_\infty(x) = \alpha \sin(k \cdot x) + \beta \cos(k \cdot x)
\]

where \( k \) is any vector of modulus \(|k| = \omega c\). Radar waves are more complex, but by Fourier decomposition they can be viewed as a linear combination of such simple unidirectional traveling waves. Now if such a wave is sent onto an object, it is reflected by it and the signal at infinity is the sum of the original wave with the reflected wave. So it is better to set an equation for the reflected wave only \( u = v - v_\infty \).

A good boundary condition for \( u \) is difficult to set; one possibility is

\[
\partial_n u + i\alpha u = 0.
\]

Indeed, when \( u = e^{i k \cdot x} \), \( \partial_n u + i\alpha u = i(d \cdot n + a) u \), so that this boundary condition is "transparent" to waves of direction \( d \) when \( a = -d \cdot n \).

If we want this boundary condition to let all outgoing waves pass the boundary, we will set \( a = -ik \cdot n \).

To summarize, we set for \( u \) the system in the complex plane:

\[
\begin{align*}
\nabla \cdot \left( \frac{1}{\mu} \nabla u \right) + \omega^2 u &= 0, \quad \text{in } \Omega, \\
\partial_n u - ik \cdot nu &= 0 \quad \text{on } \Gamma_\infty \\
u &= g \equiv -e^{ik \cdot x} \quad \text{on } S,
\end{align*}
\]

where \( \partial \Omega = S \cup \Gamma_\infty \). It can be shown that the solution exists and is unique. Notice that the variables have been rescaled, \( \omega \) is \( \omega c \), \( \mu \) is \( \mu/\mu_{\text{vacuum}} \).

Usually the criteria for optimization is a minimum amplitude for the reflected signal in a region of space \( D \) at infinity (hence \( D \) is an angular sector). For instance, one can consider

\[
\min_{S \in \Omega} \left\{ \int_{\Gamma_\infty \cap D} | \nabla u |^2 : \quad \omega^2 u + \nabla \cdot \left( \frac{1}{\mu} \nabla u \right) = 0, \quad u \Big|_S = g, \quad (-ik \cdot nu + \partial_n u) \big|_{\Gamma_\infty} = 0 \right\}
\]

(1.5)

In practice \( \mu \) is different from 1 only in a region very near \( S \) so as to model the absorbing paint that most stealth airplanes have. But constraints are aerodynamic as well, lift above a given lower limit for instance, and thus require the solution of the fluid part as well. The design variables are:

- The shape of the wing;
- The thickness of the paint;
- The material characteristics \((\epsilon, \mu)\) of the paint.
Here again, the theoretical complexity of the problem can be appreciated from the following question:
Would ribblets on the wing, of the size of the radar wave, improve the design?
Homogenization can answer the question [1, 9, 8]; it shows that an oscillatory design is indeed better. Furthermore, periodic surface irregularities are equivalent, in the far field, to new effective boundary conditions:

\[ u = 0 \text{ on an oscillatory } S^c \text{ can be replaced by } \]
\[ au + \partial_n u = 0 \text{ on a mean } S \]
for some suitable \( a \) [2].
If that is so then the optimization can be done with respect to \( a \) only. But optimization with respect to the parameters of the PDE is known to generate oscillations [43]; this topic is known as topological optimization (see below).
Optimization with respect to \( \mu \) gives also rise to complex composite structure design problems.
So an aerodynamic constraint on the lift has been added. The flow is assumed inviscid and irrotational and computed by a stream-function \( \psi \):

\[ u = \nabla \times \psi, \; p = p_{\text{ref}} - \frac{u^2}{2}, \; \Delta \psi = 0 \text{ in } \Omega, \]
\[ \psi|_S = \beta \; \psi|_{\infty} = \left( \frac{\cos \theta}{\sin \theta} \right) \times n, \]
where \( u, p \) are the velocity and pressure in the flow, \( S \) is the wing profile, \( \theta \) its angle of incidence, \( n \) its normal. The constant \( \beta \) is adjusted so that the pressure is continuous at the trailing edge [14]. The lift being proportional to \( \beta \) we impose the constraint \( \beta \geq \beta_0 \) the lift of the NACA0012 airfoil. The result after optimization is shown on Fig. 1.9.

1.2.5 Optimization of a stealth wing
Problem (1.5) has been described in section 1.2.4. The problem is that without constraint the solution is very unaerodynamic (see Fig. 1.8).

1.2.6 Optimal breakwater
Here the problem is to build a good harbor by designing an optimal breakwater. As a first approximation, the amplitude of sea waves satisfies the Helmholtz equation

\[ \nabla (\mu \cdot \nabla u) + \epsilon u = 0, \]
where \( \mu \) is a function of the water depth and \( \epsilon \) is function of the wave speed.
Fig. 1.8. Stealth wing. Optimization without aerodynamical constraint (Courtesy of A. Baron).

Fig. 1.9. Stealth wing. Optimization with aerodynamical constraint (Courtesy of A. Baron).