Evolving strategies for active flow control

Michele Milano
milano@eniac.ethz.ch
Institut für Fluiddynamik, ETH Zentrum
Zürich, CH-8092, Switzerland

Petros Koumoutsakos
petros@eniac.ethz.ch
Institut für Fluiddynamik, ETH Zentrum
Zürich, CH-8092, Switzerland
and Center for Turbulence Research
NASA Ames 202A-1
Moffett Field, CA 94035, USA

Xavier Giannakopoulos
xavier@idsia.ch
IDSIA, Galleria 2
Manno, CH-6928, Switzerland

Jürgen Schmidhuber
juergen@idsia.ch
IDSIA, Galleria 2
Manno, CH-6928, Switzerland

Abstract- About fourty years ago Rechenberg and Schwefel (Rechenberg, 1994) came up with the idea of evolution strategies for flow optimization. Since then advances in computer architectures and numerical algorithms have greatly decreased computational costs of realistic flow simulations, and today Computational Fluid Dynamics (CFD) is complementing flow experiments as a key guiding tool for aerodynamic design. Of particular interest are designs with active devices controlling the inherently unsteady flow fields, promising potentially drastic performance leaps. Here we demonstrate that CFD-based design of active control strategies can benefit from evolutionary computation. We optimize the flow past an actively controlled circular cylinder, a fundamental prototypical configuration. The flow is controlled using surface-mounted vortex generators; evolutionary algorithms are used to optimize actuator placement and operating parameters. We achieve drag reduction of up to 60 percent, outperforming the best methods previously reported in the fluid dynamics literature on this benchmark problem.

1 Introduction

The flow past a circular cylinder is a well established prototypical configuration of bluff body flows (Roshko, 1955). Here, for the first time, we use evolutionary algorithms in conjunction with CFD to achieve drag reduction in active flow control through modification of the tangential velocity on the cylinder surface.

Several control methodologies have been suggested to modify vortex shedding behind a circular cylinder and to effectuate drag reduction, either with passive geometrical modifications and/or with an open loop steady forcing. A non-exhaustive list includes studies of the effect of a splitter plate attached to the cylinder studied, among others, by (Kwon and Choi, 1996; Unal and Rockwell, 1987; Apelt and West, 1975; Apelt et al., 1973). These studies achieved drag reductions from 20 to 40% depending on the arrangement of the control devices. A free rotating cylinder attached to a splitter plate was considered in (Cimbala and Garg, 1991). The placement of a small secondary cylinder in the wake was shown to suppress the vortex shedding while reducing the drag coefficient (Strykowsky and Sreenivasan, 1990). Another possible action is base suction/blowing (Shumm et al., 1994). Rotational oscillations were studied by (Tokumaru and Dimotakis, 1991; Chen et al., 1993). At Reynolds numbers $Re = 15000$ (Re indicates the degree of flow turbulence) and for certain parameters, found after extensive experimentation, this control action was found to induce dramatic changes in the wake and a significant drag reduction of about 60 %. Due to experimental limitations, however, it was not possible to reveal the underlying governing mechanisms .

While a wealth of different possible geometry modifications and/or open loop actions have been devised and studied, little effort has been devoted to algorithms optimizing the various control parameters. Recent CFD studies of suboptimal control (Lee et al., 1998) have tried to remedy this situation for simulation of flows using mass traspiration as an actuation mechanism. Drag reduction of up to 20% was achieved using this optimization strategy for idealized blowing/suction actuators.

In practice, however, it remains an almost impossible task even to cast the optimization problem in a tractable explicit analytical form for all different types of sensor and actuator devices. The dependence of the control on the type of actuators also minimizes the generalization of certain successful control strategies. Hence, most present day design strategies rely mainly on trial and error experiments, driven by human intuition derived from physical understanding of the phenomenon. This approach has the fundamental limitation that most of this understanding is based on simulations and experiments of the *uncontrolled* flow. Moreover when the placement of the actuators needs to be optimized the cost function can be evaluated only for a certain actuator arrangement and hence no analytical objective function can be pre-

0-7803-6375-2/00/$10.00 ©2000 IEEE.

scribed *a priori* in order to be optimized with gradient based techniques. This motivates our interest in automatic optimizers of the *controlled* flow that compactly encode information about desired flow behavior in their objective function.

We conduct two-dimensional simulations for Reynolds numbers $Re = 500$ and $Re = 1000$.

Although the uncontrolled flow past a circular cylinder at these Re is three-dimensional, experiments by (Tokumaru and Dimotakis, 1991) for a cylinder undergoing rotary oscillations indicate that the controlled flows and the underlying physical mechanisms are strongly two-dimensional.

The paper is organized as follows: Section 2 presents flow equations, related numerical parameters of the flow solver and the calculation of the drag coefficient which constitutes the objective function to be minimized. In Section 3 we consider an array of actuators placed on the cylinder surface and a genetic algorithm that incorporates a mutation probability dependent on the distance from a prescribed target value of the objective function. Section 4 describes an alternative heuristic approach which generates drag-reducing control parameters based on previously observed drag values. Our methods will yield unprecedented drag reduction.

2 Governing equations and numerical method

We consider two-dimensional incompressible viscous flow past a circular cylinder. Governing equations are the Navier-Stokes equations in the velocity-pressure formulation

$$\frac{d\mathbf{v}}{dt} + (\mathbf{v}\cdot\nabla)\mathbf{v} = -\frac{1}{\rho}\nabla P + \nu\nabla^2\mathbf{v} \quad (1)$$

$$\nabla\cdot\mathbf{v} = 0 \quad (2)$$

where $\mathbf{v}$ is the velocity vector, P, ρ is the pressure and density of the flow and ν denotes the kinematic viscosity.

The boundary conditions are defined as:

$$\mathbf{v}(\mathbf{x},t) = \mathbf{V}_{ext} \qquad \text{on the cylinder surface} \quad (3)$$

$$\mathbf{v}(\mathbf{x},t) = U_\infty\,\mathbf{e}_x \qquad \text{as } |\mathbf{x}| \to \infty \quad (4)$$

where $\mathbf{V}_{ext}$ is the surface velocity induced by the actuators, and $\mathbf{e}_x$ is the unit vector in the stream-wise direction. U_∞ denotes the free-stream velocity. The Reynolds number and Strouhal frequency of the flow are defined as :

$$Re = \frac{U_\infty D}{\nu} \qquad St = \frac{f D}{U_\infty}$$

where $D = 2R$ is the diameter of the cylinder and f the shedding frequency of the flow.

We also define a normalized time, by scaling time as follows:

$$t^* = \frac{t}{t_a} = \frac{t}{(D/U_\infty)}$$

where t_a is the time taken by a fluid particle in the free stream to be advected past the cylinder. The definition of such normalized quantities allows a ready generalization of the results to differently scaled geometries.

The flow solver used in this work (Mittal, 1995) employs a staggered, second order central-difference method in generalized coordinates. The solution is advanced in time using a fractional step scheme, in which a third order Runge-Kutta scheme is used for the nonlinear convection terms and a Crank-Nicholson scheme is used for the viscous terms. A multigrid solver is used in conjuction with a Gauss-Seidel line-zebra scheme to solve the pressure Poisson equation.

We focus on the problem of reducing the drag, represented by the drag coefficient C_D calculated from quantities measured on the cylinder surface as:

$$C_D = \frac{2}{\rho U_\infty^2 D}\int_{cyl}(p\mathbf{n}_\mathbf{x} - \tau_{ix}\mathbf{n}_\mathbf{i})dl \quad (5)$$

where p is the pressure and τ_{ix} the viscous stress tensor on the surface of the cylinder and $\mathbf{n}_\mathbf{i}, i = x, y$ denotes the components of the unit normal on the suface of the cylinder.

3 Tangential actuation optimization

3.1 Problem setup

We consider flow at $Re = 500$, past a cylinder whose surface is subdivided in $n = 16$ equal size segments that are allowed to move tangentially to the cylinder surface, each with a different but steady velocity.

This configuration can be physically realized by using a set of moving belts covering the surface of the cylinder. It has been found experimentally by (Bechert et al., 1996) and (Modi, 1997) that such actuators are very efficient in achieving significant drag reduction. Fig.1 shows the location of the belts on the cylinder surface for the case study considered here.

An off-line optimization is performed here for the functional in Eq. 5; what we optimize is the time average of the drag in a fixed time interval. The average drag is computed at each simulation step, starting from a steady uncontrolled situation and averaging over 4 Strouhal periods.

We use a genetic algorithm (GA) (Holland, 1975; Goldberg, 1989) that was specifically designed to find a basin in parameter space containing all parameter configurations yielding fitness values below a given threshold (Milano and Barone, 1999).

We study whether this GA can automatically identify critical points such as the separation points of the uncontrolled flow. For $Re = 500$ these points are encompassed by segments 4 and 13.

3.2 GA Description

The GA used in this paper is a modification of Controlled Random Search (CRS) (Price, 1976). The algorithm is outlined in the following paragraphs.

Let $G(\lambda)$ the function to be minimized, $\lambda \in R^n$ the parameter vector. In a first phase S population points are initially randomly chosen according to a uniform distribution

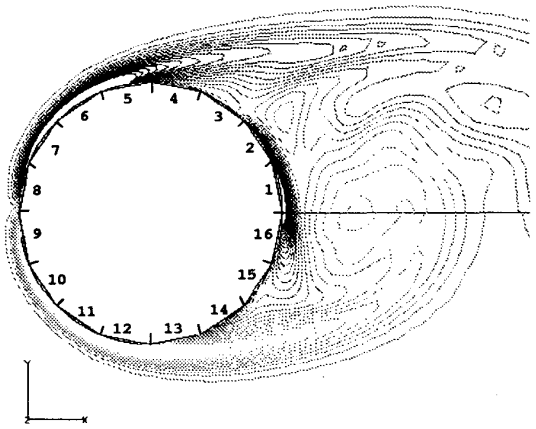


Figure 1: Location of the actuator belts on the cylinder surface. The belts are numbered as specified in the figure; also a snapshot of the uncontrolled vorticity is reported: the flow is from left to right.

within a defined search volume of dimension n. The inequality $S \gg n$ must hold for the algorithm to work properly. Then the algorithm proceeds as follows:

1) Choose the population point λ_{max} in which G reaches the maximum value:
$\lambda_{max} = arg\,[\max_{i=1,...,M} G(\lambda_i)]$;
$G_{max} = G(\lambda_{max})$;

2) Randomly choose $n+1$ different population points: $\lambda_1, ..., \lambda_{n+1}$ (breeding set). All subsequent operations are performed on this set;

3) Mutation step: for all breeding set points, with probability
$P_i = (1-\alpha^I) \cdot \left(1-\beta^{\frac{(G(\lambda_i)-G_T)}{\overline{G}}}\right) \cdot \gamma$, replace the point λ_i with one randomly chosen within the search volume limits;

4) Recombination step: for each of the $n+1$ points determine the centroid, $\underline{\lambda_i}$, of the other n points, i.e. $\underline{\lambda_i} = \frac{1}{n}\sum_{j=1}^{n} \lambda_j$;

4.a) Generate offspring $\lambda_{si} = 2\underline{\lambda_i} - \lambda_{n+1}$; if λ_{si} is not contained in the search volume, process next point in breeding set ;

4.b) Calculate $G(\lambda_{si})$: if $G(\lambda_{si}) < G_{max}$ then purge λ_{max} from population, and substitute by offspring λ_{si};

5) Compute the new G_{max}, if necessary;

6) Iterate steps 4 and 5 on the whole breeding set;

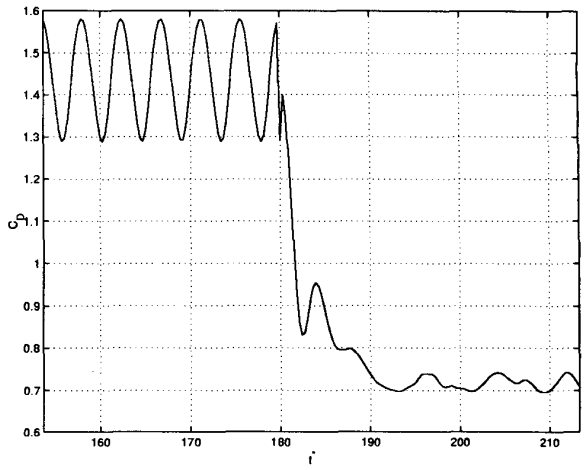


Figure 2: Validation run for the best population member. Control is switched on at $t^* = 180$.

7) If convergence test fails, return to step 1.

There are 4 parameters, i.e. α, β, γ, G_T and 2 variables: I and $\overline{G}$ to be defined in this scheme. The variable I is the number of consecutive iterations in which the population has not changed, i.e., no offspring has substituted any population member. It provides an empirical measure of the need for fresh information through increased mutation probability. The variable $\overline{G}$ is the average population fitness, used as a scaling factor.

The most important parameter is G_T, a threshold C_D value used for the convergence test: we declare convergence once all of the population's fitness values are below it. Given this criterion, population points will be clustered inside the domain defined as: $\{\lambda | G(\lambda) < G_T\}$. The final cluster provides a sampling of the minimum basin. This allows to retrieve parameter correlations and other meaningful quantities.

According to the formula defining mutation probability, the parameter $0 <= \alpha <= 1$ modulates the mutation rate during the course of the optimization process, and the parameter $0 <= \gamma <= 1$ enforces its upper bound, since $0 <= \beta <= 1$. The term containing the parameter β causes population members far from convergence (with fitness $> G_T$) to mutate more frequently. (CRS can be regarded as a GA with zero mutation probability.)

3.3 Results with the GA

For a GA population of 50 elements the parameters α and β were fixed to 0.25, the upper bound γ to 0.02 and the threshold G_T to 0.7. This value is about 50% of the drag coefficient reported for and uncontrolled flow around a cylinder at $Re = 500$ (Panton, 1996). The optimization lasted for 1500 iterations, corresponding to about 30 hours of CPU-time on a NEC SX-4 supercomputer.

On a grid twice as fine as the one used for optimization,

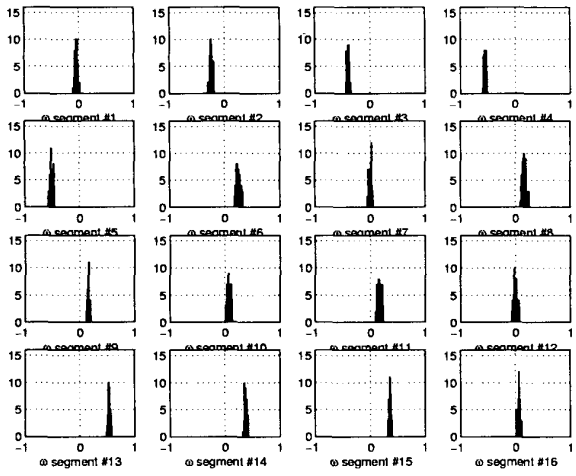


Figure 3: Histogram of the final population cluster: number of population members (ordinates) attaining a given tangential velocity, reported in the abscissas. Segments 4 and 13 contain the flow separation point.

a validation run was performed for the best candidate in the population. Fig. 2 plots the behavior of the drag coefficient during the transition from the uncontrolled and controlled phase. The transition phase is quite short, and the flow appears to settle quickly to the minimal drag configuration. The shedding frequency is drastically modified while the fluctuating amplitude in the drag coefficient is significantly reduced.

Fig. 3 shows a histogram of the converged population. We observe that the GA discovered that segments on opposite sides of the cylinder must rotate in opposite directions, in order to delay separation by allowing the flow to "slide" further on the cylinder surface.

This separation delay also becomes obvious by comparing the time averages of the controlled and uncontrolled vorticity contours, shown in Figs 4 and 5, respectively. The controlled wake is more elongated, a clear evidence of a later separation, resulting in a smaller drag.

4 Cylinder in rotary oscillations

A second set of CFD experiments was performed to optimize the parameters (amplitude, frequency) of rotary oscillations of a circular cylinder. This may be viewed as a special case of the previous experiments: all segments on the cylinder move with equal velocity.

In this case optimization was performed online, based on the same function in Eq. 5.

4.1 Problem setup - Parametrization

We consider flow at $Re = 1000$, undergoing rotary oscillations, with the tangential velocity on the surface of the cylin-

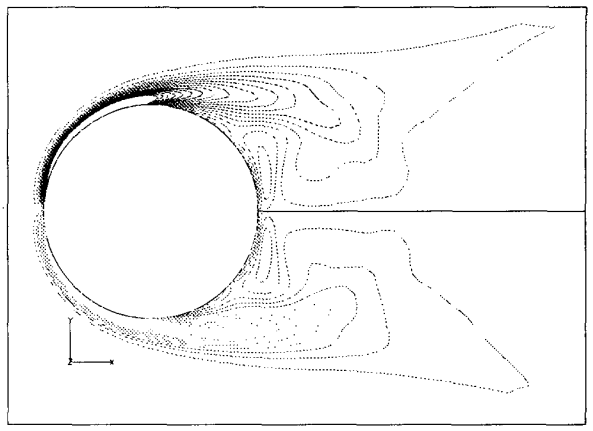


Figure 4: Vorticity contours at $Re = 500$ averaged over 4 Strouhal periods, uncontrolled case.

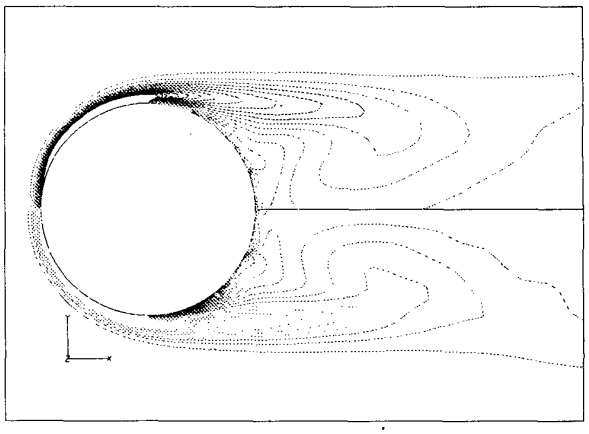


Figure 5: Steady state vorticity contours averaged over 4 Strouhal periods, controlled case.

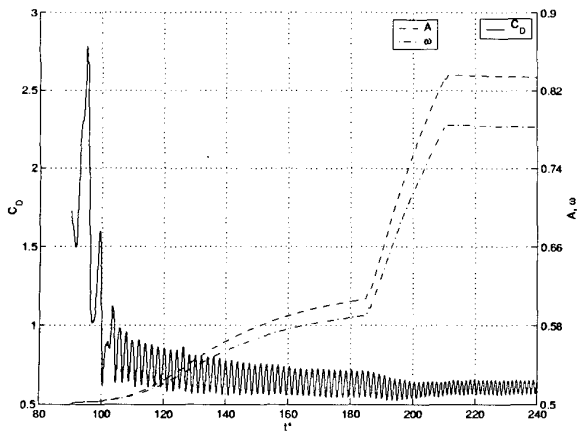


Figure 6: Evolution of the control parameters and of the drag, plotted from the onset of the algorithm.

der described by :

$$v_{slip} = A\pi sin(8\pi\omega t/St). \quad (6)$$

where A, ω denote amplitude and frequency of the rotary oscillations and St denotes the Strouhal frequency of the uncontrolled flow. The grid size used in these experiments is 80×160 (radial and azimuthal directions). The nondimensional time step used in the numerical simulations is 0.003. A typical optimisation run takes about one hour of CPU time on a NEC SX-4 supercomputer.

4.2 Empirical online optimisation

One of the biggest problems with online optimisation of bluff body flow control are the large transients after control initiation. During the flow transient the drag can fluctuate a lot before the flow stabilises. This implies that any online algorithm building on past experience has to trade off the two goals of reactivity and insensitivity to transients.

To achieve this trade-off we make an analogy between optimisation in parameter space and dynamics due to a force field. To generate a force field based on the results given by the process so far, the general guideline is that good examples attract further points while bad examples push them away.

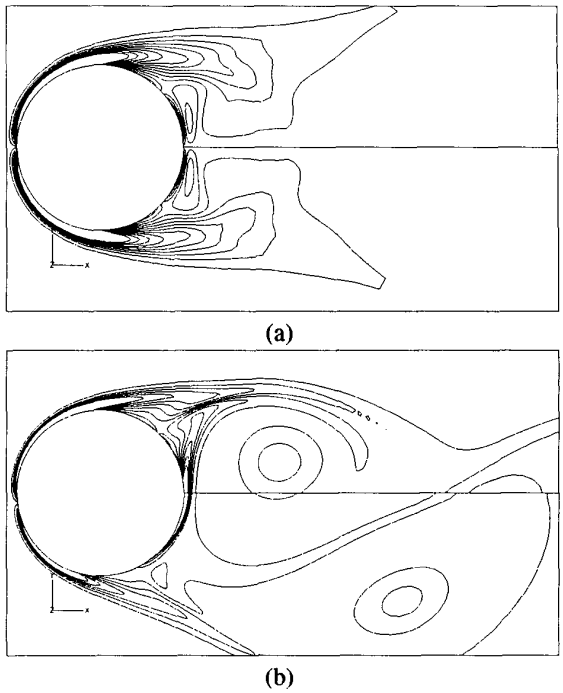
In our experiments, the force is following the line between the two parameter points, and it scales like $||1/\Delta\theta||$, where θ denotes the vector (A, ω). We define the force to be

$$F(\theta, \{j\}) = \frac{1}{|\{j\}|} \sum_j F_j(\theta) \quad (7)$$

and

$$F_i(\theta) = \frac{(C_{D\,i} - \overline{C_D})(\theta - \theta_i)}{||\theta - \theta_i||^2} \quad (8)$$

where the sum is taken over the set of values $\{j\}$. For instance $\{j\} = 1, ..., T$ means all the drag values observed so far. $\overline{C_D}$ is the mean of the drag values, θ and θ_i are base point and past point at instant i in parameter space, respectively.



(a)

(b)

Figure 7: Vorticity contour plots at $Re = 1000$, uncontrolled case. (a) Average, (b) Snapshot from the animation.

Choices that have to be made include the mappings between force and values of the optimised function, the determination of a base point for update and the update rate. Here we take as coefficient the drag minus its average, as calculated by $(C_{D\,i} - \overline{C_D})$ in Eq. 8. This choice ensures that the long range force field component is zero. After a coarse experimental search the rate α was set to 0.001. To summarize, the parameters are updated as

$$\theta_{i+1} = \theta_i + \alpha F(\theta_i, \{1, .., i-1\}) \quad (9)$$

where the force is computed over all previous values but the last one.

Although the algorithm has its own averaging effect, it is more efficient to average the drag over a short period of time. Thus, less points are collected and the computational effort is better balanced. In our case the averaging period was 100 simulation time steps.

4.3 Results of online optimization

Parameters were randomly initialized in a large area suspected to contain parameter combinations corresponding to drag reduction, following previous studies (Koumoutsakos, 1993).

The algorithm indeed discovers parameters leading to significant drag reduction (see Fig. 6). In the end of the simula-

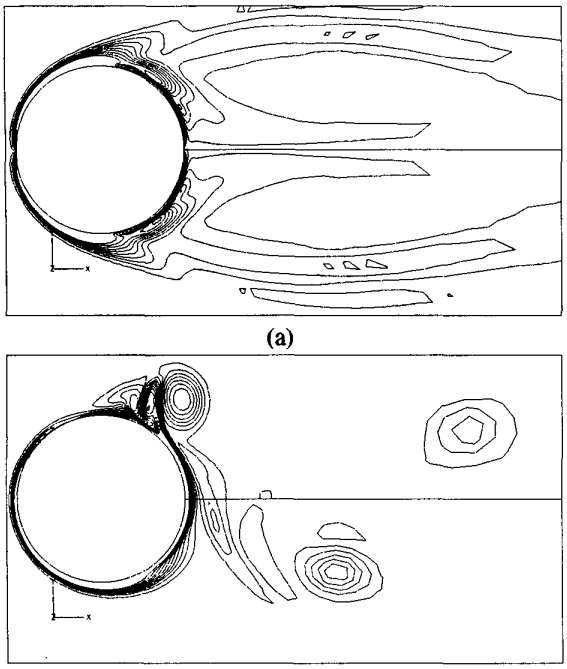


Figure 8: Vorticity contour plot at $Re = 1000$, controlled case. (a) Average, (b) Snapshot.

tion average drag is 0.61. Compared to a drag of 1.52 in the uncontrolled case, this represents a reduction of about 60%.

It is interesting to note (Fig. 6) that the two control parameters are well correlated. In fact, the relation between them almost fits

$$\omega = 0.84A + 0.08. \quad (10)$$

This corresponds to a cylinder move with an amplitude of $\pm 0.4\pi$. This value coincides with separation angle (measured from the rear stagnation point) of the uncontrolled flow, indicating an automatic identification of a critical flow parameter. The oscillations of the cylinder are such that secondary vorticity is pumped into the fluid at the primary separation point, leading to the production of vortex dipoles which in turn imply drag minimization.

A jump is observed around $t^* = 185$, implying a potential instability in the algorithm when previous points exceed the mean drag and start pushing points out of a basin instead of attracting them. The final part of the simulation shows the algorithm stabilising at a lower value. A better solution would include a more complex mapping between function value and force.

The difference arising from control is clearly visible in the vorticity contours shown in Fig. 7 and 8. One of the characteristics of control through oscillations is that tightly formed pairs of vortices (instead of single sign large scale vortices) are alternatively created near the separation point and ejected into the flow field. One such pair is displayed in Fig. 8(b). This mechanism is responsible for drastic drag reduction.

5 Conclusion

Two automatic optimizers have been applied to the problem of active flow control past a cylinder, both yielding unprecedented drag reduction of up to 60%.

Drag was reduced by two important mechanisms, namely: (a) delayed flow separation (discovered by off-line GA optimization – Section 3) and (b) drastic modification of vortical structures through formation and ejection of vortex dipoles at separation points (discovered by on-line heuristic optimization – Section 4).

An important aspect of our particular GA is the dependence of the mutation probability on the distance from a desired minimum.

Our results show for the first time that excellent solutions to benchmark problems of Computational Fluid Dynamics (CFD) can be obtained through evolutionary computation (EC). Since CFD avoids costly real-world experiments, CFD/EC combinations may become an important tool for aerodynamic designs employing on active control devices. Future work will focus on speeding up convergence of the optimizers and on three dimensional flow simulations.

Acknowledgements

This research is funded by the Swiss National Science Foundation grant No 2100-054093.98. Computational support was provided by the Swiss supercomputing center CSCS in Manno.

Bibliography

Apelt, C. and West, G. (1975). The effects of wake splitter plates on bluff body flow in the range $10^4 < r < 5 \times 10^4$: Part 2. *J. Fluid Mech.*, 71:145.

Apelt, C., West, G., and Szewczyk, A. (1973). The effects of wake splitter plates on bluff body flow in the range $10^4 < r < 5 \times 10^4$. *J. Fluid Mech.*, 61:187.

Bechert, D., Hage, W., and Brusek, M. (1996). Drag reduction with the slip wall. *AIAA Journal*, 34(5).

Chen, Y., Ou, Y., and Pealstein, A. (1993). Development of the wake behind a circular cylinder impulsively started into rotatory and rectilinear motion: Intermediate rotation rates. *J. Fluid Mech.*, 253:449.

Cimbala, J. and Garg, S. (1991). Flow in the wake of a freely rotatable cylinder with splitter plate. *AIAA Journal*, 291:1001.

Goldberg, D. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley.

Holland, J. (1975). *Adaptation in Natural and Artificial Systems*. MIT Press.

Koumoutsakos, P. (1993). Control of a circular cylinder performing rotary oscillations. In *Proc. 23rd AIAA Shear Flow Conference*, Orlando, FL, USA. AIAA.

Kwon, K. and Choi, H. (1996). Control of laminar vortex shedding behind a circular cylinder using splitter plates. *Phys. Fluids*, 8(2).

Lee, C., Kim, J., and Choi, H. (1998). Suboptimal control of turbulent channel flow for drag reduction. *J. Fluid Mech.*, 358:245–258.

Milano, M. and Barone, F. (1999). An identification procedure performing automatic sensitivity analysis based on evolutionary programming. *IEEE Transactions on Systems, Man and Cybernetics*. Submitted, available at http://www.ifd.mavt.ethz.ch/members/Milano/.

Mittal, R. (1995). Large-eddy simulation of flow past a circular cylinder. In *Annual Research Briefs*, page 107. Center for Turbulence Research, Stanford.

Modi, V. (1997). Moving surface boundary layer control: a review. *Journal of Fluids and Structures*, 11.

Price, W. (1976). A controlled random search procedure for global optimization. *The Computer Journal*, 20(4).

Rechenberg, I. (1994). *Evolutionsstrategie*. Frommann-Holzboog, Berlin.

Roshko, A. (1955). On the wake and drag of bluff bodies. *J. Aerosp. Sci.*, 22:124.

Shumm, M., Berger, E., and Monkewitz, P. (1994). Self-excited oscillations in the wake of two dimensional bluff bodies and their control. *J. Fluid Mech.*, 271:17.

Strykowsky, P. and Sreenivasan, K. (1990). On the formation and suppression of vortex shedding at low reynolds numbers. *J. Fluid Mech.*, 218:71.

Tokumaru, P. and Dimotakis, P. (1991). Rotary oscillation control of a cylinder wake. *J. Fluid Mech.*, 224:77.

Unal, M. and Rockwell, D. (1987). On vortex formation from a cylinder. part 2. Control by splitter plate interference. *J. Fluid Mech.*, 190:513.