

Explorative gradient method for active drag reduction of the fluidic pinball and slanted Ahmed body

Yiqing Li^{1,2,3}, Wenshi Cui⁴, Qing Jia^{2,3}, Qiliang Li^{2,3}, Zhigang Yang^{2,3,5,†}, Marek Morzyński⁶ and Bernd R. Noack^{1,7,†}

¹School of Mechanical Engineering and Automation, Harbin Institute of Technology, 518055 Shenzhen, PR China

²Shanghai Automotive Wind Tunnel Center, Tongji University, 201804 Shanghai, PR China

³Shanghai Key Lab of Vehicle Aerodynamics and Vehicle Thermal Management Systems, 201804 Shanghai, PR China

⁴SAIC Volkswagen Automotive Co., Ltd., 201804 Shanghai, PR China

⁵Beijing Aeronautical Science & Technology Research Institute, 102211 Beijing, PR China

⁶Department of Virtual Engineering, Poznań University of Technology, Jana Pawła II 24, PL 60-965 Poznań, Poland

⁷Hermann-Föttinger-Institut, Technische Universität Berlin, Müller-Breslau-Straße 8, D-10623, Berlin, Germany

(Received 23 September 2020; revised 29 September 2021; accepted 1 November 2021)

We address a challenge of active flow control: the optimization of many actuation parameters guaranteeing fast convergence and avoiding suboptimal local minima. This challenge is addressed by a new optimizer, called the explorative gradient method (EGM). EGM alternatively performs one exploitive downhill simplex step and an explorative Latin hypercube sampling iteration. Thus, the convergence rate of a gradient based method is guaranteed while, at the same time, better minima are explored. For an analytical multi-modal test function, EGM is shown to significantly outperform the downhill simplex method, the random restart simplex, Latin hypercube sampling, Monte Carlo sampling and the genetic algorithm. EGM is applied to minimize the net drag power of the two-dimensional fluidic pinball benchmark with three cylinder rotations as actuation parameters. The net drag power is reduced by 29 % employing direct numerical simulations at a Reynolds number of 100 based on the cylinder diameter. This optimal actuation leads to 52 % drag reduction employing Coanda forcing for boat tailing and partial stabilization of vortex shedding. The price is an actuation energy corresponding

† Email addresses for correspondence: zhigangyang@tongji.edu.cn; bernd.noack@hit.edu.cn

to 23 % of the unforced parasitic drag power. EGM is also used to minimize drag of the 35° slanted Ahmed body employing distributed steady blowing with 10 inputs. 17 % drag reduction are achieved using Reynolds-averaged Navier–Stokes simulations at the Reynolds number $Re_H = 1.9 \times 10^5$ based on the height of the Ahmed body. The wake is controlled with seven local jet-slot actuators at all trailing edges. Symmetric operation corresponds to five independent actuator groups at top, middle, bottom, top sides and bottom sides. Each slot actuator produces a uniform jet with the velocity and angle as free parameters, yielding 10 actuation parameters as free inputs. The optimal actuation emulates boat tailing by inward-directed blowing with velocities which are comparable to the oncoming velocity. We expect that EGM will be employed as efficient optimizer in many future active flow control plants as alternative or augmentation to pure gradient search or explorative methods.

Key words: aerodynamics, flow control, wakes/jets

1. Introduction

In this study, we propose an optimizer for active flow control focusing on multi-actuator bluff-body drag reduction. This optimizer combines the convergence rate of a gradient search method with an explorative method for identifying the global minimum. Actuators and sensors are becoming increasingly cheaper, powerful and reliable. This trend makes active flow control of increasing interest to industry. In addition, distributed actuation can give rise to performance benefits over single actuator solutions. Here, we focus on the simple case of open-loop control with steady or periodic operation of multiple actuators.

Even for this simple case, the optimization of actuation constitutes an algorithmic challenge. Often the budget for optimization is limited to $O(100)$ high fidelity simulations, like direct numerical simulations (DNS) or large-eddy simulations (LES) or $O(100)$ water tunnel experiments, or $O(1000)$ Reynolds-averaged Navier–Stokes (RANS) simulations, or a similar amount of wind-tunnel experiments. Moreover, the optimization may need to be performed for multiple operating conditions.

Evidently, efficient optimizers are of large practical importance. Gradient-based optimizers, like the downhill simplex method (DSM) have the advantage of rapid convergence against a cost minimum, but this minimum may easily be a suboptimal local one, particularly for high-dimensional search spaces. Random restart variants have a larger probability of finding the global minimum but come with a dramatic increase of testing. In contrast to gradient-based approaches, Latin hypercube sampling (LHS) performs an ideal exploration by guaranteeing a close geometric coverage of the search space – obviously with a poor associated convergence rate and the price of extensive evaluations of unpromising territories. Monte Carlo sampling (MCS) has similar advantages and disadvantages. Genetic algorithms (GA) elegantly combine exploration with mutation and exploitation with crossover operations. These are routinely used optimizers and the focus of our study. These optimizers are superbly described in Press *et al.* (2007). Myriad of other optimizers have been invented for different niche applications. Deterministic gradient-based optimizers may be augmented by estimators for the gradient. These estimators become particularly challenging for sparse data. This challenge is addressed by stochastic gradient methods which aim at navigating through a high-dimensional search space with insufficient derivative information. Many biologically inspired optimization methods, like ant colony and particle swarm optimization, also aim at balancing

exploitation and exploration, like GA. A new avenue is opened by including the learning of the response model from actuation to cost function during the optimization process and using this model for identifying promising actuation parameters. Another new path is ridgeline inter- or extrapolation (Fernex *et al.* 2020), exploiting the topology of the control landscape. In this study, these extensions are not included in the comparative analysis, as the additional complexity of these methods with many additional tuning parameters can hardly be objectively performed.

Our first flow control benchmark is the fluidic pinball (Ishar *et al.* 2019; Deng *et al.* 2020). This two-dimensional flow around three equal, parallel, equidistantly placed cylinders can be changed by the three rotation velocities of the cylinders. The dynamics is rich in nonlinear behaviour, yet geometrically simple and physically interpretable. With suitable rotation of the cylinders many known wake stabilizing and drag-reducing mechanisms can be realized: (i) Coanda actuation (Geropp 1995; Geropp & Odenthal 2000), (ii) circulation control (Magnus effect), (iii) base bleed (Wood 1964), (iv) high-frequency forcing (Thiria, Goujon-Durand & Wesfreid 2006), (v) low-frequency forcing (Glezer, Amitay & Honohan 2005) and (vi) phasor control (Protas 2004). In this study, constant rotations are optimized for net drag power reduction accounting for the actuation energy. This search space implies the first three mechanisms. The fluidic pinball study will foreshadow key results of the Ahmed body. This includes the drag-reducing actuation mechanism and the visualization tools for high-dimensional search spaces.

The main application focus of this study is on active drag reduction behind a generic car model using RANS simulations. Aerodynamic drag is a major contribution of traffic-related costs, from airborne to ground and marine traffic. A small drag reduction would have a dramatic economic effect considering that transportation accounts for approximately 20 % of global energy consumption (Gad-el-Hak 2007; Kim 2011; Choi, Lee & Park 2014). While the drag of airplanes and ships is largely caused by skin friction, the resistance of cars and trucks is mainly caused by pressure or bluff-body drag (Seifert *et al.* 2009). Hucho (2011) defines bodies with a pressure drag exceeding the skin-friction contribution as bluff and as streamlined otherwise.

The pressure drag of cars and trucks originates from the excess pressure at the front scaling with the dynamic pressure and a low-pressure region at the rear. The reduction of the pressure contribution from the front often requires significant changes of the aerodynamic design. Few active control solutions for the front drag reduction have been suggested (Minelli *et al.* 2020). In contrast, the contribution at the rearward side can be significantly changed with passive or active means. Drag reductions of 10 %–20 % are common, Pfeiffer & King (2014) have even achieved 25 % drag reduction with active blowing in the wind tunnel. For a car at a speed of 120 km h^{-1} , this would reduce consumption by approximately 1.8 l per 100 km. The economic impact of drag reduction is significant for trucking fleets with a profit margin of only 2 %–3 %. Two thirds of the operating costs are from fuel consumption (Brunton & Noack 2015). Hence, a 5 % reduction of fuel costs from aerodynamic drag corresponds to over 100 % increase of the profit margin.

Car and truck design is largely determined by practical and aesthetic considerations. In this study, we focus on drag reduction by active means at the rearward side. Intriguingly, most drag reductions of a bluff body fall into the categories of Kirchhoff solution and aerodynamic boat tailing. The first strategy may be idealized by the Kirchhoff solution (Pastoor *et al.* 2008), i.e. potential flow around the car with infinitely thin shear layers from the rearward separation lines, separating the oncoming flow and dead-fluid region. The low-pressure region due to curved shear layers is replaced by an elongated, ideally

infinitely long wake with small, ideally vanishing curvature of the shear layer. Thus, the pressure of the dead-water region is elevated to the outer pressure, i.e. the wake does not contribute to the drag. This wake elongation is achieved by reducing entrainment through the shear layer, e.g. by phasor control mitigating vortex shedding (Pastoor *et al.* 2008) or by energization of the shear layer with high-frequency actuation (Barros *et al.* 2016). Wake disrupters also decrease drag by energizing the shear layer (Park *et al.* 2006) or delaying separation (Aider, Beaudoin & Wesfreid 2010). Arguably, the drag of the Kirchhoff solution can be considered as an achievable limit with small actuation energy.

The second strategy targets drag reduction by aerodynamic boat tailing. Geropp (1995) and Geropp & Odenthal (2000) have pioneered this approach by Coanda blowing. Here, the shear layer originating at the bluff body is vectored inward and thus gives rise to a more streamlined wake shape. Barros *et al.* (2016) have achieved 20% drag reduction of a square-back Ahmed body with high-frequency Coanda blowing in a high-Reynolds-number experiment. A similar drag reduction was achieved with steady blowing but at higher flow coefficients.

This study focuses on drag reduction of the low-drag Ahmed body with rear slant angle of 35°. This Ahmed body simplifies the shape of many cars. Bideaux *et al.* (2011) and Gilliéron & Kourta (2013) have achieved 20% drag reduction for this configuration in an experiment. High-frequency blowing was applied orthogonal to the upper corner of the slanted rear surface. Intriguingly, the maximum drag reduction was achieved in a narrow range of frequencies and actuation velocities and its effect rapidly deteriorated for slightly changed parameters. In addition, the actuation is neither Coanda blowing nor an ideal candidate for shear-layer energization, as the authors noted.

The literature on active drag reduction of the Ahmed body indicates that small changes of actuation can significantly change its effectiveness. Actuators have been applied with beneficial effects at all rearward edges (Barros *et al.* 2016), thus further complicating the optimization task. A systematic optimization of the actuation at all edges, including amplitudes and angles of blowing, is beyond reach of current experiments. In this study, a systematic RANS optimization is performed in a rich parametric space comprising the angles and amplitudes of steady blowing of five actuator groups: one on the top, middle and bottom edge and two symmetric actuators at the corners of the slanted and vertical surfaces. High-frequency forcing is not considered, as the RANS tends to be overly dissipative to the actuation response.

The manuscript is organized as follows. The employed optimization algorithms are introduced in §2 and compared in §3. §4 optimizes the net drag power for the fluidic pinball, which features two-dimensional flow controlled in a three-dimensional actuation space based on DNS. A simulation-based optimization of actuation for the three-dimensional low-drag Ahmed body is given in §5. Here, up to 10 actuation commands controlling the velocity and direction of five rearward slot actuator groups are optimized. Our results are summarized in §6. Future directions are indicated in §7.

2. Optimization algorithms

In this section, the employed optimization algorithms for the actuation parameters are described. Let $J(\mathbf{b})$ be the cost function – here the drag coefficient – depending on N actuation parameters $\mathbf{b} = [b_1, \dots, b_N]^T$ in the domain Ω ,

$$\mathbf{b} = [b_1, \dots, b_N]^T \in \Omega \subset \mathcal{R}^N, \quad (2.1)$$

Explorative gradient method

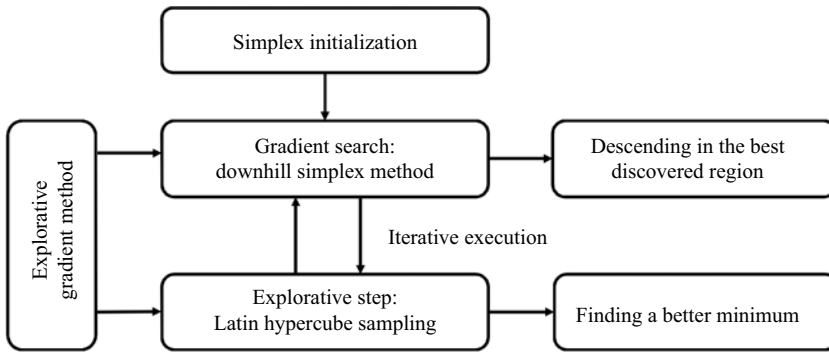


Figure 1. Sketch of the explorative gradient method. For details, see text.

where the superscript ‘T’ denotes the transpose. Permissible values of each parameter define an interval, $b_i \in [b_{i,min}, b_{i,max}]$, $i = 1, \dots, N$. In other words, optimization is performed in rectangular search space,

$$\Omega = [b_{1,min}, b_{1,max}] \times \dots \times [b_{N,min}, b_{N,max}]. \quad (2.2)$$

The optimization goal is to find the global minimum of J in Ω ,

$$\mathbf{b}^* = \arg \min_{\mathbf{b} \in \Omega} J(\mathbf{b}). \quad (2.3)$$

Several common optimization methods are investigated. Benchmark is the DSM (see, e.g. Press *et al.* 2007) as a robust data-driven representative of gradient search methods (§ 2.4). This algorithm exploits gradient information from neighbouring points to descend to a local minimum. Note that DSM is referred to as a gradient-free method in numerical textbooks because it does not require explicitly the gradient vector as input nor an estimation thereof.

Depending on the initial condition, this search may yield any local minimum. In random restart simplex (RRS), the chance of finding a global minimum is increased by multiple runs with random initial conditions. The geometric coverage of the search space is the focus of LHS (see, again, Press *et al.* 2007), which optimally explores the whole domain Ω independently of the cost values, i.e. ignores any gradient information. Evidently, LHS has a larger chance of getting close to the global minimum while DSM is more efficient descending to a minimum, potentially a suboptimal one. MCS (see, again, Press *et al.* 2007), is a simpler and more common exploration strategy by taking random values for each argument, again, ignoring any cost value information. GA starts with a MCS in the first generation but then employs genetic operations to combine explorative and exploitive features in the following generations (see, e.g. Wahde 2008).

Sections 2.1–2.3 outline the non-gradient-based explorative methods from the most explorative LHS, to MCS and the partially exploitive GA. Sections 2.4 and 2.5 recapitulate DSM and its random restart variant. These are commonly used methods for data-driven optimization with an unknown analytical cost function.

In § 2.6, we combine the advantages of DSM in exploiting a local minimum and of the LHS in exploring the global one in a new method, the explorative gradient method (EGM), by an alternative execution (see figure 1). § 2.7 discusses auxiliary accelerators which are specific to the performed computational fluid dynamics optimization.

2.1. LHS – deterministic exploration

While our DSM benchmark exploits neighbourhood information to slide down to a local minimum, LHS (McKay, Beckman & Conover 1979) aims to explore the parameter space irrespective of the cost values. We employ a space-filling variant which effectively covers the whole permissible domain of parameters. This explorative strategy (‘maximin’ criterion in Mathematica) minimizes the maximum minimal distance between the points

$$\{\mathbf{b}_m\}_{m=1}^M = \arg \max_{\mathbf{b}_m \in \Omega} \min_{\substack{i=1, \dots, M-1, \\ j=i+1, \dots, M}} \|\mathbf{b}_i - \mathbf{b}_j\|. \quad (2.4)$$

In other words, there is no other sampling of M parameters with a larger minimum distance; M can be any positive integral number.

For better comparison with DSM, we employ an iterative variant. Note that, once M sample points are created, they cannot be augmented anymore, for instance when learning by LHS was not satisfactory. We create a large number of LHS candidates \mathbf{b}_j^* , $j = 1, \dots, M^*$ for a dense coverage of the parameter space Ω at the beginning, typically $M^* = 10^6$. As first sample \mathbf{b}_1 , the centre of the initial simplex is taken. The second parameter is taken from \mathbf{b}_j^* , $j = 1, \dots, M^*$ maximizing the distance to \mathbf{b}_1 ,

$$\mathbf{b}_2 = \arg \max_{j=1, \dots, M^*} \|\mathbf{b}_j^* - \mathbf{b}_1\|. \quad (2.5)$$

The third parameter \mathbf{b}_3 is taken from the same set so that the minimal distance to \mathbf{b}_1 and \mathbf{b}_2 is maximized, and so on. This procedure allows us to recursively refine sample points and to start with an initial set of parameters.

2.2. MCS – stochastic exploration

The employed space-filling variant of LHS requires the solution of an optimization problem guaranteeing a uniform geometric coverage of the domain. In high-dimensional domains, this coverage may not be achievable. A much easier and far more commonly used exploration strategy is MCS. Here, the m th sample $\mathbf{b}_m = [b_{1,m}, \dots, b_{N,m}]^T$ is given by

$$b_{i,m} = b_{i,min} + \zeta_{i,m}(b_{i,max} - b_{i,min}), \quad (2.6)$$

where $\zeta_{i,m} \in [0, 1]$ are random numbers with uniform probability distribution in the unit domain. The relative performance of LHS and MCS is a debated topic. We will wait for the results of an analytical problem in § 3.

2.3. GA – biologically inspired exploration and exploitation

GA mimics the natural selection process. We refer to Wahde (2008) as excellent reference. In the following, the method is briefly outlined to highlight the specific version we use and the chosen parameters.

Any parameter vector $\mathbf{b} = [b_1, b_2, \dots, b_N]^T \in \Omega \subset \mathcal{R}^N$ comprises the real values b_i , also called alleles. This real value is encoded as a binary number and called the gene. The chromosome comprises alle genes and represents the parameter vector (Wright 1991).

GA evolves one generation of I parameters, also called individuals, into a new generation with the same number of parameters using biologically inspired genetic operations.

Explorative gradient method

The first generation is based on MCS, i.e. represents completely random genomes. The individuals $J_i^1, i = 1, \dots, I$ are evaluated and sorted by their costs

$$J_1^1 \leq J_2^1 \leq \dots \leq J_I^1. \quad (2.7)$$

The next generation is computed with elitism and two genetic operations. Elitism copies the N_e best performing individuals in the new generation; $P_e = N_e/I$ denotes the relative quota. The two genetic operations include mutation, which randomly changes parts of the genom, and crossover, which randomly exchanges parts of the genomes of two individuals. Mutation serves explorative purposes and crossover has the tendency to breed better individuals. In an outer loop, the genetic operations are randomly chosen with probabilities P_m and P_c for mutation and crossover, respectively. Note that $P_e + P_m + P_c = 1$ by design.

In the inner loop, i.e. after the genetic operation is determined, individuals from the current generation are chosen. Higher performing individuals have higher probability of being chosen. Following the Matlab routine, this probability is proportional to the inverse square root of its relative rank $p \propto 1/\sqrt{i}$.

GA terminates according to a predetermined stop criterion, a maximum number of generations L or corresponding number of evaluations $M = IL$. For reasons of comparison, we renumber the individuals in the order of their evaluation, i.e. $m \in \{1, \dots, I\}$ belongs to the first generation, $m \in \{I + 1, \dots, 2I\}$ to the second generation, etc.

The chosen parameters are the default values of Matlab, e.g. $P_r = 0.05, P_c = 0.8, P_m = 0.15, N_e = 3$. Further details are provided in [Appendix A](#).

2.4. *DSM – a robust gradient method*

DSM proposed by Nelder & Mead (1965) is a very simple, robust and widely used gradient method. This method does not require any gradient information and is well suited for expensive function evaluations, like the considered RANS simulation for the drag coefficients, and for experimental optimizations with inevitable noise. The price is a slow convergence for the minimization of smooth functions as compared with algorithms which can exploit gradient and curvature information.

We briefly outline the employed downhill simplex algorithm, as there are many variants. First, $N + 1$ vertices $\mathbf{b}_m, m = 1, \dots, N + 1$ in Ω are initialized as detailed in the respective sections. Commonly, \mathbf{b}_{N+1} is placed somewhere in the middle of the domain and the other vertices explore steps in all directions, $\mathbf{b}_m = \mathbf{b}_{N+1} + h \mathbf{e}_m, m = 1, \dots, N$. Here, $\mathbf{e}_i = [\delta_{i1}, \dots, \delta_{iN}]^T$ is a unit vector in the i th direction and h is a step size which is small compared with the domain. Evidently, all vertices must remain in the domain $\mathbf{b}_m \in \Omega$.

The goal of the simplex transformation iteration is to replace the worst argument \mathbf{b}_h of the considered simplex by a new better one \mathbf{b}_{N+2} . This is archived in following steps:

- (i) Ordering: without loss of generality, we assume that the vertices are sorted in terms of the cost values $J_m = J(\mathbf{b}_m): J_1 \leq J_2 \leq \dots \leq J_{N+1}$.
- (ii) Centroid: in the second step, the centroid of the best side opposite to the worst vertex \mathbf{b}_{N+1} is computed:

$$\mathbf{c} = \frac{1}{N} \sum_{m=1}^N \mathbf{b}_m. \quad (2.8)$$

- (iii) Reflection: reflect the worst simplex \mathbf{b}_{N+1} at the best side,

$$\mathbf{b}_r = \mathbf{c} + (\mathbf{c} - \mathbf{b}_{N+1}) \quad (2.9)$$

and compute the new cost $J_r = J(\mathbf{b}_r)$. Take \mathbf{b}_r as new vertex, if $J_1 \leq J_r \leq J_N$. \mathbf{b}_m , $m = 1 \dots, N$ and \mathbf{b}_r define the new simplex for the next iteration. Renumber the indices to the $1 \dots N + 1$ range. Now, the cost is better than the second worst value J_N , but not as good as the best one J_1 . Start a new iteration with step i.

- (iv) Expansion: if $J_r < J_1$, expand in this direction further by a factor 2,

$$\mathbf{b}_e = \mathbf{c} + 2(\mathbf{c} - \mathbf{b}_{N+1}). \quad (2.10)$$

Take the best vertex of \mathbf{b}_r and \mathbf{b}_e as \mathbf{b}_{N+1} replacement and start a new iteration.

- (v) Single contraction: at this stage, $J_r \geq J_N$. Contract the worst vertex half-way towards centroid,

$$\mathbf{b}_c = \mathbf{c} + \frac{1}{2}(\mathbf{b}_{N+1} - \mathbf{c}). \quad (2.11)$$

Take \mathbf{b}_c as the new vertex (\mathbf{b}_{N+1} replacement), if it is better than the worst one, i.e. $J_c \leq J_{N+1}$. In this case, start the next iteration.

- (vi) Shrink/multiple contraction: at this stage, none of the above operations was successful. Shrink the whole simplex by a factor 1/2 towards the best vertex, i.e. replace all vertices by

$$\mathbf{b}_m \mapsto \mathbf{b}_1 + \frac{1}{2}(\mathbf{b}_m - \mathbf{b}_1), m = 2, \dots, N + 1. \quad (2.12)$$

This shrunken simplex represents the one for the next iteration. It should be noted that this shrinking operation is the last resort as it is very expensive with N function evaluations. The rationale behind this shrinking is that a smaller simplex may better follow local gradients.

2.5. RRS – preparing for multiple minima

DSM of the previous section may be equipped with a random restart initialization (Humphrey & Wilson 2000). Maehara & Shimoda (2013) proposed a hybrid method of GA and DSM, which shared a similar idea, for the optimization of the chiller configuration. This method selects the elite gene with the lowest cost by GA as the starting point for the downhill simplex iteration.

As the random initial condition, we chose a MCS as main vertex of the simplex and explore all coordinate directions by a positive shift of 10% of the domain size. It is secured that all vertices are inside the domain Ω . These initial simplexes attribute the same probability to the whole search space. The chosen small edge length makes a locally smooth behaviour probable – in the absence of any other information. The downhill search is stopped after a fixed number of evaluations. We chose 50 evaluations as the safe upper bound for convergence. It should be noted that the number of simplex iterations is noticeably smaller, as one iteration implies one to $N + 2$ evaluations.

Evidently, the random restart algorithm may be improved by appreciating the many recommendations of the literature, e.g. avoiding closeness to explored parameters. We trade these improvements in all optimization strategies for simplicity of the algorithms.

2.6. EGM – combining exploration and gradient method

In this section, we combine the advantages of the exploitive DSM and the explorative LHS in a single algorithm. The source code is freely available at <https://github.com/YiqingLiAnne/egm>.

Step 0 – Initialize. First, $\mathbf{b}_m, m = 1, \dots, M + 1$ are initialized for the DSM.

Step 1 – DSM. Perform one simplex iteration (§ 2.4) with the best $M + 1$ parameters discovered so far.

Step 2 – LHS. Compute the cost J of a new LHS parameter \mathbf{b} . As described above, we take a parameter from a precomputed list which is the furthest away from all hitherto employed parameters.

Step 3 – Loop. Continue with step 1 until a convergence criterion is met.

Sometimes, the simplex may degenerate to one with small volume, for instance, when it crawls through a narrow valley. In this case, the vertices lie in a subspace and valuable gradient information is lost. This degeneration is diagnosed and cured after step 1 as follows. Let \mathbf{b}_c be the geometric centre of the simplex. Compute the distance D between each vertex and their geometric centre point. If the minimum D_{min} is smaller than half the maximum distance $D_{max}/2$, the simplex is deemed degenerated. This degeneration is removed as follows. Draw a sphere with D_{max} around the simplex centre. This sphere contains all vertices by construction. Obtain 1000 random points in this sphere. Replace the vertex with the highest cost J_{max} with one of these point to create the simplex with the largest volume. Of course, the cost of this changed point needs to be evaluated.

The algorithm is intuitively appealing. If the LHS discovers a parameter with a cost J in the top $M + 1$ values, this parameter is included in the new simplex and the corresponding iteration may slide down to another better minimum. It should be noted that LHS exploration does not come with the toll of having to evaluate the cost at $N + 1$ vertices and subsequent iterations. The downside of a single evaluation is that we miss potentially important gradient information pointing to an unexplored much better minimum. Relative to random restart gradients searches requiring many evaluations for a converging iteration, LHS exploration becomes increasingly better in rougher landscapes, i.e. more complex multi-modal behaviour.

2.7. Computational accelerators

The RANS-based optimization may be accelerated by enablers which are specific to the chosen flow control problem. The computation time for each RANS simulation is based on the choice of the initial condition, as it affects the convergence time for the steady solution. The first simulation of an optimization starts with the unforced flow as initial condition. The next iterations exploit that the averaged velocity field $\bar{\mathbf{u}}(\mathbf{x})$ is a function of the actuation parameter \mathbf{b} . The initial condition of the m th simulation is obtained with the 1-nearest-neighbour approach: the velocity field associated with the closest hitherto computed actuation vector is taken as initial condition for the RANS simulation. This simple choice of initial condition saves approximately 60% CPU time in reduced convergence time.

Another 30% reduction of the CPU time is achieved by avoiding RANS computations with very similar actuations. This is achieved by a quantization of the \mathbf{b} vector: the actuation velocities are quantized with respect to integral m s^{-1} values. This corresponds to increments of $U_\infty/30$ with $U_\infty = 30 \text{ m s}^{-1}$. All actuation vectors are rounded with respect to this quantization. If the optimization algorithm yields a rounded actuation vector

which has already been investigated, the drag is taken from the corresponding simulation and no new RANS simulation is performed. Similarly, the angles are discretized into integral degrees.

3. Comparative optimization study

In this section, the six optimization methods of § 2 are compared for an analytical function with 4 local minima.

§ 3.1 describes this function. In § 3.2, the optimization methods with corresponding parameters are discussed. § 3.3 shows the tested individuals. The learning rates are detailed in § 3.4. Finally, the results are summarized (§ 3.5).

3.1. Analytical function

The considered analytical cost function

$$J(b_1, b_2) = 1 - e^{-2(b_1-1)^2-2(b_2-1)^2} - \frac{1}{2}e^{-2(b_1+1)^2-2(b_2-1)^2} - \frac{1}{3}e^{-2(b_1-1)^2-2(b_2+1)^2} - \frac{1}{4}e^{-2(b_1+1)^2-2(b_2+1)^2} \quad (3.1)$$

is characterized by a global minimum near $[1, 1]^T$ and three local minima separately near $[1, -1]^T$, $[-1, 1]^T$ and $[-1, -1]^T$. The cost reaches a plateau $J = 1$ far away from the origin. The investigated parameter domain is $\Omega = [-3, 3] \times [-3, 3]$.

3.2. Optimization methods and their parameters

LHS is performed as described in § 2.1. We take $M^* = 10^3$ random points for the optimization of the coverage. MCS is uniformly distributed over the parameter domain Ω .

The most important parameters of GA are summarized from Appendix A: the generation size is $I = 50$ and the iterations are terminated with generation $L = 20$. The crossover and mutation probabilities are $P_c = 0.80$ and $P_m = 0.15$, respectively. The number of elite individuals $N_e = 3$ corresponds to the complementary probability $P_e = 5\%$.

DSM follows exactly the description of § 2.4 with an expansion rate of 2, single contraction rate of 1/2 and a shrink rate of 1/2. RRS (§ 2.5) has an evaluation limit of 50 for 20 random restarts. The step size for each initial simplex is $h = 0.35$. EGM builds on the LHS and downhill simplex iterations discussed above.

3.3. Tested individuals in the parameter space

Figure 2 illustrates the iteration of all six algorithms in the parameter space. LHS shows a uniform coverage of the domain. In contrast, MCS leads to local ‘lumping’ of close individuals, i.e. indications of redundant testing, and local untested regions, both undesirable features. Thus, LHS is clearly seen to perform better than MCS. GA is seen to sparsely test the plateau while densely populating the best minima. This is clearly a desirable feature over LHS and MCS.

The standard DSM converges to a local minimum in this realization, while RRS finds all minima, including the global optimal one. Clearly, the random restart initialization is a security policy against sliding into a suboptimal minimum. The proposed EGM finds all four minima and converges against the global one. By contrast, the exploration is less dense in LHS. The 1000 iterations comprise approximately 250 LHS steps and approximately 250 downhill simplex iterations with an average of three evaluations for each.

Explorative gradient method

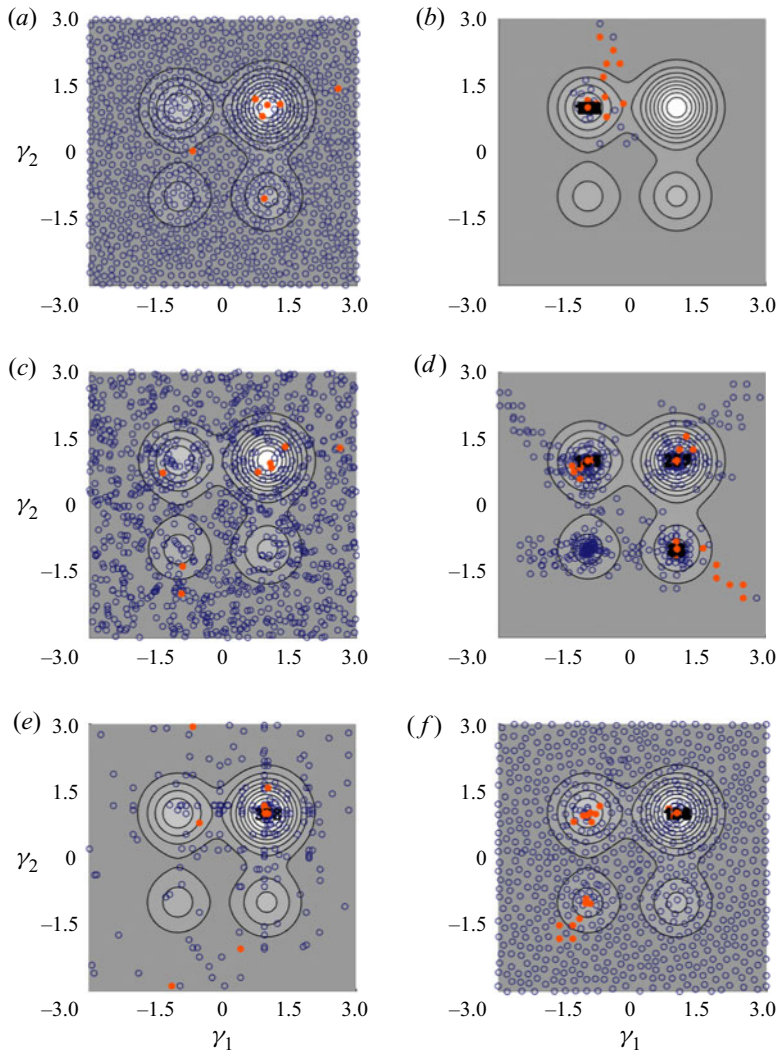


Figure 2. Comparison of all optimizers for the analytical function (3.1). Tested individuals of (a) LHS, (b) DSM, (c) MCS, (d) RRS, (e) GA and (f) EGM from a typical optimization with 1000 individuals. The red solid circles mark new local minima during the iteration while the blue open circles represent suboptimal tested parameters.

Arguably, the EGM is seen to be superior to all downhill simplex variant with more dense exploration and convergence to the global optimum. EGM also performs better than LHS, MCS and GA, as it invests in a more dense coverage of the parameter domain while approximately 75 % of the evaluations serve the convergence.

The conclusions are practically independent of the chosen realization of the optimization algorithm, except that the DSM slides into the global minimum in approximately 27 % of the cases.

We note that some of our conclusions are tied to the low dimension of the parameter space. In a cubical domain of 10 dimensions, the first $2^{10} = 1024$ LHS individuals would populate the corners before the interior is explored. A geometric coverage of higher dimensions is incompatible with a budget of 1000 evaluations.

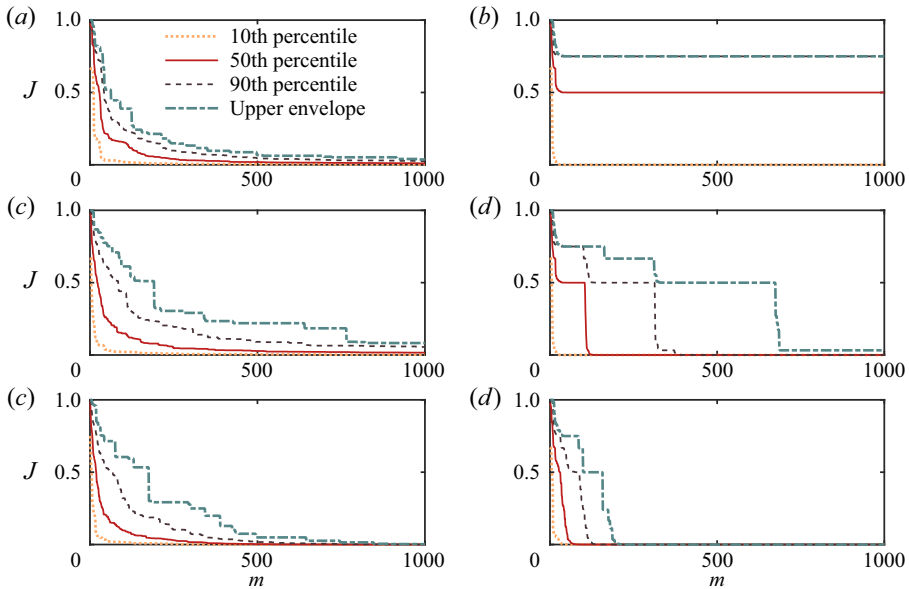


Figure 3. Comparison of all optimizers for the analytical function (3.1). Learning curve of (a) LHS, (b) DSM, (c) MCS, (d) RRS, (e) GA and (f) EGM in 100 runs. The 10th, 50th, and 90th percentiles indicate the J value below which 10, 40 and 90 per cent of runs at current evaluation fall.

3.4. The learning curve

In figure 3, we investigate the learning curve of each algorithm for 100 realizations with randomly chosen initial conditions. The learning curve shows the best cost value found with m evaluations. In this statistical analysis, the 10, 50 and 90 % percentiles of the learning curves are displayed. The 10 % percentile at m evaluations implies that 10 % of the realizations yield better and 90 % yield worse cost values. The 50 % and 90 % percentiles are defined analogously.

The gradient-free algorithms (LHS, MC, GA) in the left column show smooth learning curves. All iterations eventually converge against the global optimum as seen from the upper envelope. The 10 % and 50 % percentile curves are comparable. Focusing on the bad case (90 % percentile) and worst case performance (upper envelope), LHS is seen to beat both MCS and GA. MCS has the worst outliers, because it neither exploits the cost function, like the GA, nor comes with advantage of guaranteed good geometric coverage, like LHS.

The gradient-based algorithms reveal other features. The DSM can arrive at the global optimum much faster than any of the gradient-free algorithms. But it has also a 73 % probability terminating in one of the suboptimal local minima. The random restart version mitigates this risk to practically zero. In RRS, 50 % of the runs reach the minimum before 300 evaluations.

The learning curve of all gradient-based algorithms have jumps. Once the initial condition is in the attractive basin of one minimum, the convergence to that minimum is very fast, leading to a step decline of the learning curve. We notice that the worst case scenario does not exactly converge to zero. The reason is the degeneration of the simplex to points on a line which does not go through the global minimum. Only the EGM takes care of this degeneration with a geometric correction after step 1 as described in § 2.6. Expectedly, EGM also outperforms all other optimizers with respect to 50 % percentile,

Method	Evaluation				Failure rate
	20th	100th	500th	1000th	
LHS	0.5163	0.1456	0.0218	0.0129	0.55
MCS	0.4810	0.1863	0.0441	0.0221	0.61
GA	0.4269	0.1441	0.0065	0.0002	0
DSM	0.4893	0.4675	0.4673	0.4673	0.73
RRS	0.4893	0.3208	0.0211	0.0003	0.01
EGM	0.5121	0.0621	0.0000	0.0000	0

Table 1. Comparison of all optimizers for the analytical function (3.1). Average cost of different algorithms during $m = 20, 100, 500$ and 1000 evaluations in 100 runs.

90 % percentile and the worst case scenario. The global minimum is consistently found in less than 200 evaluations. It is much more efficient to invest 50 points in LHS exploration than to iterate into a suboptimal minimum. The gradient descent slowed down from the distraction by 25 % LHS evaluations as an insurance policy.

3.5. Discussion

The relative strengths and weaknesses of the different optimizers are summarized in table 1 for the average performance after $m = 20, 100, 500$ and 1000 evaluations. The averaging is performed over the costs of all 100 realizations after m evaluations. The iteration is considered as failed if the value is 1 %, i.e. 0.01 above the global minimum.

First, we observe that DSM has the worst failure rate with 73 %, followed by 61 % of MCS and 55 % of LHS. The failures of the DSM are more severe as the converged parameters significantly depart from the global minimum in 73 % of the runs. In case of LHS and MCS, the failure is only the result of pure convergence against the right global minimum.

Second, after 20 evaluations, the average cost of all algorithms is close to 0.50, i.e. very similar.

After 100 evaluations, algorithms with explorative steps, i.e. LHS, MCS, GA and EGM have a distinct advantage over the DSM and even over the random restart version. Approximately four restarts are necessary to avoid the convergence to a suboptimal minimum in 99 % of the cases. EGM is already better than the other algorithms by a large factor.

After 500 evaluations, EGM corroborates its distinct superiority over the other algorithms, followed by the RRS and GA. Intriguingly, GA with its exploitive crossover operation performs better than all other optimizers after 500 evaluations, except for EGM. LHS and MCS keep a significant error, lacking gradient-based optimization.

Summarizing, algorithms combining exploration and exploitation, i.e. EGM, GA and RRS, perform better than purely explorative or purely exploitive algorithms (LHS, MCS and DSM). For the ‘pure’ algorithms, LHS has the fastest decrease of cost function while DSM has the fastest convergence. EGM turns out to be the best combined algorithm by making a balance of exploration and exploitation from LHS and DSM, respectively. This superiority is already apparent after 100 evaluations.

We note that the conclusions have been drawn for a single analytical example for an optimization in a low-dimensional parameter space with few minima. From many randomly created analytical functions, we observe that EGM tends to outperform other

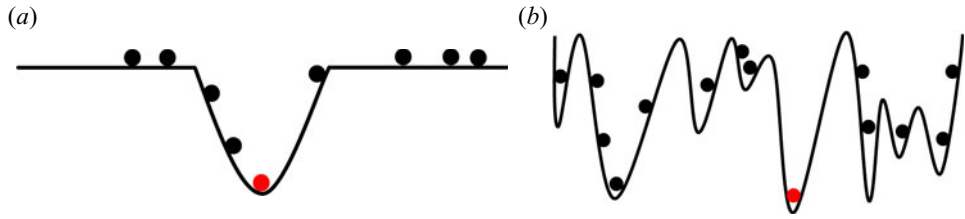


Figure 4. Sketch of the extreme landscapes with (a) one minimum and (b) many minima. The black dots denote the tested location. The red dots show the global minimum.

optimizers in the case of few smooth minima and for low-dimensional search spaces. Yet, in higher-dimensional search spaces, LHS becomes increasingly inefficient and MCS may turn out to perform better. The number of minima also has an impact on the performance. For a single minimum with parabolic growth, the DSM can be expected to outperform the other algorithms. In the case of many local shallow minima, the advantage of gradient-based approaches will become smaller and exploration will correspondingly increase in importance.

The control landscape may have from one single minimum to many minima. Exploration is often inefficient in the former scenario (figure 4a), while exploitation is likely to invest a large number of tests to descend into suboptimal minima or even be trapped in one (figure 4b). An unknown problem is more likely to have a landscape with characteristics between the two worst case scenarios for explorative and exploitive methods. The strict alternation proposed by EGM aims to minimize unnecessary evaluations in the worst case scenario to 50 % of the iterations and guarantee a gradient-based converge rate in case of an identified minimum.

The following rules of thumb can be formulated for the choice and design of the optimizers. First, the larger the number of local minima, the more effective exploration becomes over exploitation. Second, gradient search becomes more effective when the global minimum has a large domain of attraction for gradient-based descent in comparison with the search space. The volume ratio between domain of attraction and search space determines the likelihood that EGM finds the global minimum with given budget for evaluations. Third, large plateaus, e.g. from asymptotic behaviour, make gradient search inefficient. Fourth, long ‘valleys’ make gradient search inefficient, too. Fifth, space-filling exploration, like LHS or MCS, will become inefficient with increasing dimension. Ten dimensions might be considered as an upper threshold, as just the exploration of the corners of a ten-dimensional cube requires $2^{10} = 1024$ individuals. A good discretization, e.g. with 10 points per dimension is evidently impossible. Numerous other rules may be added for different control landscapes. Given the richness of possible landscapes, explorative methods come without practical and mathematically provable performance estimates.

4. Drag optimization of fluidic pinball with three actuators

As the first flow control example, EGM is applied to the two-dimensional fluidic pinball (Cornejo Maceda *et al.* 2019; Deng *et al.* 2020), the wake behind a cluster of three rotating cylinders. In § 4.1, the benchmark problem is described: minimize the net drag power with the cylinder rotations as input parameters. In § 4.2, EGM yields a surprising non-symmetric result, consistent with other fluidic pinball simulations (Cornejo Maceda *et al.* 2019) and experiments (Raibaudo *et al.* 2020). The learning process of DSM and LHS are investigated in §§ 4.3 and 4.4,

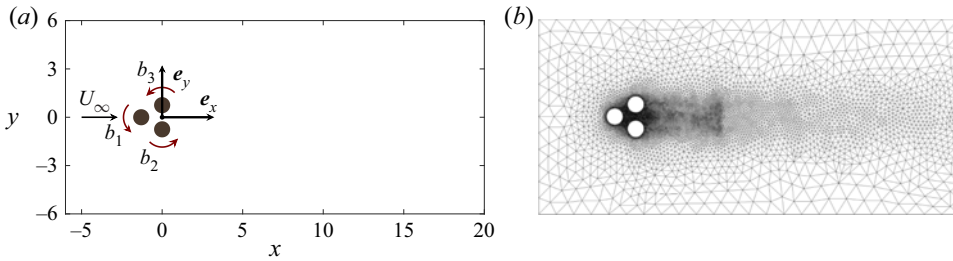


Figure 5. Fluidic pinball: (a) configuration and (b) grid.

4.1. Configuration

The fluidic pinball is a benchmark configuration for wake control which is geometrically simple yet rich in nonlinear dynamics behaviours. This two-dimensional configuration consists of a cluster of three equal, parallel and equidistantly spaced cylinders pointing in opposite to uniform flow. The wake can be controlled by the cylinder rotation. The fluidic pinball comprises most known wake stabilization mechanisms, like phasor control, circulation control, Coanda forcing, base bleed as well as high- and low-frequency forcing. In this study, we focus on steady open-loop forcing minimizing the drag power corrected by actuation energy.

The viscous incompressible two-dimensional flow has uniform oncoming flow with speed U_∞ and a fluid with constant density ρ and kinematic viscosity ν . The three equal circular cylinders have radius R and their centres form an equilateral triangle with side length $3R$ pointing upstream. Thus, the transverse dimension of the cluster reads $L = 5R$.

In figure 5(a), the flow is described in Cartesian coordinate system where the x -axis points in the direction of the flow, the z -axis is aligned with the cylinder axes and the y -axis is orthogonal to both. The origin $\mathbf{0}$ is placed in the centre of the rightmost top and bottom cylinders. Thus, the centres of the cylinders are described by

$$\left. \begin{aligned} x_1 = x_F = -3R \cos 30^\circ, & \quad y_1 = y_F = 0, \\ x_2 = x_B = 0, & \quad y_2 = y_B = -3R/2, \\ x_3 = x_T = 0, & \quad y_3 = y_T = +3R/2. \end{aligned} \right\} \quad (4.1)$$

Here, the subscripts ‘ F ’, ‘ B ’ and ‘ T ’ refer to the front, bottom and top cylinder. Alternatively, the subscripts ‘1’, ‘2’ and ‘3’ are used for these cylinders starting with the front cylinder and continuing in mathematically positive orientation.

The location is denoted by $\mathbf{x} = (x, y) = x \mathbf{e}_x + y \mathbf{e}_y$, where \mathbf{e}_x and \mathbf{e}_y are the unit vectors in the x - and y -directions. The flow velocity is represented by $\mathbf{u} = (u, v) = u \mathbf{e}_x + v \mathbf{e}_y$. The pressure and time symbols are p and t , respectively. In the following, all quantities are non-dimensionalized with cylinder diameter $D = 2R$, the velocity U_∞ and the fluid density ρ .

The corresponding Reynolds number reads $Re_D = U_\infty D / \nu = 100$. This Reynolds number corresponds to asymmetric periodic vortex shedding. Deng *et al.* (2020) have investigated the transition scenario for increasing Reynolds number. At $Re_1 \approx 18$, the steady flow becomes unstable in a Hopf bifurcation leading to periodic vortex shedding. At $Re_2 \approx 68$, both the steady Navier–Stokes solutions and the limit cycles bifurcate into two mirror-symmetric states. Chen *et al.* (2020) performed a careful parametric analysis of the gap width between the cylinders and associated this behaviour with the ‘deflected regime’, where base bleed through the rightmost cylinder are deflected upward or downward.

At $Re_3 \approx 104$ another Hopf bifurcation leads to quasi-periodic flow. After $Re_4 \approx 115$, a chaotic state emerges.

The flow properties can be changed by the rotation of cylinders. The corresponding actuation commands are denoted by

$$b_1 = U_F, \quad b_2 = U_B, \quad b_3 = U_T. \tag{4.2a-c}$$

Here, positive value denotes the anti-clockwise direction.

Following Cornejo Maceda *et al.* (2019), we aim to minimize of the averaged parasitic drag power \bar{J}_a penalizing the averaged actuation power \bar{J}_b . The resulting cost function reads

$$\bar{J} = \bar{J}_a + \bar{J}_b. \tag{4.3}$$

The first contribution $\bar{J}_a = c_D$ corresponds to drag coefficient

$$c_D = \frac{\bar{F}_D}{(1/2)\rho DU_\infty^2} \tag{4.4}$$

for the chosen non-dimensionalization. Here, \bar{F}_D denotes total averaged drag force on all cylinders per unit spanwise length. The second contribution arises from the necessary actuation torque to overcome the skin-friction resistance.

Following Deng *et al.* (2020), the flow is computed with direct numerical solution in the computational domain

$$\mathcal{D} = \{(x, y) : -6 \leq x \leq 20 \wedge |y| \leq 6 \wedge (x - x_i)^2 + (y - y_i)^2 \geq 1/4, i = 1, 2, 3\}. \tag{4.5}$$

We use an in-house implicit finite-element method solver ‘UN3’ which is of third-order accuracy in space and time. The unstructured grid in figure 5(b) contains 4225 triangles and 8633 vertices. An earlier grid convergence study identified this resolution sufficient for up to 2 per cent error in drag, lift and Strouhal number.

4.2. Optimized actuation

In the subsequent study, the actuation commands $b_1 = U_F$, $b_2 = U_B$ and $b_3 = U_T$ are bounded by 5, i.e. the search space reads

$$\Omega := \{[b_1, b_2, b_3]^T \in \mathcal{R}^3 : |b_i| \leq 5 \text{ for } i = 1, 2, 3\}. \tag{4.6}$$

Previous symmetric parametric studies have identified symmetric Coanda forcing $b_1 = 0$, $b_2 = -b_3$ around 2 as optimal for net drag reduction, both in low-Reynolds-number direct numerical simulations (Cornejo Maceda *et al.* 2019) and in high-Reynolds-number unsteady Reynolds-averaged Navier–Stokes simulations (Raibauda *et al.* 2020). The chosen bound of 5 adds a large safety factor to these values, i.e. the optimum can be expected to be in the chosen range. Steady bleed into the wake region is reported as another means for wake stabilization by suppressing the communication between the upper and lower shear layers. This study starts from the base-bleeding control in search of a different actuation from boat tailing.

LHS, DSM and EGM are applied to minimize the net drag power (4.3) with steady actuation in the three-dimensional domain (4.6). Following §§ 2.4 and 2.6, the initial simplex comprises four vertices: the individual controlled by base-bleeding actuation ($b_1 = 0, b_2 = -4.5, b_3 = 4.5$), the other three individuals are positively shifted by 0.1 for each actuation. The individuals and their corresponding costs are listed in table 2. All the

m	b_1	b_2	b_3	J
1	0	-4.5	4.5	11.8282
2	0.1	-4.5	4.5	11.8304
3	0	-4.4	4.5	11.6017
4	0	-4.5	4.6	12.0758

Table 2. Fluidic pinball: initial simplex ($m = 1, 2, 3, 4$) for the three-dimensional DSM optimization; b_i denotes the circumferential velocity of the cylinders and J corresponds to the net drag power (4.3).

individuals have a larger cost than the unforced benchmark $J = 1.8235$. The increase of the actuation amplitude from the strong base-bleeding forcing indicates higher cost. Thus, the initial condition seems to pose a challenge for optimization.

In this section, the optimization process of EGM is investigated. Figure 6(a) shows the best cost found with m simulations. EGM is quickly and practically converged after $m = 22$ evaluations and yields the near-optimal actuation at the 78th test

$$b_1^* = -0.0763, \quad b_2^* = 1.1301, \quad b_3^* = -1.1533, \quad J^* = 1.3. \quad (4.7a-d)$$

The cost function $J^* = 1.3$ reveals a net drag power saving of 29% with respect to the unforced value $J_u = 1.8235$. The large amount of suboptimal testing is indicative of a complex control landscape. The drag coefficient falls from 2.8824 for unforced flow to 1.3902 for the actuation (4.7a-d) within a few convective time units. This near-optimal actuation corresponds to 52% drag reduction. This 52% reduction of drag power requires 23% investment in actuation energy.

The best actuation mimics nearly symmetric Coanda forcing with a circumferential velocity of 1. This actuation deflects the flow towards the positive x -axis and effectively removes the dead-water region with reversal flow. The slight asymmetry of the actuation is not a bug but a feature of the optimal actuation after the pitchfork bifurcation at $Re_2 \approx 68$. This achieved performance and actuation is similar to the optimization feedback control achieved by machine learning control (Cornejo Maceda *et al.* 2019), comprising a slightly asymmetric Coanda actuation with small phasor control from the front cylinder. Also, the optimized experimental stabilization of the high-Reynolds-number regime leads to asymmetric steady actuation (Raibaudo *et al.* 2020). The asymmetric forcing may be linked to the fact that the unstable asymmetric steady Navier–Stokes solutions have a lower drag than the unstable symmetric solution.

Figure 6(b) shows the control landscape, i.e. two-dimensional proximity map of the three-dimensional actuation parameters. Neighbouring points in the proximity map correspond similar actuation vectors. The proximity map is computed with classical multi-dimensional scaling (Cox & Cox 2000). This map shows all performed simulations for figure 6(a) as solid red circles, when the evaluation improves the cost with respect to the iteration history and as open blue circles otherwise. Select new minima are highlighted with yellow circles: the first run $m = 1$ on the right side, the converged run $m = 78$ on the left and the intermediate run $m = 9$ when the explorative step jumps in new better territories. The colour bar represents interpolated values of the cost (4.3).

The meaning of the feature coordinates γ_1 and γ_2 will be revealed by the following analysis. Ten of the individuals of the control landscape are selected along the coordinates and marked with letters between A and J :

(A) unforced flow in the centre;

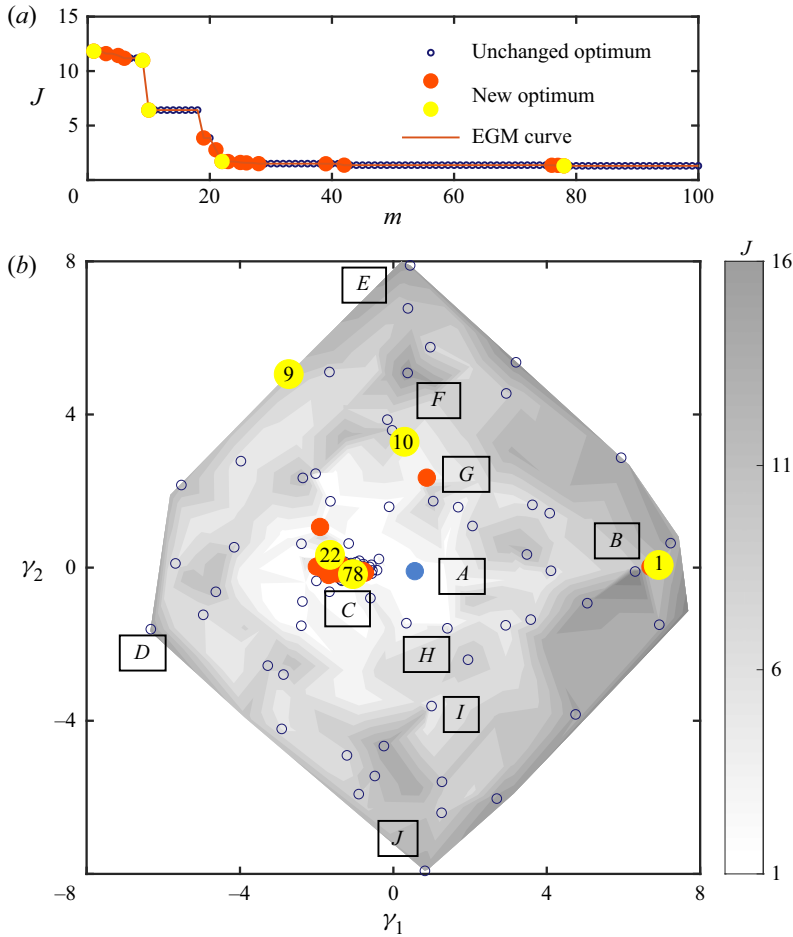


Figure 6. Optimization of the fluidic pinball actuation with EGM. The actuation parameters and cost are visualized as in figures 2 and 3. For enhanced interpretability, select new minima are displayed as solid yellow circles in the learning curve (a) and in the control landscape (b) the corresponding m index. Here, m counts the DNS for net drag power computation. The marked flows A–J are explained in the text.

- (B) base-bleeding flow $m = 1$ as the initial individual;
- (C) optimal actuation after $m = 78$ evaluations;
- (D,E,J) strong asymmetric actuations] at the boundary of the control landscape $m = 7, m = 24, m = 11$, showing a strong overall actuation;
- (F–I) random actuations along the coordinates γ_2 in the centre the control landscape corresponding to $m = 10, m = 19, m = 60, m = 51$, respectively.

The flows corresponding to actuations A–J in figure 6(b) are depicted in figure 7. The optimized actuation (C) yields a partially stabilized flow, like the machine learning feedback control by Cornejo Maceda *et al.* (2019). Actuation C corresponds to complete stabilization with strong Coanda forcing, located near $\gamma_2 \approx 0$ for small γ_1 . In contrast, flow B on the opposite side of the control landscapes represents strong base bleeding. Actuations J, G and H, located at the top and bottom of the control landscapes correspond to Magnus effects. Large positive (negative) feature coordinates γ_2 are associated with large positive (negative) total circulations and associated lift forces. Summarizing, the

Explorative gradient method

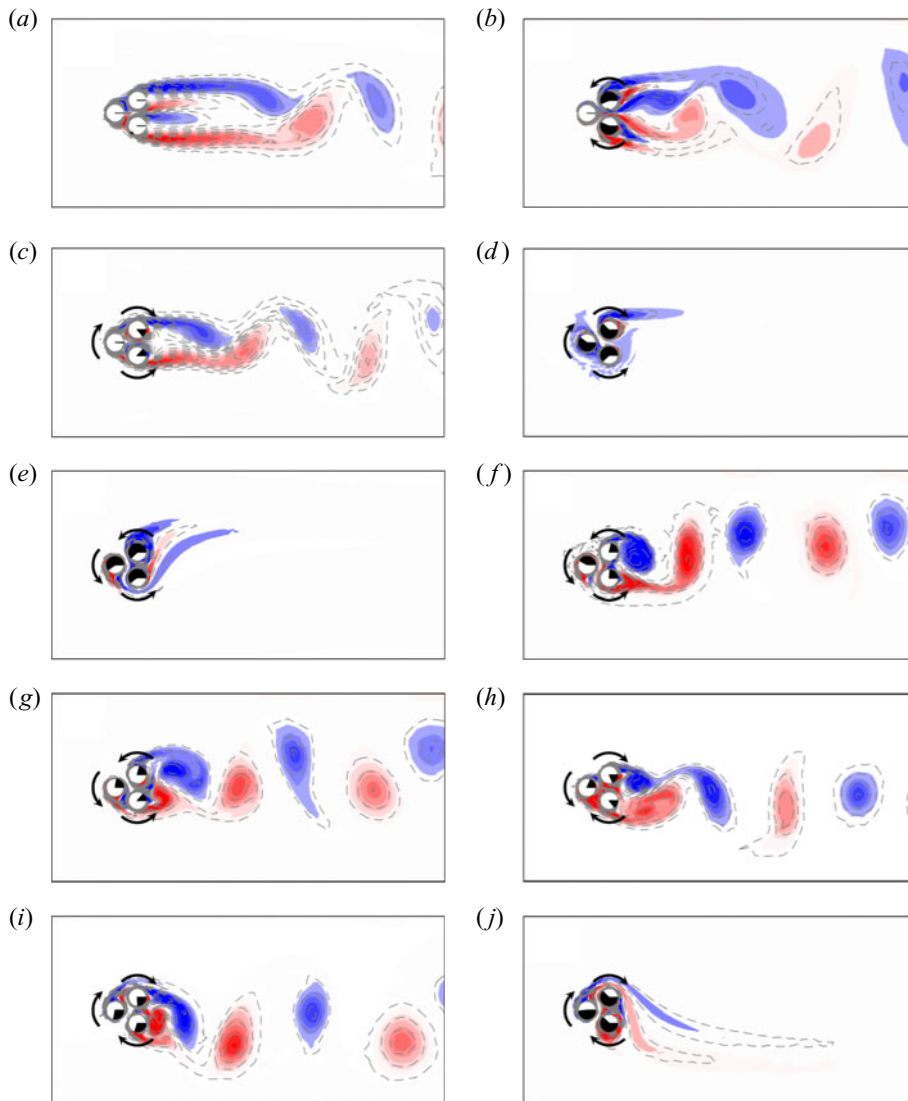


Figure 7. Fluidic pinball flows of different actuations of the control landscape (figure 6b). Panels correspond to actuations with letters A–J, respectively, and display the vorticity of the post-transient snapshot. Positive (negative) vorticity is colour coded in red (blue). The dashed lines correspond to iso-contour lines of vorticity. The orientation of the cylinder rotations is indicated by the arrows. The cylinder rotation is proportional to the angle of the black sector inside.

analysis of these points reveals that the feature coordinate γ_1 corresponds to the strength of the Coanda forcing and is hence related to the drag. In contrast, γ_2 is correlated with the total circulation of the cylinder rotations and thus with the lift. Ishar *et al.* (2019) arrives at a similar interpretation of the proximity map for differently actuated fluidic pinball simulations.

4.3. Downhill simplex method

Figure 8(a) shows the optimization process of DSM. After step-by-step descends in the former 23 simulations, the net drag cost decreases slowly during the following

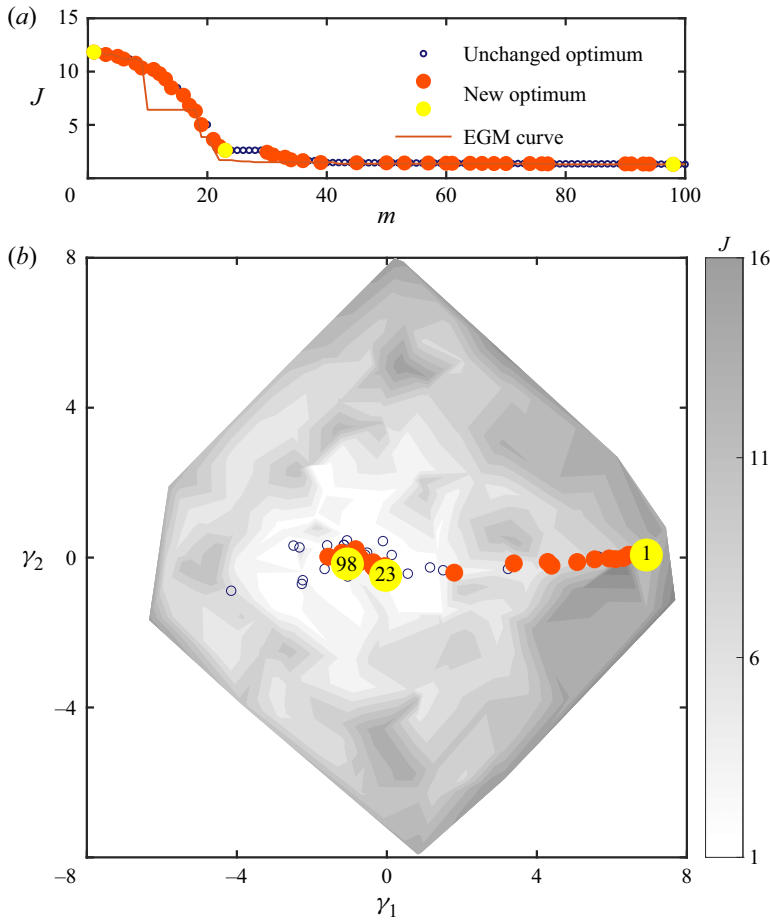


Figure 8. Same as figure 6, but with DSM.

77% computation. The optimal actuation after 100 simulations is $b_1 = -0.0721$, $b_2 = 1.0127$, $b_3 = -1.3250$ with cost $J = 1.3026$.

Figure 8(b) reveals the optimization from a broad slope to a tortuous valley after the 23th test. In face of the complex landscape on the way to global optimum (from $\boldsymbol{\gamma} = [6, 0]^T$ to $\boldsymbol{\gamma} = [-1, 0]^T$), DSM consumes relatively high computation resources. EGM is seen to outperform DSM at $m \geq 9$ because of an explorative step. For random initial conditions, DSM often performs better than EGM, because the exploration as insurance policy brings less returns for this comparatively simple control landscape.

4.4. Latin hypercube sampling

Figure 9(a) shows the performance of LHS. The algorithm is of low efficiency during most computations with only 2 new optima found. The jump at the 6th DNS does not bring significant improvement. The new optimum at 10th simulation reduces the cost by more than 85%, with the actuation $b_1 = -0.3649$, $b_2 = 1.4649$, $b_3 = -0.1458$ which is kept for the following 90% of the optimization period.

The global search is further illustrated by the uniformly tested points in figure 9(b). The algorithm starts near $\boldsymbol{\gamma} = [7, 0]^T$ and explores the feature space from the boundary

Explorative gradient method

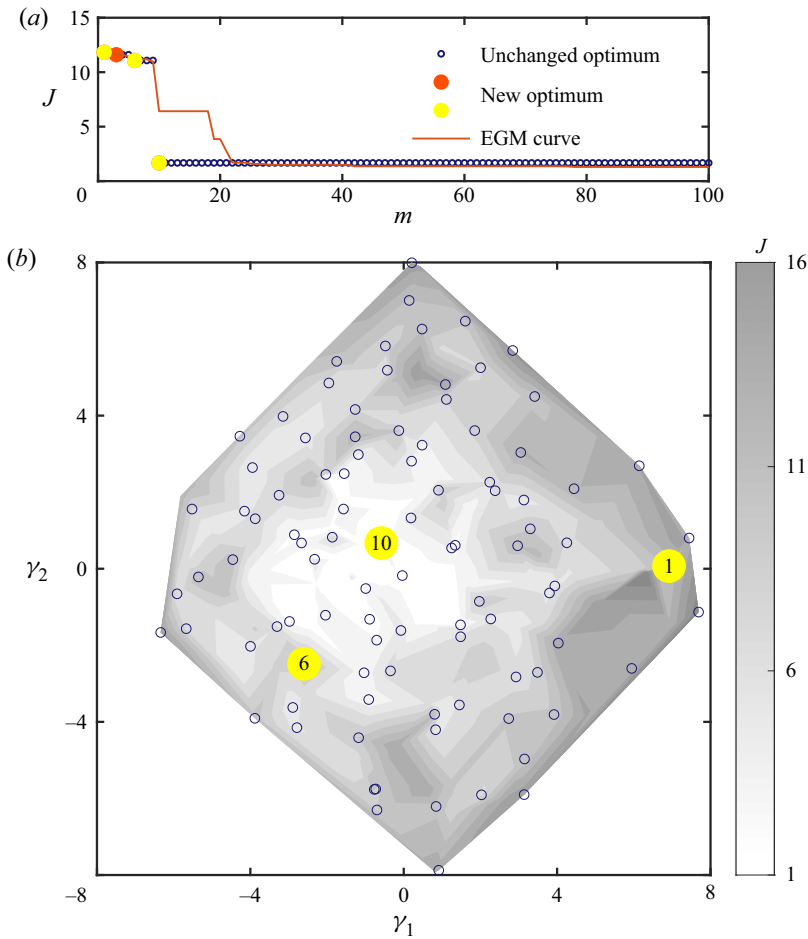


Figure 9. Same as figure 6, but with LHS.

to the central area. The second new minimum is explored early and finds an asymmetric actuation but with a cost close to the global minimum. LHS outperforms EGM at $m = 10$ before EGM leads at $m \geq 23$. Exploration is seen to have advantages at the beginning, but exploitation wins already in the midterm.

5. Drag optimization of an Ahmed body with 10 actuation parameters

The starting point of the computational fluid dynamics plant is an experimental study of a low-drag 35° Ahmed body (Li *et al.* 2018). The investigated Ahmed body configuration (§ 5.1) has the same physical dimensions. The effect of actuation is assessed with RANS simulations (§ 5.2). In § 5.3, a parametric drag study with a single streamwise-oriented actuator on the top edge is performed. In § 5.4, the streamwise blowing of all five actuator groups is optimized for drag reduction. EGM is contrasted to DSM and LHS. In § 5.5, the velocity and orientation of the five slot actuators are optimized with EGM, thus giving rise to a ten-dimensional search space. As expected drag reduction increases with the dimension of the search space, i.e. expanding actuation opportunities. The corresponding physical drag reduction mechanisms are investigated in § 5.6.

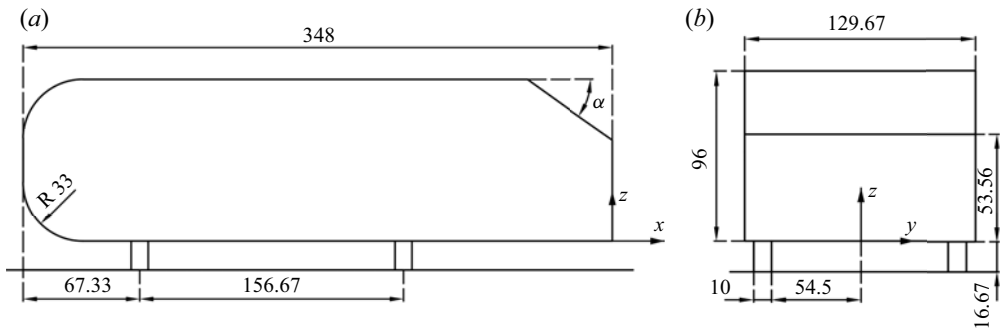


Figure 10. Dimensions of the investigated 1:3-scaled Ahmed body. (a) Side view. (b) Back view. The length unit is mm and the angle is specified in degrees.

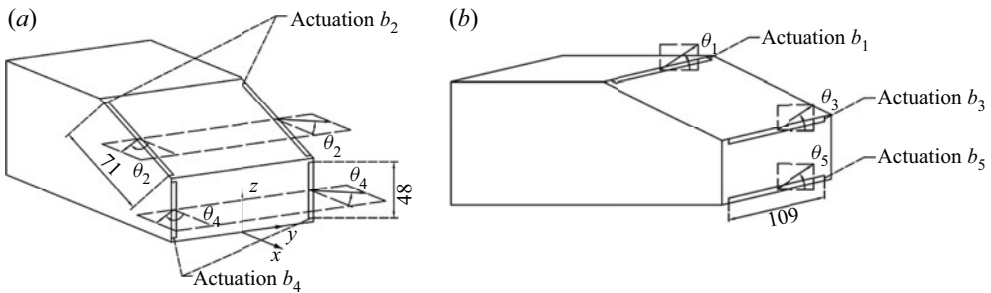


Figure 11. Deployment and blowing direction of actuators on the rear window and the vertical base. The angles $\theta_1, \theta_2, \theta_3, \theta_4$ and θ_5 are all defined to be positive when pointing outward (a) or upward (b).

5.1. Configuration

The point of departure is an experimentally investigated 1:3-scaled Ahmed body characterized by a slanted edge angle of $\alpha = 35^\circ$ with length L , width W and height H of 348 mm, 130 mm and 96 mm, respectively. The front edges are rounded with a radius of $0.344 H$. The model is placed on four cylindrical supports with a diameter equal to 10 mm and the ground clearance is $0.177 H$. The origin of the Cartesian coordinate system (x, y, z) , is located in the symmetry plane on the lower edge of the model's vertical base (see figure 10). Here, x, y and z denote the streamwise, spanwise and wall-normal coordinates, respectively. The velocity components in the x, y and z directions are denoted by u, v and w , respectively. The free-stream velocity is chosen to be $U_\infty = 30 \text{ m s}^{-1}$.

Five groups of steady blowing slot actuators (figure 11) are deployed on all edges of the rear window and the vertical base. All slot widths are 2 mm. The horizontal actuators at the top, middle and bottom sides have lengths of 109 mm. The upper and lower sidewise actuators on the upper and vertical rear window have a length of 71 mm and 48 mm, respectively. The actuation velocities U_1, \dots, U_5 are independent parameters; U_1 refers to the upper edge of the rear window, U_3 to the middle edge and U_5 to the lower edge of the vertical base; U_2 and U_4 correspond to the velocities at the right and left sides of the upper and lower windows, respectively.

Following the experiment by Zhang *et al.* (2018), all blowing angles can be varied as indicated in figure 11. This study aims at minimizing drag as represented by the drag coefficient, $J = c_D$, by varying the actuation control parameters. The actuation velocity amplitudes $U_i, i = 1, \dots, 5$ are capped by twice of the single optimum value

Explorative gradient method

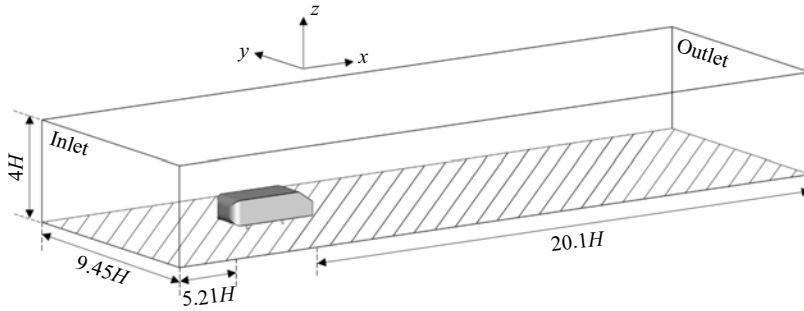


Figure 12. Computational domain of RANS and LES.

Mesh grid points	2.5 M	5 M	10 M
Drag coefficient	0.294	0.313	0.315

Table 3. Drag coefficient based on different mesh resolutions; ‘M’ denotes million.

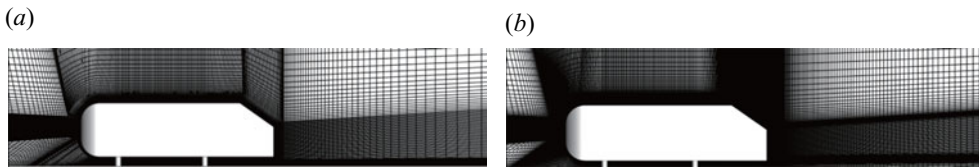


Figure 13. Side view of a part of the computational grids used for (a) RANS and (b) LES.

as discussed in § 5.3. The actuation angles θ_i , $i = 1, \dots, 5$ are fixed to 0° , i.e. streamwise direction, in a five-dimensional optimization. The actuation angles are later added into the input parameters in ten-dimensional optimization, with variable angles $\theta_1 \in [-35^\circ, 90^\circ]$, $\theta_2, \theta_3, \theta_4, \theta_5 \in [-90^\circ, 90^\circ]$.

5.2. RANS simulations

5.2.1. RANS simulation set-up

A numerical wind tunnel (figure 12) is constructed using the commercial grid generation software Ansys ICEM CFD. The rectangular computational domain is bounded by $X_1 \leq x \leq X_2$, $0 \leq z \leq H_T$, $|y| \leq W_T/2$. Here, $X_1 = -5.21H$, $X_2 = 20.17H$, $H_T = 4H$ and $W_T = 9.45H$, larger than the smallest domain suggested by Serre *et al.* (2013). Coarse, medium and fine meshes using an unstructured hexahedral computational grid are employed in order to evaluate the performance of the RANS method for the current problem with different mesh resolutions. The statistics in table 3 show that using a finer mesh can be expected to lead to negligible improvement in the accuracy of the drag coefficient. In addition, the medium mesh predicts a flow similar to time-averaged flows of LES (see Appendix C) and to experimental statistics (see Appendix D) Hence, the more economical medium mesh (figure 13) is used. This mesh consists of 5 million elements and features dimensionless wall grid sizes of $\Delta x^+ = 20$, $\Delta y^+ = 3$, $\Delta z^+ = 30$. In addition to resolving the boundary layer, the shear layers and the near-wake region, the mesh near the actuation slots is also refined.

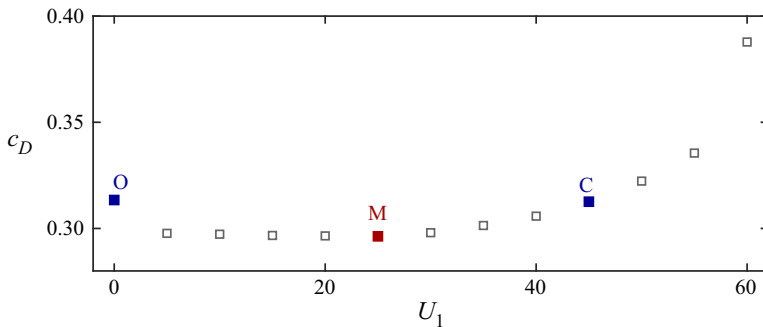


Figure 14. Drag coefficient as a function of the blowing velocity U_1 of the streamwise-oriented top actuator. Here, ‘O’ marks the drag without forcing, ‘M’ the best actuation and ‘C’ the smallest actuation with worse drag than for unforced flow.

Numerical simulations are performed using the RANS equations in conjunction with a two-equation realizable $k - \epsilon$ turbulence model employing the commercial solver Fluent. The spatial discretization is based on a second-order upwind scheme in the form of semi-implicit method for pressure linked equations scheme based on a pressure–velocity coupling method. RANS simulation has been frequently and successfully used to assess actuation effects from steady blowing (Viken *et al.* 2003; Dejoan, Jang & Leschziner 2005; Ben-Hamou, Arad & Seifert 2007; Muralidharan, Muddada & Patnaik 2013). The flow for the chosen slanted angle 35° is well predicted by RANS (Guilmineau *et al.* 2018; Viswanathan 2021). In contrast, the 25° Ahmed body with a separation bubble on the slant is difficult to capture. We deem RANS simulations to provide reasonable qualitative and approximately quantitative indications for actuator optimization of the chosen low-drag Ahmed body.

The RANS prediction of the uncontrolled and controlled cases is validated by the experiment in appendix with LES (Appendix C) and experiments (Appendix D). Partially averaged Navier–Stokes simulations (Han, Krajnović & Basara 2013) and LES (Brunn & Nitsche 2006; Krajnović 2009) are trusted higher-fidelity simulations for drag reduction with active flow control but are computationally orders of magnitudes more demanding.

5.3. The optimization problem based on streamwise blowing at the top edge

The formulation and constraints of the optimization problem are motivated by the drag reduction results from the top actuator blowing in the streamwise direction. Figure 14 shows the drag coefficient dependency on the streamwise blowing velocity, all other actuators being off. The blowing velocity varies in increments of 5 m s^{-1} from 0 to 60 m s^{-1} , i.e. reaches twice the oncoming velocity.

The drag coefficient is quickly reduced by modest blowing, has a shallow minimum near the actuation velocity $U_{b1} = 25 \text{ m s}^{-1}$ before quickly increasing with more intense blowing. This optimal value corresponds to $5/6$ of the oncoming velocity. The best drag reduction is 5% with respect to the unforced flow $C_d = 0.3134$. Near $U_1 = 45 \text{ m s}^{-1}$, the drag rapidly rises beyond the unforced value. Active separation control relies on exploiting instabilities that are inherent in the flow, generally requiring relatively small amplitude excitation (Seifert, Greenblatt & Wygnanski 2004).

This behaviour motivates the choice of actuation parameters. The first five actuation parameters are the normalized jet velocities $b_i = U_i/U_{b1}$, $i = 1, \dots, 5$ introduced in

m	b_1	b_2	b_3	b_4	b_5	J
1	1.8	1.8	1.8	1.8	1.8	0.4153
2	1.6	1.8	1.8	1.8	1.8	0.4048
3	1.8	1.6	1.8	1.8	1.8	0.4109
4	1.8	1.8	1.6	1.8	1.8	0.3996
5	1.8	1.8	1.8	1.6	1.8	0.4075
6	1.8	1.8	1.8	1.8	1.6	0.4040

Table 4. Initial simplex ($m = 1, \dots, 6$) for the five-dimensional DSM optimization; b_i are the normalized actuation velocities and J corresponds to the drag coefficient.

§ 5.1. Thus, $b_1 = 1$ corresponds to minimal drag with a single streamwise-oriented top actuator. All b_i are capped by 2: $b_i \in [0, 2]$, $i = 1, \dots, 5$. At $b_1 = 1.8$, point ‘C’ in figure 14, actuation yields already drag increase. The first vertex of the amoeba of DSM is put at $b_1 = b_2 = b_3 = b_4 = b_5 = 1.8$. From figure 14, we expect a drag minimum at lower values, hence the next five vertices test the value 1.6, e.g. $[b_1, b_2, \dots, b_5]^T = [1.8 - 0.2\delta_{1,m-1}, 1.8 - 0.2\delta_{2,m-1}, \dots, 1.8 - 0.2\delta_{5,m-1}]^T$ for $m = 2, \dots, 6$. DSM may be expected to move to the outer border of the actuation domain, if maximum drag reduction lies outside the domain, thus indicating too restrictive constraints. An example is drag reduction of wall turbulence (Fernex *et al.* 2020).

We refrain from starting already with a much larger actuation domain, as the exploration with LHS and the proposed EGM will consistently test too many large velocities. An increase of the upper velocity bound by a factor 2, for instance, implies that only 2^{-5} or approximately 3% of uniformly distributed sampling points are in the original domain and 97% of the samples are outside.

The next five parameters characterize the deflection of the actuator velocity with respect to the streamwise direction (see § 5.1), $b_{i+5} = \theta_i/(\pi/2)$, $i = 1 \dots, 5$, and are normalized with 90° . Now all b_i , $i = 1, \dots, 10$ span an interval of width 2, except for the more limited deflection b_6 of the top actuator. Summarizing, the domain for the most general actuation reads

$$\Omega := \left\{ \mathbf{b} \in \mathcal{R}^{10}: \begin{array}{ll} b_i \in [0, 2] & \text{for } i = 1, \dots, 5 \\ b_i \in [-35/90, 1] & \text{for } i = 6 \\ b_i \in [-1, 1] & \text{for } i = 7, \dots, 10 \end{array} \right\}. \quad (5.2)$$

The choice of \mathbf{b} as symbol recalls the control B -matrix in control theory and is consistent with many earlier publications of the authors, e.g. the review article by Brunton & Noack (2015).

5.4. Optimization of the streamwise trailing edge actuation

The drag of the Ahmed body is optimized with streamwise blowing from the five slot actuators. We apply DSM, LHS and EGM of §§ 5.4.1, 5.4.2 and 5.4.3, respectively.

5.4.1. DSM

Following § 5.3, DSM is centred around $b_i = 1.8$, $i = 1, \dots, 5$ as first vertex and explores a lower actuation $b_{m-1} = 1.6$ in all directions for vertices $m = 2, \dots, 6$. Table 4 shows the values of the individuals and corresponding cost. All vertices have a larger drag than for the unforced benchmark $C_d = 0.3134$. And all vertices with $b_i = 1.6$ are associated

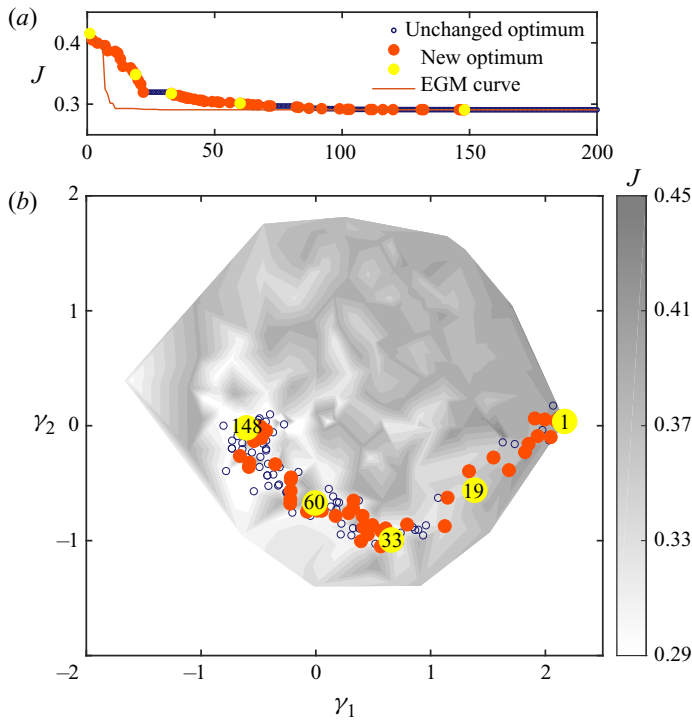


Figure 15. Optimization of five streamwise-oriented jet actuator groups with DSM. Panel (a) displays the best achieved drag reduction in terms of the number of evaluations (RANS simulations). Panel (b) shows the proximity map of all evaluated actuations. The contour plot corresponds to the interpolated cost function (drag coefficient) from all RANS simulations of this section. As in § 3, solid red circles mark newly find optima while open blue circles mark unsuccessful tests of cost functions. For better interpretability, select newly found optima are highlighted with by a yellow solid circle as in § 4. In the control landscape, these circles are marked with the index m .

with a smaller drag indicating a downhill slide to small actuation values consistent with the expectations from § 5.3.

Figure 15(a) shows the evolution of DSM with 200 RANS simulations. As in § 3, solid red circles mark newly found optima while open blue circles record the best actuation so far. The drag quickly descends after staying shortly on a plateau at $m \approx 20$. From there on, the descent becomes gradual. The optimal drag $J = 0.2908$ is reached with the 148th RANS simulation and corresponds to 7% drag reduction. The optimal actuation reads $b_1 = 0.7264$, $b_2 = 0.5508$, $b_3 = 0.1533$, $b_4 = 0.6746$, $b_5 = 0.7716$. While the middle horizontal jet has small amplitude, the other actuation velocities on the four edges of the Ahmed body are 55% to 77% of the optimal value achieved with single actuator.

Figure 15(b) illustrates the downhill search in a control landscape $J(\boldsymbol{\gamma})$ described in § 2. Here, $\boldsymbol{\gamma} = [\gamma_1, \gamma_2]^T$ feature vectors defining a proximity map of the five-dimensional actuation parameters $\boldsymbol{b} = [b_1, \dots, b_5]^T$. This landscape indicates a complex topology of the five-dimensional actuation space by many local maxima and minima in the feature plane. This complexity may explain why most simplex steps did not yield a better cost. The feature coordinates γ_1, γ_2 arise from a kinematic optimization process and have no inherent meaning. The simplex algorithm is seen to crawl from right $\boldsymbol{\gamma} \approx [2, 0]^T$ to the assumed global minimum at $\boldsymbol{\gamma} \approx [-0.6, 0]^T$ through an elongated curved valley. The simulation results for $m = 1, 19, 33, 60$ and 148 are marked with yellow solid circles.

Explorative gradient method

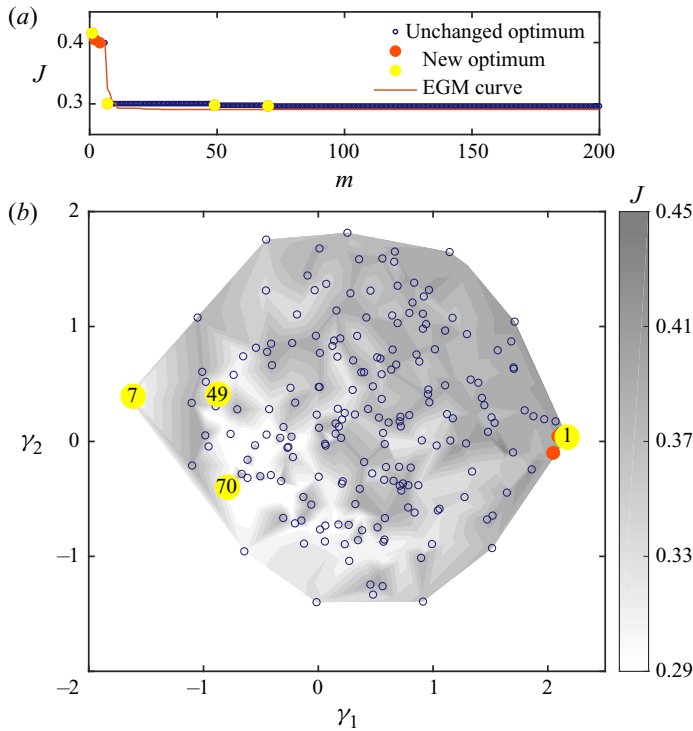


Figure 16. Same as figure 15, but with LHS.

Note that the construction of this proximity map includes also not displayed actuation data from LHS and EGM so that the control landscapes remain identical for all discussed five-dimensional actuations.

5.4.2. LHS

Figure 16(a) shows the slow learning process associated with LHS starting with the simplex reference point $b_1 = \dots = b_5 = 1.8$. Apparently, the optimization is ineffective. Only 3 new optima are successively obtained in 200 RANS simulations. The remaining simulations yield worse drags than the best discovered before. At the 70th RANS simulation, the best drag coefficient of $C_d = 0.2928$, with $b_1 = 0.0994$, $b_2 = 0.9587$, $b_3 = 0.1276$, $b_4 = 0.0289$ and $b_5 = 1.0393$ corresponding to 5% reduction like the one-dimensional top actuator $b_1 = 1, b_2 = b_3 = b_4 = b_5 = 0$. Intriguingly, only the upper side and bottom actuators have b_i amplitudes near unity while remaining parameters are less than 13% of the one-dimensional optimum. These results show that near-optimal drag reductions can be achieved with quite different actuations. Moreover, individual actuation effects are far from additive. Otherwise, the almost complimentary LHS optimum for actuators 2–5 and the one-dimensional optimum of § 5.3 should yield 10% reduction with $b_1 \approx 1, b_2 \approx 1, b_3 \approx 0.13, b_4 \approx 1$ and $b_5 \approx 1$.

Figure 16(b) shows the LHS in the control landscape. In the first iteration, LHS jumps to the opposite site of domain and finds better drag. The next successive two improvements are in a good terrain but the optimum at $m = 70$ is still far from the assuming global minimum at $\boldsymbol{\gamma} \approx [-0.6, 0]^T$ (see figure 15). The exploratory steps uniformly cover the whole range of feature vectors.

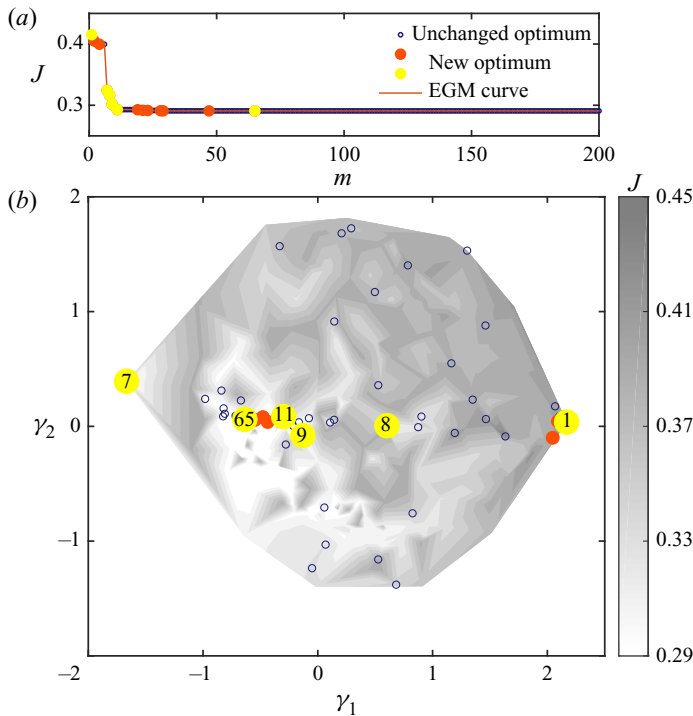


Figure 17. Same as figure 15, but with EGM.

5.4.3. EGM

From figure 17, EGM is seen to converge much faster than DSM. The best actuation is found at the 65th RANS simulation yielding the same drag coefficient $C_d = 0.2908$ as DSM with only slightly different actuation parameters $b_1 = 0.6647$, $b_2 = 0.4929$, $b_3 = 0.1794$, $b_4 = 0.7467$ and $b_5 = 0.7101$.

The fast convergence of EGM is initially surprising since up to 50% of the steps are for explorative purposes, i.e. identify distant minima. However, the control landscape in figure 16 reveals how the explorative LHS steps help the algorithm to prevent the long and painful march through the long and curved valley. At $m = 7$, an explorative step leads to the opposite side of control landscape with a better cost value. Then, the subsequent iterations quickly lead to the near-global minimum at $m = 11$. The proposed new algorithm operates like a visionary mountain climber, who performs not only local uphill steps but sends drones to the remotest location to find better mountains and terrains.

5.5. Optimization of the directed trailing edge actuation

In this section, the actuation space is enlarged by the jet directions of all slot actuators. The jets may now be directed inwards or outwards as discussed in § 5.1. The actuation optimization for drag reduction is performed with EGM.

We employ EGM as best performing method of § 5.4 for the ten-dimensional actuation optimization problem. The search is accelerated by starting with a simplex centred around the optimal actuation of the five-dimensional problem. The first vertex of table 5 contains this optimal solution. Here, the cost is $4L$ lower than the value of the previous section as the RANS integration for the first flow is re-computed and not yet fully converged. The next

m	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	b_9	b_{10}	J
1	0.6647	0.4929	0.1794	0.7467	0.7101	0	0	0	0	0	0.2895
2	0.6647	0	0	0	0	1/2	0	0	0	0	0.3268
3	0	0.4929	0	0	0	0	1/2	0	0	0	0.3226
4	0	0	0.1794	0	0	0	0	1/2	0	0	0.3168
5	0	0	0	0.7467	0	0	0	0	1/2	0	0.3476
6	0	0	0	0	0.7101	0	0	0	0	1/2	0.3060
7	0.6647	0	0	0	0	-35/90	0	0	0	0	0.3091
8	0	0.4929	0	0	0	0	-1/2	0	0	0	0.3085
9	0	0	0.1794	0	0	0	0	-1/2	0	0	0.3187
10	0	0	0	0.7467	0	0	0	0	-1/2	0	0.3001
11	0	0	0	0	0.7101	0	0	0	0	-1/2	0.3354

Table 5. Initial individuals in the optimization of the directed trailing edge actuation; b_i , $i = 1, 2, 3, 4, 5$ represents the actuation amplitudes U_i of the i th actuator; b_i , $i = 6, 7, 8, 9, 10$ denotes the actuation angle θ_i of the $(i - 5)$ th actuator; J is the drag coefficient.

five vertices represent isolated actuations at the optimal value but directed 45° outwards for the side edges and upwards for the middle horizontal actuator. The corresponding drag values are larger. The next five vertices deflect the jets in opposite direction by 45° or the maximum 35° of the top actuator, leading to smaller drag than the previous deflection. The drag of middle horizontal actuator remains close to the unforced benchmark because the jet velocity is small.

Figure 18(a) illustrates the convergence of EGM. After 289 RANS simulations, a drag coefficient of 0.2586 is achieved corresponding to a 17% drag reduction. The optimal actuation values read $b_1 = 0.8611$, $b_2 = 0.9856$, $b_3 = 0.0726$, $b_4 = 1.0089$, $b_5 = 0.8981$, $b_6 = -0.3000$ corresponding to $\theta_1 = -27^\circ$, $b_7 = -0.4666$ ($\theta_2 = -42^\circ$), $b_8 = 0.7444$ ($\theta_3 = 67^\circ$), $b_9 = -0.4888$ ($\theta_4 = -44^\circ$) and $b_{10} = 0.2444$ ($\theta_5 = -22^\circ$). All outer actuators have velocity amplitudes near unity and are directed inwards, i.e. emulate Coanda blowing. The third middle actuator blows upward with low amplitude. The strong inward blowing seems to be related to the additional 10% drag reduction as compared with the 7% of streamwise actuation.

Figure 18(b) shows the search process in a proximity map. It should be noted that this control landscape is based on data in a ten-dimensional actuation space and is hence different from the five-dimensional space in § 5.4. The algorithm quickly descends in the valley while many exploration steps probe suboptimal terrain. One reason for this quick landing in good terrain is the chosen initial simplex around the optimized actuation in the five-dimensional subspace.

The topology of the control landscape of figure 18 is investigated with discrete steepest descent lines connecting neighbouring data points in figure 19. For each investigated actuation vector, the nearest five neighbours are considered. If all neighbours have higher drag, the vector is considered as a local minimum and marked by a red point. Otherwise, a grey dashed arrow is plotted to the best of these neighbours. This steepest descent is continued until a local minimum is reached. The corresponding path is called the (discrete) steepest descent lines. Line segments shared by at least 10 of these paths may be considered as important valleys towards the minimum and are highlighted as black solid arrows. The visiting times of each individual are marked by the colour bar. The global minimum of all data points is visited most – 54 steepest descent lines end here. The resulting pathways of ‘mountain trails’ to ‘expressways’ may give an indication of the directions to be expected

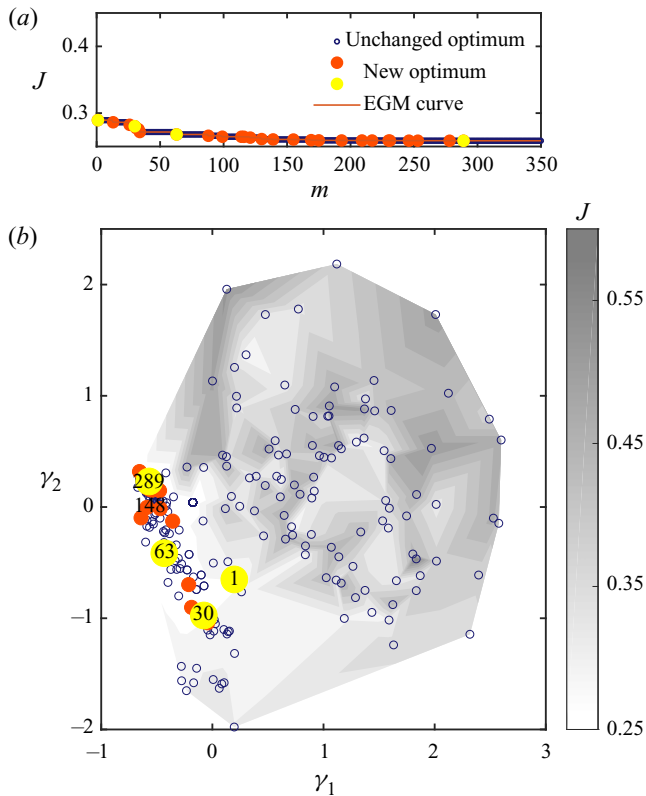


Figure 18. Same as figure 17, but for the ten-dimensional optimization of the orientable trailing edge actuation with EGM.

from local search algorithms. Moreover, crossing steepest descent lines indicate that the two-dimensional proximity maps oversimplifies a higher-dimensional landscape structure. Intriguingly, the steepest descent lines become more aligned to each other in the valleys, leading to a global data minimum, i.e. the most relevant regions for optimization.

5.6. Discussion of streamwise and directed jet actuations

In this section, the unforced and three actuated Ahmed body wakes are investigated. For brevity, we refer to flows with no, one-dimensional, five-dimensional and ten-dimensional actuation spaces as cases $0D$, $1D$, $5D$ and $10D$, respectively. The investigation includes the drag (§ 5.6.1), the surface pressure (§ 5.6.2), the near wake (§ 5.6.3) and the vortex formation (§ 5.6.4).

5.6.1. Drag reduction

First, the achieved drag reductions are discussed (see table 6). Evidently, more degrees of freedom for the actuators is associated with more opportunities for drag reduction. The drag reduces by 5% to 7% to 17% as the dimensions of the actuation parameters increase from 1 to 5 to 10, respectively. Intriguingly, the increase of drag reduction from the optimized top actuator to the best 5 streamwise actuators is only 2%. For the square-back Ahmed body, Barros (2015) experimentally observed that the individual drag reductions

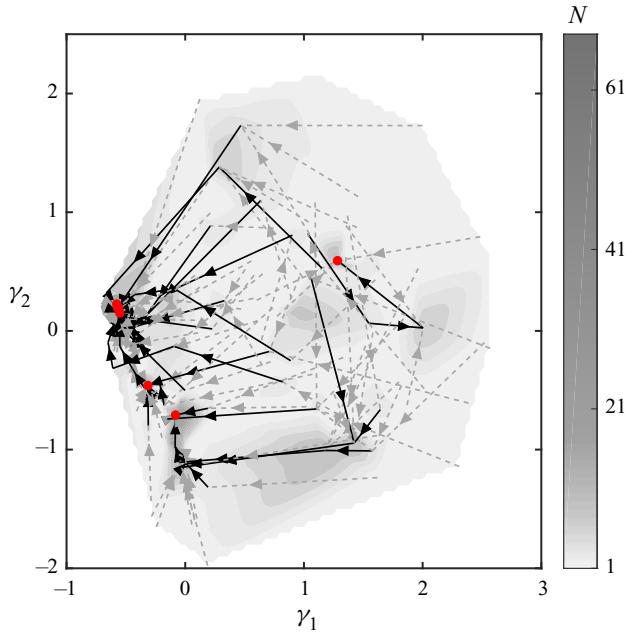


Figure 19. The steepest decent lines of the control landscape depicted in figure 18. For details see text.

Case	Drag (reduction)	Actuation parameters				
		Top	Upper side	Middle	Lower side	Bottom
(0D)	0.3134 (0%)	—	—	—	—	—
(1D)	0.2977 (5%)	$b_1 = 1$	$b_2 = 0$	$b_3 = 0$	$b_4 = 0$	$b_5 = 0$
(5D)	0.2908 (7%)	$b_1 = 0.6647$	$b_2 = 0.4929$	$b_3 = 0.1794$	$b_4 = 0.7467$	$b_5 = 0.7101$
(10D)	0.2586 (17%)	$b_1 = 0.8611$ $\theta_1 = -27^\circ$	$b_2 = 0.9856$ $\theta_2 = -42^\circ$	$b_3 = 0.0726$ $\theta_3 = 67^\circ$	$b_4 = 1.0089$ $\theta_3 = -44^\circ$	$b_5 = 0.8981$ $\theta_3 = -22^\circ$

Table 6. Investigated optimized actuations of the Ahmed body configuration in comparison with the unforced benchmark. The table shows the achieved drag reduction and corresponding actuation parameters for (0D) the unforced benchmark, and for the optimized (1D) top streamwise actuator, (5D) all streamwise actuators, (10D) all deflected actuators.

from the streamwise blowing actuators on the four trailing edges roughly add up to the total drag reduction of 10% with all pulsed jets on. This additivity of actuation effects is not corroborated for the slanted low-drag Ahmed body. Intriguingly, the inward deflection of the jet-slot actuators substantially decreases drag by additional 10%. This additional drag reduction of 10% has also been observed for the square-back Ahmed body when the horizontal jets were deflected inward with Coanda surfaces on all four edges (Barros *et al.* 2016). Improved drag reduction with inward as opposed to tangential blowing was also observed for the 35° high-drag Ahmed body (Zhang *et al.* 2018) and the square-back version (Schmidt *et al.* 2015).

Table 6 summarizes the discussed flows, associated drag reduction and actuation parameters. Cases 5D and 10D show strong peripheral blowing comparable to the optimized velocity of top actuator (case 1D). The magnitude of the middle actuation is

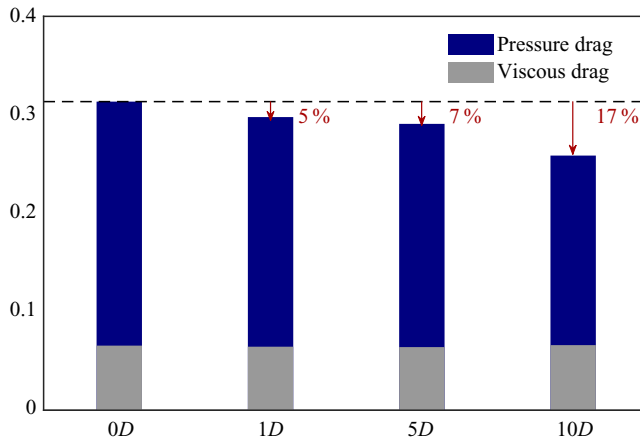


Figure 20. Pressure and viscous drag of the unforced flow (0D) and optimized actuations (1D, 5D, 10D). The total drag reduction by actuation is indicated by the red arrows and the associated percentages.

comparatively small. For optimal drag reduction (case 10D), all peripheral actuators are directed inward. The top and bottom jets have inclinations of 27° and 22° , respectively, while side jets feature stronger inward angles of 42° and 44° , respectively. Intriguingly, all optimal amplitudes of the peripheral jets become larger for inward direction (case 10D) as compared with streamwise blowing (case 5D).

For engineering purposes, the drag power needs to be corrected by the actuation power yielding the net drag power saving. The actuation power may be conservatively estimated by the energy flux through all jet actuators: $\sum_{i=1}^5 \int dA_i \rho U_i^3 / 2$.

Here, the actuation jet fluid is assumed to be accelerated from 0 to the actuation jet velocity U_i and then deflected after the outlet, e.g. via a Coanda surface. In this case, the actuation energy of cases 1D, 5D and 10D would correspond to 3.2%, 3.0% and 7.9% of the parasitic drag power, respectively. This expenditure is significantly less than the saved drag power. The ratio from saved drag power to actuation energy is comparable to a truck model where steady Coanda blowing with 7% energy expenditure yields a 25% drag reduction (Pfeiffer & King 2014). This estimate should not be taken too literally as actuation energy strongly depends on the realization of the actuator. It would be less, more precisely $\sum_{i=1}^5 \int dA_i \rho \cos(\theta_i) U_i^3 / 2$, when the actuation jet fluid leaves the Ahmed body through a slot directed with the jet velocity and can be expected much less when this fluid is taken from the oncoming flow, e.g. from the front of the Ahmed body.

5.6.2. Pressure drag

The viscous drag contributes only 20.9% to the total drag for the unforced benchmark (0D) and is hardly affected by actuation (1D, 5D, 10D) as illustrated in figure 20. In other words, effectively all drag reduction is associated with the pressure drag.

The drag reduction can be related to the C_p distribution of the rearward windows in figure 21. The 5% drag reduction in subfigure (b) for case 1D is associated with a pressure increase of the vertical surface. The additional 2% drag decrease for case 5D in (c) is accompanied by an increase over vertical and slanted surface. The aerodynamic boat tailing of case 10D with 17% drag reduction alleviates significantly the pressures on both surfaces.

Explorative gradient method

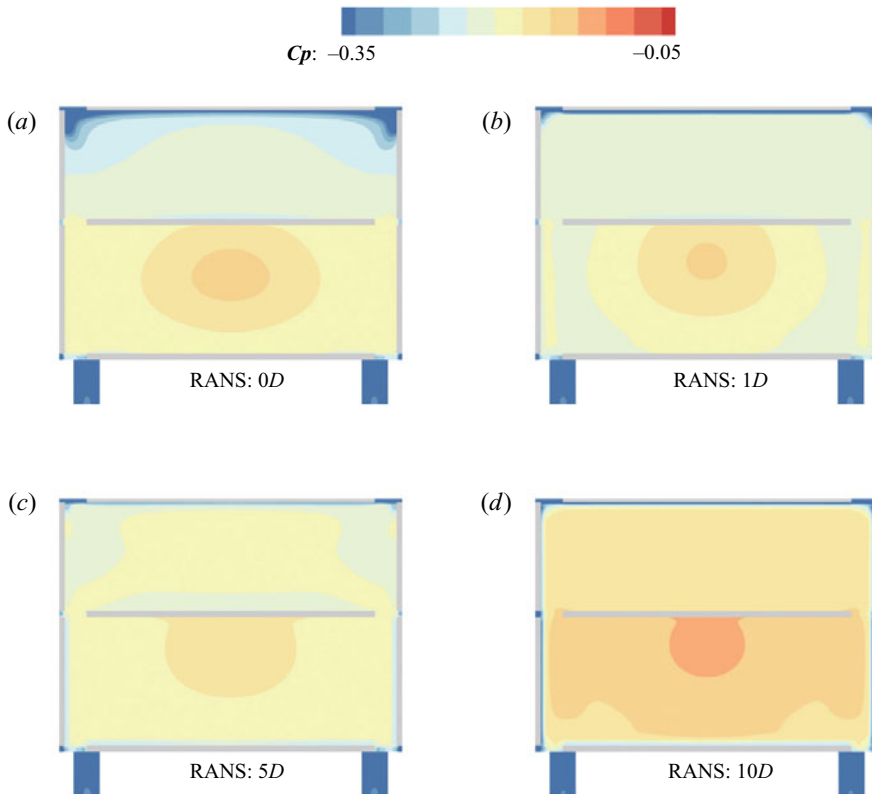


Figure 21. Pressure coefficient on the slant and vertical base predicted for the Ahmed body by RANS. (a) Without control and under (b) 1D, (c) 5D and (d) 10D control respectively.

5.6.3. Near-wake flow field

In the sequel, the mechanism behind the pressure increase on the rearward window is inferred from the flow field. Figure 22 shows the streamwise velocity component and streamlines of the transversal velocity in the same plane for the unforced and actuated cases. First, the flow topology is investigated. All flow visualizations show the streamlines of the in-plane velocity field by grey lines. Evidently, the drag reduction from actuation is associated with an elongation of the recirculation region ($(x, z)_{DW} = \arg \min\{u(x, 0, z) \leq 0\}$) marked by the solid circle. For tangential blowing (1D and 5D) the increase of drag reduction from 5 to 7% is associated with an increase of the recirculation bubble. Such correlation between length of the recirculation bubble and drag reduction has also been reported in actuated cylinder wakes (Gerhard *et al.* 2003; Thiria *et al.* 2006). In the limit of increasing extend, the wake may approximate the Kirchhoff solution with vanishing velocity in the wake. For the Kirchhoff solution, the inner pressure equals the outer pressure which might be considered as the upper limit of feasible drag reduction with small actuation energy.

The increased drag reduction by elongated recirculation region may be explained by centripetal forces. When the fluid particle follows the upper downward curved streamline it experiences a centripetal force outward which must be balanced by an equal but opposite pressure gradient. In other words, the near-wake region must have a lower pressure leading to increased drag. The more the wake is elongated the smaller the pressure gradient. Hence, the near wake shows a larger pressure and the bluff body experiences lower drag. This is

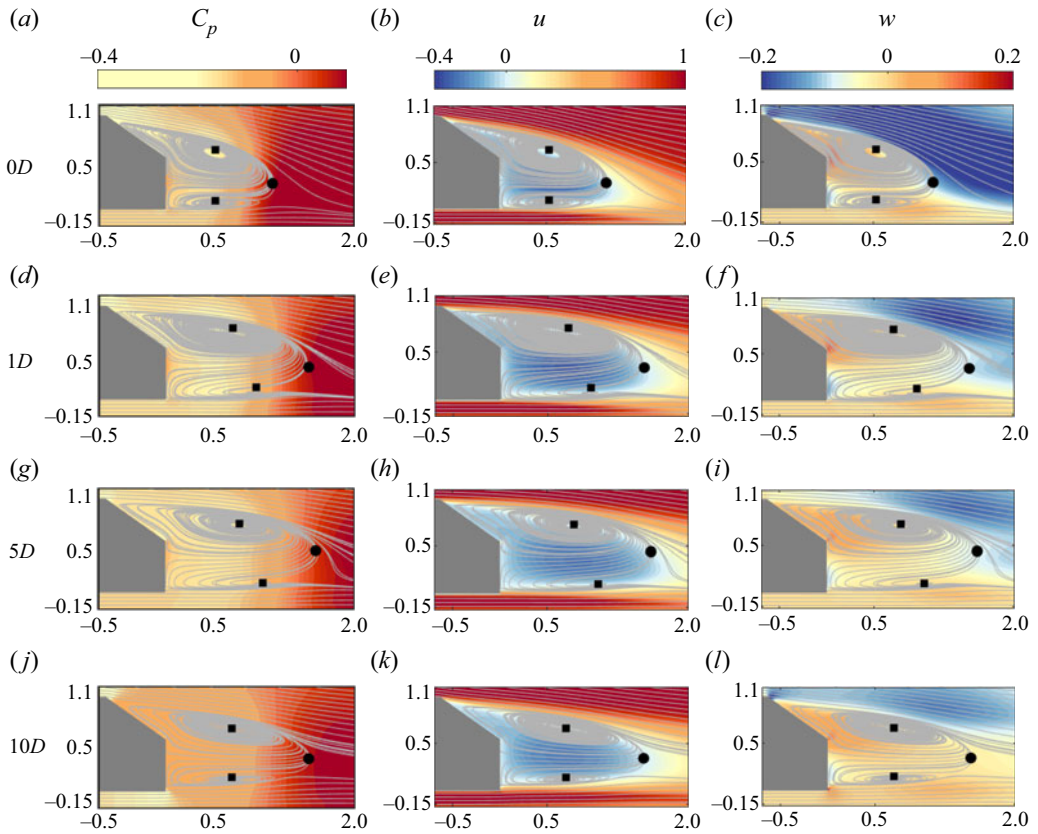


Figure 22. RANS flow visualization of the unforced Ahmed body benchmark and optimized actuations in the symmetry plane $y = 0$. The panels colour code the pressure coefficient (a, d, g, j) as well as streamwise (b, e, h, k) and vertical velocity components (c, f, i, l). From ($a-c$) to (j, k, l), the figure depicts the flows ($a-c$) without control and under ($d-f$) 1D, ($g-i$) 5D and ($j-l$) 10D control, respectively. All rows show the same streamline from the in-plane velocity components in grey for orientation. The solid circle marks the furthest downstream extend of the dead-water region $(x, z)_{DW} = \arg \max_x \{u(x, 0, z) \leq 0\}$. The squares denote in-plane velocity equilibria associated with the vortices.

confirmed by the pressure coefficient contours of figure 22(a, d, g, j). The large curvature at the end of the wake bubble is associated with small velocities, i.e. is of little relevance for this argument. More quantitative aspects of the recirculation bubble and its associated drag are beautifully summarized in the inspiring book by Hucho (2011) and have been applied to the Ahmed body wake by Barros *et al.* (2016).

For the optimized directed jets (case 10D) this wake elongation trend with increasing drag reduction is not continued, as depicted in figure 22($j-l$). Instead, the wake becomes more slender and symmetric as featured by the velocity equilibrium points marking the vortex centres (solid squares). The increased up-down symmetry is facilitated by the upward blowing of the bottom jet. This peripheral inward blowing enables aerodynamic boat tailing (Geropp 1995) as a new and evidently more effective drag reduction mechanism. Similar observations of increased drag reduction by inward actuation jet deflection have been made for the square-back Ahmed body with Coanda surfaces (Barros *et al.* 2016; Haffner *et al.* 2020), which also induce a sign change of streamline curvature leading to a local pressure increase near the wall. The pressure increase associated with a

more symmetric and slender bubble has also been reported in the vertical rear surface of a 25° Ahmed body using rounded corners by Rossitto *et al.* (2016).

The increased drag reduction by symmetrization of the wake can also be explained by the shape of the streamlines starting near the separation points. The whole wake region is pushed upward by the bottom jet blowing upward. As a consequence, the upper streamline bounding the wake region is more aligned with the streamwise direction. Hence, the transverse pressure gradient from the upper stream to the wake region is smaller, leading to a higher rearward pressure and thus smaller drag.

The pressure field illustrated in figure 22(a,d,g,j) expectedly shows that increasing drag reduction is associated with an increasing pressure towards the rearward Ahmed body windows. This trend is consistent with our streamline arguments. The visualization of the streamwise velocity component indicates a more pronounced near-wall jet with increasing drag reduction. Clearly, the velocity component towards the wake decreases in the upper wake region with increasing drag reduction. This tendency is consistent with the streamline arguments.

5.6.4. Vorticity field

The analysis of the vorticity field mirrors the discussion of the velocity field. Figure 23 displays iso-surfaces for the same Okubo–Weiss parameter value Q for all four cases. The unforced case $0D$ (figure 23a) shows pronounced C-pillar vortices extending far into the wake. Under streamwise top actuation (case $5D$, figure 23b), the C-pillar vortices significantly shorten. The next change with all streamwise actuators optimized (case $5D$) is modest, consistent with the small additional drag decrease. The C-pillar vortices are slightly shorter (see figure 23c). The inward deflection of the actuation (case $10D$) is associated with aerodynamic boat tailing, as displayed in figure 23(d). The separation from the slanted window is significantly delayed and the sidewise separation is vectored inward.

This actuation effect on the C-pillar vortices is corroborated by the streamwise vorticity contours in a transverse plane on body height downstream ($x/H = 1$). Figure 24 shows this averaged vorticity component for cases $0D$, $1D$, $5D$, $10D$ in panels a–d, respectively. The extent of the C-pillar vortices clearly shrinks with increasing drag reduction. We mention a similarity with Prandtl’s lifting line theory which relates the drag of a finite wing with the strength of the trailing edge vortices. From this theory, stronger C-pillar vortices should be related with larger lift and larger drag.

The effect of the upward-directed bottom jet can clearly be evidenced as strong vortices in the right and left near-wall regions of (d). For tangential blowing (a–c), the right and left sides feature pairs of near-wall vortices with smaller net circulation. The RANS results are validated with the LES and the experiment as detailed in Appendices C and D, respectively.

6. Conclusions

We propose a novel optimization approach for active bluff-body control exploiting local gradients with a DSM and exploring new better minima with LHS. This approach is called the explorative gradient method (EGM) as the iterations alternate between downhill simplex iteration as a robust gradient method and LHS as the most explorative step. A distinguishing feature of EGM is that it performs an ‘aggressive’ exploitation in one step and the arguably most optimal exploration in another step. Thus, both, exploitation and exploration come with optimizing principles and with an *a priori* evaluation investment

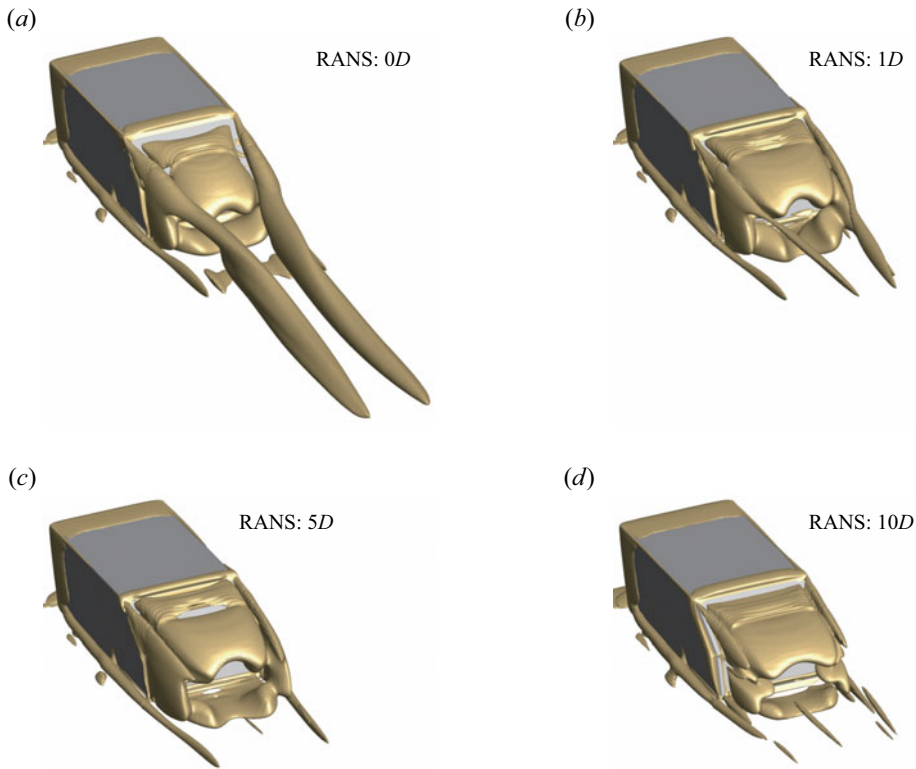


Figure 23. The iso-surfaces of Okubo–Weiss parameter $Q = 20000$ of the Ahmed body flow predicted by RANS. (a) Without control and under (b) 1D, (c) 5D and (d) 10D control respectively.

which is determined upfront. This policy has distinct advantages as illustrated in [figure 25](#) for three different control landscapes with one, few and many minima. In some cases, the exploitation (left column) will be pointless because the best minimum still needs to be found. In other cases, the exploration (middle column) will be ineffective, because gradient-based descent will quickly lead to the global minimum or because the dimension or complexity of the search space is too large. EGM (right column) hedges against both scenarios of inefficiency because high-dimensional search spaces typically have unknown characteristics. EGM seems a rational ‘all-weather’ strategy as it may perform best or, at least, does not fall much behind the winning exploitive or explorative optimizers.

This policy may be contrasted with the GA which can be remarkably effective in high dimensions, but the goal of explorative operations, like mutation, and exploitive operations, like crossover, come with no optimizing principle, like gradient-based convergence or geometric coverage of the search space. A similar observation applies to other biologically inspired optimization methods (see, e.g. [Wahde 2008](#)), like ant colony or particle swarm optimization. As another example, simulated annealing explores good minima before it increasingly exploits them. Here, again, exploration and exploitation come with no optimizing principle and the switch between exploration and exploitation is a design parameter. We argue that the radical alternation between gradient-based exploitation and maximal exploration is one of the most promising strategies in an unknown search space.

Explorative gradient method

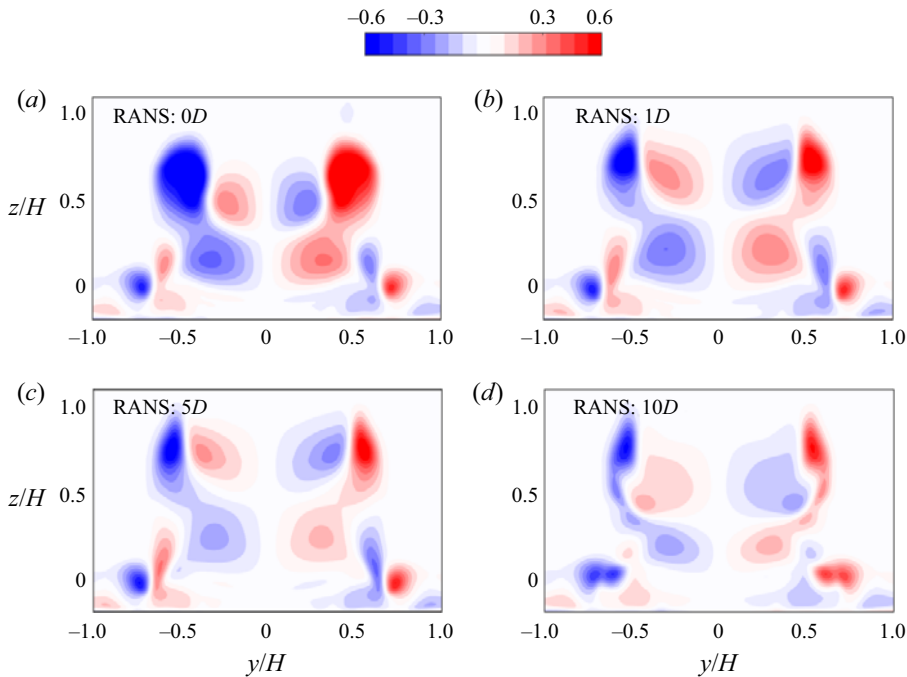


Figure 24. Streamwise vorticity component in near-wake plane $x/H = 1$ predicted by RANS. (a) Without control and under (b) 1D, (c) 5D and (d) 10D control, respectively.

Method \ Landscape	Exploit	Explore	EGM
Single minimum	Win ★ ★ ★	Lose ★	Win with price ★ ★
Several minima	Win with price ★	Win with price ★	Win ★ ★ ★
Many minima	Lose ★	Win ★ ★ ★	Win with price ★ ★

Figure 25. Simplified performance sketch of the exploitive, explorative and alternating optimizers for landscapes with one, few or many minima. Three and one stars indicate the best or ineffective strategy, respectively, for the considered control landscape. Two stars indicate a suboptimal yet practical solution.

EGM is compared with other optimizers for an analytical test function with one global and four local minima. The study includes the failure rate in finding the global optimum and the convergence rate. EGM is found to be distinctly superior in both aspects in comparison with (i) LHS, (ii) MCS, (iii) a GA, (iv) a DSM and (v) a random restart or shotgun DSM (RRS). This behaviour is made physically plausible for smooth cost functions with few minima, i.e. a typical case for active flow control.

As first flow control example, EGM is applied to the minimization of the parasitic net drag power of the multi-input fluidic pinball problem. It yields a slightly asymmetric

boat-tailing actuation with 29% net drag power reduction comprising 52% drag reduction penalized by 23% actuation energy. As very similar actuation has been found with machine learning control for feedback law ansatz (Cornejo Maceda *et al.* 2019). This boat-tailing actuation foreshadows the optimization result of the subsequent drag minimization of the Ahmed body. Intriguingly, EGM also probed base bleed and circulation control as options for drag power minimization.

EGM reduces the drag of an 35° slanted Ahmed body by 17% with independent steady blowing at all trailing edges at Reynolds number $Re_H = 1.9 \times 10^5$. The ten-dimensional actuation space includes five symmetric jet-slot actuators or corresponding actuator groups with variable velocity and variable blowing angle. The resulting drag is computed with a RANS simulation.

The approach is augmented by auxiliary methods for initial conditions, for accelerated learning and for a control landscape visualization. The initial condition for a RANS simulation with a new actuation is computed by the 1-nearest neighbour method. In other words, the RANS simulation starts with the converged RANS flow of the closest hitherto examined actuation. This cuts the computational cost by 60% as it accelerates RANS convergence. The actuation velocities are quantized to prevent testing of too similar control laws. This optional element reduces the CPU time by roughly 30%. The learning process is illustrated in a control landscape. This landscape depicts the drag in a proximity map – a two-dimensional feature space from the high-dimensional actuation response. Thus, the complexity of the optimization problem can be assessed.

The slanted Ahmed body with 1, 5 and 10 actuation parameters constitutes a more realistic plant for an optimization algorithm. First, only the upper streamwise jet actuator is optimized. This yields drag reduction of 5% with pronounced global minimum for the jet velocity. Second, the drag can be further reduced to 7% with five independent streamwise symmetric actuation jets. Intriguingly, the actuation effects of the actuator are far from additive – contrary to the experimental observation for the square-back Ahmed body (Barros 2015). The optimal parameters of a single actuator are not closely indicative for the optimal values of the combined actuator groups. The control landscape depicts a long curved valley with small gradient leading to a single global minimum. Interestingly, the explorative step is not only a security policy for the right minimum. It also helps to accelerate the optimization algorithm by jumping out of the valley to a point closer to the minimum.

A significant further drag reduction of 17% is achieved when, in addition to the jet velocities, also the jet angles are included in the optimization. Intriguingly, all trailing edge jets are deflected inward mimicking the effect of Coanda blowing and leading to fluidic boat tailing. The C-pillar vortices are increasingly weakened with one-, five- and ten-dimensional actuation. Compared with the pressure increase at C-pillar in one- and five-dimensional control, the ten-dimensional control brings a substantial pressure recovery over the entire base. The achieved 17% drag decrease with constant blowing is comparable to the experimental 20% reduction with high-frequency forcing by Bideaux *et al.* (2011) and Gilliéron & Kourta (2013).

For the 25° high-drag Ahmed body, (Zhang *et al.* 2018) have achieved 29% drag reduction with steady blowing at all sides, thus significantly outperforming all hitherto existing active flow control studies cited therein. The actuation has only been investigated for few selected actuation values. Hence, even better drag reductions are perceivable. Yet, the unforced high-drag Ahmed body has a significantly higher drag coefficient of 0.361 than the low-drag version and is hence not fully comparable. Their reduced drag coefficient of 0.256 is almost identical with the one of this study.

We expect that our RANS-based active control optimization is widely applicable for virtually all multi-input steady actuations or combinations of passive and active control (Bruneau *et al.* 2010). EGM mitigates the chances of sliding down a suboptimal minimum at an acceptable cost. The 1-nearest neighbour method for initial condition and the actuation quantization accelerate the simulations and learning processes. And the control landscape provides the topology of the actuation performance, e.g. the number of local minima, nature and shape of valleys, etc.

The current performance benefits of EGM over other commonly used optimizers has also been corroborated in preliminary bluff-body drag reduction experiments in Europe and China. Evidently, the optimizer can also be employed for cost function minimization for design parameters of passive devices or parameters of closed-loop control schemes.

7. Outlook

EGM can be algorithmically improved and generalized in several directions. The key idea of EGM is to balance exploration and exploitation by aiming to optimize each for one step in an alternating fashion. The winning choice of the exploration and exploitation method may depend on the dimensions of the search space. If this dimension is too large, the DSM may be replaced by a subplex (Rowan 1990) or a stochastic gradient method. Similarly, for large dimensions, the LHS may need to be replaced by MCS or GA (see Appendix B). Note that LHS will first explore the edge of the search space before it explores the centre.

The current version of EGM assumes complete lack of knowledge of the control landscape. The alteration between exploitation and exploration hedges against unnecessary evaluations in worst-case scenarios. For instance, exploration is unnecessary for a single radially symmetric cost function minimum, and exploitation is inefficient for large terrains with many minima. The authors actively pursue augmenting EGM from learning the response model and control landscape characteristics of the available data. This knowledge can be expected to change the algorithm, e.g. the alteration policy between explorative and exploitive steps. Machine learning (see, e.g. Brunton & Kutz 2019) provides powerful tools for these purposes.

An exciting new avenue is the generalization of EGM from a parameter optimizer to a regression problem solver: EGM has recently been applied to optimize multi-input multi-output control laws for the stabilization of the fluidic pinball and was found to be distinctly superior to genetic programming (Cornejo Maceda *et al.* 2021).

Other EGM applications of closed-loop control include the experimental stabilization of an open cavity flow Cornejo Maceda (2021), the lift increase of a high-Reynolds-number airfoil and the drag/side force reduction of a track model under transient yaw.

Summarizing, EGM is a versatile optimizer framework with numerous future applications. The algorithm can not only be applied to parameter optimization but also to model-free control law optimization, hitherto performed by genetic programming (Gautier *et al.* 2015; Ren, Hu & Tang 2020) and deep reinforcement learning (Bucci *et al.* 2019; Rabault *et al.* 2019).

A direct path is a structure identification of the control law so that the regression problem becomes a parameter optimization problem. Examples are cluster-based control laws (Nair *et al.* 2019), Taylor expansions and weights of neural networks (Lee *et al.* 1997). A second avenue is the interpolation of control laws in subspaces (Cornejo Maceda *et al.* 2021). The authors expect that future optimizers will synergize more and more methods and principles to deal with an increasing spectrum of control landscapes.

Acknowledgements. We are grateful to Q. Zhang, L. Huang, Y. Wang, Z. Li and S. Lv for performing the experimental validation for the numerical simulation. We are deeply indebted to D. Barros for illuminating discussions on the drag reduction mechanisms. We have profited from stimulating discussions with S. Brunton, V. Chernoray, G.Y. Cornejo Maceda, H. Zhou, B. Yan, R. Shen, N. Gao, S. Krajnović, H. Li, F. Lusseyran, N. Nayeri, O. Paschereit, L. Pastur, R. Semaan and W. Schröder. We express our sincere gratitude to the three anonymous referees. They have inspired a significant extension to our original submission from methodology to validation and physical discussion.

Funding. This work is supported by Shanghai Key Lab of Vehicle Aerodynamics and Vehicle Thermal Management Systems (Z.Y., grant number 18DZ2273300), by the National Natural Science Foundation of China (B.R.N., grant number 12172109), by the French National Research Agency (B.R.N., grant number ANR-17-ASTR-0022, FlowCon), by the German Science Foundation (B.R.N., grant numbers SE 2504/1-1, SE 2504/3-1) and by Polish Ministry of Science and Higher Education (MNiSW) (M.M., grant number 05/54/DSPB/6492).

Declaration of interests. The authors report no conflict of interest.

Author ORCIDs.

Yiqing Li <https://orcid.org/0000-0003-2547-5363>;

Bernd R. Noack <https://orcid.org/0000-0001-5935-1962>.

Appendix A. Genetic algorithm

This section provides further details about the chosen Matlab realization of GA.

- (i) First generation. The algorithm begins by creating an initial population with random individuals. Each parameter of each individual is taken with uniform probability from a given interval.
- (ii) Next generations. The algorithm uses the individuals in the current generation, called parents, to create individuals of the next population, called children:
 - (a) Cost evaluation. Score each member of the current population by computing its cost function. The cost function is assumed to be sorted, $J_1 \leq J_2 \leq \dots \leq J_r \leq \dots \leq J_I$. The index is called the rank.
 - (b) Scaled fitness. Scale the cost function based on relative ordering. An individual with rank r has fitness score of $1/\sqrt{r}$ (higher fitness, smaller rank).
 - (c) Parents. Select members, called parents, based on their expectation value. The selection function chooses parents for the next generation based on their expectation values. An individual can be selected more than once as a parent, in which case it contributes its genes to more than one child.
 - (d) Elitism. The best N_e individuals are copied directly into the new generation. This number corresponds to the probability $P_e = 0.05$, i.e. $N_e = I * P_e$
 - (e) Mutation and crossover. Produce children from the parents with crossover or mutation. By combining parts of genes from a pair of parents, crossover children are produced with probability $P_c = 0.8$. The remaining individuals, other than elite children, are mutation children by making random changes to a single parent. Scattered, the default crossover function, creates a random binary vector and selects the genes where the vector is a 1 from the first parent, the genes where the vector is a 0 from the second parent and combines the genes to form the child. The default mutation function adds a random number taken from a Gaussian distribution with mean 0 to each entry of the parent vector. The standard deviation of this distribution is determined by the parameters $scale = 1$ and $shrink = 1$ (Matlab 2018b).
 - (f) Next generation. This generation comprises all children as created above.

Explorative gradient method

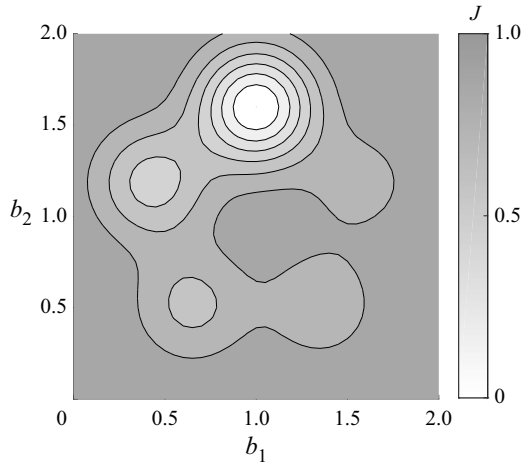


Figure 26. Two-dimensional analytical function contour.

- (iii) Termination. The algorithm stops when a stopping criterion is met. Here, the stopping criterion is the maximum generation number L .

Appendix B. Exploration in high-dimensional search spaces

A study of the exploration from low- to high-dimensional search spaces is presented. Different explorative methods are benchmarked. We show that LHS is recommended for dimensions up to five while GA performs better in higher-dimensional spaces.

Three optimization methods of § 2 are compared for analytical problems with five local minima for two-, four-, six- and eight-dimensional spaces. The corresponding analytical cost function reads

$$\begin{aligned}
 J(b_1, \dots, b_N) = & 1 - e^{\left[-10 \sum_{i=1}^N (b_1 - a_{1i})^2\right]} - \frac{1}{2} e^{\left[-10 \sum_{i=1}^N (b_2 - a_{2i})^2\right]} \\
 & - \frac{1}{3} e^{\left[-10 \sum_{i=1}^N (b_3 - a_{3i})^2\right]} - \frac{1}{4} e^{\left[-10 \sum_{i=1}^N (b_4 - a_{4i})^2\right]} \\
 & - \frac{1}{5} e^{\left[-10 \sum_{i=1}^N (b_5 - a_{5i})^2\right]}. \tag{B1}
 \end{aligned}$$

The 5 minima have the approximate locations $[a_{j1}, a_{j2}, \dots, a_{jN}]^T$, $j = 1, \dots, 5$. These minima are distributed evenly in the b_1, b_2 -plane on a circle with a radius of 0.6 around the $[1, 1]^T$ (see figure 26). In other words,

$$\begin{bmatrix} a_{j1} \\ a_{j2} \end{bmatrix} = \begin{bmatrix} 1 + 0.6 \sin \left[(6 - j) \times 2\pi/5 \right] \\ 1 + 0.6 \cos \left[(6 - j) \times 2\pi/5 \right] \end{bmatrix}, \quad j = 1, \dots, 5. \tag{B2}$$

In higher-dimensional spaces, the parameters vanish, $a_{ji} = 0$, $i = 3, \dots, N$.

The comparison includes three explorative optimization methods: LHS, MCS and GA. The corresponding parameters of the methods are the same as § 3.2. The average learning rates from 50 runs are shown in figure 27. In a low-dimensional space ($N = 2, 4$), LHS outperforms GA followed by MCS. The turning point appears around $n = 6$, from where GA significantly surpasses LHS. This study suggests replacing LHS with GA as the explorative step for EGM around $N \approx 5$.

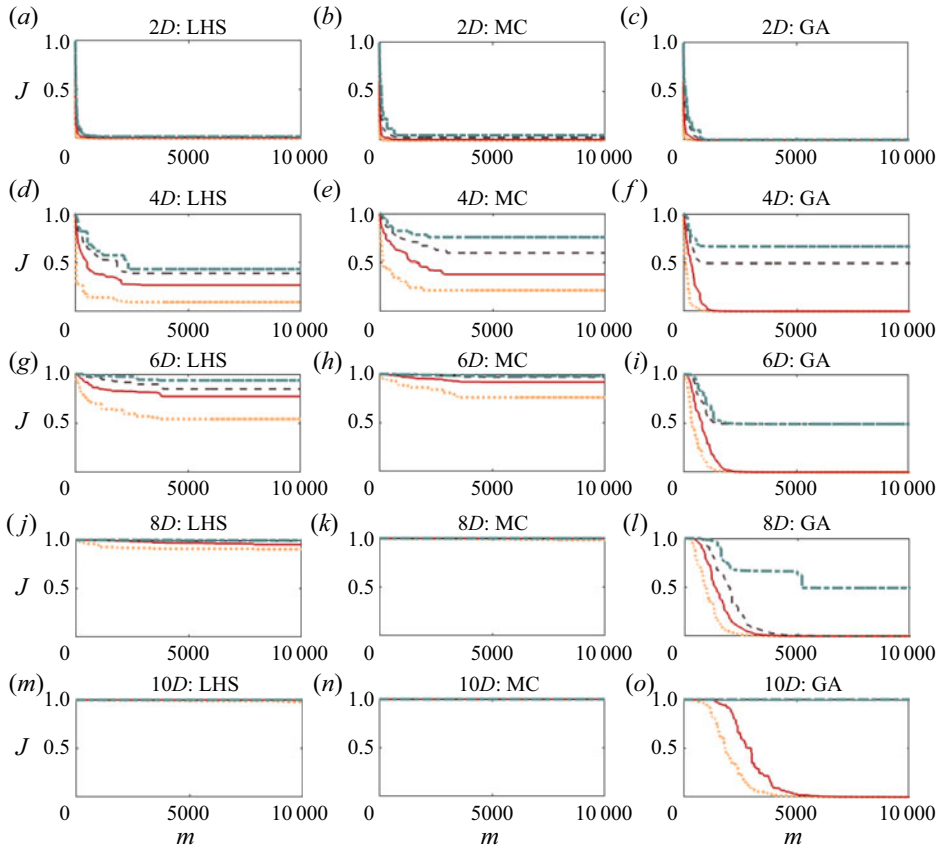


Figure 27. Comparison of explorative optimizers for the two-, four-, six- and eight-dimensional analytical functions (B1). Learning curves of (a,d,g,j,m) LHS, (b,e,h,k,n) MCS and (c,f,i,l,o) GA in 50 runs are plotted. The 10th, 50th, and 90th percentiles indicate the J value below which 10, 40 and 90 per cent of runs at current evaluation fall.

Appendix C. LES for the Ahmed body

This section discusses the first RANS validation for the unforced and actuated flow around the low-drag Ahmed body. A high-fidelity LES is performed for the unforced (0D) and optimized flows (1D, 5D and 10D) of the same configuration. The computational domain coincides the one of RANS (see § 5.2.1). The domain is discretized into an unstructured hexahedral grid with 17 million computational nodes. The cell distance in the normal direction from the body is within the range of $0.3 \leq x^+, y^+, z^+ \leq 0.7$. A wall-adjusted local eddy viscosity model is used for the subgrid-scale model (Weickert *et al.* 2010). The boundary conditions are prescribed as for RANS: no-slip wall condition on the Ahmed body and the stationary ground, steady Dirichlet inlet conditions for the incoming velocity, no-stress conditions at the sides and the top of the domain and a convective outflow condition. All simulations are computed for 250 convective time units. This corresponds to 69 downwash times over the Ahmed body.

First, the drag of unforced and actuated cases are compared. From figure 28, RANS and LES predict the trend of increasing drag reduction with increasing number of optimized actuation parameters. The deviation between RANS and LES predicted drag reductions increases with actuation complexity but remains always less than 3%. LES might yield

Explorative gradient method

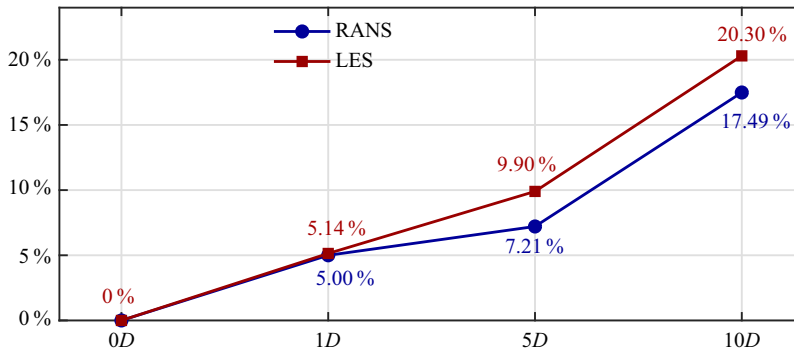


Figure 28. Drag reduction predicted by RANS and LES for the low-drag Ahmed body: 0D – the unforced benchmark, 1D – the optimized top streamwise actuator, 5D – the optimized all streamwise actuators and 10D – the optimized all deflected actuators.

higher drag reductions because it can resolve dynamic coherent effects of the unforced and actuated flows.

Second, the vorticity field of the time-averaged flow from LES is compared with RANS. In [figure 29](#), iso-surfaces of the Okubo–Weiss parameter visualize the vorticity field in analogy to [figure 23](#) depicting RANS data. The C-pillar vortices of the benchmark flow (case 0D, [figure 29a](#)) are weakened under the optimized streamwise top actuation (case 5D, [figure 29b](#)). For optimized actuation of all streamwise jets this vorticity mitigation trend is continued ([figure 29c](#)). LES predicts the aerodynamic boat tailing characterized by the inward deflection and the delayed separation associated with case 10D and illustrated in [figure 29\(d\)](#). This behaviour follows the companion [figure 23](#) of RANS.

Finally, a quantitative comparison of RANS and LES is performed and benchmarked with the companion experiment in [Appendix D](#). Summarizing, the proposed RANS-based actuation optimization faithfully distils the achievable drag reductions in the search space hierarchy. The computational savings of RANS over LES is immense, with a factor of 140. One RANS simulation employs a cluster with 52 cores for only 35 min. In contrast, one LES uses a cluster with 60 cores for 3 days. In other words, the one RANS optimization can be performed with the CPU load needed for one LES.

Appendix D. RANS simulation validation for the Ahmed body

This section provides a validation of RANS simulations of the low-drag Ahmed body. This validation comprises a grid convergence study of RANS, LES of [Appendix C](#), and a companion experiment described below.

The grid convergence of RANS is corroborated by simulations on three meshes, containing 2.5, 5 and 10 million nodes described in § 5.2.1. [Figure 30](#) presents the time-averaged streamwise velocity profiles of the corresponding RANS simulations in the symmetry plane. [Figures 30\(a\)](#) and [30\(b\)](#) show a good agreement of the flow over the slanted window and the near-wake region. The figure includes RANS profiles for the grid with 2.5 million (‘2.5M’, blue dashed curve), 5 million (‘5M’, red fine curve) curve and 10 million grid points (‘10M’, grey bold curve). The difference in RANS profiles is hardly distinguishable for the medium and fine mesh, except for a very narrow region around $x/H = -0.2$. The drag variation between the results obtained with the medium and fine meshes is only 0.6%. Hence, the medium mesh is used for the simulation during

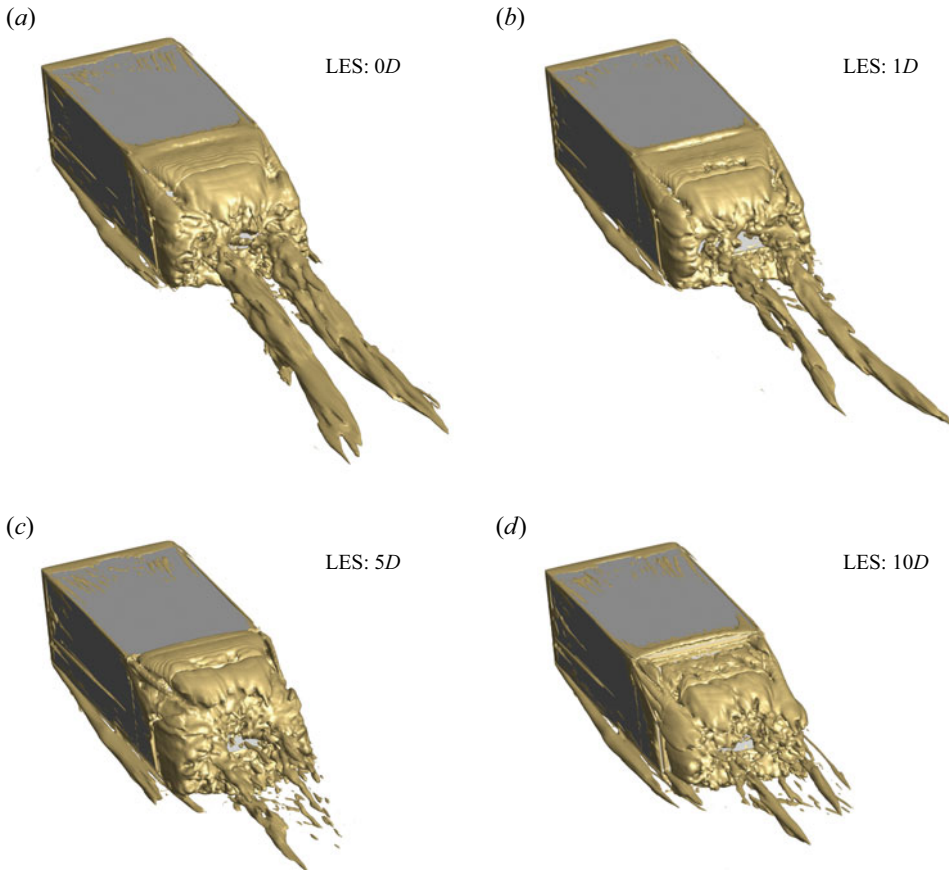


Figure 29. The iso-surfaces of Okubo–Weiss parameter $Q = 20000$ of the Ahmed body flow predicted by LES. (a) Without control and under (b) 1D, (c) 5D and (d) 10D control, respectively.

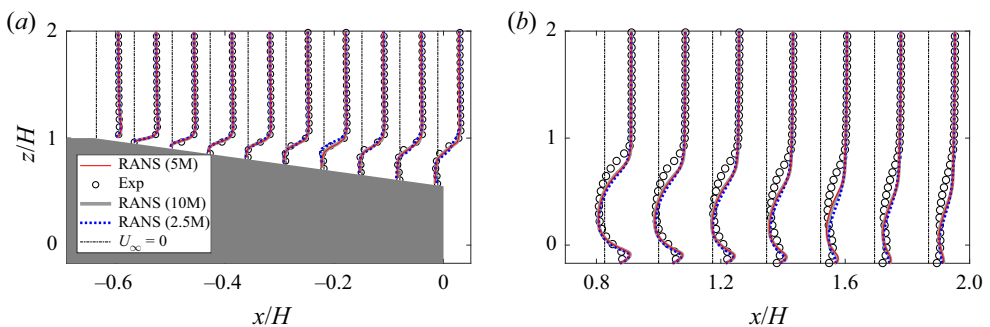


Figure 30. Time-averaged profiles obtained with the RANS model in the symmetry plane $y = 0$. (a) over the slanted window and (b) in the wake of the Ahmed body.

optimization as compromise between efficiency and accuracy. For later reference, the experimental data are included in [figure 30](#) as circles.

The experiment of the unforced configuration is carried out in a three-quarter open-jet low-speed closed-loop wind tunnel with a test section of $1520 \text{ mm} \times 1185 \text{ mm} \times 810 \text{ mm}$, belonging to Shanghai Automotive Wind Tunnel Center (SAWTC), Tongji University (see

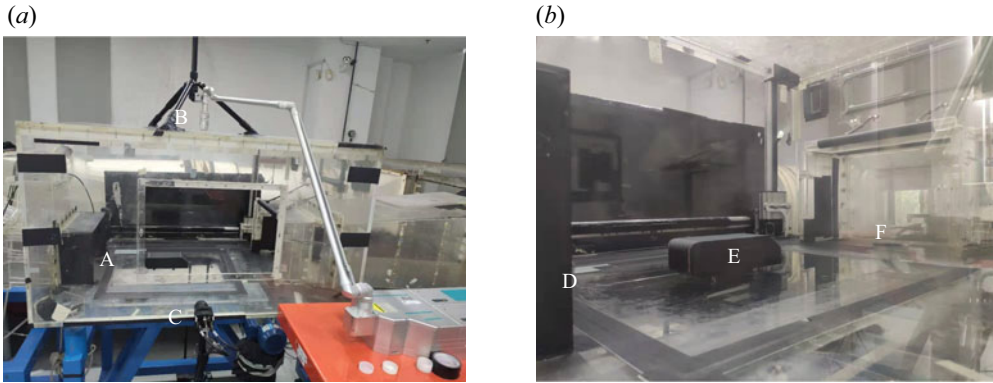


Figure 31. The set-up of PIV measurement (a) for the uncontrolled Ahmed body. The Ahmed body (E) is placed in the test section (A), with Vlite-500 pulse laser source (B) at the top and a charge-coupled device camera (C). D, F in (b) are the nozzle exit and the collector, respectively.

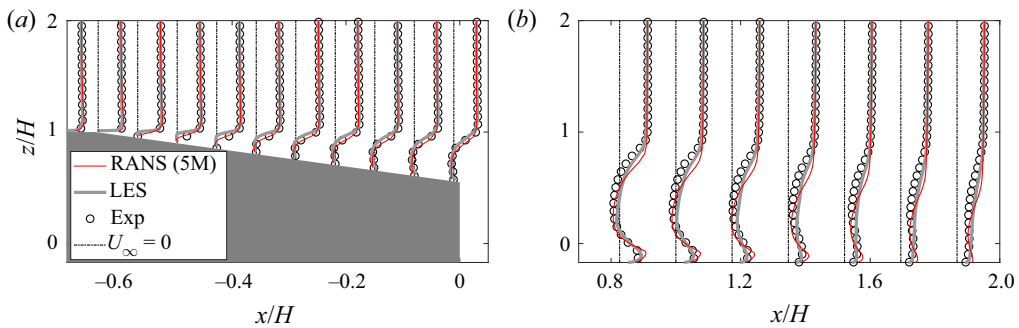


Figure 32. Velocity profiles over the slanted back (a) and in the wake (b) of the Ahmed body from RANS, LES and experimental results. For details see text.

figure 31). The maximum wind speed is approximately 49 m s^{-1} , The free-stream turbulent intensity from the nozzle ($432 \text{ mm} \times 288 \text{ mm}$) is less than 0.5%. The Reynolds number Re_H of the unforced flow is equal to 1.9×10^5 , consistent with performed RANS and LES simulations.

The velocity field of the vertical mid-plane $y = 0$ in the wake is measured by a particle image velocity (PIV) system from TSI Incorporated. The image magnification in the vertical planes is $72 \mu\text{m pixel}^{-1}$. The time interval between two successive pulses is $10 \mu\text{s}$. The mean velocity (see black circle in figure 32) is calculated from 1000 pairs of PIV images captured with the sampling frequency of 1.25 Hz. The streamwise velocity profiles over the rear slant and in the wake are plotted in figures 32(a) and 32(b), respectively. The location of the profiles follows Lienhart, Stoots & Becker (2002). The x -axis is stretched by a factor 2 to disperse the profiles. The dotted-dashed lines mark the x location and correspond to $U = 0$. The distance of each data point from its x location indicates the parameter $(U - U_\infty)/U_\infty$, where U_∞ is the incoming velocity. Both RANS and LES predict well the experimental flow profile in the symmetry plane starting with an early separation from the upper leading edge. LES shows a slightly better agreement with experiment. The drag coefficient predicted by LES is 0.3423 and closer to the experimental value 0.355 than the RANS value of 0.3134.

REFERENCES

- AIDER, J.-L., BEAUDOIN, J.-F. & WESFREID, J.E. 2010 Drag and lift reduction of a 3D bluff-body using active vortex generators. *Exp. Fluids* **48**, 771–789.
- BARROS, D. 2015 Wake and drag manipulation of a bluff body using fluidic forcing. PhD thesis, École Nationale Supérieure de Mécanique et d'Aérotechnique, Poitiers, France.
- BARROS, D., BORÉE, J., NOACK, B.R., SPOHN, A. & RUIZ, T. 2016 Bluff body drag manipulation using pulsed jets and Coanda effect. *J. Fluid Mech.* **805**, 442–459.
- BEN-HAMOU, E., ARAD, E. & SEIFERT, A. 2007 Generic transport aft-body drag reduction using active flow control. *Flow Turbul. Combust.* **78** (3–4), 365.
- BIDEAUX, E., BOBILLIER, P., FOURNIER, E., GILLIÉRON, P., EL HAJEM, M., CHAMPAGNE, J.-Y., GILOTTE, P. & KOURTA, A. 2011 Drag reduction by pulsed jets on strongly unstructured wake: towards the square back control. *Int. J. Aerodyn.* **1** (3–4), 282–298.
- BRUNEAU, C.-H., CREUSÉ, E., DEPEYRAS, D., GILLIÉRON, P. & MORTAZAVI, I. 2010 Coupling active and passive techniques to control the flow past the square back Ahmed body. *Comput. & Fluids* **39** (10), 1875–1892.
- BRUNN, A. & NITSCHKE, W. 2006 Drag reduction of an Ahmed car model by means of active separation control at the rear vehicle slant. In *New Results in Numerical and Experimental Fluid Mechanics* (ed. H.-J. Rath, C. Holze, H.-J. Heinemann, R. Henke & H. Honlinger), pp. 249–256. Springer.
- BRUNTON, S.L. & KUTZ, N. 2019 *Data-Driven Science and Engineering*. Cambridge University Press.
- BRUNTON, S.L. & NOACK, B.R. 2015 Closed-loop turbulence control: progress and challenges. *Appl. Mech. Rev.* **67** (5), 050801.
- BUCCI, M.A., SEMERARO, O., ALLAUZEN, A., WISNIEWSKI, G., CORDIER, L. & MATHELIN, L. 2019 Control of chaotic systems by deep reinforcement learning. *Proc. Roy. Soc. London A* **475**, 20190351.
- CHEN, W., JI, C., ALAM, M.M., WILLIAMS, J. & XU, D. 2020 Numerical simulations of flow past three circular cylinders in equilateral-triangular arrangements. *J. Fluid Mech.* **891**, 1–44.
- CHOI, H., LEE, J. & PARK, H. 2014 Aerodynamics of heavy vehicles. *Annu. Rev. Fluid Mech.* **46** (1), 441–468.
- CORNEJO MACEDA, G.Y. 2021 Gradient-enriched machine learning control exemplified for shear flows in simulations and experiments. PhD thesis, Université Paris-Saclay, France.
- CORNEJO MACEDA, G.Y., LI, Y., LUSSEYRAN, F., MORZYŃSKI, M. & NOACK, B.R. 2021 Stabilization of the fluidic pinball with gradient-based machine learning control. *J. Fluid Mech.* **917**, A42.
- CORNEJO MACEDA, G.Y., NOACK, B.R., LUSSEYRAN, F., DENG, N., PASTUR, L. & MORZYŃSKI, M. 2019 Artificial intelligence control applied to drag reduction of the fluidic pinball. *Proc. Appl. Math. Mech.* **19** (1), e201900268.
- COX, T.F. & COX, M.A.A. 2000 *Multidimensional Scaling*, 2nd edn. Monographs on Statistics and Applied Probability, vol. 88. Chapman and Hall.
- DEJOAN, A., JANG, Y.J. & LESCHZINER, M.A. 2005 Comparative LES and unsteady RANS computations for a periodically-perturbed separated flow over a backward-facing step. *J. Fluids Eng.* **127** (5), 872–878.
- DENG, N., NOACK, B.R., MORZYŃSKI, M. & PASTUR, L.R. 2020 Low-order model for successive bifurcations of the fluidic pinball. *J. Fluid Mech.* **884**, A37.
- FERNEX, D., SEMANN, R., ALBERS, M., MEYSONNAT, P.S., SCHRÖDER, W. & NOACK, B.R. 2020 Actuation response model from sparse data for wall turbulence drag reduction. *Phys. Rev. Fluids* **5** (7), 073901.
- GAD-EL-HAK, M. 2007 *Flow Control: Passive, Active, and Reactive Flow Management*. Cambridge University Press.
- GAUTIER, N., AIDER, J.-L., DURIEZ, T., NOACK, B.R., SEGOND, M. & ABEL, M.W. 2015 Closed-loop separation control using machine learning. *J. Fluid Mech.* **770**, 424–441.
- GERHARD, J., PASTOOR, M., KING, R., NOACK, B.R., DILLMANN, A., MORZYŃSKI, M. & TADMOR, G. 2003 Model-based control of vortex shedding using low-dimensional Galerkin models. Paper 4262. In *33rd AIAA Fluids Conference and Exhibit*. AIAA.
- GEROPP, D. 1995 Process and device for reducing the drag in the rear region of a vehicle, for example, a road or rail vehicle or the like. United States Patent US 5407245 A.
- GEROPP, D. & ODENTHAL, H.-J. 2000 Drag reduction of motor vehicles by active flow control using the Coanda effect. *Exp. Fluids* **28** (1), 74–85.
- GILLIÉRON, P. & KOURTA, A. 2013 Aerodynamic drag control by pulsed jets on simplified car geometry. *Exp. Fluids* **54** (2), 1457.
- GLEZER, A., AMITAY, M. & HONOHAN, A.M. 2005 Aspects of low- and high-frequency actuation for aerodynamic flow control. *AIAA Journal* **43** (7), 1501–1511.

- GUILMINEAU, E., DENG, G.B., LEROYER, A., QUEUTEY, P., VISONNEAU, M. & WACKERS, J. 2018 Assessment of hybrid RANS-LES formulations for flow simulation around the Ahmed body. *Comput. Fluids*. **176**, 302–319.
- HAFFNER, Y., BORÉE, J., SPOHN, A. & CASTELAIN, T. 2020 Unsteady Coanda effect and drag reduction for a turbulent wake. *J. Fluid Mech.* **899**, A36.
- HAN, X., KRAJNOVIĆ, S. & BASARA, B. 2013 Study of active flow control for a simplified vehicle model using the PANS method. *Int. J. Heat Fluid Flow* **42**, 139–150.
- HUCHO, W.H. 2011 *Aerodynamik der stumpfen Körper: Physikalische Grundlagen und Anwendungen in der Praxis*. Vieweg+Teubner.
- HUMPHREY, D.G. & WILSON, J.R. 2000 A revised simplex search procedure for stochastic simulation response surface optimization. *INFORMS J. Comput.* **12** (4), 272–283.
- ISHAR, R., KAISER, E., MORZYNSKI, M., ALBERS, M., MEYSONNAT, P., SCHRÖDER, W. & NOACK, B.R. 2019 Metric for attractor overlap. *J. Fluid Mech.* **874**, 720–752.
- KIM, J. 2011 Physics and control of wall turbulence for drag reduction. *Phil. Trans. Roy. Soc. A* **369** (1940), 1396–1411.
- KRAJNOVIĆ, S. 2009 Large eddy simulation of flows around ground vehicles and other bluff bodies. *Phil. Trans. R. Soc. A* **367** (1899), 2917–2930.
- LEE, C., KIM, J., BABCOCK, D. & GOODMAN, R. 1997 Application of neural networks to turbulence control for drag reduction. *Phys. Fluids* **9** (6), 1740–1747.
- LI, Y., CUI, W., JIA, Q. & YANG, Z. 2018 Wake control on a simplified vehicle using steady blowing. In *Proceedings of the 8th International Conference on Fluid Mechanics*, pp. 1–6.
- LIENHART, H., STOOFS, C. & BECKER, S. 2002 Flow and turbulence structures in the wake of a simplified car model (Ahmed Modell). In *New Results in Numerical and Experimental Fluid Mechanics III* (ed. S. Wagner, U. Rist, H.-J. Heinemann & R. Hilbig), pp. 323–330. Springer.
- MAEHARA, N. & SHIMODA, Y. 2013 Application of the genetic algorithm and downhill simplex methods (Nelder-Mead methods) in the search for the optimum chiller configuration. *Appl. Therm. Eng.* **61** (2), 433–442.
- MCKAY, M.D., BECKMAN, R.J. & CONOVER, W.J. 1979 Comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics* **21** (2), 239–245.
- MINELLI, G., DONG, T., NOACK, B.R. & KRAJNOVIĆ, S. 2020 Upstream actuation for bluff-body wake control driven by a genetically inspired optimization. *J. Fluid Mech.* **893**, A1.
- MURALIDHARAN, K., MUDDADA, S. & PATNAIK, B.S.V. 2013 Numerical simulation of vortex induced vibrations and its control by suction and blowing. *Appl. Math. Model.* **37** (1–2), 284–307.
- NAIR, A., YEH, C.-A., KAISER, E., NOACK, B.R., BRUNTON, S.L. & TIARA, K. 2019 Cluster-based feedback control of turbulent post-stall separated flows. *J. Fluid Mech.* **875**, 345–375.
- NELDER, J.A. & MEAD, R. 1965 A simplex method for function minimization. *J. Comput.* **7**, 308–313.
- PARK, H., LEE, D., JEON, W.-P., HAHN, S., KIM, J., CHOI, J. & CHOI, H. 2006 Drag reduction in flow over a two-dimensional bluff body with a blunt trailing edge using a new passive device. *J. Fluid Mech.* **563**, 389–414.
- PASTOOR, M., HENNING, L., NOACK, B.R., KING, R. & TADMOR, G. 2008 Feedback shear layer control for bluff body drag reduction. *J. Fluid Mech.* **608**, 161–196.
- PFEIFFER, J. & KING, R. 2014 Linear parameter varying active flow control for a 3D bluff body exposed to cross-wind gusts. Paper 2014-2406. In *32nd AIAA Applied Aerodynamics Conference*. AIAA.
- PRESS, W.H., FLAMERY, B.P., TEUKOLSKY, S.A. & VETTERLING, W.T. 2007 *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press.
- PROTAS, B. 2004 Linear feedback stabilization of laminar vortex shedding based on a point vortex model. *Phys. Fluids* **16** (12), 4473–4488.
- RABAULT, J., KUCHTA, M., JENSEN, A., RÉGLADE, U. & CERARDI, N. 2019 Artificial neural networks trained through deep reinforcement learning discover control strategies for active flow control. *J. Fluid Mech.* **865**, 281–302.
- RAIBAUDO, C., ZHONG, P., NOACK, B.R. & MARTINUZZI, R.J. 2020 Machine learning strategies applied to the control of a fluidic pinball. *Phys. Fluids* **32**, 015108.
- REN, F., HU, H. & TANG, H. 2020 Active flow control using machine learning: A brief review. *J. Hydrodyn.* **32** (2), 247–253.
- ROSSITTO, G., SICOT, C., FERRAND, V., BORÉE, J. & HARAMBAT, F. 2016 Influence of afterbody rounding on the pressure distribution over a fastback vehicle. *Exp. Fluids* **57**, 43.
- ROWAN, T. 1990 The subplex method for unconstrained optimization. PhD thesis, Department of Computer Sciences, University of Texas, USA.

- SCHMIDT, H.-J., WOSZIDLO, R., NAYERI, C.N. & PASCHEREIT, C.O. 2015 The effect of flow control on the wake dynamics of a rectangular bluff body in ground proximity. *Exp. Fluids* **56**, 151.
- SEIFERT, A., GREENBLATT, D. & WYGNANSKI, I.J. 2004 Active separation control: an overview of Reynolds and Mach numbers effects. *Aerosp. Sci. and Technol.* **8** (7), 569–582.
- SEIFERT, A., STALNOV, O., SPERBER, D., ARWATZ, G., PALEI, V., DAVID, S., DAYAN, I. & FONO, I. 2009 Large trucks drag reduction using active flow control. In *The Aerodynamics of Heavy Vehicles II: Trucks, Buses, and Trains* (ed. F. Browand, R. McCallen & J. Ross), pp. 115–133. Springer.
- SERRE, E., MINGUEZ, M., PASQUETTI, R., GUILMINEAU, E., DENG, G.B., KORNHAAS, M., SCHÄFER, M., FRÖHLICH, J., HINTERBERGER, C. & RODI, W. 2013 On simulating the turbulent flow around the Ahmed body: a French–German collaborative evaluation of LES and DES. *Comput. & Fluids* **78**, 10–23.
- THIRIA, B., GOUJON-DURAND, S. & WESFREID, J.E. 2006 The wake of a cylinder performing rotary oscillations. *J. Fluid Mech.* **560**, 123–147.
- VIKEN, S., VATSA, V., RUMSEY, C. & CARPENTER, M. 2003 Flow control analysis on the hump model with RANS tools. In *41st Aerospace Sciences Meeting and Exhibit*, p. 218. AIAA.
- VISWANATHAN, H. 2021 Aerodynamic performance of several passive vortex generator configurations on an Ahmed body subjected to yaw angles. *J. Braz. Soc. Mech. Sci.* **43**, 131.
- WAHDE, M. 2008 *Biologically Inspired Optimization Methods: An Introduction*. WIT Press.
- WEICKERT, M., TEIKE, G., SCHMIDT, O. & SOMMERFELD, M. 2010 Investigation of the LES WALE turbulence model within the lattice Boltzmann framework. *Comput. & Mathem. with Appl.* **59** (7), 2200–2214.
- WOOD, C.J. 1964 The effect of base bleed on a periodic wake. *Aeronaut. J.* **68** (643), 477–482.
- WRIGHT, A.H. 1991 Genetic algorithms for real parameter optimization. In *Foundations of Genetic Algorithms* (ed. G.J.E. Rawlins), pp. 205–218. Elsevier.
- ZHANG, B.F., LIU, K., ZHOU, Y., TO, S. & TU, J.Y. 2018 Active drag reduction of a high-drag Ahmed body based on steady blowing. *J. Fluid Mech.* **856**, 351–396.