

Linear genetic programming control for strongly nonlinear dynamics with frequency crosstalk

R. LI¹⁾, B. NOACK^{2,3,4,5)}, L. CORDIER¹⁾, J. BORÉE¹⁾,
E. KAISER⁶⁾, F. HARAMBAT⁷⁾

¹⁾*Institut Pprime, CNRS - ISAE-ENSMA - Université de Poitiers
Futuroscope Chasseneuil, France
e-mail: ruiying.li@ensma.fr*

²⁾*LIMSI, UPR 3251, Orsay, France*

³⁾*Technische Universität Braunschweig, Germany*

⁴⁾*Technische Universität Berlin, Germany*

⁵⁾*Harbin Institute of Technology
Graduate School Shenzhen, China*

⁶⁾*University of Washington
Seattle, WA, U.S.A.*

⁷⁾*Groupe PSA, Vélizy-Villacoublay, France*

WE ADVANCE GENETIC PROGRAMMING CONTROL (GPC) for turbulence flow control application building on the pioneering work of [1]. GPC is a recently proposed model-free control framework which explores and exploits strongly nonlinear dynamics in an unsupervised manner. The assumed plant has multiple actuators and sensors and its performance is measured by a cost function. The control problem is to find a control logic which optimizes the given cost function. The corresponding regression problem for the control law is solved by employing *linear genetic programming* as an easy and simple regression solver in a high-dimensional control search space. This search space comprises open-loop actuation, sensor-based feedback and combinations thereof — thus generalizing former GPC studies [2, 3]. This new methodology is denoted as linear genetic programming control (LGPC). The focus of this study is the frequency crosstalk between unforced, unstable oscillation and the actuation at different frequencies. LGPC is first applied to the stabilization of a forced nonlinearly coupled three-oscillator model comprising open- and closed-loop frequency crosstalk mechanisms. LGPC performance is then demonstrated in a turbulence control experiment, achieving 22% drag reduction for a simplified car model. In both cases, LGPC identifies the best nonlinear control achieving the optimal performance by exploiting frequency crosstalk. Our control strategy is suited to complex control problems with multiple actuators and sensors featuring nonlinear actuation dynamics. Significant further performance enhancement is envisioned in the more general field of machine learning control [4].

Key words: flow control, nonlinear dynamics, turbulent wake.

1. Introduction

CONTROL IS OF FUNDAMENTAL IMPORTANCE for most living and engineering systems. Its applications range from a small home heating controller using a single thermostat to a large industrial control system with hundreds of sensor measurements and control signals. Here, we focus on the potential of active flow control in fluid mechanics. Flow control helps to achieve important engineering goals, such as drag reduction of road vehicles, lift increase of airfoils, efficiency increase of harvesting wind and water energy and combustion processes (see for example [5]).

The rich application of control boosts the development of control theory. Over the past decades, it has progressed into a mature discipline with a sound theoretical foundation and powerful associated numerical algorithms. The most well-developed theory of control applies to a linear system or to the linearization of a nonlinear system about a fixed point or a periodic orbit. In fluid dynamics, linear control theory can be applied to stabilize shear layers over cavities [6, 7], wakes of a cylinder at low-Reynolds number [8], just to name a few benchmark configurations. In such works, an assumption of linearity can often be made, from which a model can be deduced. However, this assumption is not true for most of the real-world flows which are highly turbulent, inherently strongly nonlinear and lead to a broadband frequency spectra and a high-dimensional state space. Turbulent flows are also characterized by frequency interactions, also called frequency crosstalk: actuation at one frequency may change the whole spectrum of frequencies and thus ultimately affects the mean flow. Control of such systems poses great challenges because the linear theory rarely applies to them.

Yet, when looking at the flight maneuvers of birds, it is clear that nature has found impressive flow control solutions without apparent knowledge of the flow governing equations (Navier–Stokes equations) or reduced-order modeling. An eagle, for instance, can land gently under gusty wind conditions and in rain by moving its wings and feathers to manipulate fluid forces. This suggests an alternative way to perform flow control through optimization processes emulating nature’s evolution. Machine learning, and in particular evolutionary algorithms, can help us to achieve the control goal by mimicking the learning process of nature. The development of evolutionary computation starts from the fundamental work of [9–11] over 50 years ago. With the current advancement of big data and progress of powerful computer techniques, machine learning gets a fertile ground to grow and has been applied in a myriad of applications of control, modeling and prediction [4, 12, 13]. Genetic algorithms [9] and genetic programming [14] are the two most applied evolutionary algorithms. They learn and refine an effective control only based on the control performance (cost function) as measured on the control system. Genetic algorithms are employed for the parameter iden-

tification of controllers with a given structure like PID controller [15]. Genetic programming achieves both structure and parameter identification, thus it enables to identify arbitrary nonlinear control laws. In this case, neither a model, nor the control law structure needs to be known. The methodology of solving optimal control problems with methods of genetic programming is referred to as Genetic Programming Control (GPC), which is the focus of this study.

In this study, we target GPC for control of dynamics with strong nonlinearities—circumventing the challenge to construct corresponding models and to derive nonlinear control laws. GPC addresses the turbulent flow control challenges mentioned above using advanced methods of genetic programming. The latter is used as a powerful regression technique to explore and evolve effective control laws by learning from the training data of experiments or simulations. Successful applications of GPC include separation control [2, 16] and mixing layer control [3]. The innovations in this work include: (1) the use of *linear* genetic programming as a simpler algorithm and (2) a very general framework for control laws incorporating open-loop and sensor-based feedback control.

The paper is organized as follows. In Section 2, we present the proposed method and its implementation. Then, in Section 3, we demonstrate LGPC (linear genetic programming control) to the stabilization of a forced nonlinearly coupled three-oscillator model. This dynamical system illustrates that frequency crosstalk between actuation and dynamics can be the only enabling mechanism for stabilization—as typical in turbulence control. Moreover, the system comprises open- and closed-loop stabilization mechanisms—foreshadowing another feature of the studied experiment. In Section 4, LGPC is applied to a turbulence control experiment, achieving 22% drag reduction for a simplified car model. A landscape of the discovered control laws is visualized in Section 5 to examine its search space topology. Section 6 concludes with a summary and outlook.

2. Linear genetic programming control

We consider a multiple-input multiple-output (MIMO) system with the state $\mathbf{a} \in \mathbb{R}^{N_a}$, an input vector $\mathbf{b} \in \mathbb{R}^{N_b}$ commanding actuation and an output vector $\mathbf{s} \in \mathbb{R}^{N_s}$ sensing the state. Here, N_a , N_b and N_s denote the dimension of the state, the number of actuators and sensors, respectively. The general form of the system reads

$$(2.1a) \quad \frac{d\mathbf{a}}{dt} = \mathbf{F}(\mathbf{a}, \mathbf{b})$$

$$(2.1b) \quad \mathbf{s} = \mathbf{G}(\mathbf{a})$$

$$(2.1c) \quad \mathbf{b} = \mathbf{K}(\mathbf{s}).$$

The control \mathbf{b} directly affects the state \mathbf{a} through a general nonlinear propagator \mathbf{F} . \mathbf{G} is a measurement function comprising the sensor signals \mathbf{s} as a function of the state \mathbf{a} . The control objective is to construct a MIMO controller $\mathbf{b} = \mathbf{K}(\mathbf{s})$ so that the system has a desirable behaviour. Most control objectives can be formulated in a cost function $J(\mathbf{a}, \mathbf{b})$. The definition of J depends on the control goal. For instance, in a drag reduction problem, we define J as the drag power penalized by the actuation power.

Following [4], the control design is formulated as a regression problem: find the control law $\mathbf{b} = \mathbf{K}(\mathbf{s})$ which optimizes a given cost function J . The cost only depends on the control law, or, symbolically $J(\mathbf{K}(\mathbf{s}))$ for a well-defined initial value problem or statistically stationary actuation response. Summarizing, the control task is transformed into an optimization problem via cost minimization and is equivalent to finding \mathbf{K}^{opt} such that

$$(2.2) \quad \mathbf{K}^{\text{opt}}(\mathbf{s}) = \underset{\mathbf{K}}{\operatorname{argmin}} J(\mathbf{K}(\mathbf{s})).$$

The sensor-feedback law maps N_s sensor signals onto N_b actuation commands. Such feedback can be expected to be approximated by a finite number of elementary operations $(+, -, \times, \div, \dots)$ acting on the sensor signals \mathbf{s} and a finite number of fixed constants. Thus, the search space of permissible control laws is finite, yet of astronomical cardinality. Hence, an exhausting testing in an experiment or numerical calculation is not an option. Instead, we employ genetic programming (GP) as a powerful evolutionary search algorithm. GP yields optimal or near-optimal control laws in the search space with a high probability for suitable parameters, yet with no mathematically assured convergence. The original tree-based genetic programming (TGP) formulates the mapping by a binary tree structure [14]. Here, we propose to apply a more recent alternative to TGP, called linear genetic programming (LGP) [17]. LGP formulates the control law as a sequence of instructions operating on a set of sensors \mathbf{s} , elementary operators $(+, -, \times, \div, \dots)$ and constants (for details, see App. A in [18]). In contrast, the control law in TGP is represented as a recursive tree where the root holds the output variable, each branching node holds an elementary operator $(+, -, \times, \div, \dots)$ and the leaves contain the sensors \mathbf{s} and constants. TGP and LGP are equivalent in the sense that any LGP-law can be expressed in TGP and vice versa. The difference is the linear versus recursive coding of LGP and TGP, respectively. LGP describes a directed acyclic graph [19] and is not constrained as a tree structure. Hence, it provides more freedom for the function construction than TGP. Linear genetic programming is a terminus technique in computer sciences. The term ‘linear’ in LGP refers to the linear sequence of instructions, and not to the superposition principle like in differential equations. The resulting functions of LGP are typically nonlinear. As presented before, we

refer to this method as *linear genetic programming control* (LGPC). For details of LGPC, see [18].

The implementation of LGPC for feedback control is sketched in Fig. 1. The fast real-time control occurs in the inner loop with a control law proposed by LGPC. The control law is evaluated in the dynamical system over an evaluation time T . Then, a cost J is measured quantifying the performance of the control law. The cost value for each control law is sent to the slow outer learning loop, where LGPC evolves these laws.

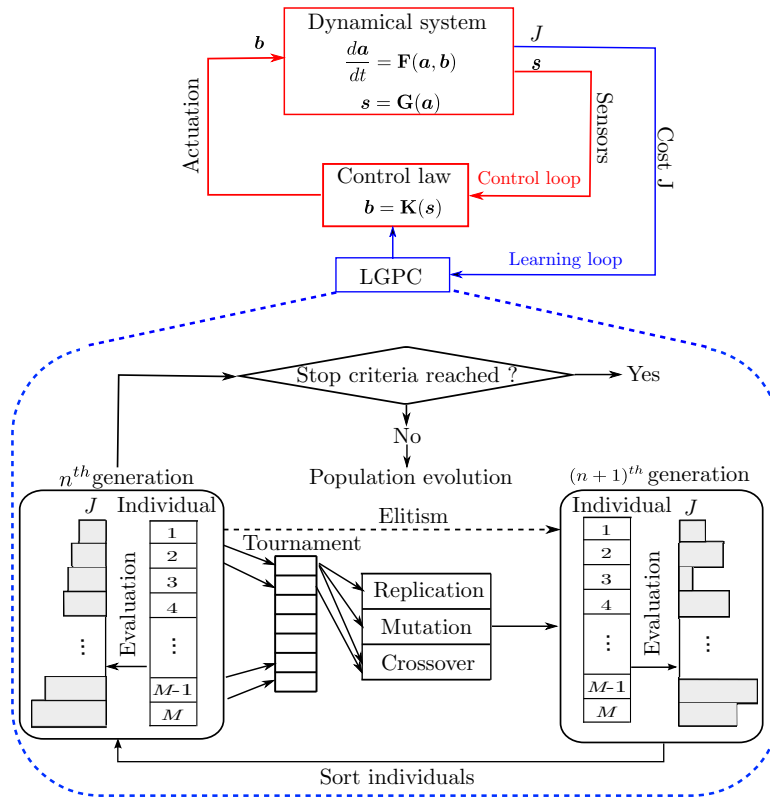


FIG. 1. LGPC implementation. The real-time closed-loop control is performed in the inner loop (red lines). The control plant feeds back the sensor output s to the control law. This control law proposed by LGPC computes the actuation command based on s and sends it back to the plant. A cost J is attributed to the control law after its evaluation during the time T . In the outer learning loop, LGPC uses these costs J to evolve the new population of control laws. The LGPC learning process is depicted in the lower part. On the leftmost side, an evaluated generation with M individuals is sorted in ascending order based on J . If the stopping criterion is met, the learning process is terminated. If not, the next generation (on the rightmost side) is evolved by genetic operators (elitism, replication, mutation, and crossover). After being evaluated, this generation is sorted as indicated by the arrow at the bottom. We repeat the process from left to right until the stopping criterion is met.

The learning process of LGPC is detailed in the lower part of Fig. 1. An initial population of control law candidates, called *individuals*, is generated randomly like in a Monte-Carlo method (see App. A in [18]). Each individual is evaluated in the inner loop and a cost J is attributed to them. After the whole generation is evaluated, its individuals are sorted in ascending order based on J . The next generation of individuals is then evolved from the previously evaluated one by elitism and genetic operators (replication, mutation, and crossover). Elitism is a deterministic process which copies a given number of top-ranking individuals directly to the next generation. This ensures that the best individual in the next generation will not perform worse than that of the previous one. The remaining genetic operations are stochastic in nature and have specified selection probabilities. The individual(s) used in these genetic operators is (are) selected by a tournament process: N_t randomly chosen individuals compete in a tournament and the winner(s) (based on J) is (are) selected. Replication copies a statistically selected number of individuals to the next generation. Thus better performing individuals are memorized. Crossover involves two statistically selected individuals and generates a new pair of individuals by exchanging randomly their instructions. This operation contributes to breeding better individuals by searching the space around well-performing individuals. In the mutation operation, random elements in the instructions of a statistically selected individual are modified. Mutation serves to explore potentially new and better minima of J . After the new generation is filled, the evaluation of this generation can be pursued in the plant. This learning process will continue until a stopping criterion is met. Different criteria are used. Ideally, the process is stopped when a known global minimum is obtained (which is unlikely in an experiment). Alternatively, the evolution terminates upon no improvement from one generation to the next or when a predefined maximum number of generations is reached. By definition, the targeted optimal control law is the best individual of the last generation.

LGPC can also be used to explore open-loop control by including time-periodic functions \mathbf{h} in the inputs of the control law, i.e. $\mathbf{b} = \mathbf{K}(\mathbf{h})$. This method permits to search a much more general multi-frequency control which is hardly accessible to a parametric study of single frequency. Furthermore, the range of LGPC can be extended by comprising both the sensors \mathbf{s} and time-periodic functions \mathbf{h} into the inputs of \mathbf{K} . This results in a non-autonomous control law $\mathbf{b} = \mathbf{K}(\mathbf{s}, \mathbf{h})$. This generalization permits to select between open-loop actuation $\mathbf{b} = \mathbf{K}(\mathbf{h})$, sensor-based feedback $\mathbf{b} = \mathbf{K}(\mathbf{s})$ or combinations thereof $\mathbf{b} = \mathbf{K}(\mathbf{s}, \mathbf{h})$ depending on which performs better. In the following, we term the approach optimizing open-loop frequency combinations $\mathbf{b} = \mathbf{K}(\mathbf{h})$ as LGPC-1. The approach to optimize autonomous controllers $\mathbf{b} = \mathbf{K}(\mathbf{s})$ is referred to as LGPC-2. The generalized non-autonomous control design $\mathbf{b} = \mathbf{K}(\mathbf{s}, \mathbf{h})$ is denoted as LGPC-3.

3. Model of three coupled oscillators

In this section, we illustrate LGPC to stabilize a forced dynamical system with three nonlinearly coupled oscillators at three incommensurable frequencies extending the generalized mean-field model [20] (see Chapter 5 of [4]). The goal is to stabilize the first unstable, amplitude-limited oscillator, while the forcing is performed on the second and third oscillator (see Fig. 2). The second oscillator has also unstable, amplitude-limited dynamics and destabilizes the first oscillator. The third oscillator has linear stable dynamics and has a stabilizing effect on the first. The stabilization of the first oscillator can be performed by closed-loop suppression of the second oscillator or open-loop excitation of the third one. In the following, we formulate the control problem mathematically (Section 3.1), parametrically explore the effect of periodic forcing like in many turbulence control experiments (Section 3.2), and apply LGPC (Section 3.3).

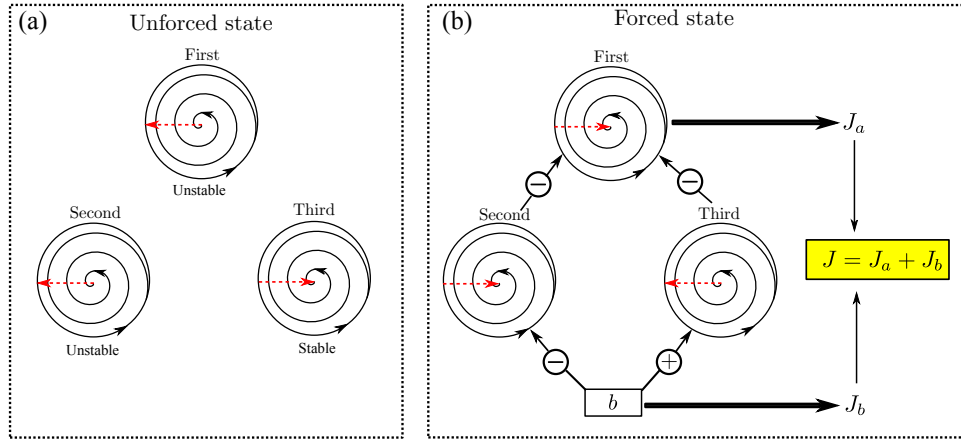


FIG. 2. Illustration of the three-oscillator model: (a) unforced state and (b) forced state. The red dashed arrows indicate the tendency of amplitudes of oscillators. The sign ‘-’ and ‘+’ in (b) represent the suppression and excitation of oscillators, respectively.

3.1. Problem formulation

The system has three oscillators at frequency $\omega_1 = 1$, $\omega_2 = \pi$ and $\omega_3 = \pi^2$, the coordinates of which being (a_1, a_2) , (a_3, a_4) and (a_5, a_6) , respectively. The evolution equation of the state $\mathbf{a} = (a_1, a_2, \dots, a_6)$ reads:

$$\begin{aligned}
 (3.1) \quad & \frac{da_1}{dt} = \sigma_1 a_1 - a_2, & \frac{da_3}{dt} = \sigma_2 a_3 - \pi a_4, & \frac{da_5}{dt} = \sigma_3 a_5 - \pi^2 a_6, \\
 & \frac{da_2}{dt} = \sigma_1 a_2 + a_1, & \frac{da_4}{dt} = \sigma_2 a_4 + \pi a_3 + b, & \frac{da_6}{dt} = \sigma_3 a_6 + \pi^2 a_5 + b, \\
 & \sigma_1 = -r_1^2 + r_2^2 - r_3^2, & \sigma_2 = 0.1 - r_2^2, & \sigma_3 = -0.1,
 \end{aligned}$$

where $r_1^2 = a_1^2 + a_2^2$, $r_2^2 = a_3^2 + a_4^2$ and $r_3^2 = a_5^2 + a_6^2$ denote the fluctuation level of the three oscillators, respectively. The growth rate for each oscillator is denoted by $\sigma_i, i = 1, \dots, 3$. Without forcing $b \equiv 0$, the first and second system are linearly unstable and damped by a Landau-type cubic term to asymptotic amplitudes $r_1^u = r_2^u = \sqrt{0.1}$. Here, and in the following, the superscript ‘ u ’ refers to asymptotic values for unforced dynamics. The third system is linear and stable, i.e. converges to the vanishing amplitude $r_3^u = 0$. The forcing b is only applied on the second and third oscillators. A linearization of Eqs. (3.1) around the fixed point $\mathbf{a} = \mathbf{0}$ yields a system in which the first oscillator is uncontrollable.

The effect of the forcing on the first oscillator can be inferred from the growth rate formula for σ_1 (see first column in Eqs. (3.1)). The fluctuation level r_2 of the second system destabilizes the first oscillator, while the third system stabilizes it with increasing fluctuation level r_3 . Hence, stabilization of the first oscillator may be achieved by exploiting one of two frequency crosstalk mechanisms: stabilizing the second system or exciting the third one. Evidently stabilization of the second system requires feedback $b = K(\mathbf{a})$ while excitation of the stable oscillator can be performed with the periodic forcing $b(t) = B \sin(\pi^2 t)$ at the resonance frequency and sufficiently large amplitude B .

The cost function to be minimized is the averaged energy of the unstable oscillator $J_a = \overline{a_1^2 + a_2^2}$ penalized by the actuation cost $J_b = \overline{b^2}$. Here, the temporal averaging is indicated by the overbar. Without forcing, $J_a^u = (r_1^u)^2$ and $J_b \equiv 0$. We normalize the total cost by the unforced value J_a^u of the first oscillator to characterize the relative benefit of actuation:

$$(3.2) \quad J = \frac{J_a + \gamma J_b}{J_a^u},$$

with $\gamma = 1$ as the penalization coefficient. By definition, $J = 1$ for the unforced system.

The numerical evaluation of J is based on the integration of the dynamical system (3.1) with the initial condition $\mathbf{a}(0) = (0.1, 0, 0.1, 0, 0.1, 0)$ at $t = 0$. In the first 10 periods of the target oscillator, i.e. for $t \in [0, t_0]$ with $t_0 = 10 \frac{2\pi}{\omega_1} = 20\pi$, no forcing is applied and the system converges to unforced quasi-periodic dynamics $(r_1^u)^2 = 0.1$, $(r_2^u)^2 = 0.1$, $r_3^u = 0$. The cost function is evaluated in the next 500 periods, $t \in [20\pi, 1020\pi]$. This time interval contains an actuated transient but is dominated by the post-transient dynamics, i.e. sufficient for statistical averaging.

3.2. Open-loop periodic forcing

First, the open-loop periodic forcing is studied, following a practice of many turbulence control experiments. The goal is to minimize the cost function

Eq. (3.2) with the periodic forcing $b_{\text{OL}}(t) = B \sin(\omega t)$ employing a parametric variation of the amplitude B and frequency ω in the range of $[0, 1]$ and $[0, 4\pi]$, respectively. The performance (Eq. (3.2)) at amplitude B and frequency ω is scanned with increments 0.01 and 0.01π , respectively. The corresponding colormap of J is shown in Fig. 3. This figure displays a local minimum of $J^\circ = 0.031$. The corresponding parameters are denoted by the superscript ‘ \circ ’ in the following. The low value indicates a stabilization by over one order of magnitude in the fluctuation level, accounting for the actuation expense. The minimum J is reached at the eigenfrequency of the third oscillator $\omega^\circ = \pi^2$, as $\sigma_1 < 0$ for $r_3^2 > 0.1$, numerically observing that the second oscillator is hardly affected by the forcing at a non-resonant frequency, $r_2^\circ \approx r_2^u = \sqrt{0.1}$. The optimal amplitude $B^\circ = 0.07$ is numerically determined as the best trade-off between the achieved stabilization and actuation cost. This amplitude leads to $r_3^2 \approx 0.12$ and $\sigma_1 \approx -0.02$. For a larger time evaluation horizon, the current results suggest a better performance at lower actuation $B \approx 0.05$ leading to $r_3^2 \approx 0.1$ which just neutrally stabilizes the first oscillator $\sigma_1 \approx 0$, exploiting that the second oscillator is unaffected by forcing. The corresponding analytical approximations are described in Chapter 5 of [4].

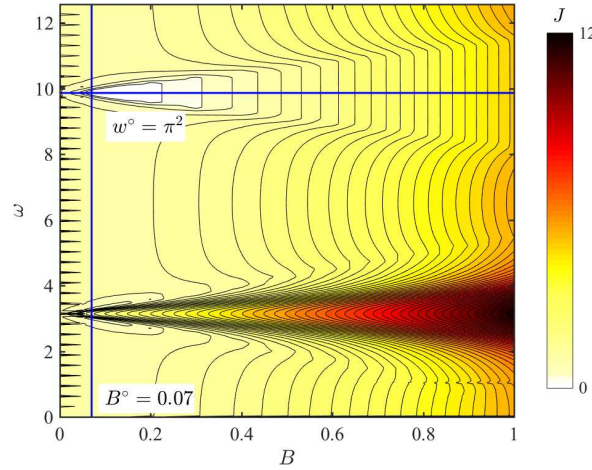


FIG. 3. Colormap of cost value J under the periodic forcing $b(t) = B \sin(\omega t)$.

On the other hand, the maximal J value is associated with the forcing at the eigenfrequency of the second oscillator $\omega = \pi$, as the excitation of r_2 leads to $\sigma_1 > 0$, resulting in an increase of r_1 . These results show that the enabler of open-loop control is the third oscillator rather than the second.

The unforced transient and actuated dynamics of the system are illustrated in Fig. 4 under the optimal periodic forcing $b^\circ(t) = 0.07 \sin(\pi^2 t)$. The unforced

state during the time window $t \in [0, 20\pi]$ is depicted by a blue dashed line and the forced one at $t > 20\pi$ by a red curve. For clarity, only the first 110 periods are shown in Fig. 4(a-d). Figure 4(e,f) covers the whole time interval $t \in [0, 1020\pi]$. When unforced, the unstable oscillators self-amplify towards the limit cycle $r_1^2 = r_2^2 = 0.1$, whilst the stable oscillator vanishes to $r_3^2 = 0$. Convergence is implied by $\sigma_1 = 0$ and $\sigma_2 = 0$. Once b starts at $t_0 = 20\pi$, r_3 is rapidly excited to an energy level of $r_3^2 = 0.12$, while r_2 keeps its original fluctuation level $r_2^2 = 0.1$. The resulting system yields $\sigma_1 < 0$ which leads consequently to the stabilization of (a_1, a_2) , i.e. $r_1^2 \approx 0$. The phase portraits in Fig. 4(e) and (f) illustrate the interactions between different oscillators. The circle indicates the initial point and the arrows the time direction. The forced trajectories represent low-pass filtered data, i.e. do not resolve cycle-to-cycle variation. In particular,

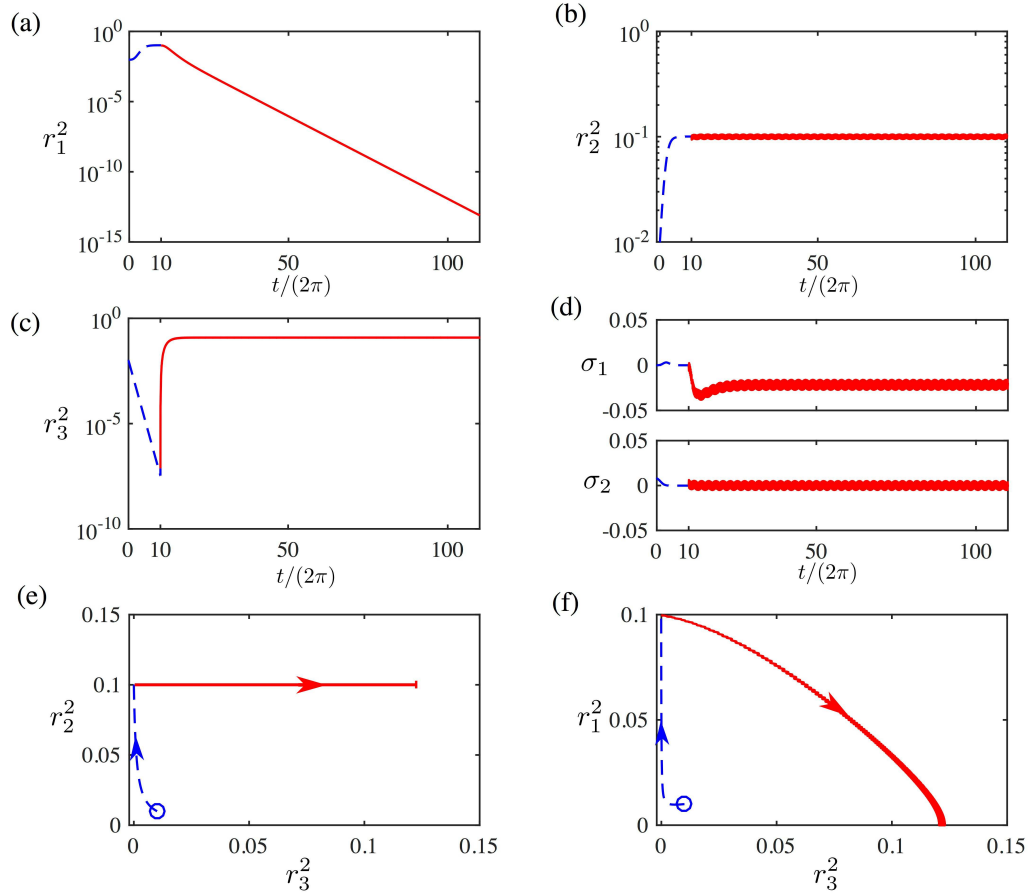


FIG. 4. Dynamics of the model system (3.1) with the optimal periodic forcing $b^\circ(t) = 0.07 \sin(\pi^2 t)$ applied at $t/(2\pi) \geq 10$. Unforced state: blue dashed line; forced state: red line. (a-d) Time evolution of r_1^2 , r_2^2 , r_3^2 , σ_1 and σ_2 . Only the first 110 periods are shown here for clarity. (e) Phase portrait of r_2^2 against r_3^2 and (f) r_1^2 against r_3^2 .

Fig. 4(f) shows clearly that r_1^2 decreases with the increase of r_3^2 , corroborating that a high-frequency forcing stabilizes a low-frequency unstable oscillator via frequency crosstalk.

3.3. Results of LGPC

LGPC is applied to solve the control problem of Section 3.1. For all LGPC tests, up to $N = 50$ generations with $M = 500$ individuals in each are evaluated. Hereafter, we denote the cost value of the m th individual in the n th generation by J_m^n ($m = 1, \dots, M; n = 1, \dots, N$). After generating the individuals, each is pre-evaluated based on the state \mathbf{a} of the unforced system. The resulting actuation command is an indicator for their feedback control performance. If no actuation ($b = 0, \forall t$) is obtained in the pre-evaluation, this individual cannot change the unforced state. As a consequence, the individual is not subjected to a testing and is assigned a high cost value. This pre-evaluation step saves numerical testing time.

The parameters of linear genetic programming are similar to those of most GPC studies (see, e.g. the textbook [4]). Elitism is set to $N_e = 1$, i.e. the best individual of a generation is copied to the next one. The probabilities for replication, crossover and mutation are 10%, 60% and 30%, respectively. The individuals on which these genetic operations are performed are determined from a tournament selection of size $N_t = 7$. The instruction number in the initial generation is selected between 2 to 30 with a uniform probability distribution. In the following generations, the maximum instruction number for each individual is capped by 100. Elementary operations comprise $+$, $-$, \times , \div , \sin , \cos , \tanh and \ln . The operation \div and \ln are protected, i.e. the absolute value of the denominator of \div is set to 10^{-2} when $|x| < 10^{-2}$. Similarly, $\ln(x)$ is modified to $\ln(|x|)$ where $|x|$ is set to 10^{-2} when $|x| < 10^{-2}$. In addition, we choose six random constants in the range $[-10, 10]$ with the uniform probability distribution.

In the following, we introduce successively the results of open-loop multi-frequency forcing LGPC-1 (Section 3.4), full-state feedback control LGPC-2 (Section 3.5) and non-autonomous control LGPC-3 (Section 3.6).

3.4. LGPC-1

First, we search for generalizing the open-loop control by including the best periodic forcing at all eigenfrequencies, i.e. $b = K(\mathbf{h})$ where $\mathbf{h} = (h_1, h_2, h_3) = (\sin(t), \sin(\pi t), \sin(\pi^2 t))$. This approach, called LGPC-1, contains the best periodic forcing frequency $\omega^\circ = \pi^2$, thus it should be at least as good as the optimal periodic forcing b° . Figure 5 displays the ‘spectrogram’ of the cost values for the whole collection of control laws. Each generation n is seen to consist of a large range of cost values. The decreasing J values towards the right bottom with increasing generation evidences the learning of increasingly better control laws.

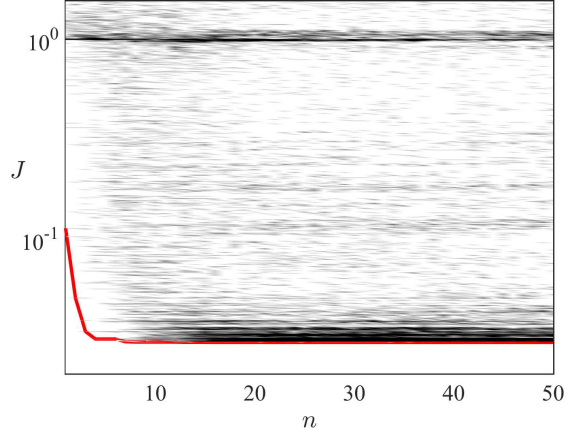


FIG. 5. ‘Spectrogram’ of all computed J_m^n ($m = 1, \dots, M; n = 1, \dots, N$) for LGPC-1. For each generation n , J_m^n is ordered with respect to their cost $J_1^n \leq J_2^n \leq \dots \leq J_M^n$. The color shows the distribution of cost values. Darker color indicates larger proportion. The red line highlights the best cost value of each generation J_1^n .

The best cost value of each generation is highlighted by a red line. The best individual ($m = 1$) in the last generation ($n = 50$) reads

$$(3.3) \quad b^\odot(t) = 0.37 \sin(0.18 \sin(\pi^2 t)).$$

Here, and in the following, the superscript ‘ \odot ’ refers to LGPC-1. When applying a first order approximation on b^\odot , we get $b^\odot(t) \approx 0.067 \sin(\pi^2 t)$. This expression resembles that of the optimal periodic forcing $b^\circ(t) = 0.07 \sin(\pi^2 t)$, and leads to a slightly better cost $J^\odot = 0.03$ as a better amplitude with a higher precision is explored by LGPC-1. The dynamics of the system with b^\odot are similar to Fig. 4 and are not shown here for brevity.

If we increase the precision of B to 0.001 in the parameter scan of the periodic forcing in Section 3.2, we should find the same result. However, the number of evaluations raises to $N_B \times N_\omega = 1001 \times 401 = 401000$ (N_B and N_ω being the number of the amplitudes and frequencies to be tested, respectively) which is 16 times that of LGPC-1 which equals $M \times N = 500 \times 50 = 25000$. In summary, LGPC-1 identifies automatically the optimal frequency $\omega^\odot = \pi^2$ and the optimal amplitude $B^\odot = 0.067$ by employing less time than that for the periodic forcing with an exhaustive parameter sweep.

3.5. LGPC-2

Next, an autonomous full-state feedback law (LGPC-2) is optimized,

$$b = K(\mathbf{a}) = K(a_1, a_2, a_3, a_4, a_5, a_6).$$

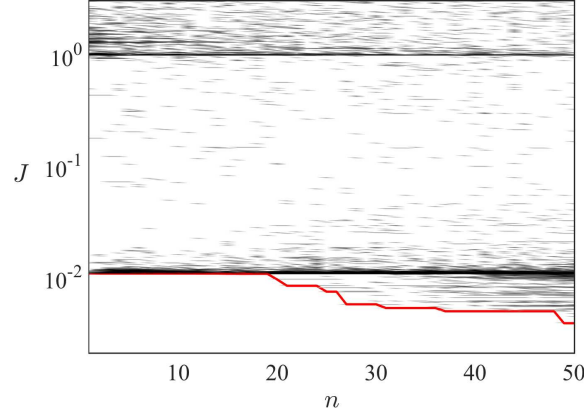


FIG. 6. Same as Fig. 5, but for LGPC-2.

The ‘spectrogram’ of the cost values is shown in Fig. 6. The successive jumps of the best cost value for each generation (red line) reflect the evolution process to better individuals. The targeted LGPC-2 feedback law, i.e. the best individual in the last generation, reads as follows:

$$(3.4) \quad b^{\blacksquare} = \tanh \left(\sin \left(\tanh \left(\tanh \left(\tanh \left(\left(\ln(a_4) + \frac{5.8}{\frac{a_6}{1-a_6}} a_4 \right) \right) \right) \right) \right) \right).$$

Here, and in the following, the superscript ‘ \blacksquare ’ refers to LGPC-2. The corresponding cost $J^{\blacksquare} = 0.0038$ is more than seven times better than the value achieved with optimal open-loop control b° . Closed-loop control b^{\blacksquare} leads to both, a smaller fluctuation level J_a and a lower actuation energy J_b . The corresponding dynamics are depicted in Fig. 7.

Instead of the regular excitation of periodic forcing, Fig. 7(a) shows that b^{\blacksquare} gives a strong initial ‘kick’ on the system by exciting the third oscillator to a high energy level of $r_3^2 = 0.5$ (see Fig. 7(d), (f) and (g)), while simultaneously stabilizing the second oscillator, $r_2^2 \approx 0$ (see Fig. 7(c) and (f)). The first oscillator exhibits consequently a fast decay as σ_1 has decreased to $\sigma_1 = -0.5$ due to the change in r_2^2 and r_3^2 (see Fig. 7(b), (e) and (g)). This fast transient takes about one period $\Delta t = 2\pi$, see the close view of forcing b in Fig. 7(a). It should be emphasized that LGPC-2 discovers and exploits both frequency crosstalk mechanisms, the excitation of the third oscillator for a quick transient and the suppression of the second oscillator to sustain the low fluctuation level of the target dynamics.

Following this fast transient, the first and second oscillators enter into a quasi-stable state at nearly vanishing fluctuation levels. Subsequently, the control command vanishes as the full-state feedback shows no need to actuate after the

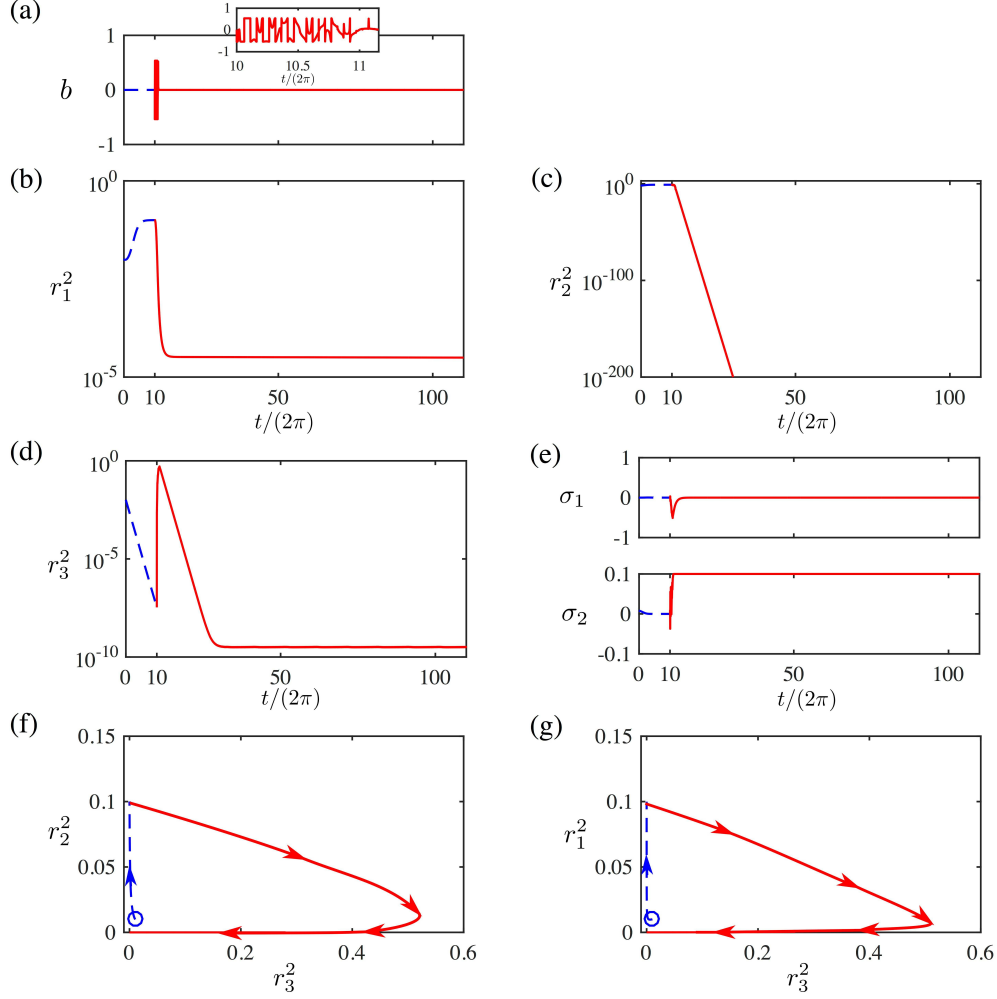


FIG. 7. Dynamics of the dynamical system (3.1) with the LGPC-2 control b^\blacksquare applied at $t/(2\pi) \geq 10$. Unforced state: blue dashed line; forced state: red line. (a-e) Time evolution of b , r_1^2 , r_2^2 , r_3^2 , σ_1 and σ_2 . Only the first 110 periods are shown here for clarity. (f) Phase portrait of r_2^2 against r_3^2 and (g) r_1^2 against r_3^2 .

energy is defeated. With vanishing b , the third oscillator decays exponentially fast. This transient process converges to the fixed point as depicted in Fig. 7(f) and (g). Now, the first oscillator has a stabilizing growth rate $\sigma_1 \approx -r_1^2$. LGPC-2 shows an example of feedback control better than the open-loop control. With only a tiny investment of actuation energy at the very beginning of the control, the whole system remains stabilized without actuation even after thousands of periods.

It should be noted that closed-loop control is not necessarily better than open-loop actuation. Suppose the growth-rate of the first oscillator reads

$$(3.5) \quad \sigma_1 = 0.1 - r_1^2 + r_2^2/100 - r_3^2.$$

In this case, exciting the third oscillator is the only effective stabilizing mechanism and this excitation can already be performed with the open-loop forcing.

3.6. LGPC-3

Finally, we explore a more general class of control laws which combines full-state feedback \mathbf{a} and the best periodic forcing at all eigenfrequencies $\mathbf{h} = (\sin(t), \sin(\pi t), \sin(\pi^2 t))$, as discussed in Section 2. Then, the generalized LGPC-3 control law $b = K(\mathbf{a}, \mathbf{h})$ includes the pure full-state feedback and the best periodic forcing frequency ω° . Hence, it should be at least as good as LGPC-2. The learning process is similar to Fig. 6, thus we do not show the convergence of cost values here for brevity. The optimal control law from LGPC-3 reads

$$(3.6) \quad b^\bullet(t) = \tanh \left(\sin \left(\tanh \left((3a_2 \sin(t) \sin(\pi^2 t) - a_4) \right) \right) \right).$$

Here, and in the following, the superscript ‘ \bullet ’ refers to LGPC-3 results. This control law achieves a better cost value $J^\bullet = 0.0025$ compared to LGPC-1 with similar dynamics. Hence, the results are not detailed here to avoid redundancies.

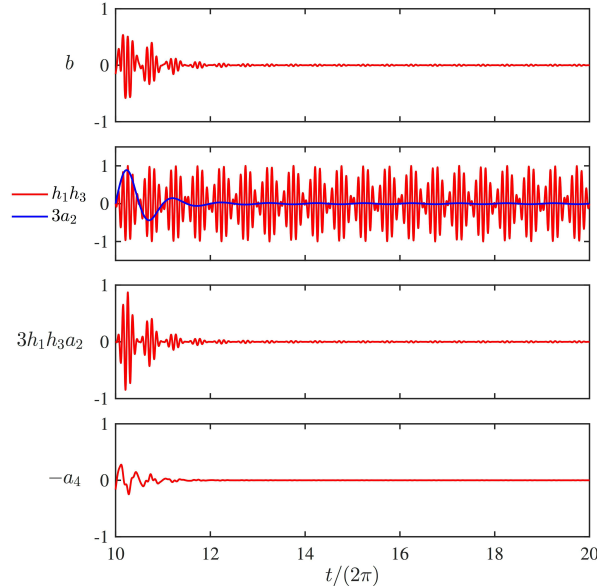


FIG. 8. Time evolution of b^\bullet , $h_1 h_3$, a_3 and a_4 .

It is worth to note that Eq. (3.6) can also be expressed as $b^\bullet = K_1(3a_2h_1h_3 - a_4)$ where K_1 represents the operator ‘ $\tanh(\sin(\tanh(\cdot)))$ ’. To shed light on the contribution of each term to b^\bullet , Fig. 8 displays the temporal evolution of the actuation command b^\bullet and the relevant input from the states and from the harmonic functions. It shows that the harmonic component h_1h_3 destabilizes the stable oscillator by a quasi-periodic forcing while the states a_2 and a_4 act as an amplitude regulator.

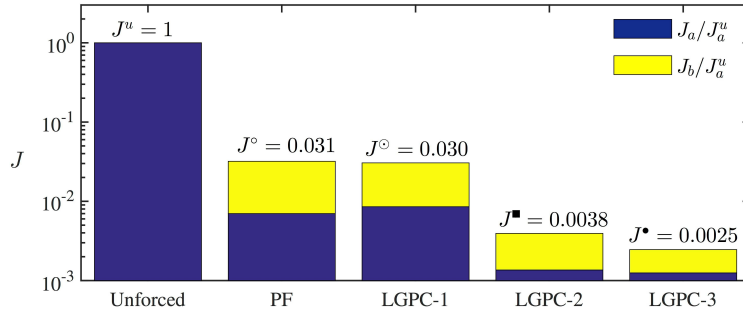


FIG. 9. Synthesis of J for different controls.

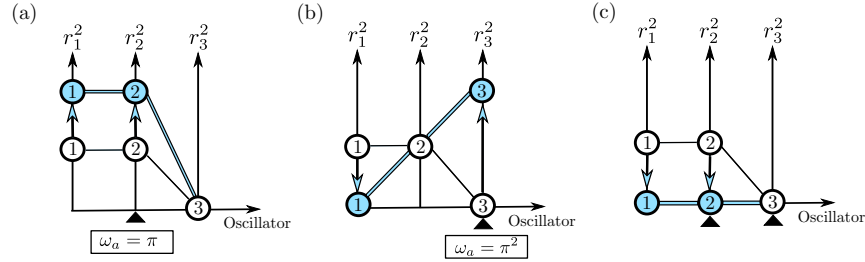


FIG. 10. Synthesis of system dynamics under the forcing. The energy level of each oscillator is qualitatively indicated by circles. Unforced state: white circles connected by black line; forced state: colored circles connected by colored line. (a) Open-loop forcing at actuation frequency $\omega_a = \pi$. (b) Open-loop forcing at actuation frequency $\omega_a = \pi^2$. (c) Feedback control. The triangles indicate the oscillator(s) contributing to alter the first oscillator. The arrows show the transition state when control is applied.

To summarize, optimal periodic forcing (PF), open-loop multi-frequency forcing (LGPC-1), full-state feedback (LGPC-2), and generalized feedback (LGPC-3) are compared. The contributions to the cost function are depicted in Fig. 9, showing that the generalized feedback outperforms optimal periodic forcing and full-state feedback. The stabilizing mechanisms are schematically depicted in Fig. 10.

4. Drag reduction using LGPC

In this section, we apply LGPC to a turbulence control experiment targeting the drag reduction of a simplified car model. Given that the drag of a ground vehicle is dominated by pressure drag, we aim to increase the base pressure and thus reduce the drag. For that, active control is applied on the wake flow using fluidic actuators. Unlike three-oscillator dynamical system before, no model is established here for the three-dimensional turbulent wake. The modeling of such a flow constitutes a significant challenge even without forcing [21]. Here, we simply apply LGPC to search automatically effective nonlinear control laws within few hours of testing time. In the following, the experimental setup is presented in Section 4.1. The implementation and results of LGPC are discussed in Section 4.2. Section 4.3 illustrates the effect of the optimal forcing on the near wake dynamics.

4.1. Experimental setup

A sketch of the experimental setup is shown in Fig. 11. The experiment is performed in a closed-circuit wind tunnel, the test section of which is $2.4 \text{ m} \times 2.6 \text{ m} \times 6 \text{ m}$. The model is similar to the square-back Ahmed body [22] and has the following dimensions: height $H = 0.297 \text{ m}$, width $W = 0.350 \text{ m}$ and length $L = 0.893 \text{ m}$. The ground clearance is set to $G = 0.05 \text{ m} \approx 0.17H$ as in [22]. The experiment is conducted with the constant free-stream velocity $U_\infty = 15 \text{ m} \cdot \text{s}^{-1}$ corresponding to the Reynolds number $Re_H = U_\infty H / \nu = 3 \times 10^5$. The actuator consists of four independent actuation slits along the trailing edges, as shown in Fig. 11(b). The slit thickness is $h_{\text{slit}} = 1 \text{ mm} \approx 0.003H$. Pulsed jets, which are driven by 32 solenoid valves and supplied by a compressed air reservoir (see Fig. 11(a)), are blown tangentially to the free-stream velocity through these slits. The solenoid valves are distributed homogeneously along the trailing edges, and the zone between the valves and the slit exit is specifically designed so that the

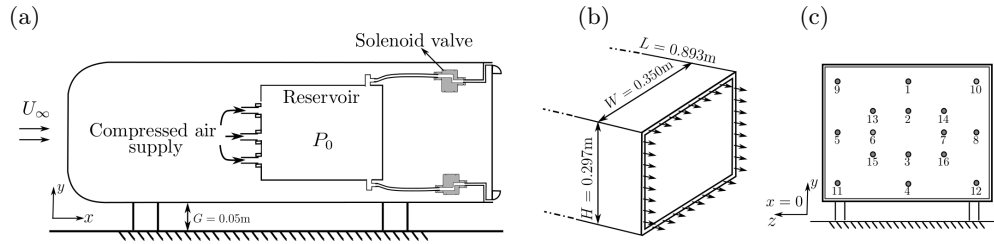


FIG. 11. Experimental setup. (a) A slice of the model illustrating the actuation setup. (b) Side-view of pulsed jets. (c) Distribution and number of pressure sensors over the base surface. The first 12 pressure sensors are used for the feedback. x, y, z represent the streamline, wall-normal and spanwise directions respectively.

exiting flow is continuous along the periphery of four edges. The solenoid valve generates pulsed jet in ON/OFF mode within the frequency range [0,500] Hz. Besides, a rounded surface of radius $9h_{\text{slit}}$ is installed immediately beneath each slit as an additional passive device, as shown in Fig. 11(a). In the present study, all actuation slits are controlled simultaneously. The actuation command b is binary. The valves are closed at $b = 0$ and open at $b = 1$. The flow is monitored by 16 pressure sensors distributed over the base surface, 12 of which are used as feedback sensors, see Fig. 11(c). Particle Image Velocimetry (PIV) is performed to capture the flow dynamics in the near wake and to identify the control effects. The measured plane is the vertical (normal to ground) symmetry plane downstream the base. The first and second order statistics of the streamwise (along x) and cross-stream (along y) velocity are computed based on 1000 images with a spatial resolution of 0.8% of the model's height. For more details on the experimental setup, see [23].

4.2. Results of LGPC

In the following, we apply LGPC on the plant for the purpose of increasing the base pressure. We define the cost function J as

$$(4.1) \quad J = \frac{C_{p_b}^a}{C_{p_b}^u},$$

where $C_{p_b}^a$ and $C_{p_b}^u$ represent the time- and area-averaged base pressure coefficients in the actuated and unforced flow, respectively. For estimating these quantities, all the pressure sensors in the base surface are used. By definition, $J = 1$ for the unforced flow. $J < 1$ ($J > 1$) represents the increase (decrease) of the base pressure.

The included sensors are $\mathbf{s}' = [s'_1, \dots, s'_{12}]$, where s'_i is the fluctuating component of the i th pressure sensor signal. As the control command b is binary, we apply the Heaviside function H to transform the continuous output of a control law to a binary output, i.e. $b = H(K(\mathbf{s}'))$ where $H(x) = 0$, if $x \leq 0$; $H(x) = 1$, otherwise. The control law is evaluated for a time period of $T = 10$ s. This value is approximately 500 convective time units defined by H/U_∞ . This period has been found to be sufficient for good statistical accuracy [18].

Before testing LGPC, a preliminary periodic forcing is performed as benchmark. The tested frequencies and duty cycles (DC) are constrained by the sampling rate $F_s = 2$ kHz, see details in [18]. The optimal periodic forcing b° is found at $St_H^\circ = f^\circ H/U_\infty = 6.6$ and $DC^\circ = 33\%$, resulting in $J = 0.66$ and increasing the base pressure by 33%. Hereafter, this optimal periodic forcing will be used as reference.

First, we explore the open-loop multi-frequency control (LGPC-1) optimizing the frequency combination. Let \mathbf{h} comprise 9 harmonic functions $h_i(t) =$

Table 1. Harmonic functions $h_i(t) = \sin(2\pi f_i t)$ used as inputs of LGPC-1.

Controller input	h_1	h_2	h_3	h_4	h_5	h_6	h_7	h_8	h_9
f_i (Hz)	10	20	50	100	200	250	333	400	500
St_{H_i}	0.2	0.4	1	2	4	5	6.6	8	10

$\sin(2\pi f_i t)$, $i = 1, \dots, 9$ listed in Table 1. In this case, the control law reads $b = H(K(\mathbf{h}))$. Up to $N = 4$ generations with $M = 50$ individuals in each are evaluated. We stop at the fourth generation because half of the individuals have similar J values near the optimal one. The optimal control law reads:

$$(4.2) \quad b^\odot = H(h_5/h_8 - 0.622).$$

The resulting cost $J^\odot = 0.65$ beats the optimized periodic forcing, leading to 35% base pressure recovery associated with 22% drag reduction. The actuation energy defined by the time-averaged momentum of pulsed-jets is about 7% for both control laws. The optimal control law contains two frequencies, indicating that LGPC-1 explores a multi-frequency forcing which outperforms the reference periodic forcing.

The results for LGPC-2, $b = H(K(\mathbf{s}'))$, have been discussed in an earlier study [18] and are not shown here. Intriguingly, LGPC-2 provides a sensor optimization by reproducibly selecting only one sensor s'_4 near the centre of bottom edge in the optimal control law. The corresponding control emulates the optimal high-frequency periodic forcing but is slightly worse ($J^\blacksquare = 0.72$). A similar observation has been made for stabilization of the mixing layer [3], where the optimized high-frequency periodic forcing has outperformed GPC-optimized sensor-based feedback in stabilizing the flow. At high frequencies, time delays and noise in sensor-based feedback give rise to low-frequency actuation components which are detrimental to the cost function. We could even change the dynamical system (3.1) to have an unbeatable periodic forcing, as discussed at the end of Section 3.5.

Finally, a test of the generalized non-autonomous control LGPC-3 is performed by combining the sensors \mathbf{s}' and the optimal harmonic forcing $h^\circ(t) = \sin(2\pi f^\circ t)$, i.e. $b = H(K(\mathbf{s}', h^\circ))$. Up to $N=5$ generations are evaluated with 50 individuals in each generation. LGPC-3 converges to the optimal periodic forcing b° from the second generation and no improvement is observed in subsequent generations. The finding is in agreement with the LGPC-2 result where the optimal control emulates the optimal periodic forcing but is slightly worse. LGPC-3 prefers to select the optimal periodic forcing to the sensor feedback. Upon these results, we do not pursue LGPC-3 $b = H(K(\mathbf{s}', \mathbf{h}))$ by including multiple frequencies in this experiment. We assume the result will be the same with LGPC-1.

In summary, LGPC identifies an open-loop multi-frequency forcing as the best control for drag reduction. The underlying dynamics is presented in the following section. Note that this control has been identified by testing only 200 individuals in less than one hour. The required optimization time is less than that for finding the best frequency and duty cycle for the periodic reference with a thorough parameter scan.

4.3. Near wake dynamics of LGPC-1

In this section, we investigate the impact of the best control b^\odot from LGPC-1 on the near wake dynamics. To illustrate the actuation characteristics of b^\odot , Fig. 12 displays (a) its phase-averaged jet velocity over one period and (b) its power spectral density S_b . The results of b° are also presented for comparison. Intriguingly, b^\odot exhibits a multi-frequency dynamic, showing two frequencies at $St_H = 4$ and $St_H = 8$, respectively.

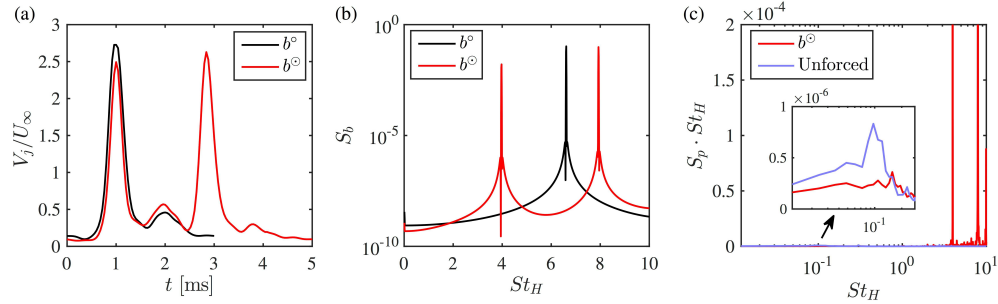


FIG. 12. (a) Phase-averaged jet velocity V_j for the optimal periodic forcing b° and the optimal LGPC-1 control b^\odot . (b) Power spectral density S_b of b° and b^\odot . (c) Power spectral density S_p of the area-averaged pressure coefficient.

It has been reported that forcing at frequencies several times that of the natural vortex shedding can stabilize the wake fluctuations by inducing large dissipation and inhibiting the entrainment of fluid into the recirculation region [23–25]. Here, LGPC-1 exploits similar actuations in an unsupervised manner. The actuation frequencies in b^\odot are one order of magnitude larger than that of the natural vortex shedding frequency $St_H^{vs} = 0.2$. The impact of the actuation on the wake dynamics can be further inferred from the base pressure fluctuation. We use the area-averaged base pressure coefficient $\langle C_p \rangle$ as a global indicator of the dynamics. Figure 12(c) compares the spectral energy of $\langle C_p \rangle$ for the unforced and optimal forced flows, where S_p represents its power spectral density. The high-frequency forcing has two major effects: (1) it significantly excites the frequencies over $St_H > 1$, and (2) it suppresses a range of frequencies below $St_H < 0.2$.

The high level of energy around $St_H = 0.1$ in the unforced flow is associated with the bubble pumping frequency, which is induced by an axial oscillation of the recirculation bubble [26]. It seems that the damping of this pumping mode contributes to reduce the drag. The benefit in drag reduction by the suppression of this mode has been also observed in [27]. This result is a good illustration of the frequency crosstalk between low- and high-frequencies, and corroborates the mechanisms proposed in [24].

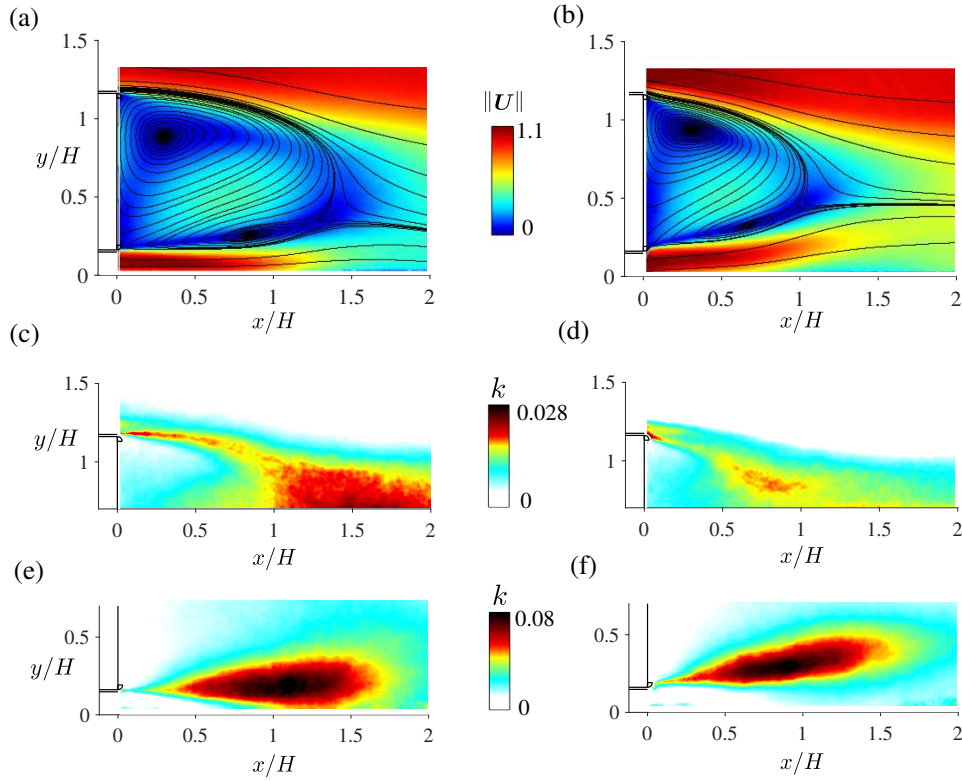


FIG. 13. Near wake dynamics for the unforced baseline flow (a, c, e) and forced flow (b, d, f). (a, b) Time-averaged velocity norm $\|\mathbf{U}\|$ and 2D streamlines; (c, d) 2D estimation of the turbulent kinetic energy k for the upper shear layer; (e, f) k for the lower shear layer.

Now, we focus on the effects of the best LGPC-1 control b^\odot on the wake dynamics identified from the PIV measurements. Figure 13 shows the color map of the time-averaged velocity norm $\|\mathbf{U}\| = \sqrt{\bar{u}^2 + \bar{v}^2}$ overlapped with 2D streamlines (a, b) and 2D estimation of the turbulent kinetic energy $k = \frac{1}{2}(\overline{u'^2} + \overline{v'^2})$ (c-f) for the baseline (a,c,e) and controlled flow (b,d,f). \bar{u} and \bar{v} represent the time-averaged streamwise and cross-stream velocity, respectively; u' and v' are

their corresponding velocity fluctuations. The values of these quantities are normalized by U_∞ .

The mean wake of the baseline flow consists of two counter-rotating structures with a very low velocity inside, leading to a recirculating bubble extending up to $L_r/H \approx 1.42$, where $L_r = \max_x(\bar{u}(x) = 0)$ denotes the bubble length. The upper recirculating structure dominates the wake and results in an asymmetry in the cross-stream direction. The distribution of k is concentrated in the shear layers, indicating its important role in the wake dynamics. In addition, higher values of k are noticeable at the lower shear layer near the ground which corroborates the asymmetry observed above. Such asymmetry is ascribed to the presence of ground as a perturbation.

The forcing induces significant changes in the wake. First, the shear layers are highly deviated toward the model base, resulting in a thinner and shorter recirculation bubble, the length of which is $L_r/H \approx 1.06$, reduced by 25% compared with the baseline flow. The vectorization of the shear layer modifies the streamline curvature near the separation and thus results in a local rise in base pressure. Secondly, the vectorization of shear layers is accompanied by an overall reduction of turbulent kinetic energy inside the recirculation bubble, which can be qualitatively observed in Fig. 13(d) and (f). The injection of high-frequency jet structures enhances the interaction of the small- and large-structures, and thus diminish the turbulent dynamics [23], resulting a more stabilized wake. In summary, it is the combined effect of shear layer deviation and wake stabilization that lead to ultimately a base pressure recovery and drag reduction.

5. Visualization of control laws

In this section, we illustrate the control laws and cost function values by an easily interpretable 'topological landscape', generalizing earlier work [28]. First (Section 5.1), the visualisation technique is described, employing a control-law distance metric and multidimensional scaling for feature extraction. Then, (Section 5.2), the LGPC laws for the dynamical system and the turbulence control experiment are depicted.

5.1. Multidimensional scaling

LGPC systematically explores the control law space by generating and evaluating a large number of control laws from one generation to the next. An assessment of the similarity of control laws gives additional insights into their diversity and convergence to optimal control laws, i.e. into the explorative and exploitative nature of LGPC. For that purpose, we rely on Multidimensional

Scaling (MDS) [29], a method classically used to visualize abstract data in a low-dimensional space. The main purpose of MDS is to visualize the (dis)similarity of objects or observations. MDS comprises a collection of algorithms to detect a meaningful low-dimensional embedding given a dissimilarity matrix. Here, we employ Classical Multidimensional Scaling (CMDS) which originated from the works of [30] and [31].

Let us define N_K as the number of objects to visualize, and $\mathbf{D} = (D_{ij})_{1 \leq i, j \leq N_K}$ as a given distance matrix of the original high-dimensional data. The aim of CMDS is to find a centred representation of points $\mathbf{\Gamma} = [\gamma_1 \ \gamma_2 \ \dots \ \gamma_{N_K}]$ with $\gamma_1, \dots, \gamma_{N_K} \in \mathbb{R}^r$, where r is typically chosen to be 2 or 3 for visualization purposes, such that the pairwise distances of the points approximate the true distances, i.e. $\|\gamma_i - \gamma_j\|_2 \approx D_{ij}$.

We choose to visualize all control laws in a two-dimensional space $r = 2$. Thus, the number of objects is $N_K = M \times N$, where M is the number of individuals in a generation, and N is the total number of generations. The distance between two control laws b_i and b_j , $i, j \in \{1, \dots, N_K\}$ shall measure their ‘effective difference’. Let us consider the non-autonomous feedback $b_i = b_i(\mathbf{s}_i, \mathbf{h}_i)$. Here, $\mathbf{s}_i(t)$ are the sensor readings collected when b_i is applied and $\mathbf{h}_i(t)$ are the harmonic arguments of the control law b_i . The squared difference between b_i and b_j is defined as

$$(5.1) \quad D_{ij}^2 = \frac{\langle |b_i(\mathbf{s}_i(t), \mathbf{h}_i(t)) - b_j(\mathbf{s}_i(t), \mathbf{h}_i(t))|^2 + |b_i(\mathbf{s}_j(t), \mathbf{h}_j(t)) - b_j(\mathbf{s}_j(t), \mathbf{h}_j(t))|^2 \rangle}{2} + \alpha |J_i - J_j|.$$

The time average $\langle \cdot \rangle$ is taken over the evaluation time interval. Thus, D_{ij}^2 represents the difference between the i th and j th control law in an average sense evaluated in the relevant sensor space. The permutation of control laws b_i and b_j with its arguments guarantees that the distance matrix is symmetric. More importantly, this ensures that the control laws are compared in the relevant sensor space with an equal probability of both forced attractors.

The second term in (5.1) penalizes the difference of their achieved costs J with the coefficient α . The penalization coefficient α is chosen as the ratio between the maximum difference of two control laws (first term of D_{ij}^2) and the maximum difference of the cost function (second term of D_{ij}^2). Thus, the dissimilarities between control laws and between the cost functions have comparable weights in the distance matrix D_{ij} . This penalization evidently smoothes the control landscape $J(\gamma)$.

A problem may arise for the comparison of two pure open-loop forcings b_i and b_j . We expect, for instance, that $b_i = \cos t$ and $b_j = \sin t$ give rise to the

same actuation response modulo a time shift $\tau = \pi/2$ and would consider these control laws as equivalent. Even for sensor-based feedback enriched by a harmonic input, we expect the actuation response to be 'in phase' or synchronized with the harmonic input. This expectation is taken into account by minimizing the difference between two control commands modulo a minimizing time shift:

$$(5.2) \quad D_{ij}^2 = \min_{\tau} \frac{1}{2} \left(\langle |b_i(\mathbf{s}_i(t), \mathbf{h}_i(t)) - b_j(\mathbf{s}_i(t - \tau), \mathbf{h}_i(t - \tau))|^2 + |b_i(\mathbf{s}_j(t), \mathbf{h}_j(t)) - b_j(\mathbf{s}_j(t - \tau), \mathbf{h}_j(t - \tau))|^2 \rangle \right) + \alpha |J_i - J_j|.$$

Evidently, (5.1) and (5.2) coincide at $\tau = 0$.

Summarizing, the square of the distance matrix $\mathbf{D}^2 = (D_{ij}^2)$ is defined as follows:

- (1) If both control laws have a non-trivial harmonic input (are non-autonomous), (5.2) defines the distance.
- (2) Otherwise, (5.1) is employed.

Applying CMDS to the distance matrix \mathbf{D} , each control law b_i is associated with a point $\gamma_i = (\gamma_{i,1}, \gamma_{i,2})$ such that the distance between different γ_i emulates the distance between control laws defined by (5.1) and (5.2). More generally, γ_i are feature vectors which coefficients represent those features that contribute most on average to the discrimination of different control laws.

5.2. Control landscapes for the LGPC runs

Figure 14 visualizes the control laws determined by LGPC-3 for the three-oscillator model (a), and LGPC-1 for the simplified car model (b). Due to the huge number of control laws in the three-oscillator model ($N_K = 500 \times 50 = 25000$), we present every 10th individual in every 10th generation for clarity. The full ensemble of individuals are shown for the simplified car model as its number is moderate ($N_K = 50 \times 4 = 200$). Each symbol represents a control law which is color-coded with respect to its performance ranking, for instance the dark color represents the best 10% of the presented control laws. The control laws in the first generation cover a significant portion of the control space, like in a Monte-Carlo search. When the value of n increases, we observe a global movement of control laws towards the minimum where better performance is obtained (darker color). Moreover, the distances between control laws of different generations are also decreased resulting in a dense distribution. This is illustrated in Fig. 14(a) where the inserted figure gives a close view of the control laws near the origin point, where the best control law(s) are found at $[\gamma_1, \gamma_2] \approx [-0.18, 0.02]$. These observations show that LGPC has effectively explored the control space, evidenced

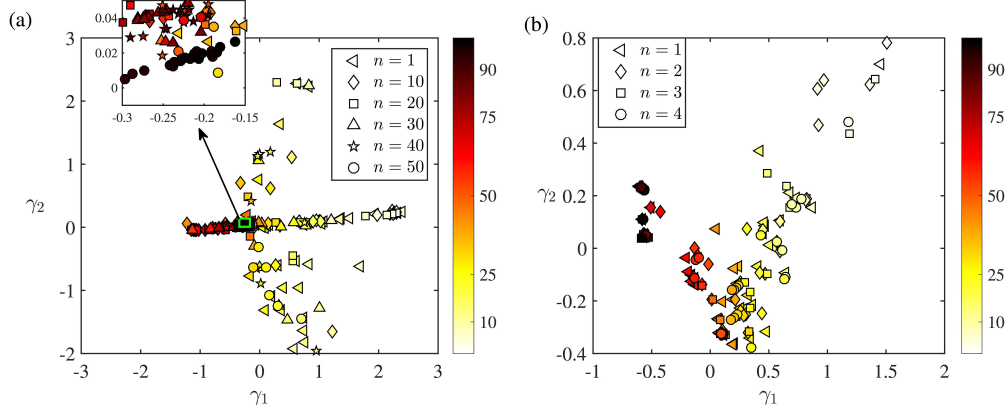


FIG. 14. Visualization of the control laws obtained for (a) the three-oscillator model by LGPC-3 (Sec. 3); and (b) the simplified car model by LGPC-1 (Section 4); n represents the generation number. The color scheme corresponds to the percentile rank of the control laws with respect to their performance J . Darker color presents a better performance. The control law b_i is presented by the point $\gamma = (\gamma_1, \gamma_2)$. The distance between two control laws, i.e. two points, approximates their respective dissimilarity.

by the extended distribution of control laws. In summary, the visualization provides not only a simple and revealing picture of the exploration and exploitation characteristics of the control approach, but also inspires further improvement of the methodology.

6. Conclusions and outlook

We have demonstrated that *linear genetic programming control (LGPC)* is a simple yet effective model-free control strategy for strongly nonlinear dynamics with frequency crosstalk. This nonlinearity is a considerable challenge for the model-based control design due to the difficulties in the corresponding mathematical modeling and limited knowledge about the flow in experiments. LGPC is shown to discover and exploit the most effective nonlinear open- and closed-loop control mechanisms in dynamical systems and turbulence control experiments in an automated unsupervised manner without any model or knowledge of the plant.

Three categories of LGPC are investigated in this work: an open-loop multi-frequency control $b = K(\mathbf{h})$, named LGPC-1, an autonomous sensor-based feedback control $b = K(\mathbf{s})$, termed LGPC-2, and a generalized non-autonomous control $b = K(\mathbf{s}, \mathbf{h})$ comprising the sensors \mathbf{s} and time-periodic functions \mathbf{h} , called LGPC-3. All of them are successfully applied to the stabilization of a forced

nonlinearly coupled three-oscillator model (Section 3). The obtained control laws stabilize the first unstable oscillator by exploiting two frequency crosstalk mechanisms: (1) the excitation of the third oscillator by a hard 'kick' for a quick transient and (2) the suppression of the second oscillator to sustain the low fluctuation level of the target dynamics. Following the quick transient, the first and second oscillators enter into a quasi-stable state at nearly vanishing fluctuation levels. Hence, the full-state feedback hardly needs to actuate and the control command starts to vanish. The whole system is stabilized with only a small investment of the actuation energy at the very beginning of the control. Thus, LGPC laws show a performance over the optimal open-loop control as both a lower fluctuation level and a lower actuation energy are obtained. The explored control demonstrate the vital importance of frequency crosstalk for control design.

LGPC is applied to a turbulence control experiment targeting drag reduction of a car model (Section 4). It finds that multi-frequency forcing beats optimized periodic forcing by 22% over 19%, the current benchmark for this car configuration. This performance increase of 3% pays for almost half of the invested actuation energy. Perhaps surprisingly, the maximum actuation frequency is about 33 times that of the von Kármán vortex shedding. This high-frequency forcing leads to a broadband suppression at very low frequencies of base pressure signals and a global attenuation of averaged and turbulent kinetic energy in the near wake, resulting in a more stabilized wake. On the other hand, the mean wake geometry is modified such that the shear layers are deviated towards the center, resulting in a shorter, narrower, more stream-lined shaped bubble. The drag reduction is ultimately achieved by the combined effect of the wake stabilization and the shear layer deviation and can legitimately be called *fluidic boat tailing*.

One of the many benefits of LGPC is that it explores automatically the control space with little or no prior knowledge of the system being controlled. Moreover, LGPC-3 for the control law can make the evolutionary algorithm choose between sensor-based feedback, multi-frequency forcing and combinations thereof. In addition, the number of control laws evaluations for the Ahmed body drag reduction was quite comparable to a single frequency optimization but yields a much more general multi-frequency actuation which is hardly accessible to a parametric study. In an even more general framework, noise signals \mathbf{n} could also be included in the control law arguments, leading to $\mathbf{b} = \mathbf{K}(\mathbf{s}, \mathbf{h}, \mathbf{n})$. Thus stochastic forcing and its generalizations are included. Another generalization is the use of temporal filters as considered operations. In [4], a filter-enriched GPC has successfully discovered the optimal linear quadratic gaussian control for the stabilization of a noise-driven oscillator. In summary, LGPC can work in a search space which includes in principle any perceivable control logic with finite amount of operations.

Visualization of the ensemble of the control laws in a two-dimensional plane sheds light on the explorative and exploitative nature of LGPC, and thus addresses the need to monitor the search space and guide the improvement of the algorithm. The example given in Fig. 14 indicates clearly the search space topology and distills the local extrema in this feature space. Evidently, in a future development of LGPC, this feature space has been shown to estimate the cost function of an untested control law [32] and may be used to avoid the redundant testing of control laws in unpromising terrain. Thus, experimental testing time can be reduced. The visualization is becoming an important component of LGPC for on-line decisions during a control experiment. Further increases in the learning rate are expected from incorporating over 50 years of experience in evolutionary algorithms [33, 34].

The authors currently improve the LGPC methodology, and pursue car model experiments for reducing the drag and yaw moment during cross-wind gusts. LGPC opens refreshingly new paths in fluid mechanics, as estimation, prediction and control tasks are all regression problems minimizing a cost function. LGPC exploits that control is a mapping from the plant sensors (output) to actuations (input) optimizing aerodynamic or other goals. Prediction is the mapping from the state to its time derivative or future state. And estimation maps sensor signals to flow fields. Evidently all these tasks can be solved with LGP. Moreover, a single LGPC run yields already rich actuation response data for the computation of a control-oriented nonlinear black-box model [35]. Soon, we will see turbulence control experiments in which control laws, dynamical models and flow estimations are learned in few hours of wind tunnel testings—instead weeks or months’ time of parameter studies or model development. LGPC and, more generally, machine learning control [4] can be expected to be a game changer in future flow control and in fluid mechanics in general.

Acknowledgements

We warmly thank the great support during the experiment by J.-M. Breux, J. Laumonier, P. Braud and R. Bellanger. The thesis of RL is supported by PSA Groupe in the context of OpenLab Fluidics (fluidics@poitiers). We also acknowledge the funding of the former Chair of Excellence ‘Closed-loop control of turbulent shear layer flows using reduced-order models’ (TUCOROM, ANR-10-CHEX-0015) supported by the French National Research Agency (ANR) and the funding, and excellent working conditions of the Collaborative Research Center (CRC880) ‘Fundamentals of High Lift for Future Civil Aircraft’ funded by the German Science Foundation (DFG) and hosted by the Technical University of Braunschweig, Germany. This work is supported by a public grant overseen by the French National Research Agency (ANR) as part of the “Investissement

d’Avenir” program, through the “iCODE Institute project” funded by the IDEX Paris-Saclay, ANR-11-IDEX-0003-02, by the ANR grant ‘ACTIV_ROAD’. LC acknowledges the funding of the ONERA/Carnot project INTACOO (INnova-Tive ACTuators and mOdels for flow cOntrol). EK gratefully acknowledges funding by the Moore/Sloan foundation, the Washington Research Foundation and the eScience Institute. We appreciate valuable stimulating discussions with Diogo Barros, Steven Brunton, Thomas Duriez and Andreas Spohn.

References

1. D.C. DRACOPOULOS, *Evolutionary Learning Algorithms for Neural Adaptive Control*, Springer, London, 2013.
2. N. GAUTIER, J.-L. AIDER, T. DURIEZ, B.R. NOACK, M. SEGOND, W.M. ABEL, *Closed-loop separation control using machine learning*, Journal of Fluid Mechanics, **770**, 424–441, 2015.
3. V. PAREZANOVIC, L. CORDIER, A. SPOHN, T. DURIEZ, B.R. NOACK, J.-P. BONNET, M. SEGOND, M. ABEL, S. BRUNTON, *Frequency selection by feedback control in a turbulent shear flow*, Journal of Fluid Mechanics, **797**, 247–283, 2016.
4. T. DURIEZ, S. BRUNTON, B.R. NOACK, *Machine Learning Control—Taming Nonlinear Dynamics and Turbulence*, Fluid Mechanics and Its Applications, **116**, Springer, 2016.
5. S.L. BRUNTON, B.R. NOACK, *Closed-loop turbulence control: Progress and challenges*, Applied Mechanics Review, **67**, 5, 050801:01–48, 2015.
6. C.W. ROWLEY, D.R. WILLIAMS, T. COLONIUS, R.M. MURRAY, D.G. MACMYNOWSKI, *Linear models for control of cavity flow oscillations*, Journal of Fluid Mechanics, **547**, 317–330, 2006.
7. P.J. SCHMID, D. SIPP, *Linear control of oscillator and amplifier flows*, Physical Review Fluids, **1**, 4, 040501, 2016.
8. K. ROUSSOPOULOS, *Feedback control of vortex shedding at low Reynolds numbers*, Journal of Fluid Mechanics, **248**, 267–296, 1993.
9. J.H. HOLLAND, *Outline for a logical theory of adaptive systems*, Journal of the Association of Computing Machinery, **9**, 3, 297–314, 1962.
10. I. RECHENBERG, *Cybernetic Solution Path of an Experimental Problem*, Roy. Airc. Establ., Libr. Transl., **1122**, Hants, Farnborough, 1965.
11. H.P. SCHWEFEL, *Projekt MHD-Staustrahlrohr: Experimentelle Optimierung einer Zweiphasendüse. Teil I*, **11**, Technical Report, AEG Forschungsinstitut, Berlin, 1968.
12. D.C. DRACOPOULOS, S. KENT, *Genetic programming for prediction and control*, Neural Computing and Applications, **6**, 214–228, 1997.
13. P.J. FLEMING, R.C. PURSHOUSE, *Evolutionary algorithms in control systems engineering: a survey*, Control Engineering Practice, **10**, 11, 1223–1241, 2002.
14. J. KOZA, *Genetic Programming: on the Programming of Computers by Means of Natural Selection*, vol. 1, MIT Press, U.S.A., 1992.

15. N. BENARD, P.J. PONS, J.F. PERIAUX, C. BUGEDA, J.-P. BONNET, E. MOREAU, *Multi-input genetic algorithm for experimental optimization of the reattachment downstream of a backward-facing step with surface plasma actuator*, 46th AIAA Plasmadynamics and Lasers Conference, 1–23, AIAA, 2015.
16. A.DEBIEN, K.A.F.F. VON KRBK, N. MAZELLIER, T. DURIEZ, L. CORDIER, B.R. NOACK, M.W. ABEL, A. KOURTA, *Closed-loop separation control over a sharp-edge ramp using genetic programming*, Experiments in Fluids, **57**, 40, 1–19, 2016.
17. M. BRAMEIER, W. BANZHAF, *Linear Genetic Programming*, Springer Science & Business Media, Berlin, 2007.
18. R. LI, B.R. NOACK, L. CORDIER, J. BORÉE, F. HARAMBAT, *Drag reduction of a car model by linear genetic programming control*, Experiments in Fluids, **58**, 103, 1–20, 2017.
19. K. THULASIRAMAN, M.N.S. SWAMY, *5.7 Acyclic directed graphs*, Graphs: Theory and Algorithms, **118**, 1992.
20. D.M. LUCHTENBURG, B. GÜNTER, B.R. NOACK, R. KING, G. TADMOR, *A generalized mean-field model of the natural and actuated flows around a high-lift configuration*, Journal of Fluid Mechanics, **623**, 283–316, 2009.
21. J. ÖSTH, S. KRAJNOVIĆ, B.R. NOACK, D. BARROS, J. BORÉE, *On the need for a non-linear subscale turbulence term in POD models as exemplified for a high Reynolds number flow over an Ahmed body*, Journal of Fluid Mechanics, **747**, 518–544, 2014.
22. S.R. AHMED, G. RAMM, G. FALTIN, *Some salient features of the time averaged ground vehicle wake*, Society of Automotive Engineers, SAE Inc., 840300, 1984.
23. D. BARROS, J. BORÉE, B.R. NOACK, A. SPOHN, T. RUIZ, *Bluff body drag manipulation using pulsed jets and Coanda effect*, Journal of Fluid Mechanics, **805**, 422–459, 2016.
24. A.R. OXLADE, J.F. MORRISON, A. QUBAIN, G. RIGAS, *High-frequency forcing of a turbulent axisymmetric wake*, Journal of Fluid Mechanics, **770**, 305–318, 2015.
25. SH.J. SCHMIDT, R. WOSZIDLO, C.N. NAYERI, C.O. PASCHEREIT, *The effect of flow control on the wake dynamics of a rectangular bluff body in ground proximity*, Experiments in Fluids, **59**, 6, 107, 2018.
26. E. BERGER, D. SCHOLZ, M. SCHUMM, *Coherent vortex structures in the wake of a sphere and a circular disk at rest and under forced vibrations*, Journal of Fluids and Structures, **4**, 3, 231–257, 1990.
27. B. KHALIGHI, S. ZHANG, C. KOROMILAS, S.R. BALKANYI, L.P. BERNAL, G. IACCARINO, P. MOIN, *Experimental and computational study of unsteady wake flow behind a bluff body with a drag reduction device*, Technical Report, SAE Technical Paper, 2001.
28. E. KAISER, B.R. NOACK, A. SPOHN, L.N. CATTAFESTA, M. MORZYŃSKI, *Cluster-based control of nonlinear dynamics*, Theoretical and Computational Fluid Dynamics, (online), 1–15, 2017,
29. K.V. MARDIA, J.T. KENT, J.M. BIBBY, *Multivariate Analysis*, Academic Press, U.S.A., 1979.
30. I.J. SCHOENBERG, *Remarks to Maurice Fréchet’s article ‘Sur la définition axiomatique d’une classe d’espaces distanciés vectoriellement applicable sur l’espace de Hilbert’*, Annals of Mathematics, **38**, 724–732, 1935.

31. G. YOUNG, A.S. HOUSEHOLDER, *Discussion of a set of points in terms of their mutual distances*, Psychometrika, **3**, 19–22, 1938.
32. E. KAISER, R. LI, B.R. NOACK, *On the control landscape topology*, [in:] The 20th World Congress of the International Federation of Automatic Control pp. 1–5, (IFAC), Toulouse, France, 2017.
33. Z. MICHALEWICZ, M. SCHOENAUER, *Evolutionary algorithms for constrained parameter optimization problems*, Evolutionary Computation, **4**, 1, 1–32, 1996.
34. Z. MICHALEWICZ, D. DASGUPTA, R.G. LE RICHE, M. SCHOENAUER, *Evolutionary algorithms for constrained engineering problems*, Computers & Industrial Engineering, **30**, 4, 851–870, 1996.
35. J.-C. LOISEAU, B.R. NOACK, S.L. BRUNTON, *Sparse reduced-order modelling: sensor-based dynamics to full-state estimation*, Journal of Fluid Mechanics, **844**, 459–490, 2018.

Received June 29, 2018; revised version November 2, 2018.
