AIAA SciTech Forum
7-11 January 2019, San Diego, California
AIAA Scitech 2019 Forum

10.2514/6.2019-0167

Plant and Controller Optimization in the Context of Chaotic Dynamical Systems

Luiz V. R. Cagliari*, Sandipan Mishra.†, and Jason E. Hicken‡
Rensselaer Polytechnic Institute, Troy, New York, 12180

This paper presents a gradient-based framework for plant and controller design of chaotic dynamical systems. The proposed approach uses a least-squares-shadowing method to perform the sensitivity analysis of the chaotic system and a first-order optimization algorithm to carry out the optimization process. Two different optimization strategies are examined. The first is a sequential approach, where the plant and control optimization problems are solved in series. The second is a simultaneous approach, where a single optimization problem is solved that considers both the plant and control disciplines at once. The two optimization strategies are applied to a simplified flow-control problem consisting of N potential vortices trapped inside a rectangular domain and controlled by one or more control vortices.

I. Introduction

In the context of dynamic-systems design, the standard practice has been to divide the design process into two steps: physical plant design and controller design, where physical plant design refers to any non-control component of the system, e.g. structural design. In this conventional design approach, these two steps are performed sequentially, meaning, the plant is designed first and then the controller is optimized for that particular plant model. This step-wise separation makes the design process easier; however, the resulting design is no longer optimal in a multidisciplinary sense [1], because the plant and controller problems are coupled. In order to achieve system-level optimality, the interdependency between the plant and controller design must be taken into consideration. This is illustrated in Figure 1.

Following the literature[2], we will refer to any multidisciplinary design problem that involves control as a co-design problem. There are several different co-design approaches that attempt to account for the coupling between the plant and controller. These include iterative, nested/bi-level, and simultaneous methods [3, 4]; however, according to Fathy [1], from an optimization standpoint, system-level optimality is only guaranteed for the nested and simultaneous approaches to co-design.

Although co-design has been applied to a wide range of areas, including electric motors [3] and active suspensions [2], in the field of aerospace engineering most of the work in co-design has been limited to structural problems [5–10] that are non-chaotic in nature. The long-term objective of this work is to expand the use of co-design to turbulent fluid flows,which are inherently chaotic. This introduces challenges beyond those present in typical multidisciplinary design optimization (MDO) applications. In particular, the sensitive dependence on parameters causes conventional sensitivity analysis methods, such as the tangent and adjoint methods, to fail to produce usable derivatives for chaotic systems [11]. To address sensitivity analysis in the context of chaotic dynamical systems, we use the least-squares-shadowing adjoint method (LSS)[12], which produces bounded sensitivities that are sufficiently accurate for optimization.

The remainder of the paper is organized as follows. Section II introduces and describes our co-design framework. Section III reviews sensitivity analysis and our choice of optimization algorithm. Section IV describes a model problem used to test the proposed co-design framework. The results are presented in Section V and the paper concludes with a discussion in Section VI.

II. Methodologies for Plant-Control Optimization

In this section, we introduce the class of problems that we are interested in solving using an integrated co-design formulation. We develop the co-design problem by first introducing the, typically independent, statements for optimal plant design and optimal control design.

*Ph.D Student, Department of Mechanical, Aerospace and Nuclear Engineering, and AIAA Student Member.
†Associate Professor, Department of Mechanical, Aerospace and Nuclear Engineering.
‡Associate Professor, Department of Mechanical, Aerospace and Nuclear Engineering, and AIAA Member.

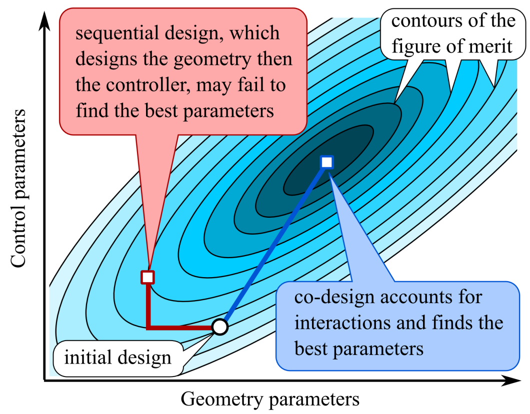


Fig. 1 Sequential design vs Co-design

The approach taken in this paper consists of solving the two problems, physical-plant and controller optimization, as one single coupled system of equations, in order to take full advantage of the interdependencies between the disciplines as illustrated in Fig. 1. The problem is optimized for a common objective with respect to the optimization variables, namely the plant and controller design parameters.

Consider the generalized system representation illustrated in Fig. 2. As seen in this figure, dynamical systems are often composed of a physical plant (structure, actuators and sensors) and a controller corresponding to some digital process. The controller interprets data from the sensors, based on an established control law, and defines the actuation rate, i.e, the input signals sent to the actuators. The system states are represented as $x \in \mathbb{R}^n$, where $n = p + c$ and the first p variables are the plant states, $x_p \in \mathbb{R}^p$, and the last c variables are the control states, $x_c \in \mathbb{R}^c$. The measured output is represented as $y \in \mathbb{R}$ and the design parameters are defined as $z \in \mathbb{R}^d$; example design parameters include the physical geometry, sensor and actuator distribution, and controller gains.

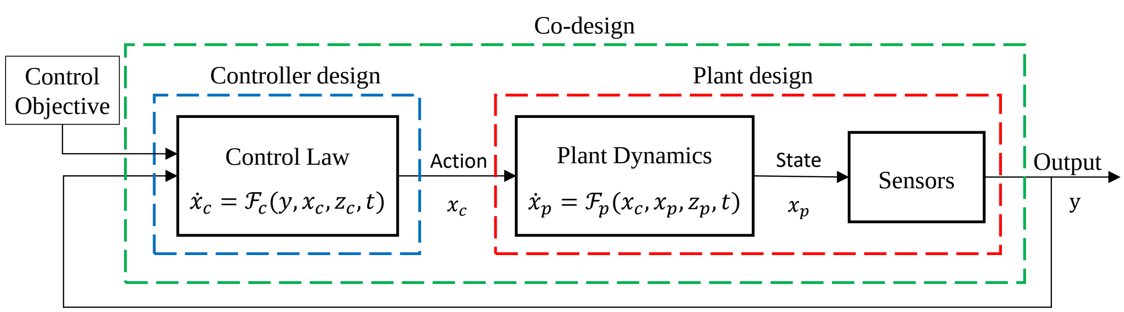


Fig. 2 Typical controlled system block diagram

A. Optimization Problem

From Fig. 2, we see that each dashed block can be related to an individual optimization problem. The red-dashed block corresponds to the plant optimization problem. Depending on the system being studied it could be a single-discipline optimization problem, or a more general MDO problem, e.g. aerostructural optimization. The blue-dashed block corresponds to the controller optimization problem and is only concerned in finding the best control law to optimize the control objective (tracking error, control effort, or some weighted combination thereof).

In order to better illustrate these optimization problems, consider Fig. 2 and the following general optimization statement:

$$\begin{aligned} \underset{z}{\text{minimize}} \quad & \mathcal{J}(x(z), z) \\ \text{governed by} \quad & \mathcal{R}(x(z), z, t) \equiv \dot{x} - \mathcal{F}(x, z, t) = 0, \ \forall t \in [0, T] \\ & x(0) = x_{ic}. \end{aligned} \tag{1}$$

$\mathcal{R}(x(z), z, t)$ is the residual of the system dynamics and $\mathcal{F}(x, z, t)$ defines those dynamics; for example, $\mathcal{F}(x, z, t)$ may represent the spatial discretization of the Navier-Stokes equations. Note that (1) is a reduced-space formulation, in which the state is an implicit function of the design parameters, $x = x(z)$, via the system dynamics. Finally, $\mathcal{J}(x(z), z) \in \mathbb{R}$ is the objective function that we assume is a time-averaged quantity of interest of the form

$$\mathcal{J}(x, z) = \frac{1}{T} \int_0^T \mathcal{G}(x, z, t) dt, \tag{2}$$

where $\mathcal{G}(x, z, t)$ is some time-dependent function and T is the terminal time.

1. Physical Plant Optimization Problem

In the context of plant optimization, the parameters of interest are the ones related to the physical characteristics of the system, for example, the geometric parameters. These physical parameters are denoted as z_p. Therefore, referring to the optimization statement in (1), the optimization parameter of interest can be defined as $z \equiv z_p$. In this case the residual $\mathcal{R}$ corresponds to open-loop dynamics and, as previously mentioned, it could constitute a single- or multi-disciplinary analysis. To illustrate, in the case of an aerostructural analysis, the optimization statement would be of the following form:

$$\begin{aligned} \underset{z_p}{\text{minimize}} \quad & \mathcal{J}(x, z) \equiv \mathcal{P}(x_s, x_a, z_p) = \frac{1}{T} \int_0^T \mathcal{G}(x_s, x_a, z_p, t) dt \\ \text{governed by} \quad & \mathcal{R}(x_s, x_a, z_p, t) \equiv \begin{cases} \mathcal{R}_s \equiv \dot{x}_s - \mathcal{F}_s(x_s, x_a, z_p, t) = 0 \\ \mathcal{R}_a \equiv \dot{x}_a - \mathcal{F}_a(x_s, x_a, z_p, t) = 0 \end{cases} \quad \forall t \in [0, T] \\ & x_s(0) = x_s^{ic}, \ x_a(0) = x_a^{ic}, \end{aligned}$$

where x_s and x_a denote the structural and aerodynamic states, respectively.

2. Controller Optimization Problem

When considering controller design, the goal of the optimization problem is to find the control input that produces the best possible system performance based on a given cost function. In optimal control, we seek a control design that minimizes cost functions of the form

$$\mathcal{J}(x, z) \equiv C(x(t), u(t)) = \phi(x(T), T) + \int_0^T L(x(t), u, t) dt,$$

where $\phi(x(T), T)$ corresponds to the terminal cost and $L(x(t), u, t)$ is the Lagrange running cost.

There are several approaches for solving the optimal control problem [13]:

- Indirect Methods: An application of Pontryagin's Maximum Principle, where the optimality conditions are applied to identify the optimal control trajectory, $u_{op}(t)$, that minimizes C.
- Direct Methods: The infinite-dimensional optimal control problem is first discretized and rewritten as a non-linear program. An example of this approach is the direct transcription method, which is a special case of the all-at-once (or simultaneous analysis-and-design) MDO formulation.

In most practical applications, the control system corresponds to a feedback law, and the optimization process is used to find the best gains for this control architecture. For instance, suppose that our control architecture is a simple proportional feedback law, such as $u = -z_c y$, where z_c is the controller gain and y is the measured output of the system. Assuming this control structure, the optimal control problem can be solved for the controller gain z_c; therefore, referring to the statement (1), the optimization parameter of interest is $z \equiv z_c$. The well-known Linear Quadratic Regulator (LQR) is the n-state generalized version of this control law that optimizes a quadratic cost.

3. Co-design Optimization Problem

Now consider the most general form of the optimization problem, as shown by statement (1). Here both $\mathcal{J}(x, z)$ and $\mathcal{R}(x, z)$ are functions of the plant physical design parameters, z_p, and also of the controller parameters, z_c. In this case, choosing to minimize the objective function using both design parameters, z_p and z_c, results in the co-design optimization problem. Referring to the optimization statement (1), the optimization parameter can be partitioned as $[z] \equiv \left[z_p^T, z_c^T\right]^T$ and the optimization problem becomes the following statement:

$$\begin{aligned} &\underset{z_c, z_p}{\text{minimize}} && \mathcal{J}(x_p, x_c, z_p, z_c) \\ &\text{governed by} && R(x_p, x_c, z_p, z_c, t) \equiv \begin{cases} \dot{x}_p - \mathcal{F}_p(x_p, x_c, z, t) = 0, \ \forall t \in [0, T] \\ \dot{x}_c - \mathcal{F}_c(x_p, x_c, z, t) = 0, \ \forall t \in [0, T] \end{cases} \\ & && x_p(0) = x_p^{ic}, \ x_c(0) = x_c^{ic}, \end{aligned} \tag{3}$$

where $\mathcal{F}_p(\cdot)$ is the plant governing equation and $\mathcal{F}_c(\cdot)$ is the control governing equation.

The objective function for the co-design problem (3) can be a combination of the plant optimization and control optimization objective functions, e.g. $\mathcal{J}(\cdot) = \mathcal{P}(\cdot) + C(\cdot)$, or it could be a completely distinct objective function that does not directly relate to $\mathcal{P}(\cdot)$ and $C(\cdot)$.

As previously mentioned in the introduction, only the nested and simultaneous co-design approaches are guaranteed to achieve system-level optimality. The differences between these two approaches are described below:

Nested: In the nested approach, two nested optimization loops are used to solve for the combined objective function described in 3. The outer loop minimizes the objective by optimizing only the plant-design variables, while the inner loop optimizes the control variables for each candidate plant design generated by the outer loop. Therefore, we have two different optimization problems, the outer one that handles the co-design objective, and the inner one that deals with the control objective alone [1].

Simultaneous: In the simultaneous plant and controller optimization approach, we want to find the best solution by directly solving problem 3. This can be mathematically and computationally challenging due to the increased dimensionality of the problem [1]. Furthermore, even if the plant and control subproblems are convex, the combined problem may be non-convex. Nevertheless, this approach takes full advantage of the coupling between the control and plant states and has the potential to require fewer optimization iterations.

In this work we will adopt the simultaneous approach for the optimization of a chaotic system.

III. Sensitivity Analysis

We are interested in solving integrated plant and controller design optimization problems (co-design problems) with large numbers of design parameters, such as the aerodynamic shape and active-flow control optimization of high-lift devices. Gradient-based optimization algorithms are attractive in this context, because they scale favorably with the number of design variables. However, the chaotic nature of our target applications raises several challenges for conventional gradient-based approaches [14, 15]. Consequently, specialized sensitivity analysis methods are necessary to compute useful derivatives, and noise-tolerant optimization algorithms are needed to cope with the fluctuations in the objective and its gradient. These matters are discussed in greater detail in this section.

A. Sensitivity Analysis

In the context of design optimization, sensitivity analysis is the study of how perturbations in an input of a model are translated into perturbations in an output, i.e. total differentiation. In this section, we review the least-squares-shadowing adjoint method, which is a sensitivity analysis method developed for chaotic dynamical systems [12]; however, we begin with a review of the conventional adjoint since both methods share some elements.

1. Conventional Adjoint Method

In order to minimize the functional $\mathcal{J}(x(z), z)$ with respect to z using a gradient-based optimization algorithm, we need to be able to efficiently compute the total derivative $\nabla_z \mathcal{J}$. As can be seen from its general form in Eq. (2), $\mathcal{J}(x, z)$ can have both explicit and implicit dependencies on z; the latter arises because the state depends implicitly on z. In order to account for the implicit dependencies, we introduce the Lagrangian

$$\mathcal{L}(x, z, \lambda) = \mathcal{J}(x, z) + \int_0^T \lambda^T \mathcal{R}(x, z)\, dt, \tag{4}$$

where λ^T is the Lagrange multiplier, or adjoint. The total derivative of $\mathcal{J}(x, z)$ with respect to z can be found by partial differentiation of the Lagrangian:

$$\nabla_z \mathcal{J}(x, z) \equiv \frac{\partial \mathcal{L}(x, z, \lambda)}{\partial z} = \frac{\partial \mathcal{J}(x, z)}{\partial z} + \int_0^T \lambda^T \frac{\partial \mathcal{R}(x, z)}{\partial z}\, dt. \tag{5}$$

However, this expression for $\nabla_z \mathcal{J}(\cdot)$ contains λ, which is as yet unknown. The adjoint can be determined by recognizing that the first variations of $\mathcal{L}(\cdot)$ with respect to x should vanish for a given z. From this, we can conclude that

$$\delta_x \mathcal{L} = 0 \quad \Rightarrow \quad \frac{1}{T} \int_0^T \frac{\partial \mathcal{G}(x, z, t)}{\partial x} \delta_x\, dt + \int_0^T \lambda^T \frac{\partial \mathcal{R}(x, z)}{\partial x} \delta_x\, dt = 0 \qquad \forall \delta_x. \tag{6}$$

From Eq. (6) it is possible to determine the adjoints λ and, consequently from Eq. (5), the gradient of the objective function.

2. Adjoint Method Discretization

In this section, we derive the discretized form of the adjoint equation that is often used in practice. In this work we use the Crank-Nicolson time marching scheme to discretize the problem into M uniform time-steps; therefore, $\Delta t = T/M$ and $i = 1, 2..., M+1$. Thus the discretized Lagrangian is

$$\begin{aligned}\mathcal{L}(x, z, \lambda) = {} & \frac{1}{T} \sum_{i=1}^{M} \frac{\Delta t}{2} \left[\mathcal{G}(x_i, z_i, t_i) + \mathcal{G}(x_{i+1}, z_{i+1}, t_{i+1}) \right] + \\ & \sum_{i=1}^{M} \lambda_{i+1}^T \left\{ (x_{i+1} - x_i) - \frac{\Delta t}{2} \left[\mathcal{F}(x_i, z_i, t_i) + \mathcal{F}(x_{i+1}, z_{i+1}, t_{i+1}) \right] \right\} + \lambda_1^T (x_1 - x_{IC}),\end{aligned} \tag{7}$$

where we have explicitly included the initial conditions in the Lagrangian. Differentiating $\mathcal{L}(\cdot)$ with respect to variable x_{M+1} and equating it to zero we find that the equation for the adjoint at the terminal time is

$$\left(I - \frac{\Delta t}{2} \left. \frac{\partial \mathcal{F}}{\partial x} \right|_{M+1} \right)^T \lambda_{M+1} = - \frac{\Delta t}{2T} \left. \frac{\partial \mathcal{G}}{\partial x} \right|_{M+1}, \tag{8}$$

where I is the $n \times n$ identity matrix. Similarly, differentiating the Lagrangian with respect to x_i, we obtain the equation for the discrete adjoint when $1 < i < M+1$:

$$\left(I - \frac{\Delta t}{2} \left. \frac{\partial \mathcal{F}}{\partial x} \right|_{i} \right)^T \lambda_i = \left(I + \frac{\Delta t}{2} \left. \frac{\partial \mathcal{F}}{\partial x} \right|_{i} \right)^T \lambda_{i+1} - \frac{\Delta t}{T} \left. \frac{\partial \mathcal{G}}{\partial x} \right|_{i}, \qquad \forall i = 2, 3, \ldots, M. \tag{9}$$

Finally, differentiating the Lagrangian with respect to x_1, we obtain the equation for the adjoint at $t = 0$:

$$\lambda_1 - \left(I + \frac{\Delta t}{2} \left. \frac{\partial \mathcal{F}}{\partial x} \right|_{1} \right)^T \lambda_2 = - \frac{\Delta t}{2T} \left. \frac{\partial \mathcal{G}}{\partial x} \right|_{1}. \tag{10}$$

Note that the terminal adjoint, λ_{M+1}, is solved first and then the remaining adjoints are solved backward in time. Once the adjoint is calculated at each time step, the gradient of the objective function can be obtained by discretizing Eq. (5):

$$\nabla_z \mathcal{J}(x, z) = \frac{\Delta t}{2} \left[\frac{1}{T} \sum_{i=1}^{M} \left(\left. \frac{\partial \mathcal{G}}{\partial z} \right|_{i} + \left. \frac{\partial \mathcal{G}}{\partial z} \right|_{i+1} \right) + \sum_{i=1}^{M} \lambda_{i+1}^T \left(\left. \frac{\partial \mathcal{F}}{\partial z} \right|_{i} + \left. \frac{\partial \mathcal{F}}{\partial z} \right|_{i+1} \right) \right]. \tag{11}$$

It will be helpful in the subsequent sections to introduce the following compact representation for the adjoint equation:

$$A^T \lambda = -b,$$

where

$$b^T = \left[\frac{\Delta t}{2T} \frac{\partial \mathcal{G}}{\partial x}\bigg|_{i=1}, \quad \frac{\Delta t}{T} \frac{\partial \mathcal{G}}{\partial x}\bigg|_{i=2}, \quad \ldots, \quad \frac{\Delta t}{T} \frac{\partial \mathcal{G}}{\partial x}\bigg|_{i=M}, \quad \frac{\Delta t}{2T} \frac{\partial \mathcal{G}}{\partial x}\bigg|_{i=M+1} \right], \tag{12}$$

and A denotes the Jacobian of the discretized state equation over all time steps:

$$A = \begin{bmatrix} I & & & & \\ E_1 & G_1 & & & \\ & E_2 & G_2 & & \\ & & \ddots & \ddots & \\ & & & E_M & G_M \end{bmatrix},$$

where E_i and G_i are defined as

$$E_i = -I - \frac{\Delta t}{2} \frac{\partial \mathcal{F}}{\partial x}\bigg|_i \qquad \text{and} \qquad G_i = I - \frac{\Delta t}{2} \frac{\partial \mathcal{F}}{\partial x}\bigg|_{i+1}.$$

Note that the identity matrix in the first row of A, which makes A square, is due to the initial condition.

3. Least-Squares-Shadowing Adjoint Method

The adjoint differential equation becomes linearly unstable for chaotic dynamical systems. Consequently, for such systems, the adjoint grows unbounded (backward) in time, and it produces unusable gradients. The least-squares shadowing (LSS) adjoint was developed to mitigate this issue [15]. The LSS sensitivity method relies on the shadowing lemma [16], which states that for a given parameter perturbation one can find a perturbed state that is "nearby" a given reference state at all time, i.e. a shadow trajectory.

To find the shadow trajectory we begin by assuming the chaotic system is ergodic and the objective function is an infinite time-averaged output. Then, among all the trajectories that satisfy the governing equations, an approximation to the shadow trajectory can be found by solving the following optimization problem.

$$\begin{aligned} \underset{\tau, x}{\text{minimize}} \quad & \frac{1}{2} \int_0^T \left(\| x(\tau(t)) - x_r(t) \|^2 + \beta^2 \left(\frac{d\tau}{dt} - 1 \right)^2 \right) dt \\ \text{subject to} \quad & \frac{dx}{d\tau} = \mathcal{F}(x, z), \end{aligned} \tag{13}$$

where $x_r(t)$ is a reference trajectory that satisfies the governing equation at a given z, $x(t)$ is the shadowing trajectory at a slightly different value of z, and β^2 is a scaling factor for the so-called time-dilation term, $\tau(t)$. The minimization problem described by Eq. (13) is a least-squares problem. Since $x(\tau(t))$ satisfies both the perturbed governing equations and the LSS minimization problem, it can be used to calculate the objective function $\mathcal{J}(x, z)$ and its derivatives.

In order to use the (approximate) shadowing trajectory for sensitivity analysis, we first linearize the LSS-problem to obtain

$$\begin{aligned} \underset{\eta, x'}{\text{minimize}} \quad & \frac{1}{2} \int_0^T \left(\| x' \|^2 + \beta^2 \eta^2 \right) dt \\ \text{subject to} \quad & \frac{dx'}{dt} = \frac{\partial \mathcal{F}}{\partial x} x' + \frac{\partial \mathcal{F}}{\partial z} + \eta \mathcal{F}(x_r, z), \end{aligned} \tag{14}$$

where x' and η are the solutions to the linearized LSS-problem corresponding to x and τ in Eq. (13) [12].

Problem (14) defines the tangent (or direct) version for the LSS method and it can be used to find total derivatives; however, we would need to solve one tangent LSS problem for each design variable. This motivates the adjoint version of the LSS method. In order to construct the LSS adjoint method it is necessary to modify the functional portion of the linearized LSS problem, creating a composite objective function and a minimization problem of the form:

$$\begin{aligned} &\underset{\eta,x'}{\text{minimize}} \quad \hat{\mathcal{J}}(x',\eta,z) + \frac{1}{2}\int_0^T \left(\|x'\|^2 + \beta^2\eta^2\right)dt \\ &\text{subject to} \quad \hat{\mathcal{R}}(x',\eta,z) \equiv \frac{dx'}{dt} - \frac{\partial \mathcal{F}}{\partial x}x' - \frac{\partial \mathcal{F}}{\partial z} - \eta\mathcal{F}(x_r,z) = 0, \end{aligned} \tag{15}$$

where $\hat{\mathcal{J}}$ is of the form

$$\hat{\mathcal{J}}(x',\eta,z) = \frac{1}{T}\int_0^T \frac{\partial \mathcal{G}}{\partial x}x' + \eta(t)\big(\mathcal{G}(x_r,z) - \mathcal{J}(x_r,z)\big)dt.$$

$\hat{\mathcal{J}}$ corresponds to the objective function linearized about the reference trajectory, x_r, with a perturbation size x'. From the linearized LSS problem we can obtain the LSS adjoint equation by constructing the Lagrangian:

$$\hat{\mathcal{L}}(x',\eta,\lambda,z) = \hat{\mathcal{J}}(x',\eta,z) + \int_0^T \left(\frac{1}{2}\big(\|x'\|^2 + \beta^2\eta^2\big) + \lambda^T\hat{\mathcal{R}}(x',\eta,z)\right)dt. \tag{16}$$

Optimality conditions require that the first variations of the Lagrangian with respect to $\delta x'$,$\delta\eta$ and $\delta\lambda$, be zero. From these conditions we obtain the following equations, that define the LSS adjoint:

$$\begin{aligned} \delta_{x'}\hat{\mathcal{L}} = 0 &\longrightarrow \int_0^T \left(x' + \lambda^T\frac{\partial\hat{\mathcal{R}}(x',\eta,z)}{\partial x'}\right)\delta_{x'}dt + \frac{1}{T}\int_0^T \frac{\partial \mathcal{G}}{\partial x}\delta_{x'}dt = 0 \\ \delta_{\eta}\hat{\mathcal{L}} = 0 &\longrightarrow \int_0^T \left(\beta^2\eta + \lambda^T\frac{\partial\hat{\mathcal{R}}(x',\eta,z)}{\partial \eta}\right)\delta_{\eta}dt + \frac{1}{T}\int_0^T \Big(\mathcal{G}(x_r,z) - \mathcal{J}(x_r,z)\Big)dt = 0 \\ \delta_{\lambda}\hat{\mathcal{L}} = 0 &\longrightarrow \int_0^T \hat{\mathcal{R}}(x',z)\delta_{\lambda}dt = 0. \end{aligned} \tag{17}$$

Eq.(17) corresponds to the system of equations that characterizes the LSS-adjoint.

4. Least-Squares-Shadowing Adjoint Discretization

The same discretization method used in section III.A.2 will be employed in this section; therefore, to avoid repetitiveness, we can discretize Eq. (17) as

$$\begin{bmatrix} Q & 0 & \hat{A}^T \\ 0 & \beta^2 I & F^T \\ \hat{A} & F & 0 \end{bmatrix}\begin{bmatrix} x' \\ \eta \\ \lambda \end{bmatrix} + \begin{bmatrix} b' \\ \bar{b} \\ 0 \end{bmatrix} = 0, \tag{18}$$

where $\hat{A}$ represents the Jacobian of the discretized state equation over all time steps and is related to the Jacobian A from the conventional adjoint method by removing the first block row (see below). F represents the dependency on the time dilation term, and is defined below. The block matrix Q is composed of weights from the trapezoid rule, which is associated with the regularization term x' in Eq. (17). Therefore,

$$\hat{A} = \begin{bmatrix} E_1 & G_1 & & & \\ & E_2 & G_2 & & \\ & & \ddots & \ddots & \\ & & & E_M & G_M \end{bmatrix}, \ F = \begin{bmatrix} f_1 & & & \\ & f_2 & & \\ & & \ddots & \\ & & & f_M \end{bmatrix}, \ Q = \begin{bmatrix} \frac{\Delta t}{2} & & & & \\ & \Delta t & & & \\ & & \ddots & & \\ & & & \Delta t & \\ & & & & \frac{\Delta t}{2} \end{bmatrix},$$

where E_i and G_i are as defined in Section III.A.1, $f_i = x'_{i+1} - x'_i$, and

$$b' = \begin{bmatrix} \frac{\Delta t}{2T}\frac{\partial \mathcal{G}}{\partial x'}\Big|_i & \frac{\Delta t}{T}\frac{\partial \mathcal{G}}{\partial x'}\Big|_{i+1} & \cdots & \frac{\Delta t}{T}\frac{\partial \mathcal{G}}{\partial x'}\Big|_{M-1} & \frac{\Delta t}{2T}\frac{\partial \mathcal{G}}{\partial x'}\Big|_M \end{bmatrix}^T, \ \bar{b} = \begin{bmatrix} \big(\mathcal{G}(x_r,z) - \mathcal{J}(x_r,z)\big)\Big|_i & \cdots & \big(\mathcal{G}(x_r,z) - \mathcal{J}(x_r,z)\big)\Big|_M \end{bmatrix}^T.$$

Once Eq. (18) is solved for λ we can substitute it into Eq. (11) — which is the same equation used with the conventional adjoint — to obtain the desired sensitivity:

$$\nabla_z \mathcal{J}(x', z) = \frac{\Delta t}{2T} \sum_{i=1}^{M} \left(\frac{\partial \mathcal{G}}{\partial z}\bigg|_i + \frac{\partial \mathcal{G}}{\partial z}\bigg|_{i+1} \right) + \frac{1}{T} \sum_{i=1}^{M} \lambda_i^T \frac{\partial \hat{\mathcal{R}}}{\partial z}\bigg|_i . \tag{19}$$

B. Sensitivity Analysis Verification

In order to verify the implementation of the least-squares-shadowing adjoint, the method was applied to the well-known Lorenz oscillator, which is defined by the following set of dynamics.

$$\begin{aligned} \frac{dx}{dt} &= \sigma(y - x), \\ \frac{dy}{dt} &= x(\rho - z) - y, \\ \frac{dz}{dt} &= xy - \beta z, \end{aligned} \tag{20}$$

with initial conditions $x(0) = x_0$ and objective function

$$\mathcal{J} = \frac{1}{T} \int_0^T z dt. \tag{21}$$

We chose the Lorenz oscillator as a test case to verify the LSS adjoint implementation because it is a well studied system with known values for the approximate derivative of the objective function (21) with respect to the parameter ρ [11].

The LSS sensitivity analysis procedure described above was applied to Eq. (20) and (21). In all simulations, the integration period was $T = 16$, a time step of $\Delta t = 10^{-3}$ was adopted, and the scaling factor for the time-dilation term on the LSS adjoint is $\beta = 100$. In addition, we used a spin-up period of $T = 10$, to avoid effects due to initial transients and ensure the trajectory is near the attractor. The initial conditions were randomly perturbed with the perturbation being sampled from a zero mean normal distribution with standard deviation of 0.05. We used 20 random initial conditions to demonstrate the statistical behavior of the LSS gradient.

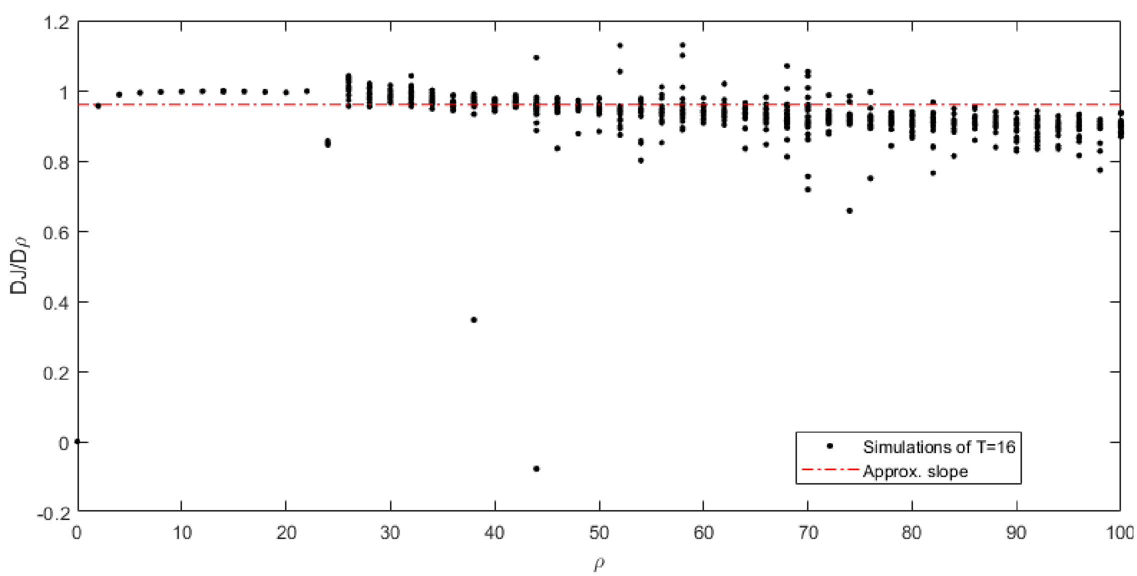


Fig. 3 Statistical gathering of objective function w.r.t parameter ρ

The information presented in Fig. 3 consists of a statistical gathering of the total derivative of the functional Eq. (21) with respect to a specific design variable (ρ). Each black dot corresponds to a simulation at a given (ρ) with a given initial condition. The sampled derivatives tend to cluster around the value 0.95, which is consistent with the approximate value of $D\mathcal{J}/D\rho$ [11, 12], therefore verifying the method implementation.

C. Optimization Method

As shown above, it is possible to compute useful gradient information using the LSS adjoint method; nevertheless, the objective function and gradient still contain noise-like fluctuations. More sophisticated optimization algorithms such as fast gradient descent (FGM) and other accelerated methods show a performance degradation that can be related to the level of noise in the objective function and consequently the amount of error present in its gradients [17].

In many cases the performance of such methods is worse than classical methods in terms of convergence rate and in some cases the optimization algorithm fails to converge entirely. For that reason we will focus on using a classical first-order approach such as the gradient descent method with a backtracking line search routine and a quadratic interpolation procedure to select the initial step length.

IV. A Flow-Control Model Problem

To compare the performance of the co-design and sequential approaches for chaotic dynamical systems, we consider a system of inviscid and incompressible vortices. One subset of these vortices is free to move within the domain and has pre-determined circulation strengths, while another subset consists of control vortices that are fixed in space and have a circulation strength that can be adjusted in real-time based on sensor readings and a prescribed control algorithm. The flow field created by the vortices is modeled using potential theory, and this field is then used in a dynamical system governing the vortex positions. All vortices are located within a rectangular domain of fixed dimensions, and the free-vortices move around within the domain based on the velocity field created as a result of their interaction with each other, the domain boundaries, and the geometrically fixed control vortices. Figure 4 shows a schematic of the problem setup.

For this system we use N to denote the total number of vortices inside the domain, and X_v, Y_v, X_c, and Y_c as the vectors of coordinates for the positions of the free and control vortices, respectively. The circulation strengths are denoted by Γ_v for the free-vortices and Γ_c for the control vortices. Finally, the sensor positions and domain dimensions are represented by (X_s, Y_s) and (a, b), respectively.

The objective of the problem is to find the optimal system configuration, i.e. the control vortex positions, sensor locations, circulation strengths, and controller parameters, that minimizes the weighted distance between the vortices and a given target location within the domain or on the boundary wall, while exerting minimal control effort. Thus we can define our state and design variables as follows:

$$
\begin{aligned}
x &= \left[X_v^T, Y_v^T, \Gamma_c^T\right]^T, \\
z &= \left[X_c^T, Y_c^T, \theta, \alpha_0, \alpha_1, \alpha_2\right],
\end{aligned}
\tag{22}
$$

where θ, α_0, α_1, and α_2 are control parameters described later.

This problem can be cast as a co-design problem, because the design parameters involve plant design variables (control vortex and sensor positions) and control parameters (θ and α_i). We have observed that the system can exhibit chaotic dynamics if the number of vortices $N \geq 3$.

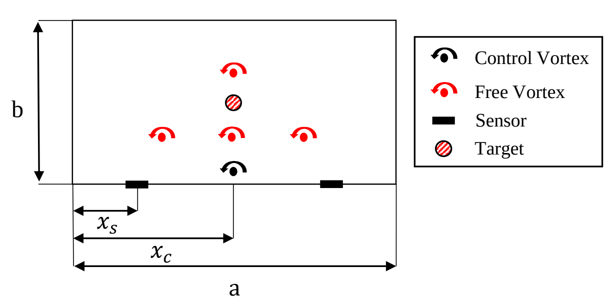


Fig. 4 Flow-control model problem structure illustration

A. Dynamics Model

1. Potential Flow

Recall that a potential vortex produces a circular flow field, centered around the vortex core, that is irrotational everywhere except at the core where a singularity exists. The circulation of any simple closed curve that encloses the core is Γ. In cylindrical coordinates the velocity distribution can be described as

$$V_r = 0 \quad \text{and} \quad V_\theta = \frac{\Gamma}{2\pi R}. \tag{23}$$

We can generalize this equation to compute the induced velocity of $N-1$ point vortices at the position of the Nth vortex. This can be repeated for the first $N-1$ vortices, and then a time marching method can be used to predict each vortex trajectory. In particular, the velocity at the ith vortex is given by Eq. (24).

$$V(X_i, Y_i) = \sum_{j\neq i}^{N} \frac{\Gamma}{2\pi R_{ij}^2}(Y_i - Y_j, X_j - X_i), \quad \forall i = 1, 2, \ldots, N \tag{24}$$

where $R_{ij} = \sqrt[2]{(X_j - X_i)^2 + (Y_i - Y_j)^2}$.

2. Infinite Image Method

The infinite image method takes advantage of the principle of superposition, in order to represent solid walls, for example. By superimposing mirrored potential flows, lines of symmetry that can be interpreted as streamlines are created between the true and the image flow. Such streamlines can then be interpreted as a boundary that satisfy the no-penetration condition.

The method of images can also be used to determine the potential flow due to a vortex inside a finite rectangular domain; however, in this case an infinite number of image vortices are necessary. Fortunately, a closed-form solution for this velocity field was proposed by Choi and Humphrey[18], which makes use of elliptic functions to transform the infinite summations into an analytical solution. Further development of the equations used to model this system can be found in the Appendix.

B. Control Strategy

We assume the feedback control law is a proportional feedback on the velocity measurement at a given sensor location. This can be expressed as follows:

$$\dot{\Gamma}_c = \mathcal{F}_c(x, z, t) = -\theta\Gamma_c + \alpha_0 + \alpha_1\bar{V}(Y_s) + \alpha_2\bar{V}(Y_s), \tag{25}$$

where the coefficients θ, α_0, α_1 and $\alpha_2 \in \mathbb{R}$ are the controller design parameters. The pair (X_s, Y_s) corresponds to the coordinates of a sensor and $\bar{V}(Y_s)$ is the sensor readings defined as

$$\bar{V}(Y_s) = \frac{1}{2\pi}\sum_{i=1}^{N-N_{cv}} \frac{\Gamma_i}{R_i^2}(Y_i - Y_s)\,, \tag{26}$$

where N is the total number of vortices inside the domain, and N_{cv} is a subset of N corresponding to the number of control vortices within the domain. R_i is the ith euclidean distance between vortex and sensor, and Γ_i is the ith vortex circulation strength . Note that Eq. (26) only depends on the Y coordinate; this is because we chose to use only the X-component of the velocity induced at the sensor by the free vortices as a measurement for the control law.

C. Objective Function

The objective is to minimize the sum of the weighted distance between vortices and a given target location within the bounded domain while accounting for the control effort. The instantaneous form of the first part of the objective is given by

$$\sum_{i=1}^{N} |\Gamma_i|\, \|X_i - X_{\text{target}}, Y_i - Y_{\text{target}}\|. \tag{27}$$

To account for the control effort, we add the 2-norm of the control variable to Eq. (27).

As previously mentioned, we are interested in objectives that are time-averages. Thus, the resulting objective is of the form

$$\mathcal{J} = \frac{1}{T}\sum_{j=1}^{T}\left[\sum_{i=1}^{N} w_p\left(|\Gamma_i|\|X_i - X_{\text{target}}, Y_i - Y_{\text{target}}\|\right) + w_c\,\|u_j\|\right]\Delta t, \tag{28}$$

where w_p is the plant-objective weighting factor, and w_c is the control-objective weighting factor; these weights are necessary in order to help us adjust the optimization priorities between the plant and control terms.

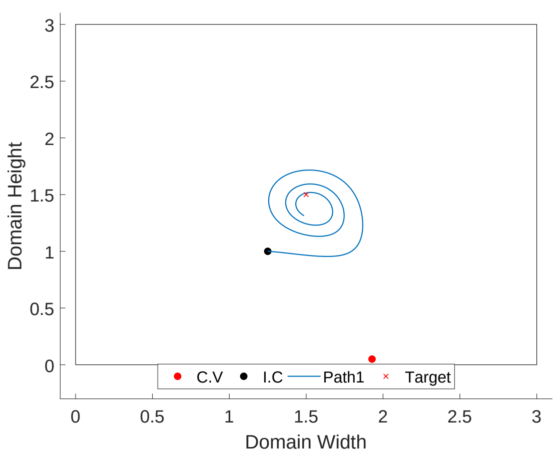
V. Numerical Results

In this section, we present the optimization results for non-chaotic and chaotic problems using both the sequential and the simultaneous approaches. In section V.A the non-chaotic problem is presented; the non-chaotic problem is amenable to conventional sensitivity analysis methods, so it is used to provide verification of the expected performance improvements provided by using the co-design approach over the traditional sequential approach. In section V.B, the more difficult chaotic problem is studied.

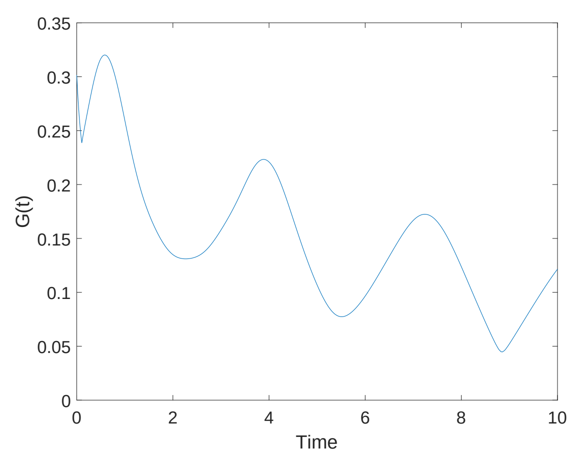
For all study cases, the time horizon is $T = 10$, and the scaling factor for the time dilation term on the LSS-adjoint is $\beta = 40$. In addition, the domain dimensions are fixed at $(a, b) = [3, 3]$, the target point is $T_p = [1.5, 1.5]$, and the objective function weights are $w_p = 0.9$ and $w_c = 0.1$. For the chaotic cases the spin-up period is $T_{spin} = 20$.

A. Non-chaotic system optimization

For the first study case, a single free-vortex is considered; its initial position is $(X_v, Y_v) = (1.25, 1.0)$, and its circulation strength is $\Gamma_v = 5.0$; the position of the sensors are $(X_s,\ Y_s) = ([0.5,\ 2.5], [0.0,\ 0.0])$; the control vortex position is $(X_c,\ Y_c) = (1.5,\ 0.15)$, and its initial circulation strength is $\Gamma_{cv} = -5.0$. The optimization begins with the control parameters set to $(\theta,\ \alpha_0,\ \alpha_1, \alpha_2) = (10.0,\ -0.1,\ 0.1,\ -0.5)$. This particular choice of control parameters creates a nearly constant control input to the system, similar to what a passive flow-control device, such as a vortex generator, would do.



(a) Vortex dynamics

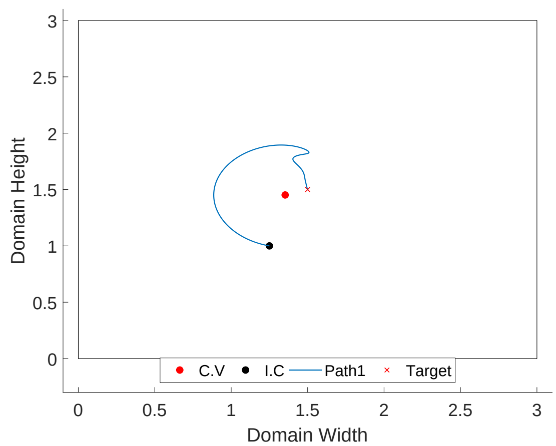


(b) Objective dynamics

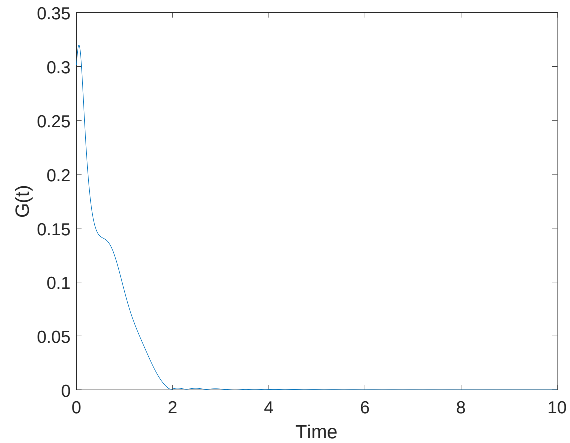
Fig. 5 Sequential optimization

In Figure 5a, it is possible to observe the vortex dynamics obtained as a result of the sequentially optimized parameters. The black dot corresponds to the initial position of the free vortex, the red dot is the position for the control vortex, and the red cross is the target point. Fig. 5b plots the integrand of the objective function as a result of the vortex and control dynamics.

In the sequential approach, the optimizer can only deal with a subset of the design variables at a time. For this particular case, since the control exerts a constant input that would drive the free vortex away from the target, and the optimizer only had access to the plant variables, the only solution is to move the control vortex away from the free vortex. Recall that, in order to enforce the boundary conditions, our formulation uses "mirror" vortices on the outside of the domain. Consequently, by moving the control vortex towards the the wall the optimizer negates the control vortex's effects by "creating" a doublet-like flow, which leads to a reduction in the objective function. However, while this improved the objective from a plant point of view, it also deteriorated the system's control authority, which made it difficult for a first-order method to converge to a optimal solution during the second part of the sequential method.



(a) Vortex dynamics



(b) Objective dynamics

Fig. 6 Simultaneous optimization (Co-design)

Figure 6a, shows the vortex dynamics obtained by the simultaneously optimizing all parameters. As before, the black dot is the free-vortex initial position, the red dot is the control vortex position, and the red cross is the target point. Also as before, Fig. 6b show the temporal behavior of the objective function as a result of the vortex and control dynamics.

In a simultaneous approach, the optimizer has access to all design parameters at once, and the optimization step is taken by using sensitivity information from all variables. This makes the problem harder to solve due to its increased dimensionality, but it also allows the optimizer to exploit the coupling between parameters. For this particular case, exploiting the coupling was beneficial, because it allowed the optimizer to move the control vortex away from the boundaries and create an orbit-like trajectory that minimized the objective rapidly.

Table 1 shows the initial and optimized values for the design parameters and the corresponding objective function value. The simultaneous approach not only succeeded in driving the free-vortex to the target point but also achieved a 95% reduction in the objective function value; in contrast the sequential approach managed only a 68% reduction.

Parameter	Initial System	Sequential Optimization	Co-design Optimization
X_c	1.5	1.93	1.35
Y_c	0.15	0.05	1.45
θ	10.0	9.94	10.03
α_0	-0.1	-0.63	-0.96
α_1	0.1	1.29	0.79
α_2	-0.5	0.13	0.06
$\mathcal{J}$	**4.81**	**1.52**	**0.21**

Table 1 A comparison between design parameters and objective function value for the initial, sequentially optimized and simultaneously optimized non-chaotic case 1

Figure 7 shows the convergence histories for the two optimization methods. The long plateaus are a characteristic of the gradient descent method adopted for this study. A Newton or quasi-Newton optimization method would eliminate these plateaus.

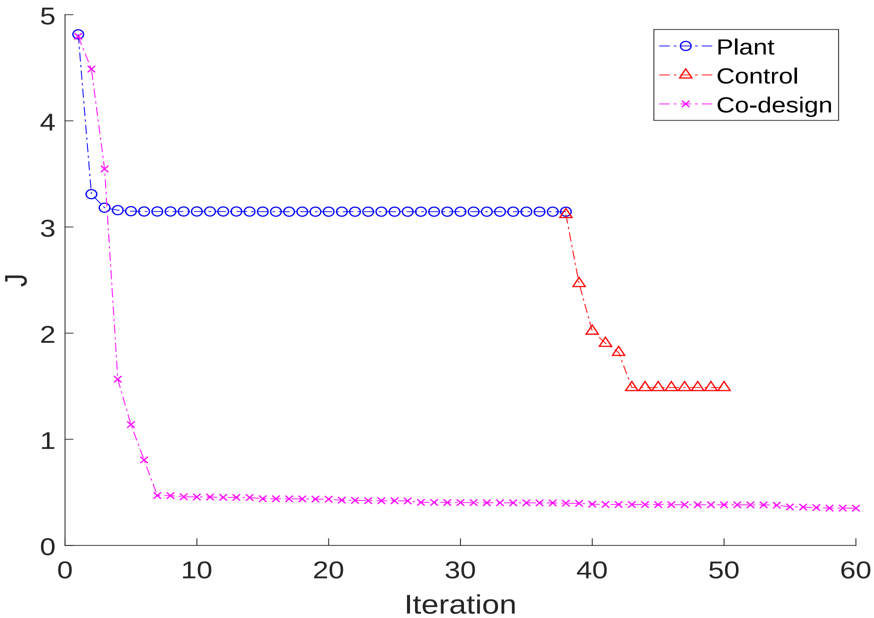
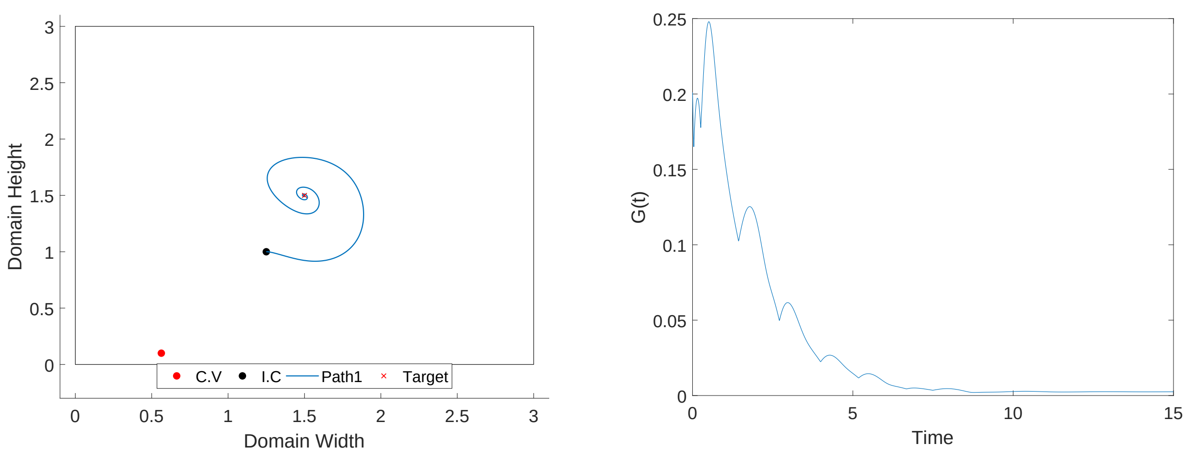


Fig. 7 Comparison between sequential and simultaneous optimization approaches non-chaotic case 1

For the second non-chaotic case, we use an identical set of baseline plant parameters and initial conditions; however, we set the control vortex initial circulation strength to $\Gamma_{cv} = 5.0$; and the control parameters are set to $(\theta,\ \alpha_0,\ \alpha_1, \alpha_2) = (10.0,\ -5.0,\ 5.0,\ -1.0)$.

This choice of control parameters creates a dynamic feedback control, similar to what an active flow-control device might use. This case is intended to check if a better initial control law would reduce the performance gap between the sequential and simultaneous approaches.

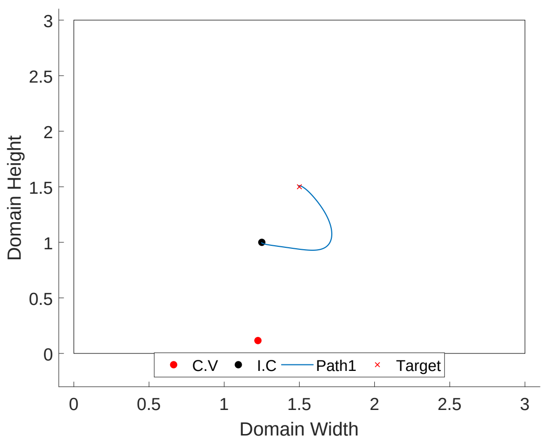


(a) Vortex dynamics **(b) Objective dynamics**

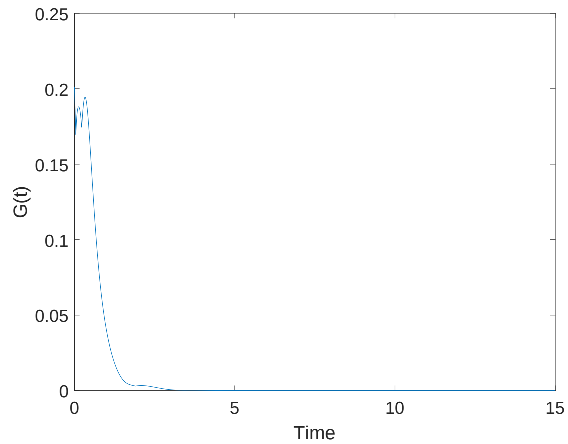
Fig. 8 Sequential optimization

In Fig. 8a, the vortex dynamics for the sequentially optimized system is plotted. As usual, the black dot is the free-vortex initial position, the red dot is the position for the control vortex, and the red cross is the target point. Figure 5b plots the integrand of the objective function versus time.

For this particular case, the initial system has sufficient control authority to drive the free vortex to a point other than the target inside the domain. Since the first part of the sequential approach only has access to the plant parameters, the optimizer compensates for the off-set convergence point by moving the control vortex to the left of the target-point but maintaining a certain distance from the boundary. This made it possible for the control optimization loop to correct its parameters and ensure terminal state convergence to the target.



(a) Vortex dynamics



(b) Objective dynamics

Fig. 9 Simultaneous optimization

Figures 9a and 9b are analogous plots to Figs. 5a and 8b for the simultaneous (co-design) optimization. For this case, the optimizer took advantage of the initial control authority and consequently did not have to compensate for the control position as much.

Table 2 lists the initial and optimized values for the design parameters along with the corresponding objective function value. As we can see, both the sequential and simultaneous approaches succeeded in driving the free vortex to target point; however, the sequential method only managed a 75% reduction in the objective function value, while the simultaneous approach produced a 92% reduction.

Parameter	Initial System	Sequential Optimization	Co-design Optimization
X_c	1.5	0.56	1.22
Y_c	0.15	0.10	0.12
θ	10.0	8.81	10.03
α_0	-5.0	-5.67	-4.74
α_1	5.0	6.74	5.31
α_2	-1.0	-0.37	-0.21
$\mathcal{J}$	**2.41**	**0.61**	**0.19**

Table 2 A comparison between design parameters and objective function value for the initial, sequentially optimized and simultaneously optimized systems non-chaotic case 2

Figure 10 shows the convergence histories of each method for the second non-chaotic case. As we can see, the behavior is qualitatively similar to the previous study case. As before, the long plateaus are expected for a first-order optimization method, and better convergence should be possible with more advanced algorithms.

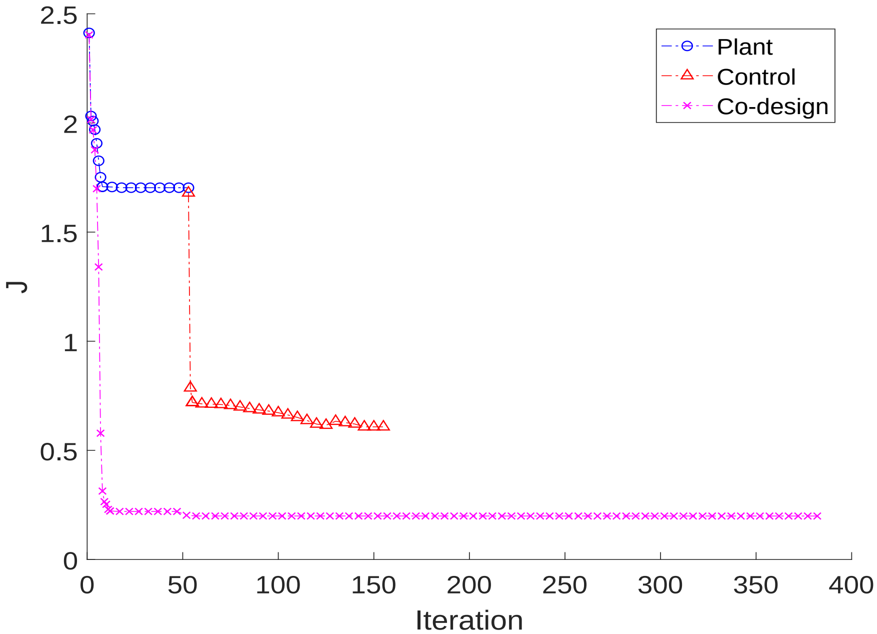


Fig. 10 Comparison between sequential and simultaneous optimization approaches non-chaotic case 2

B. Chaotic system optimization

We now turn to the more important, and difficult, chaotic-system optimizations. We begin this subsection by verifying the LSS adjoint for our chaotic model problem. Then we proceed to demonstrate the behavior of the open-loop system, and finally the optimization study is performed.

1. LSS Adjoint Applied to the Model Problem

While the LSS adjoint implementation was verified previously for the Lorenz problem in Section III.B, the vortex model problem is significantly more complex and demands an independent verification. For the simulation conditions described in the second chaotic study case below, the objective function was sampled for the control parameter α_0.

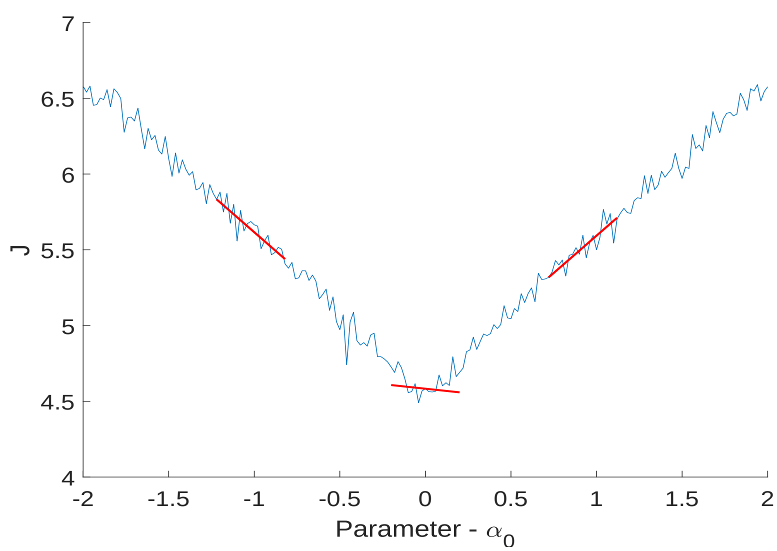
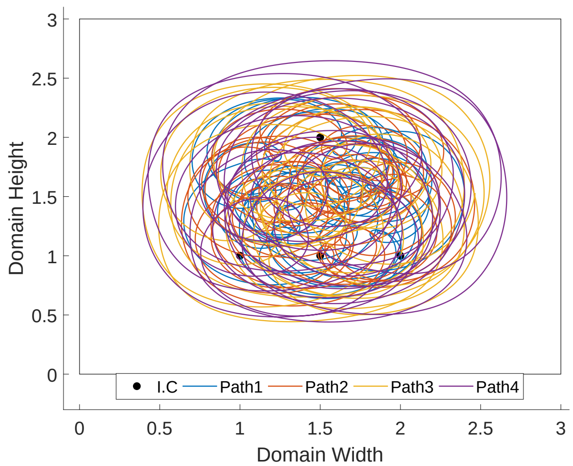


Fig. 11 LSS adjoint - Gradient verification

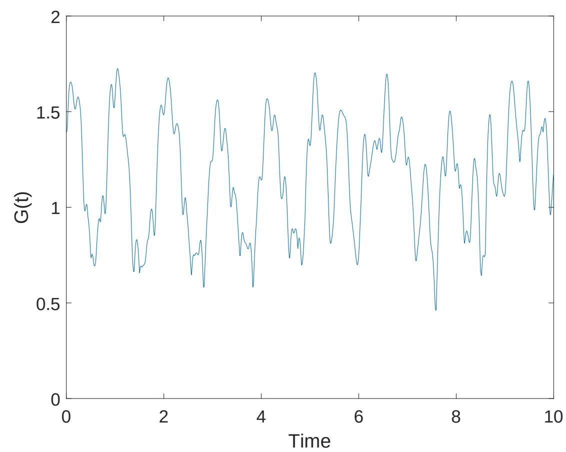
In Fig. 11, the objective function is sampled along the parameter α_0 design space and the LSS-calculated gradient is overlaid in red at a few representative locations. As we can see, despite the obvious noise-like fluctuations in the objective, there is good agreement between the observed slope and the LSS-predicted gradient.

2. Open-Loop Behavior

In order to have a better understanding of the chaotic system we seek to optimize, we first investigate the behavior of the free vortices and the objective in the absence of control.



(a) Open-loop vortex dynamics



(b) Open-loop objective dynamics

Fig. 12 Open-loop system dynamics

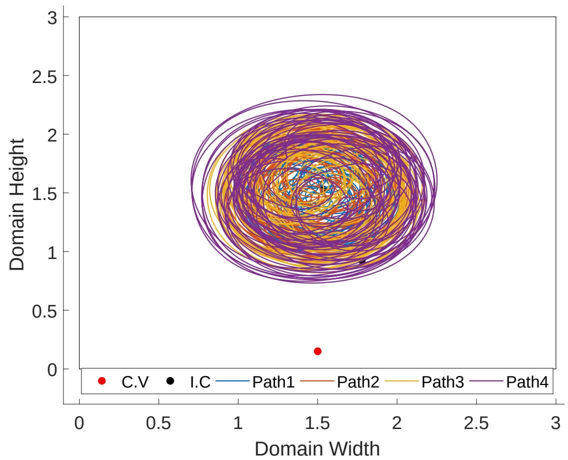
In Fig. 12a, the different colored lines correspond to the trajectories taken by each free vortex within the domain, and the black dots correspond to the initial positions of the vortices. It is easy to see that the trajectories are irregular and non-periodic. In Fig. 12b, the integrand of the objective function is plotter versus time, and its behavior suggests that the system is ergodic.

3. Closed-loop optimization

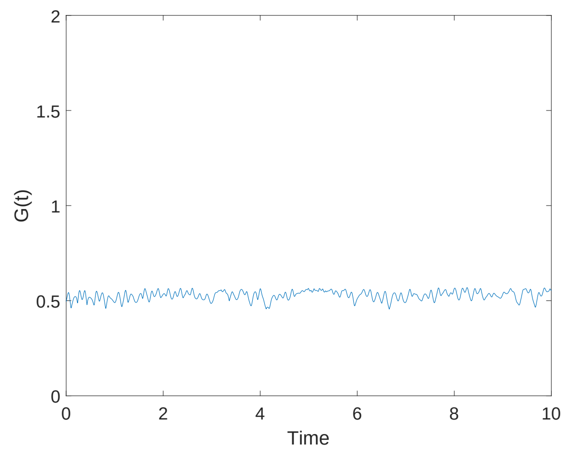
Just as in section V.A, two cases will be analyzed for both the sequential and simultaneous approaches. For the first case, the number of free vortices is $N = 4$; the initial positions for the free vortices are $(X_v, Y_v) = \big([1.0\ 1.5\ 2.0\ 1.0\ 1.5], [1.0\ 1.0\ 1.0\ 1.5]\big)$, and their circulation strengths are $\Gamma_v = [5.0,\ 4.0,\ 3.0,\ 2.0]$. The sensor positions are given by $(X_s,\ Y_s) = \big([0.5,\ 2.5], [0.0,\ 0.0]\big)$. The control vortex position is $(X_c,\ Y_c) = \big([1.5,\ 0.15]\big)$, and its initial circulation strength is $\Gamma_{cv} = -5.0$. Finally, the control parameters are initially $(\theta,\ \alpha_0,\ \alpha_1, \alpha_2) = \big(10.0,\ -0.1,\ 0.1,\ -0.5\big)$. As before, this set of control parameters creates a nearly constant control input to the system, similar to the effects of a passive flow-control device.

Figures 13a and 14a show the free-vortex trajectories obtained using the sequential and simultaneous optimization methods, respectively. As with the non-chaotic cases, the black dots are the free-vortices, initial positions, the red dot is the control-vortex position and, as before, the target point is at the center of the domain. Figures 13b and 14b plot the objective-function integrand versus time.

From Fig. 13 and Fig. 14, we can see that the vortex trajectories are closer to the center of the domain (i.e. the target location). Furthermore, there is a subtle difference between the resulting dynamics obtained with the sequential optimization and the simultaneous optimization. However, unlike the non-chaotic cases, both the sequential and simultaneous approaches produce similar reductions in the objective function; compare Figs.13b and 14b.

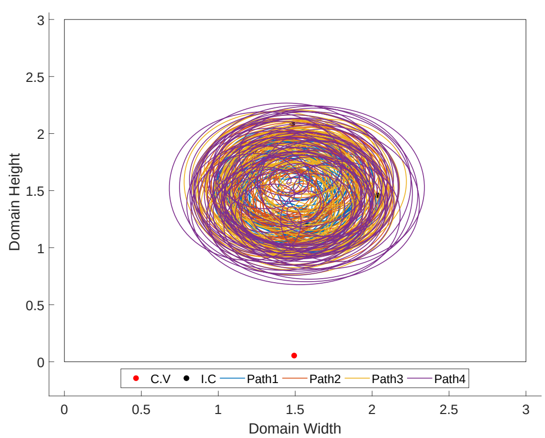


(a) Vortex dynamics

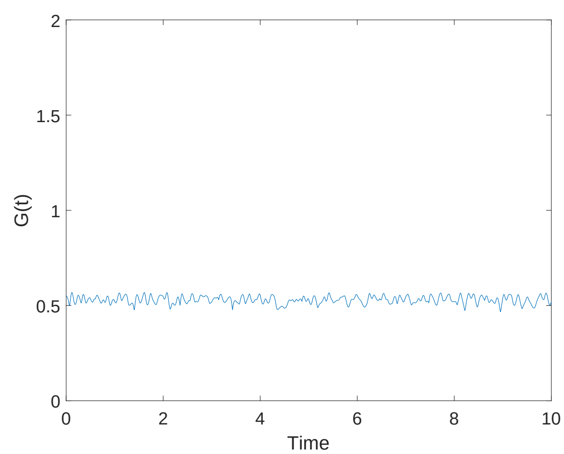


(b) Objective dynamics

Fig. 13 Sequential optimization - chaotic case 1



(a) Vortex dynamics



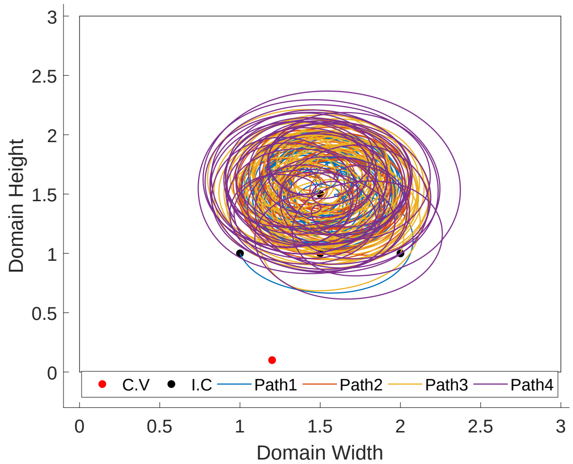
(b) Objective dynamics

Fig. 14 Simultaneous optimization - chaotic case 1

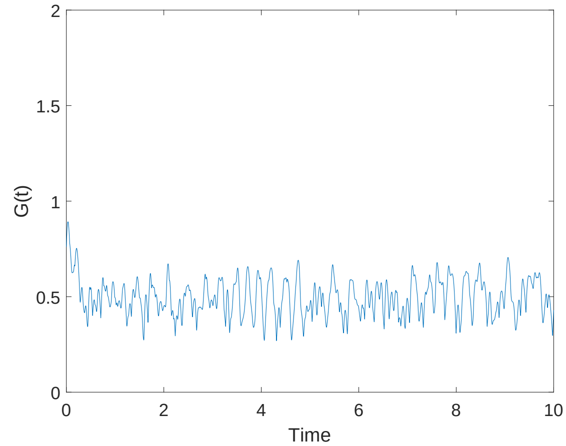
For the second chaotic case, we use an identical set of baseline plant parameters and initial conditions; however, we set the control vortex initial circulation strength to $\Gamma_{cv} = 5.0$; the control parameters are $(\theta,\ \alpha_0,\ \alpha_1, \alpha_2) = \big(10.0,\ -5.0,\ 5.0,\ -1.0\big)$. As before, this choice of control parameters creates a dynamic feedback control, similar to what an active flow-control device would do, however there is still room for improvement, as the optimization results show.

As before, the different lines in Fig. 15a and 16a correspond to the trajectories taken by each vortex within the domain, the black dots represent the initial positions of the free-vortices, and the red dot is the position of the control vortex. By comparing the open-loop system behavior with the sequentially optimized closed-loop behavior, it is possible to notice significant improvement. The vortex trajectories are more centralized within the domain and there is a significant reduction in the objective function, as shown by Figs. 15b and 16b.

From Fig. 15 and Fig. 16, it is possible to see that a reduction in the objective function was also achieved, and mimicking the non-chaotic case, the results achieved here were better than the ones obtained with the first study case, even if marginally.

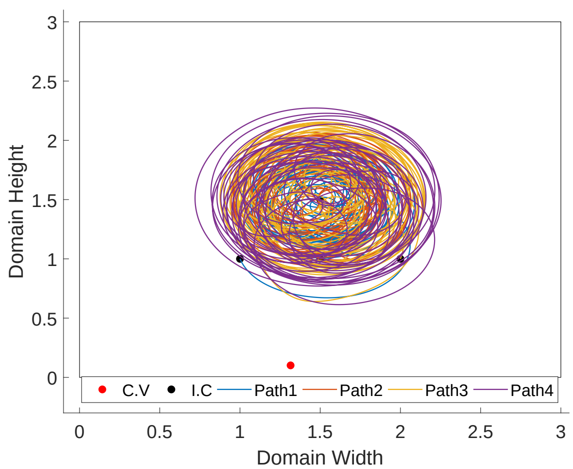


(a) Vortex dynamics

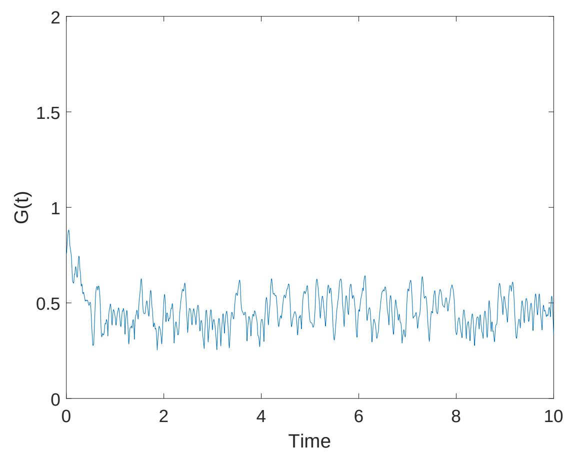


(b) Objective dynamics

Fig. 15 Sequential optimization - chaotic case 2



(a) Vortex dynamics



(b) Objective dynamics

Fig. 16 Simultaneous optimization - chaotic case 2

Table 3 shows the initial and optimized values of the design parameters for the first chaotic case study, along with the corresponding objective function values. As can be seen, both the sequential and the simultaneous approaches succeeded in reducing the objective function; however in both cases the resulting system was incapable of driving the vortices center of mass to the target position. This is expected, since we are using one control-vortex to manipulate several free vortices. The sequential approach managed a reduction of 16%, while the simultaneous approach managed a reduction of 15%.

Table 4, is analogous to 3, but for the second chaotic test case. In this case, both approaches also succeeded in reducing the objective function; however, the sequential approach, in contrast with what was observed for the non-chaotic cases, slightly outperformed the simultaneous approach, achieving a reduction in the objective function of 18.6% while the simultaneous approach only managed a 17.4% reduction.

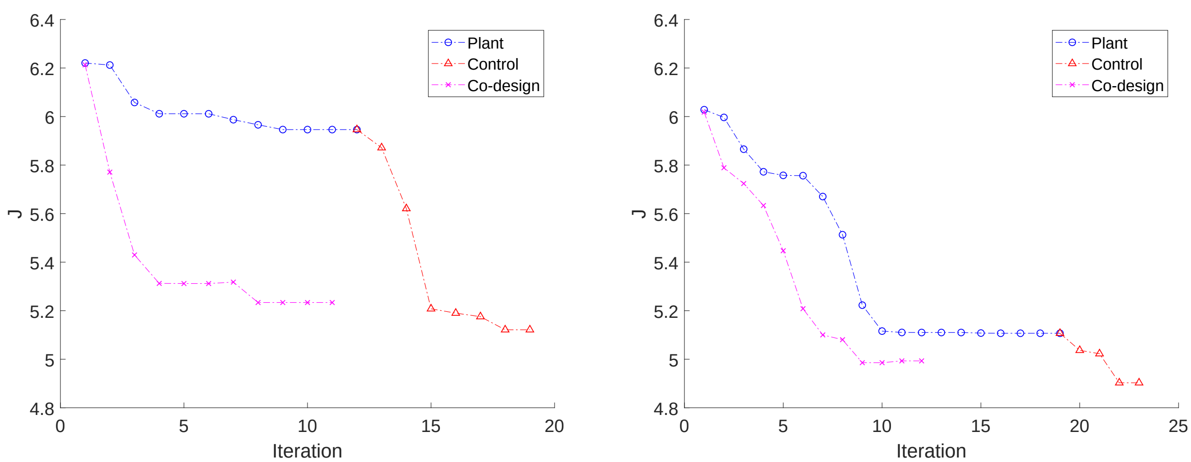
Parameter	Initial System	Sequential Optimization	Co-design Optimization
X_c	1.5	1.5	1.49
Y_c	0.15	0.15	0.05
θ	10.0	10.06	10.01
α_0	-0.1	0.04	0.02
α_1	0.1	0.26	0.33
α_2	-0.5	-0.35	-0.32
$\mathcal{J}$	**6.2**	**5.19**	**5.23**

Table 3 A comparison between design parameters and objective function value for the initial, sequentially optimized and simultaneously optimized systems chaotic case 1

Parameter	Initial System	Sequential Optimization	Co-design Optimization
X_c	1.5	1.20	1.32
Y_c	0.15	0.10	0.10
θ	10.0	10.02	10.02
α_0	-5.0	-4.77	-4.68
α_1	5.0	5.24	5.33
α_2	-1.0	-0.77	-0.67
$\mathcal{J}$	**6.03**	**4.91**	**4.98**

Table 4 A comparison between design parameters and objective function value for the initial, sequentially optimized and simultaneously optimized systems chaotic case 2

Figure 17 shows the convergence histories of each optimization method. As we can see the overall behavior is similar in both cases.



(a) Chaotic case 1 **(b) Chaotic case 2**

Fig. 17 Comparison between sequential and simultaneous optimization approaches

VI. Conclusions

Chaotic dynamical systems are widespread in many engineering disciplines, and this motivates research on the application of co-design methods for these types of problems; however the optimization of such systems with gradient-based methods is challenging, because conventional sensitivity analyses fail to produce useful gradients.

In this paper we presented a simplified model problem for chaotic flow-control and applied the least-squares-shadowing (LSS) adjoint sensitivity method to obtain usable derivatives information. This information was used in a simultaneous approach for plant-control design optimization of chaotic dynamical systems and compared to a more traditional sequential approach. We found that for non-chaotic problems, the performance of the co-design method was far superior to the traditional approach; however, both methods performed equally well for the optimization of the chaotic problem.

This work shows the potential of optimization in the context of chaotic systems, for future works we will consider computational fluid dynamics (CFD) models of the flow. The challenges associated with using CFD are mostly related with the need for a cost efficient sensitivity analysis method, since the LSS is not yet practical for CFD.

VII. Acknowledgments

This work was conducted during a scholarship supported by CNPq - National Council for Scientific and Technological development within the Ministry of Science and Technology of Brazil at Rensselaer Polytechnic Institute - RPI.

Appendix

In accordance with the work of Choi and Humphrey[18], Eq. (24) can be modified and extended, for the image method, in terms of Jacobi elliptic functions $sn(U,k)$, $cn(U,k)$ and $dn(U,k)$ by Eq. (29).

$$\begin{aligned} V_x &= P_y(X - X_1, Y - Y_1) + P_y(X - 2a + X_1, Y - 2b + Y_1) \\ &\quad - P_y(X - X_1, Y - 2b + Y_1) - P_y(X + 2a + X_1, Y - Y_1), \\ V_y &= -P_x(X - X_1, Y - Y_1) - P_x(X - 2a + X_1, Y - 2b + Y_1) \\ &\quad + P_x(X - X_1, Y - 2b + Y_1) + P_x(X - 2a + X_1, Y - Y_1). \end{aligned} \tag{29}$$

where,

$$\begin{aligned} P_y &= -\frac{K\Gamma}{a} \frac{cn^2(\frac{KX}{a}, k) cn(\frac{KY}{a}, k') sn(\frac{KY}{a}, k') dn(\frac{KY}{a}, k')}{[1 - cn^2(\frac{KX}{a}, k) cn^2(\frac{KY}{a}, k')]}, \\ P_x &= -\frac{K\Gamma}{a} \frac{cn^2(\frac{KY}{a}, k') cn(\frac{KX}{a}, k) sn(\frac{KX}{a}, k) dn(\frac{KX}{a}, k)}{[1 - cn^2(\frac{KX}{a}, k) cn^2(\frac{KY}{a}, k')]}, \end{aligned} \tag{30}$$

and,

$$K = \int_0^{\pi/2} \frac{d\phi}{\sqrt{1 - k^2 sin^2\phi}}, K' = \int_0^{\pi/2} \frac{d\phi}{\sqrt{1 - k'^2 sin^2\phi}} \tag{31}$$

Note that k corresponds to the elliptic function modulus, $k' = \sqrt{1 - k^2}$, and the ratio between $K/K' = a/b$ where a and b are the domain's length and height, respectively.

References

[1] Fathy, H. K., Reyer, J. A., Papalambros, P. Y., and Ulsov, A., "On the coupling between the plant and controller optimization problems," *American Control Conference, 2001. Proceedings of the 2001*, Vol. 3, IEEE, 2001, pp. 1864–1869.

[2] Allison, J. T., Guo, T., and Han, Z., "Co-design of an active suspension using simultaneous dynamic optimization," *Journal of Mechanical Design*, Vol. 136, No. 8, 2014, p. 081003.

[3] Reyer, J. A., and Papalambros, P. Y., "Combined optimal design and control with application to an electric DC motor," *Journal of Mechanical Design*, Vol. 124, No. 2, 2002, pp. 183–191.

[4] Reyer, J. A., and Papalambros, P. Y., "An investigation into modeling and solution strategies for optimal design and control," *The Proceedings of the 2000 ASME Design Engineering Technical Conferences, Baltimore, MD*, 2000.

[5] Kajiwara, I., and Haftka, R. T., "Integrated design of aerodynamics and control system for micro air vehicles," *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, Vol. 43, No. 3, 2000, pp. 684–690.

[6] Gupta, S., and Joshi, S. M., "An integrated control/structure design method using multi-objective optimization," 1991.

[7] Onoda, J., and Haftka, R. T., "An approach to structure/control simultaneous optimization for large flexible spacecraft," *AIAA journal*, Vol. 25, No. 8, 1987, pp. 1133–1138.

[8] Zhang, J., Smith, E. C., and Wang, K., "Active-Passive Hybrid Optimization of Rotor Blades with Trailing Edge Flaps," *Journal of the American Helicopter Society*, Vol. 49, No. 1, 2004, pp. 54–65.

[9] Hiramoto, K., and Grigoriadis, K., "Integrated design of structural and control systems with a homotopy like iterative method," *International Journal of Control*, Vol. 79, No. 9, 2006, pp. 1062–1073.

[10] Dhingra, A., and Lee, B., "Multiobjective design of actively controlled structures using a hybrid optimization method," *International journal for numerical methods in engineering*, Vol. 38, No. 20, 1995, pp. 3383–3401.

[11] Lea, D. J., Allen, M. R., and Haine, T. W., "Sensitivity analysis of the climate of a chaotic system," *Tellus A: Dynamic Meteorology and Oceanography*, Vol. 52, No. 5, 2000, pp. 523–532.

[12] Wang, Q., Hu, R., and Blonigan, P., "Least Squares Shadowing sensitivity analysis of chaotic limit cycle oscillations," *Journal of Computational Physics*, Vol. 267, 2014, pp. 210–224. doi:http://dx.doi.org/10.1016/j.jcp.2014.03.002, URL `http://www.sciencedirect.com/science/article/pii/S0021999114001715`.

[13] Rao, A. V., "A survey of numerical methods for optimal control," *Advances in the Astronautical Sciences*, Vol. 135, No. 1, 2009, pp. 497–528.

[14] Sobieszczanski-Sobieski, J., and Haftka, R. T., "Multidisciplinary aerospace design optimization: survey of recent developments," *Structural optimization*, Vol. 14, No. 1, 1997, pp. 1–23.

[15] Wang, Q., "Forward and adjoint sensitivity computation of chaotic dynamical systems," *Journal of Computational Physics*, Vol. 235, 2013, pp. 1–13. doi:10.1016/j.jcp.2012.09.007, URL `http://dx.doi.org/10.1016/j.jcp.2012.09.007`.

[16] Pilyugin, S. Y., *Shadowing in dynamical systems*, Springer, 2006.

[17] Devolder, O., Glineur, F., and Nesterov, Y., "First-order methods of smooth convex optimization with inexact oracle," *Mathematical Programming*, Vol. 146, No. 1-2, 2014, pp. 37–75.

[18] Choi, Y., and Humphrey, J. A. C., "Analytical prediction of two-dimensional potential flow due to fixed vortices in a rectangular domain," *Journal of Computational Physics*, Vol. 56, No. 1, 1984, pp. 15–27.