# Active Flow Control Using Adaptive Model Predictive Control with Online, Closed-Loop System Identification

Jacob C. Vander Schaaf,* Krzysztof J. Fidkowski,† and Dennis S. Bernstein‡
*University of Michigan, Ann Arbor, Michigan 48109*

https://doi.org/10.2514/1.G009500

**Flow control is challenging for model-based and deep-learning control methods due to the complex, nonlinear dynamics of fluids. The present paper addresses this challenge by using predictive cost adaptive control (PCAC), which is an indirect adaptive control technique. PCAC performs continual, online, closed-loop linear system identification to facilitate output-feedback model predictive control. Operating from cold start with no prior modeling of the system dynamics, PCAC adapts automatically to changing flow conditions. This work assesses the performance of PCAC for several 2D flow-control scenarios. First, PCAC is used to control the velocity within a viscous boundary layer; next, PCAC is used to suppress turbulence. Finally, PCAC is used to specify the flow-separation location in order to delay stall on a high-angle-of-attack airfoil. The results demonstrate the applicability of PCAC to flow-control problems without prior control-oriented modeling or offline controller training.**



## I. Introduction

ACTIVE flow control is a scientifically challenging problem with significant potential benefits for the performance of flight vehicles [1–11]. The nonlinear dynamics of fluids, along with their complex behavior at different Reynolds and Mach numbers, make traditional controller design challenging. For model-based feedback control, available models may possess errors that degrade closed-loop performance [12,13]. These modeling errors include imprecise physical models and parameter values, as well as errors due to model reduction and temporal or spatial discretization. Controller-synthesis techniques used for flow control include optimal control [14,15], robust control [12], and machine-learning methods [10,16]. Machine-learning methods require significant offline training of the controller and may be valid only for narrow ranges of operating conditions.

The present paper focuses on indirect adaptive control, where continual, online system identification is performed in response to actual, changing conditions with no prior modeling information. In particular, the present paper focuses on predictive cost adaptive control (PCAC) [17], which is a discrete-time, indirect adaptive extension of model predictive control. Unlike model-based control, PCAC uses no prior model, and, unlike deep and reinforcement learning, PCAC uses no preflight data or offline training. PCAC uses recursive least squares (RLS) with variable-rate forgetting (VRF) to perform continual, online, closed-loop system identification to update a linear input–output model for use by output-feedback model predictive control (MPC). PCAC was applied to vibration control in [18], experimental thermoacoustic oscillation suppression in [19], and an aeroelastic model in [20]. In [21], PCAC increased the lift-to-drag ratio of an airfoil by reducing separation in the low-Reynolds-number regime, where the flow is dominated by unsteady flow structures and vortices. Additionally, in [22,23], PCAC was used to suppress unsteady aerodynamic effects in ground-effect flight. As a precursor to the present paper, PCAC was applied to flow separation for a 2D airfoil in [24] and velocity control in the boundary layer in [25].

This work applies PCAC to several 2D flow-control problems. For each example, PCAC begins from cold-start conditions with no prior modeling of the fluid dynamics; PCAC identifies the dynamics of each flow-control scenario during closed-loop operation. In the first example, the streamwise velocity within a laminar boundary layer in a duct is controlled, followed by a more challenging case where harmonic command following is performed for the velocity in the boundary layer of an axial cylinder. Next, turbulence suppression in a duct with a backward-facing step and in a duct with crossflow jets is considered. Finally, a practical application involving control of the separation location and stall mitigation in a high-angle-of-attack airfoil is demonstrated. Within each example, various aspects of PCAC are examined to assess the performance of PCAC. In particular, the tuning hyperparameters are adjusted, the effect of multiple simultaneous control objectives is tested, the performance of PCAC using different measurement resolutions is assessed, and the ability of PCAC to adapt to changing operating conditions is investigated. Table 1 lists the examples and their corresponding flow-control objectives and PCAC studies.

## II. Review of Predictive Cost Adaptive Control

As shown in Fig. 1, predictive cost adaptive control (PCAC) combines online identification with output-feedback MPC. PCAC uses no a priori modeling information aside from a suitable model order for system identification, nor does it use probing signals.

### A. Online Identification

To perform online, closed-loop system identification, consider the MIMO input–output model

$$\hat{y}_k = -\sum_{i=1}^{\hat{n}} \hat{F}_i y_{k-i} + \sum_{i=1}^{\hat{n}} \hat{G}_i u_{k-i} \tag{1}$$

where $k \geq 0$ is the time step, $\hat{n} \geq 1$ is the identification data window, $\hat{F}_i \in \mathbb{R}^{p\times p}$ and $\hat{G}_i \in \mathbb{R}^{p\times m}$ are the estimated model coefficients, and $u_k \in \mathbb{R}^m$, $y_k \in \mathbb{R}^p$, and $\hat{y}_k \in \mathbb{R}^p$ are the inputs, outputs, and predicted outputs, respectively. At step $k$, $\hat{F}_{i,k}$, and $\hat{G}_{i,k}$ are estimated online using recursive least squares with variable-rate forgetting (RLS/VRF) by minimizing the cost function [26]

$$J_k(\hat{\theta}) = \sum_{i=0}^{k} \frac{\rho_i}{\rho_k} z_i^T(\hat{\theta}) z_i(\hat{\theta}) + \frac{1}{\rho_k}(\hat{\theta} - \theta_0)^T P_0^{-1}(\hat{\theta} - \theta_0) \tag{2}$$

Presented as Paper 2025-0030 at the 2025 AIAA SciTech Forum, Orlando, FL, January 6–10, 2025; received 8 August 2025; accepted for publication 24 March 2026; published online 29 May 2026. 

*Ph.D. Candidate, Department of Aerospace Engineering; jacobcvs@umich.edu (Corresponding Author).
†Professor, Department of Aerospace Engineering; kfid@umich.edu.
‡Professor, Department of Aerospace Engineering; dsbaero@umich.edu.

**Table 1 Flow-control examples**

| Example | Flow-control objective | PCAC study |
|---|---|---|
| 1 | Velocity control in duct boundary layer | MPC cost function weighting |
| 2 | Velocity control in axial-cylinder boundary layer | RLS model order |
| 3 | Turbulence suppression in backward-facing step | MPC horizon |
| 4 | Turbulence suppression in duct with crossflow jets | Additional control objectives |
| 5 | Airfoil flow separation: single setpoint | Baseline case |
| 6 | Airfoil flow separation: multiple setpoints | Measurement resolution |
| 7 | Airfoil flow separation: changing angle of attack | Adaptability to operating conditions |

where $\rho_k \triangleq \prod_{j=0}^{k} \lambda_j^{-1} \in \mathbb{R}$, $\lambda_k \in (0,1]$ is the forgetting factor, $P_0 \in \mathbb{R}^{[\hat{n}p(m+p)]\times[\hat{n}p(m+p)]}>0$ is the initial RLS covariance matrix, $\theta_0 \in \mathbb{R}^{[\hat{n}p(m+p)]}$ is the initial estimate of the coefficient vector, and the performance variable $z_i(\hat{\theta}) \in \mathbb{R}^p$ is defined as

$$z_k(\hat{\theta}) = y_k - \phi_k \hat{\theta} \tag{3}$$

The vector $\hat{\theta} \in \mathbb{R}^{[\hat{n}p(m+p)]}$ of coefficients to be estimated is

$$\hat{\theta} \triangleq \mathrm{vec}\begin{bmatrix} \hat{F}_1 & \cdots & \hat{F}_{\hat{n}} & \hat{G}_1 & \cdots & \hat{G}_{\hat{n}} \end{bmatrix} = \mathrm{vec}\begin{bmatrix} \hat{\theta}_{\hat{F}} & \hat{\theta}_{\hat{G}} \end{bmatrix}$$

where $\hat{\theta}_{\hat{F}}$ and $\hat{\theta}_{\hat{G}}$ are the estimates of the coefficients, defined by

$$\hat{\theta}_{\hat{F}} \triangleq \mathrm{vec}\begin{bmatrix} \hat{F}_1 & \cdots & \hat{F}_{\hat{n}} \end{bmatrix} \tag{4}$$

$$\hat{\theta}_{\hat{G}} \triangleq \mathrm{vec}\begin{bmatrix} \hat{G}_1 & \cdots & \hat{G}_{\hat{n}} \end{bmatrix} \tag{5}$$

with the regressor matrix $\phi_k \in \mathbb{R}^{p\times[\hat{n}p(m+p)]}$ defined by

$$\phi_k \triangleq \begin{bmatrix} -y_{k-1}^T & \cdots & -y_{k-\hat{n}}^T & u_{k-1}^T & \cdots & u_{k-\hat{n}}^T \end{bmatrix} \otimes I_p$$

The global minimizer $\theta_{k+1} \triangleq \mathrm{argmin}_{\hat{\theta}} J_k(\hat{\theta})$ of Eq. (2) is

$$L_k = \lambda_k^{-1} P_k \tag{6}$$

$$P_{k+1} = L_k - L_k \phi_k^T (I_p + \phi_k L_k \phi_k^T)^{-1} \phi_k L_k \tag{7}$$

$$\theta_{k+1} = \theta_k + P_{k+1} \phi_k^T (y_k - \phi_k \theta_k) \tag{8}$$

The variable-rate forgetting factor $\lambda_k$ is given by [27]

$$\lambda_k = \frac{1}{1 + \eta g(z_{k-\tau_d}, \ldots, z_k) \mathbf{1}[g(z_{k-\tau_d}, \ldots, z_k)]} \tag{9}$$

where $\mathbf{1}: \mathbb{R} \to \{0,1\}$ is the unit step function, and

$$g(z_{k-\tau_d}, \ldots, z_k) \triangleq \sqrt{\frac{\tau_n}{\tau_d} \frac{(\Sigma_{\tau_n}(z_{k-\tau_n}, \ldots, z_k)\Sigma_{\tau_d}(z_{k-\tau_d}, \ldots, z_k)^{-1})}{c}} - \sqrt{f}$$

where $\eta > 0$ and $p \le \tau_n < \tau_d$ represent numerator and denominator window lengths. $\Sigma_{\tau_n}$ and $\Sigma_{\tau_d}$ are the sample variances of the respective window lengths, and the threshold constant $f$ is described in [27,28]. The constant $c$, based on the window lengths, is described in [27]. The estimator coefficients $\hat{\theta}$ can be written in the block observable canonical form at step $k$ with matrices $\hat{A}_k$, $\hat{B}_k$, and $\hat{C}_k$ given by

$$\hat{A}_k \triangleq \begin{bmatrix} -\hat{F}_{1,k} & I_p & \cdots & \cdots & 0_{p\times p} \\ \vdots & 0_{p\times p} & \ddots & & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0_{p\times p} \\ \vdots & \vdots & & \ddots & I_p \\ -\hat{F}_{\hat{n},k} & 0_{p\times p} & \cdots & \cdots & 0_{p\times p} \end{bmatrix}, \qquad \hat{B}_k \triangleq \begin{bmatrix} \hat{G}_{1,k} \\ \hat{G}_{2,k} \\ \vdots \\ \hat{G}_{\hat{n},k} \end{bmatrix} \tag{10}$$

$$\hat{C}_k \triangleq \begin{bmatrix} I_p & 0_{p\times p} & \cdots & 0_{p\times p} \end{bmatrix} \tag{11}$$

where $\hat{F}_{i,k}$ and $\hat{G}_{i,k}$ denote the estimates $\hat{F}_i$ and $\hat{G}_i$ at step $k$, respectively.

### B. Model Predictive Control

MPC is a standard control technique that has been widely studied and applied in a variety of contexts [29–32]. In summary, MPC uses a model of the system to optimize future performance over a time horizon. The optimization yields a sequence of controls, the first of which is implemented, and the procedure is repeated at subsequent steps. By performing constrained optimization, MPC enforces constraints on the control input and plant outputs. At step $k$, PCAC uses the identified model $\hat{A}_k$, $\hat{B}_k$, and $\hat{C}_k$. As in [17], receding-horizon optimization is performed using quadratic programming (QP), which is a convex optimization technique. This optimization determines the

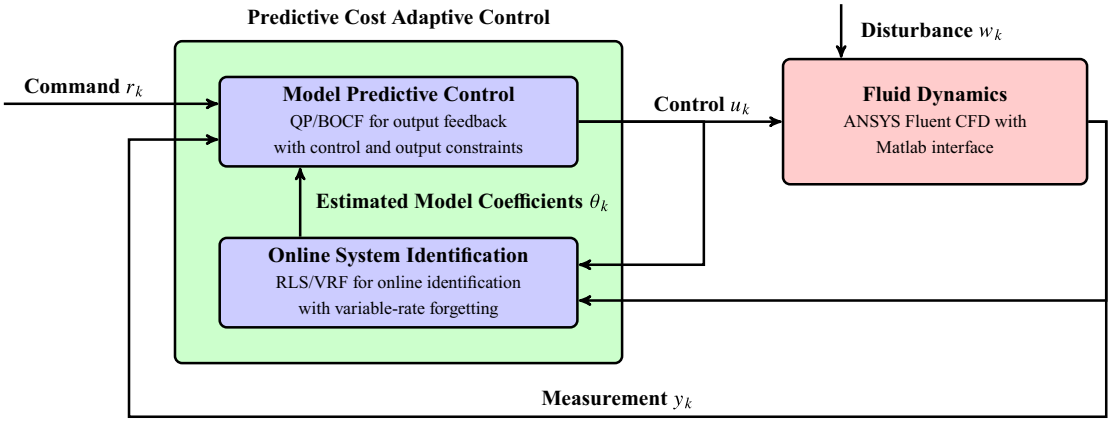


**Fig. 1 PCAC block diagram. Online system identification based on recursive least squares (RLS) provides estimated model coefficients $\theta_k$ used by output-feedback model predictive control.**

control input $u_{k+1}$ at the next time step while also attempting to satisfy constraints on the state and control input.

To describe QP-based MPC, let $\mathcal{R}_{k,\ell} \triangleq [\, r_{k+1}^T \;\cdots\; r_{k+\ell}^T \,]^T \in \mathbb{R}^{\ell p_t}$ be the vector of future commands over the horizon $\ell$; let $Y_{1|k,l}$ be the corresponding $l$-step predicted output for a sequence of $l$ future controls, $U_{1|k,l}$; and let $Y_{t,1|k,\ell} \triangleq C_{t,\ell} Y_{1|k,\ell}$ be the $\ell$-step predicted output, where $C_{t,\ell} \triangleq I_\ell \otimes C_t \in \mathbb{R}^{\ell p_t \times \ell p}$, $\otimes$ is the Kronecker product, and $C_t y_{i|k}$ computes the tracking outputs from $y_{i|k}$. Let $\mathcal{C}_\ell \triangleq I_\ell \otimes (\mathcal{C} C_c) \in \mathbb{R}^{\ell n_c \times \ell p}$, where $C_c y_{i|k}$ creates the constrained outputs from $y_{i|k}$, let $\mathcal{D}_\ell \triangleq 1_\ell \otimes \mathcal{D} \in \mathbb{R}^{\ell n_c}$, and define the sequence of differences of control inputs as

$$\Delta U_{1|k,\ell} \triangleq \begin{bmatrix} (u_{1|k} - u_k)^T & \cdots & (u_{\ell|k} - u_{\ell-1|k})^T \end{bmatrix}^T \in \mathbb{R}^{\ell m} \quad (12)$$

The QP-based MPC optimization problem is then given by

$$\min_{U_{1|k,\ell}} (Y_{t,1|k,\ell} - \mathcal{R}_{k,\ell})^T Q (Y_{t,1|k,\ell} - \mathcal{R}_{k,\ell}) + \Delta U_{1|k,\ell}^T R \Delta U_{1|k,\ell} \quad (13)$$

subject to

$$\mathcal{C}_\ell Y_{1|k,\ell} + \mathcal{D}_\ell \leq 0_{\ell n_c} \quad (14)$$

$$U_{\min} \leq U_{1|k,\ell} \leq U_{\max} \quad (15)$$

$$\Delta U_{\min} \leq \Delta U_{1|k,\ell} \leq \Delta U_{\max} \quad (16)$$

where

$$Q \triangleq \begin{bmatrix} \bar{Q} & 0_{p_t \times p_t} \\ 0_{p_t \times p_t} & \bar{P} \end{bmatrix} \in \mathbb{R}^{\ell p_t \times \ell p_t}$$

is the output weighting, $\bar{Q} \in \mathbb{R}^{(\ell-1)p_t \times (\ell-1)p_t}$ is the cost-to-go output weighting, $\bar{P} \in \mathbb{R}^{p_t \times p_t}$ is the terminal output weighting, $R \in \mathbb{R}^{\ell m \times \ell m}$ is the control-move-size weighting, $U_{\min} \triangleq 1_\ell \otimes u_{\min} \in \mathbb{R}^{\ell m}$, $U_{\max} \triangleq 1_\ell \otimes u_{\max} \in \mathbb{R}^{\ell m}$, $\Delta U_{\min} \triangleq 1_\ell \otimes \Delta u_{\min} \in \mathbb{R}^{\ell m}$, and $\Delta U_{\max} \triangleq 1_\ell \otimes \Delta u_{\max} \in \mathbb{R}^{\ell m}$.

## III. Simulation Setup

All examples in the paper are computed using ANSYS Fluent. Fluent is interfaced with the MATLAB-based PCAC control code using a custom Fluent user-defined function (UDF). At each time step $k$, the UDF obtains measurements $y_k$ from the flowfield and provides them to PCAC. PCAC uses $y_k$ to compute the performance variable $z_k$, which it uses to compute the control inputs $u_k$. The UDF then sends $u_k$ to Fluent, which applies the control input by modifying the associated actuator boundary conditions. Specific details of each case, including the geometry and control objective, are discussed within the examples.

## IV. Boundary-Layer Velocity Control

In this section, the flow-control objective is to specify the velocity at a specified location within a viscous boundary layer. A flow-velocity sensor is placed at the location of interest to measure the $x$-velocity $y_k$, and PCAC specifies the uniform duct-inlet velocity $u_k$ in order to achieve the velocity setpoint $r_k$. As in all of the examples in this paper, no prior model of the fluid dynamics is available to PCAC. As described above, PCAC uses RLS with VRF to perform online, closed-loop system identification.

*Example 1: x-velocity setpoint in a 2D duct boundary layer.* Consider a 2D duct with a width of 0.5 m and a length of 1 m. An $x$-velocity setpoint is commanded at the location of a velocity sensor, which is positioned halfway along the duct and within the boundary layer. The left boundary condition is a velocity inlet with a uniform velocity profile specified by the control input $u_k$, and the right boundary condition is a pressure outlet. A no-slip wall boundary condition is enforced at the bottom of the duct, and a symmetry boundary condition is enforced at the midheight line of the duct. The velocity sensor measures the flow $x$-velocity $y_k$ at the sensor location. The sensor measurements are used by PCAC to compute the command-following error $z_k$ between the measurement and the setpoint. The fluid computation is incompressible, unsteady, viscous, and laminar, with dynamic viscosity $\mu = 1.7894 \times 10^{-5}$ kg/(m · s).

The flowfield velocity in the duct is initialized to be 0 m/s at all locations, and the flow state is advanced forward in time with a fixed time step of 0.01 s using first-order implicit time stepping. PCAC samples the velocity sensor with a matching time step of 0.01 s. To investigate the response of PCAC, the MPC cost function weightings on the performance variable $z_k$ are varied while the weighting on the change $\Delta u_k$ of the control input is fixed. In particular, the cost-to-go and terminal weightings $Q$ and $P$, respectively, on $z_k$ are equal and set to 100, 50, 10, or 0.5. The weighting $R$ on $\Delta u_k$ is fixed at 0.1. Figure 2 shows the resulting sensor measurements and the control inputs for the selected values of $Q$ and $P$. Note that, as $Q$ and $P$ are increased, the command-following performance improves at the expense of larger transients when the controller is enabled. Figure 3 shows the sensor measurement $y_k$, the control input $u_k$, the estimated model coefficients $\theta_k$, and the forgetting factor $\lambda_k$ for the weightings $Q = P = 10$ and $R = 0.1$, which balance command-following performance with overshoot.

As an independent check of the simulation, the classical Blasius laminar boundary-layer analysis, which describes the boundary layer over a flat plate, is used to determine the inlet velocities that produce the setpoint velocities at the sensor location. In particular, the Blasius differential equation is given by

$$f''' + \frac{1}{2} f f'' = 0 \quad (17)$$

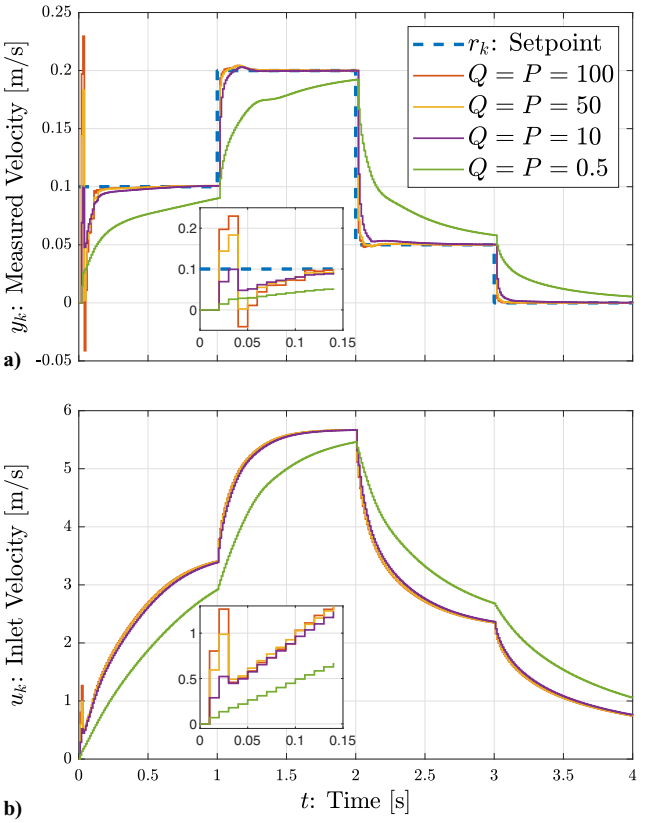


**Fig. 2 Example 1: As the cost-to-go and terminal weightings $Q$ and $P$ increase, panel (a) shows that the initial transient increases and subsequent performance improves.**

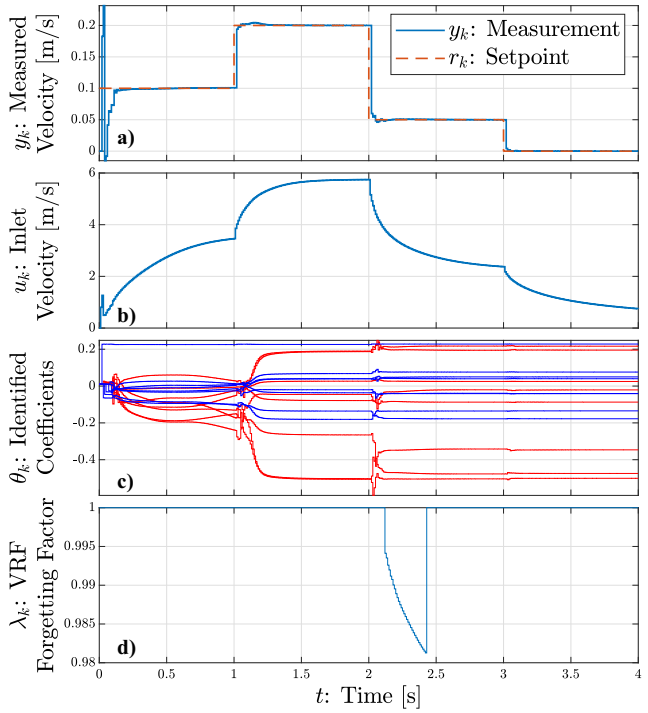


**Fig. 3 Example 1: With $Q = P = 10$, panel (a) shows that the measurement converges to the setpoint after 0.25 s.**

where the nondimensional stream function $f$ is a function of the similarity variable $\eta$, which describes the location of interest within the boundary layer and is defined in [33] (p. 234).

$$\eta = y\sqrt{\frac{u_\infty}{\nu x}} \tag{18}$$

where $y$ is the distance from the plate, $u_\infty$ is the freestream velocity, $\nu = \mu/\rho$ is the kinematic viscosity, and $x$ is the distance along the plate. For the boundary conditions

$$f(0) = 0, \quad f'(0) = 0, \quad f'(\infty) = 1 \tag{19}$$

solving Eq. (17) yields the normalized velocity profile in the boundary layer. The velocity $u$ at the sensor location is defined using this profile and is given by

$$u = f'(\eta)u_\infty \tag{20}$$

To determine the inlet velocity that results in the setpoint velocity at the sensor location, the command $r_k$ is substituted for the velocity $u$ at the sensor location, and the inlet velocity $u_k$ is substituted for the freestream velocity $u_\infty$. Then, an iterative method is used to solve for the inlet velocity that corresponds to the setpoint. Figure 4 shows the relationship between the setpoint velocity at the sensor location and the required inlet velocity obtained from the Blasius solution. The above procedure yields the inlet velocities of 2.55, 4.05, and 6.42 m/s, which are close to the converged inlet velocities of 2.37, 3.46, and 5.74 m/s obtained from PCAC. PCAC achieves these values with no prior model of the fluid dynamics. An exact match is not expected since this case is a duct and not purely a flat plate. Additionally, at the sensor location, the Reynolds numbers $Re_x$ based on $x$ are $8.12 \times 10^4$, $1.18 \times 10^5$, and $1.97 \times 10^5$ using the converged inlet velocities, and the Blasius solution becomes more accurate at higher Reynolds numbers, consistent with the assumptions of the Blasius equation.

The results in Fig. 3 show that the streamwise velocity measurement converges to the setpoint in approximately 0.25 s, whereas the PCAC-specified inlet velocity is nearly converged after 1 s, at which point the setpoint changes. Since the fluid calculation is

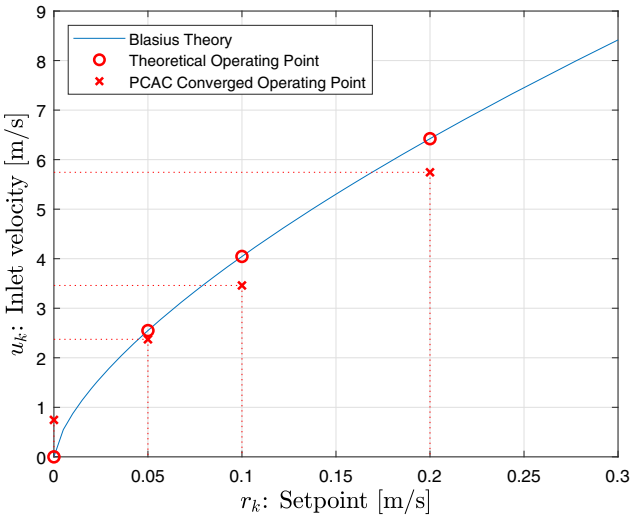


**Fig. 4 Example 1: Blasius theoretical inlet velocity as a function of the setpoint.**

incompressible, changes in the duct-inlet velocity are immediately propagated to the entire duct, and thus the velocity at the sensor location changes immediately with changes in the duct-inlet velocity. While the change in the inlet velocity is sensed immediately, the boundary layer takes time to develop with respect to the new bulk velocity. This is evident in Fig. 5, where the velocity at the sensor location is plotted as the duct-inlet velocity undergoes step changes. At each step change, a fast-time-scale change in the measured velocity due to the change in bulk velocity is visible in the first 0.25 s, and a slow-time-scale change in the measured velocity due to the boundary-layer development is visible over the remaining 1.75 s. In order to maintain the setpoint velocity at the sensor location, the inlet velocity needs to change as the boundary layer develops. This process is visible in Figs. 6 and 7, where the boundary profile at the sensor $x$-position is plotted at different time instances. The velocity at the sensor location is held nearly constant by PCAC while the boundary-layer profile develops.

System performance with PCAC is compared with performance using proportional–integral (PI) control in Fig. 8. PI control is appealing since it requires the tuning of only two gains. However, the gains must be chosen to ensure system stability. A difficulty with applying PI control to this case is due to the different time scales in the response to changes in the inlet velocity. The slow development of the boundary layer requires large gains to achieve fast convergence to the setpoint. However, the instantaneous change in bulk velocity due to the incompressible solution, coupled with high PI controller gains, results in oscillations in the response. By comparison, PCAC produces a larger initial transient as it identifies the system dynamics but afterward gives faster convergence to the

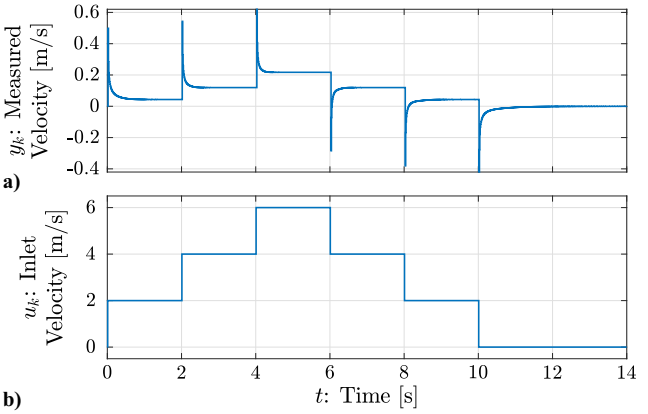


**Fig. 5 Example 1: Sensor flow-velocity measurements in response to step changes in duct-inlet velocity.**

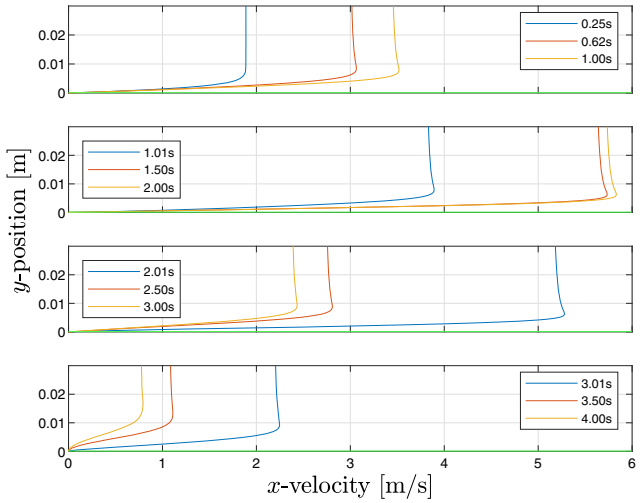


**Fig. 6 Example 1: Boundary-layer profiles at the sensor $x$-position using PCAC.**

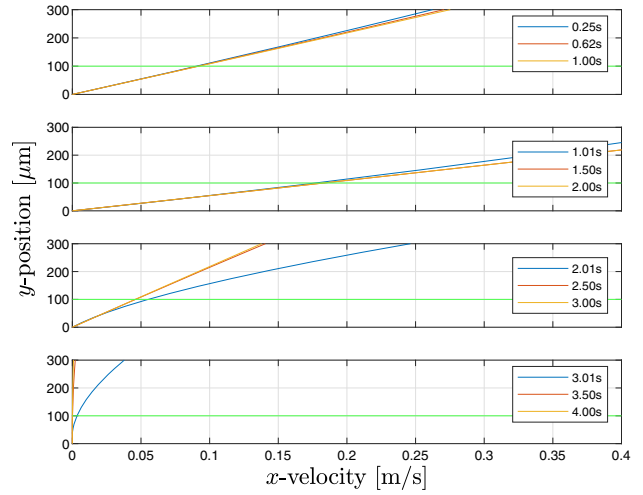


**Fig. 7 Example 1: Zoomed boundary-layer profiles at the sensor $x$-position using PCAC. The velocity at the sensor height remains nearly constant.**

setpoint with less oscillation. Through adaptation, PCAC is able to handle these challenging dynamics.

*Example 2: x-velocity command following in the boundary layer of an axial cylinder in a 2D duct.* Reconsider the duct geometry in Example 1 with the addition of an axial cylinder with a radius of 0.1 m, as shown in Fig. 9. In this example, a harmonic $x$-velocity command is specified at a location within the boundary layer on the leading surface of the axial cylinder. Namely, the command is given by

$$r_k = A\sin(\omega k) + 0.3 \tag{21}$$

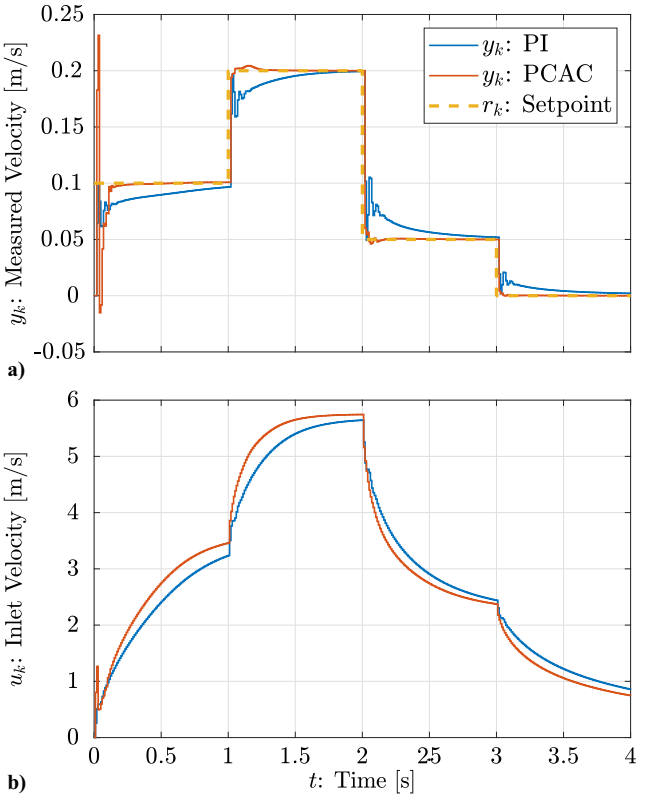


**Fig. 8 Example 1: Compared to PI, panel (a) shows that PCAC achieves faster convergence and less severe oscillations after changes in the setpoint.**

where $\omega = 0.5$ rad/step and $A = 0.01$ m/s. As in Example 1, constant setpoints may be applied to this system to achieve steady flow. However, a harmonic command is now specified to demonstrate the ability of PCAC to follow harmonic commands. This case is a more complex version of Example 1, where Blasius theory no longer applies due to both the modified geometry and the harmonic command.

The flowfield is initialized to have a velocity of 0 m/s at all locations within the duct. The fluid computation is incompressible, unsteady, viscous, and turbulent using the $k-\omega$ turbulence model. The dynamic viscosity is $\mu = 1.7894 \times 10^{-5}$ kg/(m · s). At the inlet, the turbulent intensity is 5%, and the turbulent viscosity ratio is 10. The flow state is advanced forward in time with a fixed time step of 0.01 s using first-order implicit time stepping. PCAC samples the velocity sensor with a matching time step of 0.01 s.

To investigate the performance of PCAC, the identified model order $\hat{n}$ in the RLS system identification is varied. In particular, $\hat{n}$ is set to be 2, 4, 8, 16, and 32. All other PCAC parameters remain fixed. Figure 10 shows the sensor measurements and the control inputs for the selected values of $\hat{n}$. PCAC is insensitive to the model order for this case, and all selected $\hat{n}$ values give similar results. Figure 11 shows the same quantities over the last 0.35 s, where it can be seen that higher values of $\hat{n}$ give slightly better

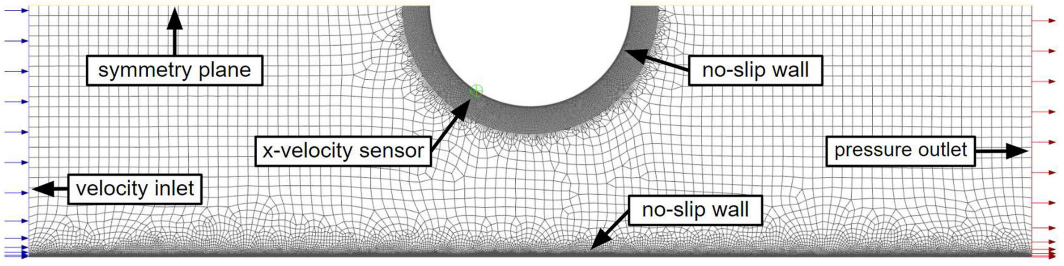


**Fig. 9 Example 2: Mesh and boundary conditions for the axial cylinder within a duct.**

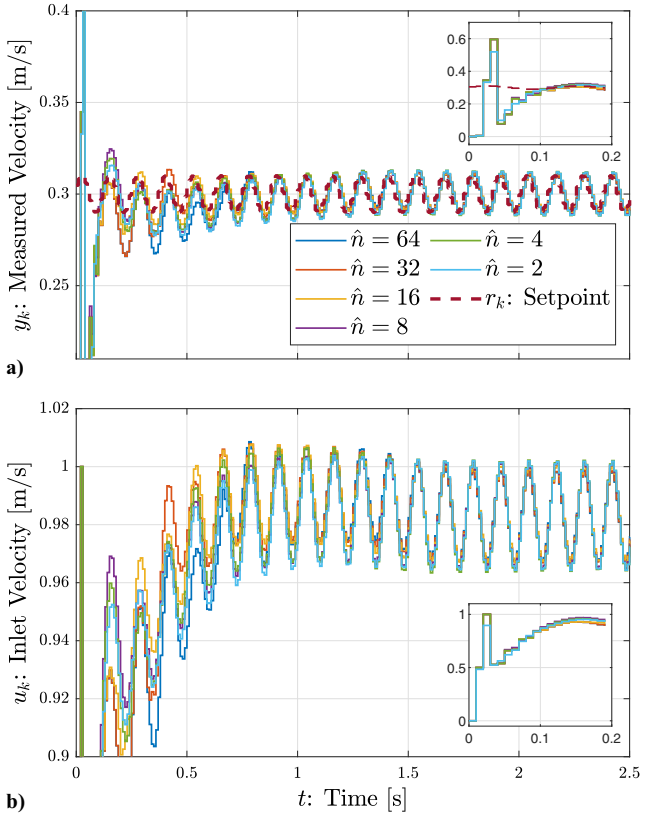


**Fig. 10 Example 2: panel (a) shows similar performance for each identified model order $\hat{n}$ with slight improvements with increased $\hat{n}$.**

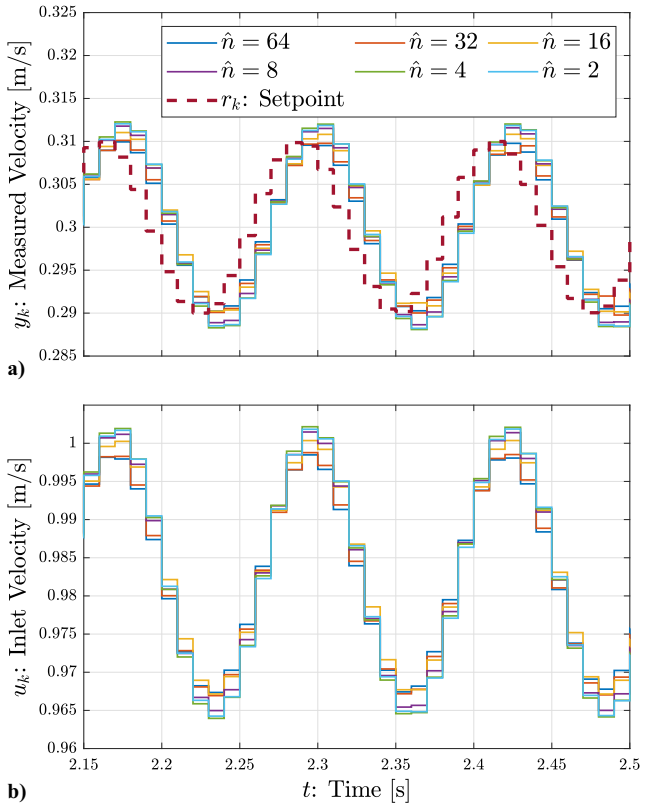


**Fig. 11 Example 2: Sensor measurements (a) and control inputs (b) over the last 0.35 s. As $\hat{n}$ is increased, the command-following performance improves.**

command-following performance. As in Example 1, the incompressibility together with the growth of the viscous boundary layer creates fast and slow time scales that PCAC must deal with in order to follow the command.

## V. Turbulence Suppression

In this section, the flow-control objective is to reduce the turbulent kinetic energy (TKE) at the sensor location. TKE is the mean kinetic energy per unit mass due to turbulent eddies in the flow and can be computed using measurements of velocity fluctuations in time. In contrast to Sec. IV, these examples have a constant, uncontrolled inlet velocity in addition to a second inlet where the velocity is specified by PCAC. One or more TKE sensors are placed at locations of interest to measure the TKE $y_k$, and PCAC specifies a control-inlet velocity $u_k$ in order to achieve the TKE setpoint $r_k$. Note that 0 J/kg is specified as the TKE setpoint at all sensor locations in each of the following examples. While it is not expected that the TKE could be stabilized at 0 J/kg, this setpoint is chosen so that the controller attempts to achieve the lowest TKE possible. In these Reynolds-averaged simulations using the $k-\omega$ model, the TKE is a modeled quantity. As in Sec. IV, no prior model of the fluid dynamics is available to PCAC; PCAC uses RLS with VRF to perform online, closed-loop system identification.

*Example 3: TKE suppression near a backward-facing step.* Reconsider the duct geometry from Example 1 with the addition of a backward-facing step of length 0.2 m and height 0.075 m at the inlet side of the duct, as shown in Fig. 12. The horizontal surface of the step and the lower surface of the duct are no-slip walls, the midheight line of the duct is a symmetry condition, and the right boundary is a pressure outlet. The left boundary, above the step, is a velocity inlet with a velocity that is a prescribed function of time. This velocity defines the freestream and is 5 m/s for $t < 5$ s, 10 m/s for $5 \le t < 10$ s, and 15 m/s for $10 \le t < 15$ s. These velocities correspond to Reynolds numbers of $3.4 \times 10^5$, $6.8 \times 10^5$, and $1.0 \times 10^6$. The vertical edge of the step is a velocity inlet where the velocity is the control input specified by PCAC. At all inlets, the turbulent intensity is 5%, and the turbulent viscosity ratio is 10.

Flow over a backward-facing step is a canonical flow-control problem that has been widely studied [34,35]. While the geometry of the backward-facing step is simple, it serves as a rich fluids problem because it displays separation, reattachment, boundary-layer development, and recirculation, which appear in more complex geometries of engineering interest [35]. Suppression of turbulence resulting from the backward-facing step thus has practical relevance.

The flowfield is initialized to have a velocity of 0 m/s at all locations within the duct. The fluid computation is incompressible, unsteady, viscous, and turbulent using the $k-\omega$ turbulence model. The dynamic viscosity is $\mu = 1.7894 \times 10^{-5}$ kg/(m · s). The flow state is advanced forward in time with a fixed time step of 0.01 s using first-order implicit time stepping. PCAC samples the TKE sensor with a matching time step of 0.01 s. In the absence of control, the separated, fast-moving flow from above the step interacts with the slow-moving, recirculating flow behind the step. This interaction leads to the formation of a shear layer where the mixing of the fast and slow fluid streams produces a region of high TKE. Figure 13 shows the TKE contour for the uncontrolled flow, where a region of high TKE is present downstream and below the step.

For the controlled case, a TKE sensor is placed near the location of the maximum TKE in the uncontrolled case, as indicated by the white crosshair in Fig. 13. PCAC starts at $t = 1$ s. To investigate the performance of PCAC and the effect of the MPC horizon, $\ell$ is varied while all other PCAC parameters are held constant. Figure 14 shows the sensor measurements for $\ell = 10, 25, 50, 100, 200, 300$, and 400. The closed-loop performance is poor for small MPC horizons where, in some instances, the TKE is increased rather than suppressed. Furthermore, long MPC horizons result in poor transient performance when the freestream velocity changes. $\ell = 200$ results in the best performance of the tested horizons.

Figure 15 shows the sensor measurement $y_k$, the control input $u_k$, the estimated model coefficients $\theta_k$, and the forgetting factor $\lambda_k$ for

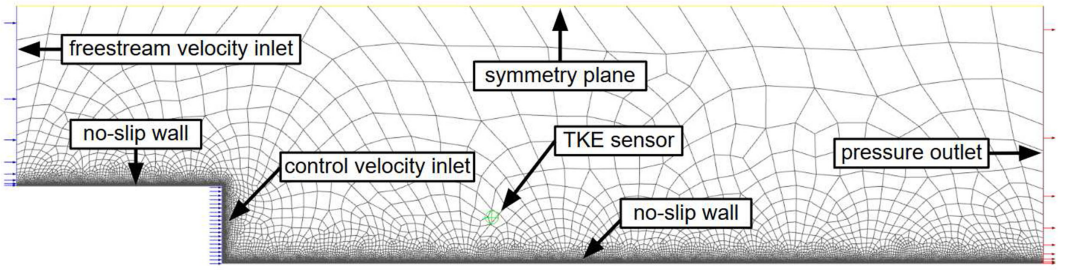


**Fig. 12 Example 3: Mesh and boundary conditions for the backward-facing step within a duct.**

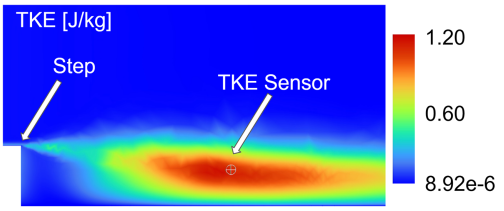


**Fig. 13 Example 3: Partial TKE contour for the uncontrolled flow. The sensor is placed near the maximum TKE.**

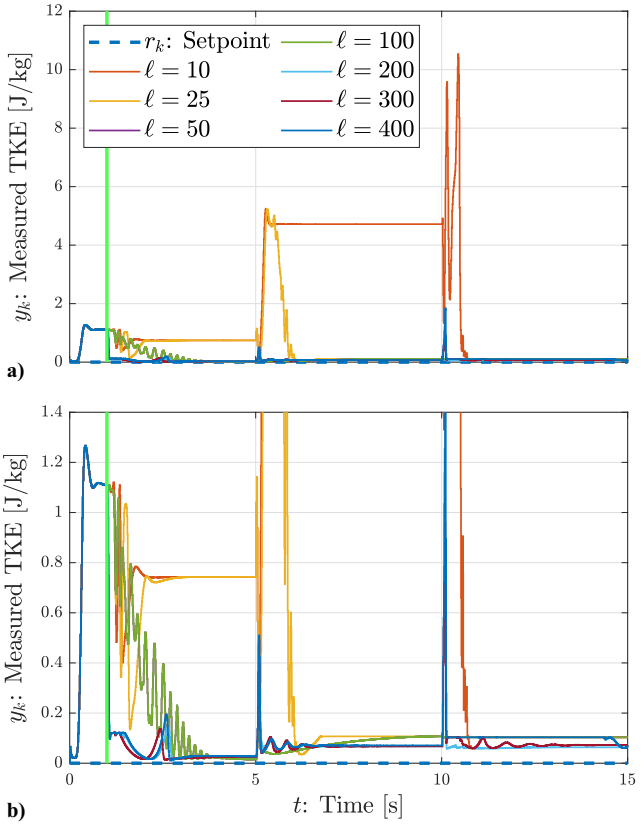


**Fig. 14 Example 3: As the MPC horizon $\ell$ is varied, panels (a) and (b) show that $\ell = 200$ gives the best transient and subsequent response.**

$\ell = 200$. PCAC suppresses turbulence downstream of the backward-facing step by injecting flow through the control jet. This additional flow behind the step reduces the velocity difference between the flow above and below the upper surface of the step. The reduction in the velocity difference reduces the shear between the layers and results in less mixing and recirculation. As a result, the TKE is reduced as shown at $t = 4.98$ s in Fig. 16. As the prescribed freestream velocity is increased, PCAC adapts and increases the control-inlet velocity to continue to suppress the TKE.

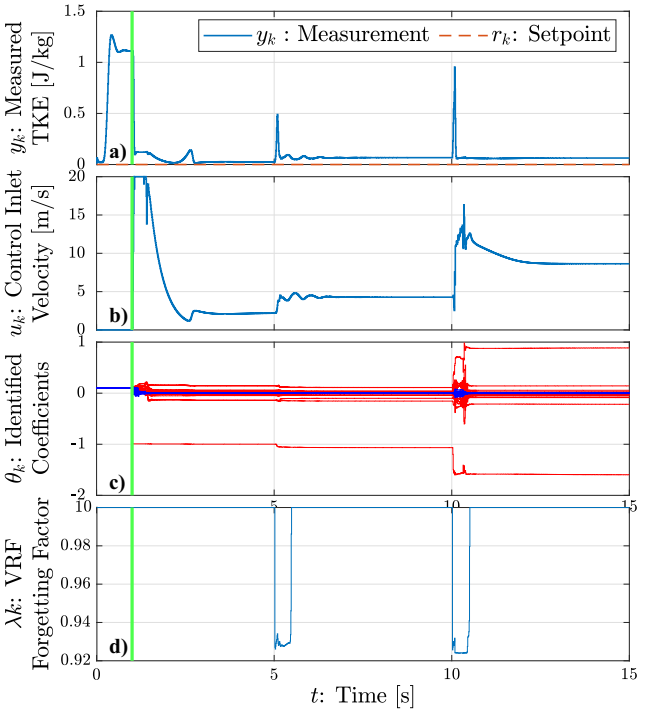


**Fig. 15 Example 3: With $\ell = 200$, a) PCAC reduces the measured TKE. b) As the freestream-inlet velocity is increased, PCAC increases the control-inlet velocity.**

*Example 4: TKE suppression in a 2D duct with crossflow jets.* Consider a 2D duct with a width of 0.25 m and a length of 1 m. The top and bottom boundaries are no-slip walls. The left boundary is a constant, uniform 10 m/s velocity inlet. Based on this velocity, the Reynolds number is $6.81 \times 10^5$. The right boundary is a pressure outlet. On the bottom boundary, 0.333 m downstream of the inlet, is an uncontrolled disturbance jet modeled as a velocity inlet with a constant velocity of 50 m/s. On the top boundary, 0.500 m downstream of the inlet, is a control jet modeled as a velocity inlet with a velocity specified by PCAC. At all inlets, the turbulent intensity is 5%, and the turbulent viscosity ratio is 10. Figure 17 shows the duct

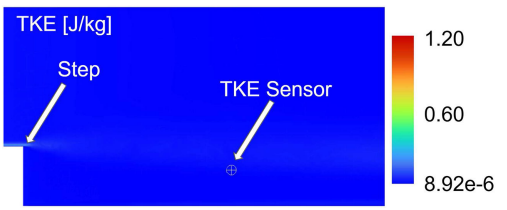


**Fig. 16 Example 3: TKE for the controlled flow. The mixing region present in the uncontrolled flow is removed.**

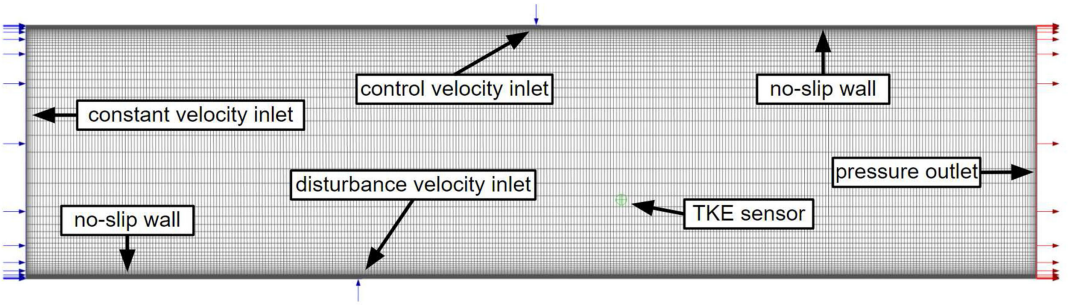


**Fig. 17 Example 4: Mesh and boundary conditions for the duct with crossflow jets.**

mesh, the location of the constant, uncontrolled disturbance jet, and the location of the PCAC-actuated control jet.

The flowfield is initialized to have a velocity of 0 m/s at all locations within the duct. The fluid computation is incompressible, unsteady, viscous, and turbulent using the $k-\omega$ turbulence model. The dynamic viscosity is $\mu = 1.7894 \times 10^{-5}$ kg/(m · s). The flow state is advanced forward in time with a fixed time step of 0.01 s using first-order implicit time stepping. The disturbance jet injects flow normally to the freestream direction, causing separation immediately downstream of the jet, as shown in the velocity contour in Fig. 18. This separation, and the resulting recirculation of the flow, produces a shear layer downstream of the jet and near the midheight line of the duct. As in Example 3, this shear layer results in high TKE. In the absence of control, this region of high TKE is shown in Fig. 19.

To enable feedback control, a TKE sensor, indicated by the white crosshair, is placed in this high-TKE region. PCAC is enabled at 0.5 s, allowing the flow to develop before the controller starts. In this example, the maximum control-jet velocity is constrained to be 40 m/s. Figure 20 shows the sensor measurement $y_k$, the control input $u_k$, the estimated model coefficients $\theta_k$, and the forgetting factor $\lambda_k$. PCAC reduces the measured TKE from 43.63 to 0.38 J/kg. PCAC achieves this reduction by requesting flow through the control jet. This actuation shifts the high-TKE mixing region caused by the disturbance jet upstream and away from the TKE sensor, as shown in the TKE contour in Fig. 21.

While PCAC reduced the measured TKE, Fig. 21 shows that PCAC does not globally reduce the TKE. Instead, the mixing region is only moved upstream. In addition, the flow injected by the

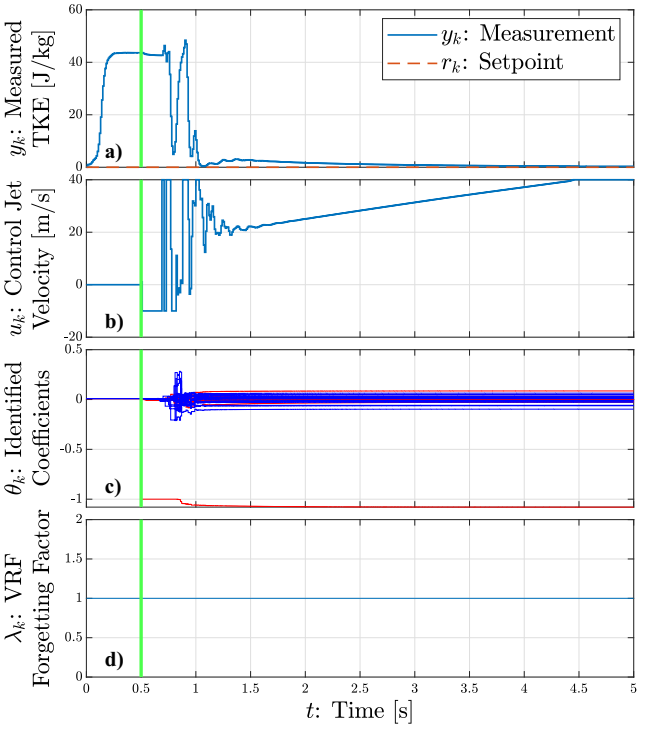


**Fig. 20 Example 4: a) The measured TKE is reduced from 43.63 to 0.38 J/kg. b) The control-jet velocity reaches the 40 m/s constraint.**

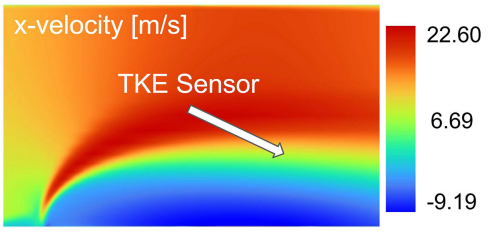


**Fig. 18 Example 4: Partial $x$-velocity contour for the uncontrolled flow. The disturbance jet causes separation.**

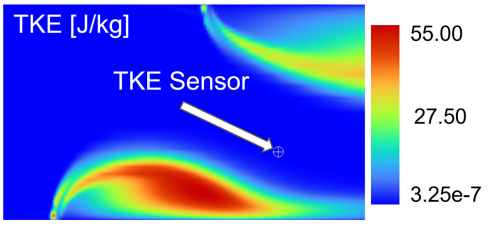


**Fig. 21 Example 4: TKE for the controlled flow. PCAC uses the control jet to move the high-TKE region away from the sensor.**

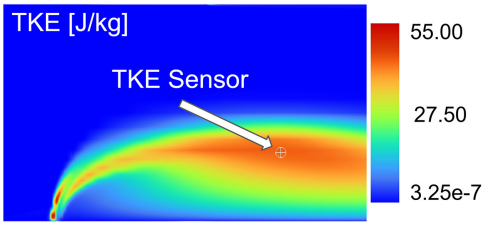


**Fig. 19 Example 4: Partial TKE contour for the uncontrolled flow. The sensor is placed in near the maximum TKE.**

PCAC-actuated control jet induces flow separation and creates another shear layer on the upper surface of the duct, which results in an additional high-TKE region. As a result, a global decrease in the TKE is not achieved. This case shows that PCAC optimizes only the defined control objective to the best of its ability using the given actuator and sensor configuration.

Since PCAC can be used with multiple sensors and actuators, more complex control objectives can be defined. For example, additional sensors can be placed in the duct to allow the location of the turbulence to be specified with greater control. Figure 22

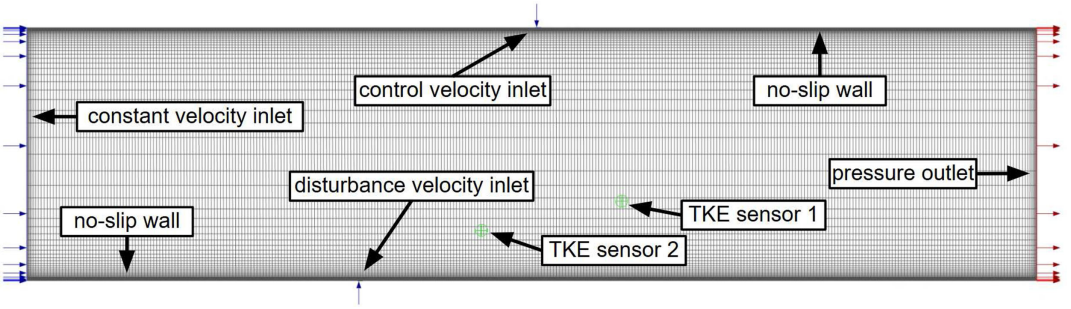


**Fig. 22 Example 4: Duct with 2 TKE sensors indicated with green crosshairs.**

shows the placement of two TKE sensors, where the first is placed in the high-TKE region of the uncontrolled flow and the second is placed upstream in the high-TKE region shown in Fig. 21. Additionally, a control-input constraint is applied such that $0 \text{ m/s} \leq u_k \leq 40 \text{ m/s}$, where the lower limit prevents the actuator from applying suction.

Figure 23 shows the sensor measurements $y_k$, the control input $u_k$, the estimated model coefficients $\theta_k$, and the forgetting factor $\lambda_k$ for the case where the performance at both locations is weighted equally in the MPC cost function. Relative to the uncontrolled case, the TKE at sensor 1 is reduced at the cost of higher TKE at sensor 2. However, compared to the case where only sensor 1 is used, the TKE at sensor 1 is higher since PCAC must now balance the performance at two locations. Specifically, the control objective is to simultaneously reduce the TKE with equal weight on the measured TKE at both locations.

By adjusting the cost-to-go and terminal-value weights applied to the performance variables, the control objective can be modified, and the high-TKE mixing region can be moved relative to the locations of the two sensors. Figure 24 shows the performance with different relative weighings on each sensor. As shown, increasing the weight on the performance at a sensor moves the region of high TKE away from that sensor. When the cost-to-go and terminal-value weights applied to the performance at sensor 1 are 10 times larger

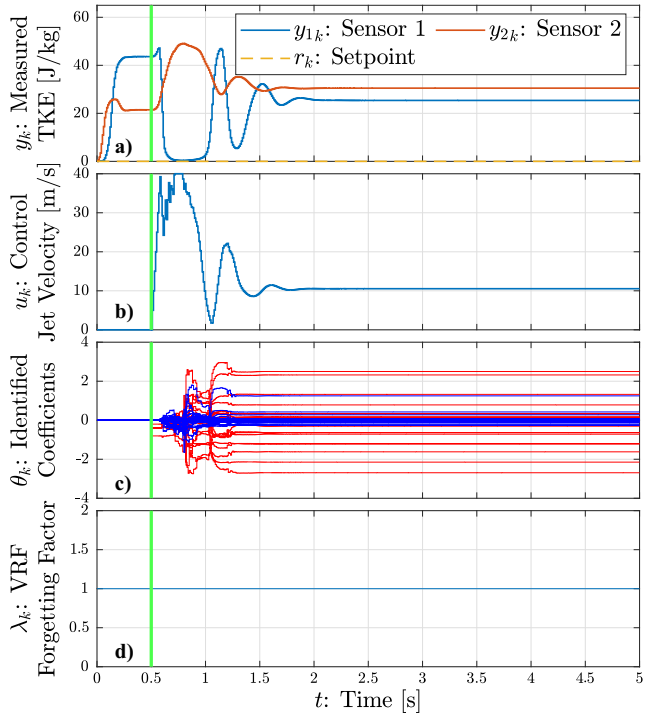


**Fig. 23 Example 4: With equal weight on the performance at both sensors, a) the TKE at sensor 1 decreases while the TKE at sensor 2 increases.**

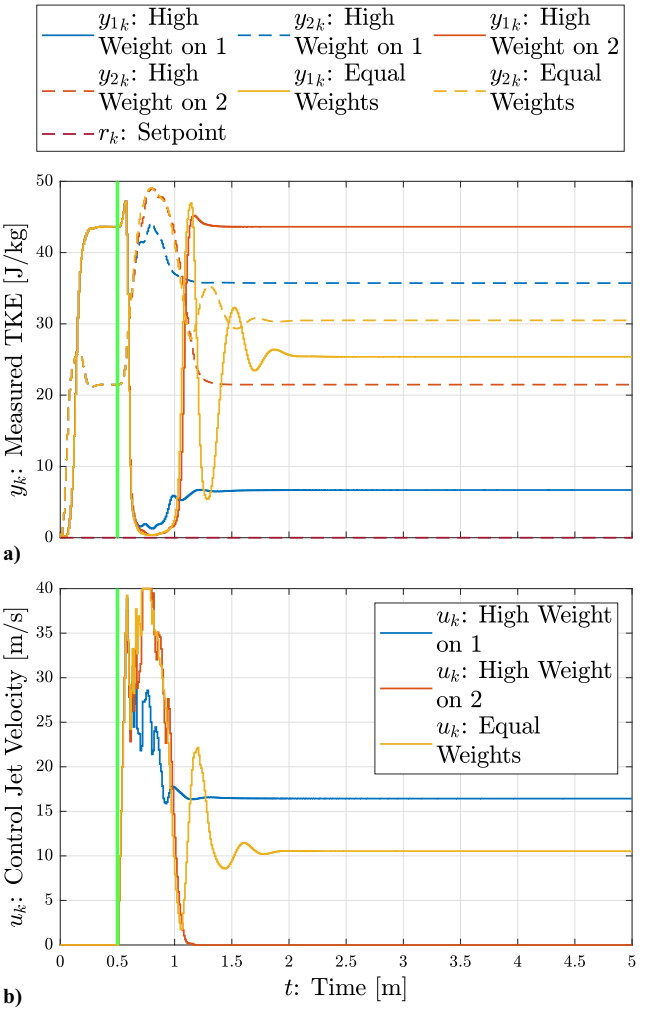


**Fig. 24 Example 4: panel (a) shows that the TKE at each sensor decreases as the weight at that sensor is increased.**

than those at sensor 2, the TKE at sensor 1 is decreased from 43.62 to 4.16 J/kg, and the TKE at sensor 2 is increased from 21.47 to 35.72 J/kg. When the weights applied to the performance at sensor 2 are 10 times larger than those at sensor 1, the TKE at sensors 1 and 2 is unchanged from the uncontrolled values, and the requested velocity at the control inlet is 0 m/s, which is equal to the lower constraint on the control-jet velocity. When the performance at both sensor locations is weighted equally, the TKE at sensor 1 is reduced to 25.38 J/kg, and the TKE at sensor 2 is increased to 30.50 J/kg. In this case, the TKE at both sensors is nearly equal. Figure 25 shows the TKE contours for each set of weights, where the position of the mixing region relative to the sensor locations is shown.

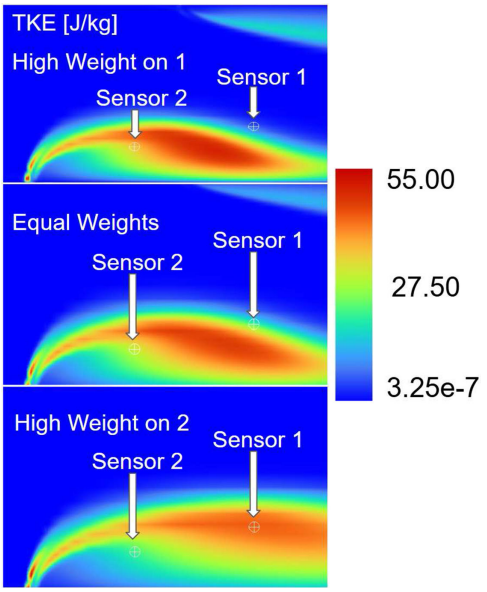


**Fig. 25 Example 4: Partial TKE contours for different relative performance weightings. With equal weighting, the high TKE region is placed nearly equally between the sensors.**

## VI. Airfoil Flow-Separation Control

In this section, PCAC is applied to flow-control problems involving flow separation. This scenario represents a fundamental challenge in aerodynamics, where, under a high angle of attack, the flow becomes detached from the airfoil, leading to a potentially catastrophic loss of lift. By controlling the location of the flow separation, lift can be maintained at a higher angle of attack, and the risk of aerodynamic stall can be reduced. These examples consider suction near the leading edge of the airfoil in conjunction with velocity sensors along the upper surface. By assessing the flow velocity along the upper surface of the airfoil, the location of flow separation can be estimated. Using this estimate, PCAC controls the location at which the flow separates from the airfoil.

Leading-edge suction is a classical actuation approach to influencing airfoil flow separation [36–38]. Removing slow-moving fluid from near the airfoil surface through suction leads to its replacement by faster-moving fluid from farther away from the wall, which strengthens the boundary-layer profile and delays separation. Closed-loop separation control is considered in [39,40]. A fundamental difficulty for controller design is that the nonlinear fluid dynamics, and thus the flow-separation behavior and the effect of the suction slot, vary as functions of Reynolds number. Furthermore, changes in the airfoil angle of attack affect the flow-separation location. Thus, a controller designed for one operating condition may degrade performance at another. Gain scheduling can be used to compensate for changing dynamics by modifying controller behavior as a function of the operating conditions, such as flight speed, angle of attack, or altitude. However, gain scheduling relies on high-fidelity modeling of the airfoil at each operating condition, the design or tuning of controllers for each operating condition, and additional sensors to provide measurements of the operating condition.

This section considers the novel application of PCAC to adaptively modulate the amount of suction in a closed loop to achieve separation at a specified location along the airfoil chord. In contrast to model-based approaches, PCAC uses no prior modeling of the system and instead relies entirely on online identification of the system. Through adaptation, PCAC can automatically compensate for changes in operating conditions without the need for additional measurements or the high-fidelity modeling needed for gain scheduling. The location of flow separation has a significant impact on airfoil performance metrics such as lift and drag. The ability to specify the separation location provides a practical mechanism for altering the flight characteristics of an airfoil. For example, situations that demand high lift, and thus attached flow, include low-speed landing and maneuvers. By contrast, lower lift, and thus separated flow, is desirable to bleed altitude or maneuver without deflecting a control surface.

The following examples consider a NACA 4412 airfoil of chord length 1 m at a high angle of attack. The fluid simulations are 2D incompressible unsteady RANS using the SA turbulence model. The discretization is second-order finite-volume in space, with the mesh shown in Fig. 26, and first-order implicit in time, with a fixed time step of 0.01 s. In the following examples, the nominal freestream velocity is 15 m/s. This nominal freestream velocity corresponds to a Reynolds number of $1.03 \times 10^6$.

An array of $x$-velocity sensors is used to estimate the flow-separation location. This sensor array can be viewed as a discrete measurement of the flow-separation location. At the cost of a larger number of sensors, the improved resolution of the sensor array allows the separation location to be estimated more accurately, as shown in the following examples. These sensors are placed along the upper surface of the airfoil at evenly spaced locations in the range $x \in [0.2000, 0.9800]$ m, where $x = 0$ m denotes the leading edge and $x = 1$ m denotes the trailing edge. By measuring the $x$-velocity at multiple locations, the flow-separation location can be estimated as the position of the forward-most sensor measuring a negative $x$-velocity, where a negative velocity indicates that the flow has reversed direction and is moving toward the airfoil leading edge. Using this estimate, PCAC computes the suction velocity $u_k$, which is applied by modifying the boundary condition at the location of the suction slot. The suction slot is oriented at a 45 deg angle relative to the positive $x$-direction, as shown in Fig. 27. A negative control-jet velocity is defined to represent suction into the slot.

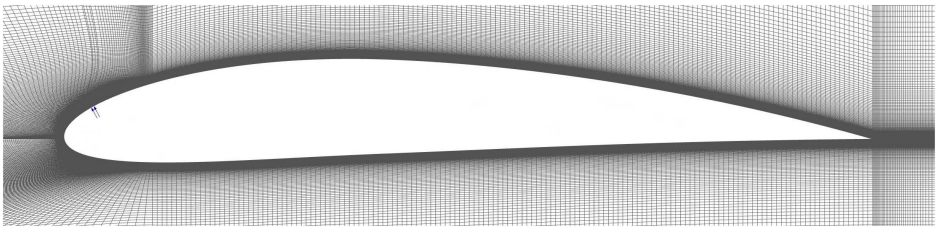

**Fig. 26 Zoomed view of the NACA 4412 mesh. The full mesh of 196,300 nodes extends to approximately 8 chords in all directions.**

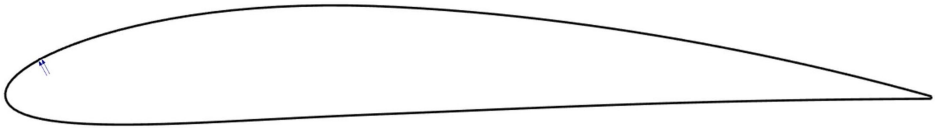

**Fig. 27 Airfoil with the suction-slot location (blue arrows) indicated. A negative suction-slot flow is oriented at 180 deg relative to the arrows.**

*Example 5: Single separation-location setpoint*. In this example, a single separation-location setpoint is specified. A sensor array of 39 sensors, shown in Fig. 28, is used to assess the flow-separation location. The angle of attack of the airfoil is 20 deg, and the freestream velocity is 15 m/s, both of which are constant. Before the controller is enabled, the separation-location estimate is $x = 0.3437$ m. PCAC is enabled at 3 s with the setpoint $x = 0.6721$ m, which coincides with the location of an $x$-velocity sensor. Figure 29 shows the sensor measurement $y_k$, the control input $u_k$, the estimated model coefficients $\theta_k$, and the forgetting factor $\lambda_k$. Under closed-loop control, Fig. 29 shows that, in approximately 7.5 s, the estimated separation location moves from $x = 0.3437$ to 0.6721 m.

*Example 6: Sequence of separation-location setpoints*. This example extends Example 5 by specifying a sequence of separation-location setpoints. The angle of attack of the airfoil is 20 deg, and the freestream velocity is 15 m/s, both of which are constant. PCAC is applied to the airfoil at 3 s with the sequence of setpoints $x = 0.5079$, 0.5900, 0.6721, and 0.5079 m. These setpoints are chosen to coincide with the locations of the sensors. Figure 30 shows the sensor measurement $y_k$, the control input $u_k$, the estimated model coefficients $\theta_k$, and the forgetting factor $\lambda_k$. Under closed-loop control, Fig. 30 shows that, in approximately 9 s, the separation-location estimate is moved from $x = 0.3232$ to 0.4874 m. This estimated location is at the position of the sensor immediately forward of the setpoint position. Following each change in setpoint, the separation-location estimate oscillates between the sensors adjacent to the setpoint in about 5 s. Following the final change in setpoint, the separation-location estimate settles to the setpoint value after 21 s.

To explore the response of PCAC to measurement resolution and to reduce the oscillation around the setpoints, the case is repeated using arrays of 79 and 319 $x$-velocity sensors for comparison. The sensor placement for these arrays is compared with the original 39-sensor array in Fig. 31. The setpoints used with each array are adjusted such that the setpoint always coincides with the location of a sensor.

The results for each array are compared in Fig. 32, where the oscillation near the setpoint is reduced by using an array with higher resolution. In the case of 319 sensors, the oscillation is removed, and instead a small error is present when the separation-location estimate is at the location of a sensor adjacent to the setpoint position. Regardless, the error, defined as the difference between the separation-location estimate and the setpoint, is reduced and stabilized by using the higher-resolution array. Figure 33 shows the sensor measurement $y_k$, the control input $u_k$, the estimated model coefficients $\theta_k$, and the forgetting factor $\lambda_k$ for the 319-sensor array. At the cost of additional sensors, the system performance is improved.

*Example 7: Separation-location setpoint with time-varying angle of attack*. This example extends Example 5 by specifying a single separation-point setpoint and varying the airfoil angle of attack, as shown in Fig. 34. In this case, an array of 319 sensors is used. PCAC is applied to the airfoil after 3 s with the setpoint $x = 0.6709$ m, which coincides with the location of an $x$-velocity sensor. The freestream velocity is held constant at 15 m/s. Initially, the angle of attack of the airfoil is 17 deg. At 40 s, the angle of attack increases to 20 deg in 10 s. As the airfoil angle of attack is increased, the flow separation becomes more severe, and larger suction velocities are necessary to maintain the desired flow-separation location. Figure 34 shows the sensor measurement $y_k$, the control input $u_k$, the estimated model coefficients $\theta_k$, the forgetting factor $\lambda_k$, and the angle of attack. In approximately 12 s, the separation-location estimate is moved from $x = 0.5532$ m to the setpoint, $x = 0.6709$ m. Furthermore, at approximately 15 s after the angle of attack reaches 20 deg,

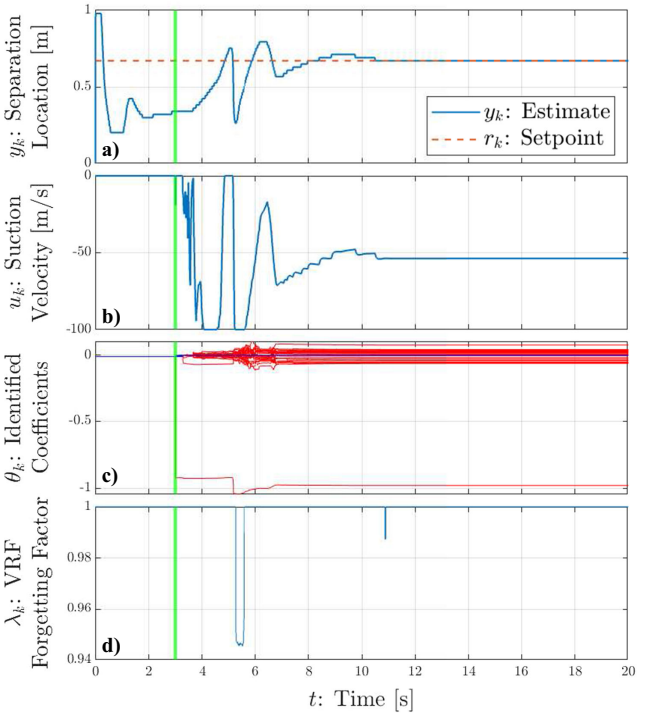


**Fig. 29 Example 5: a) the location of flow separation is moved to the setpoint location in approximately 7.5 s.**

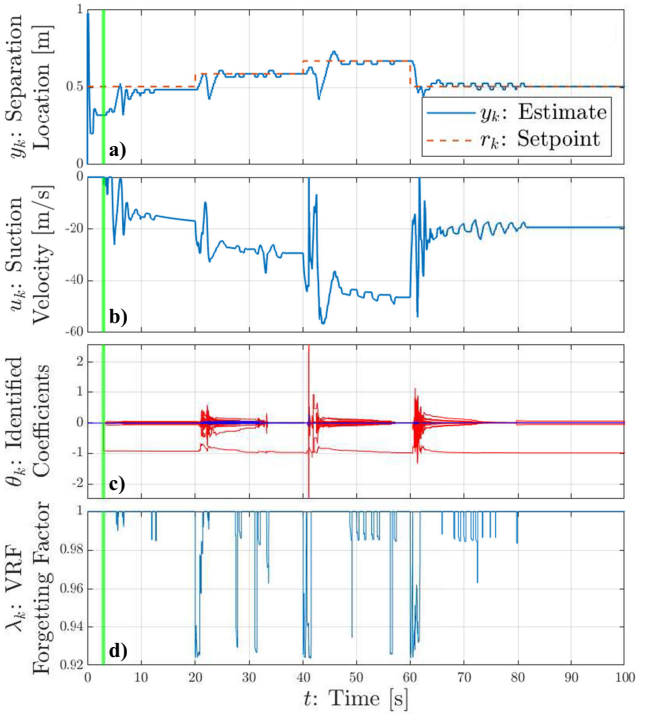


**Fig. 30 Example 6: With PCAC using a low-resolution discrete measurement, a) the separation location oscillates near the setpoint.**

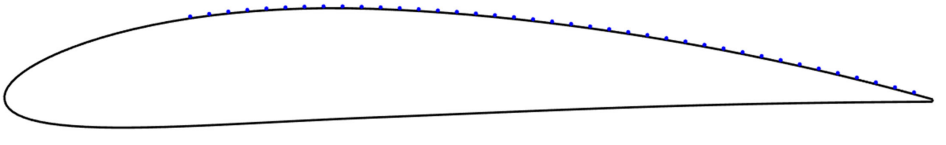

**Fig. 28 Example 5: Airfoil with sensor array. The blue dots represent the positions of the 39 $x$-velocity sensors.**

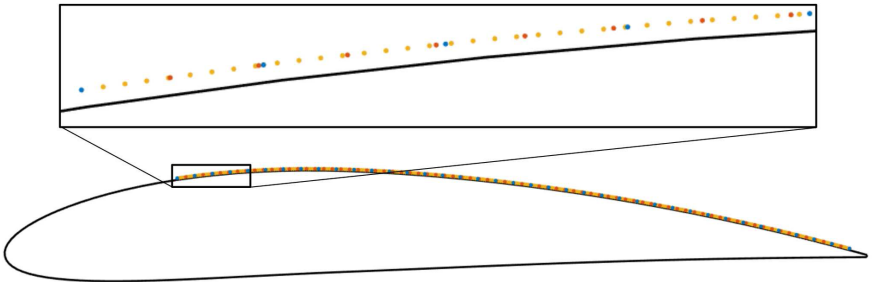

Fig. 31 Example 6: Airfoil with 3 sensor arrays of varying resolution.

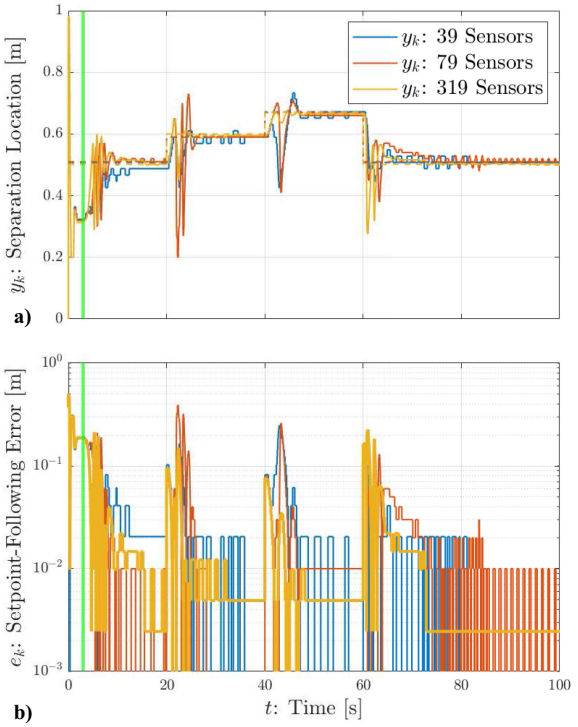


Fig. 32 Example 6: panels (a) and (b) show that the separation location is stabilized to a location near the setpoint as the measurement resolution is increased.

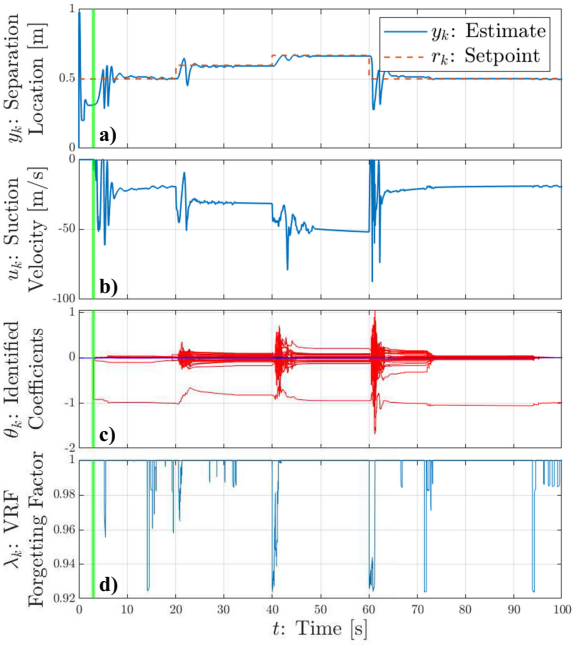


Fig. 33 Example 6: Using the 319-sensor array, a) PCAC moves the separation location near the setpoint.

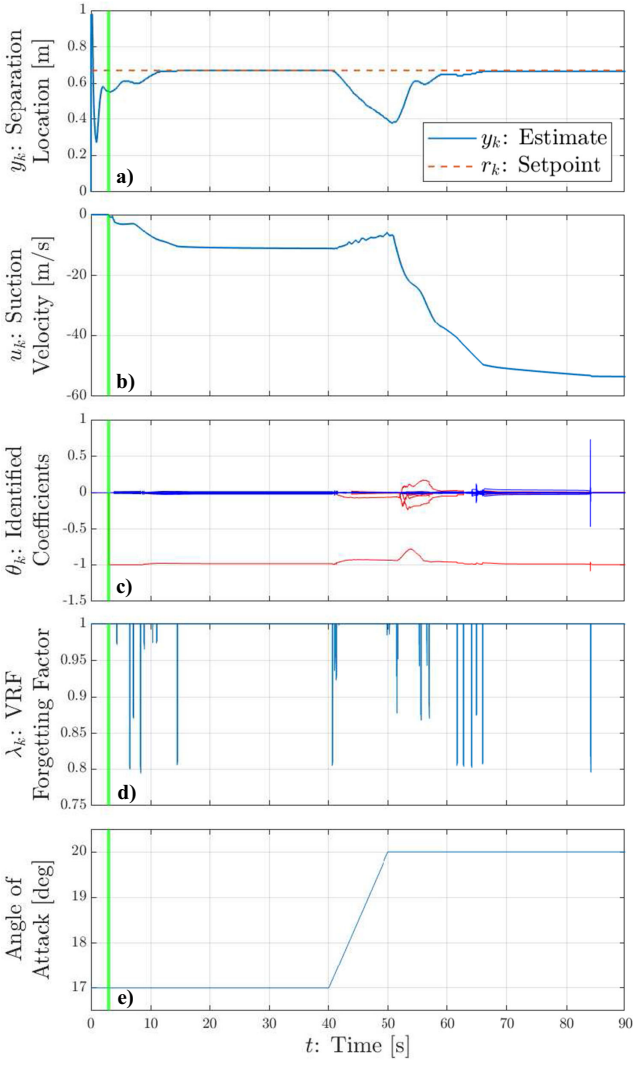


Fig. 34 Example 7: With the angle of attack varying as shown in (e), panel (a) shows that the separation location approaches the setpoint.

PCAC moves the separation-location estimate to $x = 0.6660$ m, which is near the setpoint. PCAC adapts to the changing angle of attack without the need for gain scheduling and using only the estimated location of the flow separation.

## VII. Conclusions

This work presented a numerical study of active flow control using PCAC coupled with a series of 2D flow-control cases simulated in ANSYS Fluent. PCAC, which combines continual, online system identification with output-feedback MPC, was used to control complex, nonlinear fluid dynamics in several flow-control scenarios without the need for any prior modeling of the system. While fluid behavior varies substantially as a function of Mach and

Reynolds numbers, PCAC automatically adapted to the changing flow conditions. In particular, PCAC required no prior modeling of the system dynamics in any of the examples, nor did it require the computationally expensive offline training needed for reinforcement-learning approaches.

PCAC controlled the flow velocity within the viscous boundary layer in a duct by specifying the duct inlet velocity. Compared to PI control, PCAC was able to more quickly bring the measured velocity to the setpoint velocity. Harmonic command following was demonstrated for the velocity within the boundary layer of an axial cylinder. PCAC performed turbulence suppression by reducing the TKE at a specified location within a duct with a backward-facing step. The ability of PCAC to adapt was demonstrated by varying the freestream velocity. Next, a duct with crossflow jets, one constant and one actuated by PCAC, was tested, and it was shown that, by using multiple sensors, the high-turbulence region could be moved by modifying the cost-to-go and terminal-value weights in the MPC cost function. Finally, a practically-relevant case involving flow separation over an airfoil at a high angle of attack was tested, where PCAC used discrete estimates of the flow-separation location in order to actuate a suction slot on the airfoil leading edge. It was shown that increased measurement resolution resulted in improved performance. The ability of PCAC to automatically adapt to changing airfoil angles of attack without the need for measurements of the angle of attack or gain scheduling was demonstrated.

PCAC was shown to be a practical alternative to model-based flow-control methods, which require substantial control-oriented modeling for each flow condition, and it was shown that a strength of the method is the ability to control these types of systems without the need for extensive modeling, training, or tuning. Instead, the control design process required only limited trials to select controller hyperparameters, while the controller implementation relied on continual, closed-loop system identification. The ability of PCAC to automatically adapt to changing dynamics further simplifies controller design by eliminating the need for gain scheduling. This research shows that PCAC has the potential for enabling more complex flow-control applications with significant impacts on the efficiency and performance of future flight vehicles.

Future extensions of this work will include the application of PCAC to increasingly challenging flow-control scenarios of practical engineering relevance. In particular, 3D cases will be considered, as well as multi-input, multi-output control architectures employing multiple sensors and actuators. The relationship between separation setpoint and airfoil performance will be studied to enable the development of a method for systematically choosing the setpoint to optimize airfoil performance. Higher-Mach-number cases will be studied to assess the performance of PCAC when applied to transonic compressible flow problems such as buffeting. Finally, flow-control experiments will be performed to assess the performance of PCAC in practice.

## Acknowledgment

This research was supported by the Office of Naval Research under Grant N00014-22-1-2457.

## References

[1] Bewley, T. R., "Flow Control: New Challenges for a New Renaissance," *Progress in Aerospace Sciences*, Vol. 37, No. 1, 2001, pp. 21–58. https://doi.org/10.1016/S0376-0421(00)00016-6

[2] Aamo, O. M., and Krstic, M., *Flow Control by Feedback: Stabilization and Mixing*, Springer, Berlin, 2003. https://doi.org/10.1115/1.1584420

[3] Gunzburger, M. D., *Perspectives in Flow Control and Optimization*, SIAM, Philadelphia, PA, 2002. https://doi.org/10.1137/1.9780898718720

[4] Gad-el Hak, M., *Flow Control: Passive, Active, and Reactive Flow Management*, Cambridge Univ. Press, New York, 2000. https://doi.org/10.1017/CBO9780511529535

[5] Joslin, R. D., and Miller, D. N. (eds.), *Fundamentals and Applications of Modern Flow Control*, AIAA, Reston, VA, 2009. https://doi.org/10.2514/4.479892

[6] Barbu, V., *Stabilization of Navier–Stokes Flows*, Springer, Berlin, 2011. https://doi.org/10.1007/978-0-85729-043-4

[7] Cattafesta III, L. N., and Sheplak, M., "Actuators for Active Flow Control," *Annual Review of Fluid Mechanics*, Vol. 43, No. 1, 2011, pp. 247–272. https://doi.org/10.1146/annurev-fluid-122109-160634

[8] Luhar, M., Sharma, A. S., and McKeon, B. J., "Opposition Control within the Resolvent Analysis Framework," *Journal of Fluid Mechanics*, Vol. 749, May 2014, pp. 597–626. https://doi.org/10.1017/jfm.2014.209

[9] Brunton, S. L., and Noack, B. R., "Closed-Loop Turbulence Control: Progress and Challenges," *Applied Mechanics Reviews*, Vol. 67, No. 5, 2015. https://doi.org/10.1115/1.4031175

[10] Duriez, T., Brunton, S. L., and Noack, B. R., *Machine Learning Control—Taming Nonlinear Dynamics and Turbulence*, Springer, Berlin, 2017. https://doi.org/10.1007/978-3-319-40624-4

[11] Wang, J., and Feng, L., *Flow Control Techniques and Applications*, Cambridge Univ. Press, New York, 2019. https://doi.org/10.1017/9781316676448

[12] Bewley, T. R., and Liu, S., "Optimal and Robust Control and Estimation of Linear Paths to Transition," *Journal of Fluid Mechanics*, Vol. 365, June 1998, pp. 305–349. https://doi.org/10.1017/S0022112098001281

[13] Mushtaq, T., Seiler, P., and Hemati, M. S., "Feedback Control of Transitional Flows: A Framework for Controller Verification Using Quadratic Constraints," *AIAA Aviation Forum, virtual*, AIAA Paper 2021-2825, 2021. https://doi.org/10.2514/6.2021-2825

[14] Joshi, S. S., Speyer, J. L., and Kim, J., "A Systems Theory Approach to the Feedback Stabilization of Infinitesimal and Finite-Amplitude Disturbances in Plane Poiseuille Flow," *Journal of Fluid Mechanics*, Vol. 332, Feb. 1997, pp. 157–184. https://doi.org/10.1017/S0022112096003746

[15] Joshi, S. S., Speyer, J. L., and Kim, J., "Finite Dimensional Optimal Control of Poiseuille Flow," *Journal of Guidance, Control, and Dynamics*, Vol. 22, No. 2, 1999, pp. 340–348. https://doi.org/10.2514/2.4383

[16] Gautier, N., Aider, J.-L., Duriez, T., Noack, B., Segond, M., and Abel, M., "Closed-Loop Separation Control Using Machine Learning," *Journal of Fluid Mechanics*, Vol. 770, No. 5, 2015, pp. 442–457. https://doi.org/10.1017/jfm.2015.95

[17] Nguyen, T. W., Islam, S. A. U., Bernstein, D. S., and Kolmanovsky, I. V., "Predictive Cost Adaptive Control: A Numerical Investigation of Persistency, Consistency, and Exigency," *IEEE Control Systems*, Vol. 41, No. 6, 2021, pp. 64–96. https://doi.org/10.1109/MCS.2021.3107647

[18] Kamaldar, M., Mohseni, N., Islam, S. A. U., and Bernstein, D. S., "A Numerical and Experimental Investigation of Predictive Cost Adaptive Control for Noise and Vibration Suppression," *Mechanical Systems and Signal Processing*, Vol. 221, Dec. 2024, pp. 1–42. https://doi.org/10.1016/j.ymssp.2024.111711

[19] Parades, J., and Bernstein, D. S., "Experimental Application of Predictive Cost Adaptive Control to Thermoacoustic Oscillations in a Rijke Tube with Unknown Input Delay," *Proceedings of the American Control Conference*, 2025, pp. 1864–1869. https://doi.org/10.23919/ACC63710.2025.11107843

[20] Richards, R. J., Islam, S. A. U., and Bernstein, D. S., "Predictive Cost Adaptive Control of the NASA Benchmark Flutter Model," *Journal of Guidance, Control, and Dynamics*, Vol. 48, No. 12, 2025, pp. 2663–2679. https://doi.org/10.2514/1.G008859

[21] Stout, E. N., Kumar, H., Schaaf, J. C. V., Islam, S. A. U., Fidkowski, K., and Bernstein, D., "Lift and L/D Enhancement in Vortex-Shedding Flows Using Predictive Cost Adaptive Control," *AIAA SciTech 2026 Forum*, AIAA Paper 2026-1697, 2026. https://doi.org/10.2514/6.2026-1697

[22] Vander Schaaf, J. C., Lu, Q., Kumar, H., Ozmadenci, B., Fidkowski, K., and Bernstein, D., "Predictive Cost Adaptive Control of In-Ground-Effect Flight Using Active Flow Control with Unmodeled, Unsteady Aerodynamics," *AIAA SciTech 2025 Forum*, AIAA Paper 2025-0033, 2025. https://doi.org/10.2514/6.2025-0033

[23] Vander Schaaf, J. C., Fidkowski, K., and Bernstein, D., "Adaptive Control of 2D In-Ground-Effect Flight with Distributed Pressure Sensing," *AIAA*

*SciTech 2026 Forum*, AIAA Paper 2026-1505, 2026. https://doi.org/10.2514/6.2026-1505

[24] Vander Schaaf, J. C., Lu, Q., Fidkowski, K. J., and Bernstein, D. S., "Data-Driven Model Predictive Control of Airfoil Flow Separation," *Proceedings of the American Control Conference*, 2024, pp. 1568–1573. https://doi.org/10.23919/ACC60939.2024.10644206

[25] Vander Schaaf, J., Fidkowski, K. J., and Bernstein, D. S., "Active Internal Flow Control Using Adaptive Model Predictive Control with Online, Closed-Loop System Identification," *2025 AIAA SciTech Forum*, AIAA Paper 2025-0030, 2025. https://doi.org/10.2514/6.2025-0030

[26] Bruce, A. L., Goel, A., and Bernstein, D. S., "Convergence and Consistency of Recursive Least Squares with Variable-Rate Forgetting," *Automatica*, Vol. 119, 2020, Paper 109052. https://doi.org/10.1016/j.automatica.2020.109052

[27] Mohseni, N., and Bernstein, D. S., "Recursive Least Squares with Variable-Rate Forgetting Based on the F-Test," *Proceedings of the American Control Conference*, 2022, pp. 3937–3942. https://doi.org/10.23919/ACC53348.2022.9867849

[28] McKeon, J. J., "F Approximations to the Distribution of Hotelling's $T_0^2$," *Biometrika*, Vol. 61, No. 2, 1974, pp. 381–383. https://doi.org/10.1093/biomet/61.2.381

[29] Kwon, W., and Han, S., *Receding Horizon Control: Model Predictive Control for State Models*, Springer, Berlin, 2006. https://doi.org/10.1007/b136204

[30] Rawlings, J. B., Mayne, D. Q., and Diehl, M. M., *Model Predictive Control: Theory, Computation, and Design*, Nob Hill, 2020, https://books.google.com/books/about/Model_Predictive_Control.html?id=fDLz0AEACAAJ

[31] Tao, J., Ma, L., and Zhu, Y., "Improved Control Using Extended Non-Minimal State Space MPC and Modified LQR for a Kind of Nonlinear Systems," *ISA Transactions*, Vol. 65, Nov. 2016, pp. 319–326. https://doi.org/10.1016/j.isatra.2016.08.015

[32] Eren, U., Prach, A., Koçer, B. B., Raković, S. V., Kayacan, E., and Açıkmeşe, B., "Model Predictive Control in Aerospace Systems: Current State and Opportunities," *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 7, 2017, pp. 1541–1566. https://doi.org/10.2514/1.G002507

[33] White, F. M., *Viscous Fluid Flow*, 2nd ed., McGraw Hill, New York, 1991.

[34] Xiang, J., Zong, H., Wu, Y., Li, J., and Liang, H., "Experimental Control of the Flow Separation Behind a Backward Facing Step Using Deep Reinforcement Learning," *Physics of Fluids*, Vol. 36, No. 10, 2024. https://doi.org/10.1063/5.0231459

[35] McQueen, T., Burton, D., Sheridan, J., and Thompson, M. C., "Active Control of Flow over a Backward-Facing Step at High Reynolds Numbers," *International Journal of Heat and Fluid Flow*, Vol. 93, Feb. 2022, Paper 108891. https://doi.org/10.1016/j.ijheatfluidflow.2021.108891

[36] Atik, H., Kim, C.-Y., Van Dommelen, L. L., and Walker, J. D., "Boundary-Layer Separation Control on a Thin Airfoil Using Local Suction," *Journal of Fluid Mechanics*, Vol. 535, July 2005, pp. 415–443. https://doi.org/10.1017/S002211200500501X

[37] Zhang, W., Zhang, Z., Chen, Z., and Tang, Q., "Main Characteristics of Suction Control of Flow Separation of an Airfoil at Low Reynolds Numbers," *European Journal of Mechanics - B/Fluids*, Vol. 65, 2017, pp. 88–97. https://doi.org/10.1016/j.euromechflu.2017.01.010

[38] Yousefi, K., and Saleh, R., "Three-Dimensional Suction Flow Control and Suction Jet Length Optimization of NACA 0012 Wing," *Meccanica*, Vol. 50, No. 6, 2015, pp. 1481–1494. https://doi.org/10.1007/s11012-015-0100-9

[39] Obeid, S., Ahmadi, G., and Jha, R., "NARMAX Identification Based Closed-Loop Control of Flow Separation over NACA 0015 Airfoil," *Fluids*, Vol. 5, No. 3, 2020, p. 100. https://doi.org/10.3390/fluids5030100

[40] Lee, B., Kim, M., Choi, B., Kim, C., Kim, H. J., and Jung, K. J., "Closed-Loop Active Flow Control of Stall Separation Using Synthetic Jets," *31st AIAA Applied Aerodynamics Conference*, AIAA Paper 2013-2925, 2013. https://doi.org/10.2514/6.2013-2925

C. A. Woolsey
*Associate Editor*