OPTIMIZATION COURSE — PROJECT PROPOSALS

# Optimizing Simple Flows: Algorithms and Starting Papers

*Seven algorithm families, each with a short explanation, a set of papers and resources to start from, and a suggested project for students with a fluid-mechanics background.*

**A note on the references.** This list was compiled without access to a literature database. The papers and books are well-known works in their fields, but volume numbers and the exact titles of some items should be checked against the publisher or a library catalogue before relying on them. Items with a higher chance of error are marked "(verify)".

## Overview

| # | Topic | Core idea in one line | Suggested test bed | Difficulty |
|---|---|---|---|---|
| **1** | Adjoint-based passive control | One adjoint solve gives the cost gradient with respect to all design parameters | Obstacle in Stokes or low-Re flow; cylinder wake, Re 40–100 | Hard |
| **2** | Extended Kalman filter | Predict with the model, correct with measurements; estimates feed feedback control | Stuart–Landau or Ginzburg–Landau model; 1D Burgers | Medium |
| **3** | Ensemble variational methods | Variational cost, with gradients and covariances estimated from an ensemble instead of an adjoint | 1D Burgers; 2D laminar flow | Hard |
| **4** | Kriging / neural-network surrogates | Fit a cheap model to a few CFD runs, then optimize the model | Parameterized airfoil (6–10 parameters) | Easy–Medium |
| **5** | Genetic algorithms for flow control | Population search over controller parameters; no gradients | Rotating or oscillating cylinder, Re ≈ 100 | Easy–Medium |
| **6** | Reinforcement learning | Learn a feedback policy by trial and error against a reward | Kuramoto–Sivashinsky equation, then 2D cylinder wake | Hard (compute) |
| **7** | Data assimilation for flows | Combine a model with sparse, noisy data to reconstruct the flow or its parameters | Identical-twin experiment on a cylinder wake | Medium–Hard |

## 1. Adjoint-based passive control

### The algorithm in brief

Each optimization iteration needs one forward flow solve and one adjoint solve. The adjoint solution delivers the gradient of a cost (drag, shedding amplitude, pressure loss) with respect to every design parameter simultaneously, at a cost that does not grow with the number of parameters. A gradient-based optimizer (steepest descent, BFGS) then updates the design. "Passive" means no energy is added: the design variable is a shape, a topology, or the position of a small fixed object. This is the Lagrangian and adjoint machinery from Lecture 3 applied to a flow.

### Starting papers and resources

- **Jameson (1988)**, *Aerodynamic design via control theory*, J. Sci. Comput. 3. The classic adjoint shape-optimization paper.
- **Giles & Pierce (2000)**, *An introduction to the adjoint approach to design*, Flow, Turbulence and Combustion 65. Short and readable; a good first read.
- **Marquet, Sipp & Jacquin (2008)**, *Sensitivity analysis and passive control of cylinder flow*, J. Fluid Mech. 615. Adjoint sensitivity used to place a small control cylinder in the wake — an excellent template for a student project.
- **Giannetti & Luchini (2007)**, *Structural sensitivity of the first instability of the cylinder wake*, J. Fluid Mech. 581.
- **Borrvall & Petersson (2003)**, *Topology optimization of fluids in Stokes flow*, Int. J. Numer. Methods Fluids 41. A small, tractable problem.
- **Gunzburger (2003)**, *Perspectives in Flow Control and Optimization*, SIAM (book). Also Mohammadi & Pironneau, Applied Shape Optimization for Fluids.
- **Software:** dolfin-adjoint with FEniCS (Farrell et al., SIAM J. Sci. Comput. 2013) generates adjoints automatically; OpenFOAM-based adjoint frameworks such as DAFoam exist (verify the reference).

> **Possible project:** Optimize the shape or position of an obstacle in Stokes or low-Reynolds-number laminar flow; a natural second step is minimizing cylinder drag at Re ≈ 40–100.

## 2. Extended Kalman filter for estimation, optimization and control

### The algorithm in brief

The Kalman filter alternates a predict step, in which the model advances the state and its covariance, with an update step, in which a measurement corrects the state through a gain matrix. The extended version linearizes a nonlinear model about the current estimate. It is used in two ways relevant here: (i) as a state estimator whose output feeds a feedback law, as in LQG-style control; and (ii) as an optimizer, since the EKF update is a recursive Gauss–Newton step and can estimate parameters or train a neural network. Its weakness is the cost of storing and propagating the full covariance, which is infeasible for large flows and motivates the ensemble methods of topic 3.

### Starting papers and resources

- **Hœpffner, Chevalier, Bewley & Henningson (2005)**, *State estimation in wall-bounded flow systems. Part 1. Perturbed laminar flows*, J. Fluid Mech. 534.
- **Chevalier, Hœpffner, Bewley & Henningson (2006)**, *State estimation in wall-bounded flow systems. Part 2. Turbulent flows*, J. Fluid Mech. 552.
- **Colburn, Cessna & Bewley (2011)**, *State estimation in wall-bounded flow systems. Part 3. The ensemble Kalman filter*, J. Fluid Mech. 682. Links directly to topic 3.
- **Semeraro, Bagheri, Brandt & Henningson (2013)**, *Transition delay in a boundary layer flow using active control*, J. Fluid Mech. 731. Estimator-based feedback control.
- **Illingworth, Morgans & Rowley (2011)**, *Feedback control of flow resonances using balanced reduced-order models*, J. Sound Vib. 330.

- **Rowley & Dawson (2017)**, *Model reduction for flow analysis and control*, Annu. Rev. Fluid Mech. 49. Review.
- **Brunton & Noack (2015)**, *Closed-loop turbulence control: progress and challenges*, Appl. Mech. Rev. 67. Review.
- **Bell & Cathey (1993)**, *The iterated Kalman filter update as a Gauss–Newton method*, IEEE Trans. Automat. Control 38. For the optimizer view (verify).

> **Possible project:** Use a Stuart–Landau or Ginzburg–Landau reduced model of vortex shedding, or 1D Burgers. Build an EKF from sparse sensors, then close the loop with a simple feedback law.

# 3. Ensemble variational methods

## The algorithm in brief

Variational assimilation or optimization minimizes a cost function J(x) using an adjoint gradient, which requires writing and maintaining adjoint code. Ensemble methods instead run N forward simulations with perturbed inputs and estimate gradients and covariances statistically from the spread. Ensemble-variational (EnVar) methods combine the two: they keep the variational cost function but replace the tangent-linear and adjoint models by ensemble regression. The price is more forward runs; the benefits are no adjoint code and trivially parallel execution. The ensemble-gradient idea is closely related to the evolution-strategy methods of Lecture 4 (cross-entropy, NES, CMA-ES).

## Starting papers and resources

- **Evensen (2009)**, *Data Assimilation: The Ensemble Kalman Filter*, Springer (book).
- **Evensen (2018)**, *Analysis of iterative ensemble smoothers for solving inverse problems*, Comput. Geosci. 22.
- **Emerick & Reynolds (2013)**, *Ensemble smoother with multiple data assimilation*, Comput. Geosci. 55. Easy to implement.
- **Chen, Oliver & Zhang (2009)**, *Efficient ensemble-based closed-loop production optimization*, SPE J. 14. "EnOpt" — from reservoir engineering but transferable.
- **Bannister (2017)**, *A review of operational methods of variational and ensemble-variational data assimilation*, Q. J. R. Meteorol. Soc. 143.
- **Mons, Chassaing, Gomez & Sagaut (2016)**, *Reconstruction of unsteady viscous flows using data assimilation schemes*, J. Comput. Phys. 316. Compares variational and ensemble approaches on flows.
- **Mons et al. (2021)**, *Ensemble-variational assimilation of statistical data in large-eddy simulation*, Phys. Rev. Fluids 6. (verify)
- **Xiao et al. (2016)**, *Quantifying and reducing model-form uncertainties in RANS simulations using EnKF*, J. Comput. Phys. 324.

> **Possible project:** Estimate an unknown inflow or forcing in 1D Burgers or 2D laminar flow, and compare an adjoint approach with an ensemble approach on accuracy and cost.

# 4. Kriging or neural-network surrogate models

## The algorithm in brief

Run a limited set of expensive CFD simulations, fit a cheap statistical model to the results, and optimize the model instead of the simulator. Kriging (a Gaussian process) also returns its own uncertainty, which enables Bayesian optimization: the next simulation is chosen with an acquisition function such as expected improvement, balancing exploitation of good regions against exploration of uncertain ones. Neural networks scale better to many parameters and large data sets but give less natural uncertainty estimates. Chapters 14–16 of the course textbook (Algorithms for Optimization) cover surrogate models, Gaussian processes and surrogate optimization.

## Starting papers and resources

- **Jones, Schonlau & Welch (1998)**, *Efficient global optimization of expensive black-box functions*, J. Global Optim. 13. The EGO paper.
- **Forrester, Sóbester & Keane (2008)**, *Engineering Design via Surrogate Modelling*, Wiley (book).
- **Forrester & Keane (2009)**, *Recent advances in surrogate-based optimization*, Prog. Aerosp. Sci. 45.
- **Queipo et al. (2005)**, *Surrogate-based analysis and optimization*, Prog. Aerosp. Sci. 41.
- **Raissi, Perdikaris & Karniadakis (2019)**, *Physics-informed neural networks*, J. Comput. Phys. 378.
- **Brunton, Noack & Koumoutsakos (2020)**, *Machine learning for fluid mechanics*, Annu. Rev. Fluid Mech. 52. The best overview to start from.
- **Thuerey et al. (2020)**, *Deep learning methods for Reynolds-averaged Navier–Stokes simulations of airfoil flows*, AIAA J. 58.
- **Software:** SMT, the Surrogate Modeling Toolbox (Bouhlel et al., Adv. Eng. Softw. 2019); scikit-learn or GPyTorch for Gaussian processes; BoTorch for Bayesian optimization.

> **Possible project:** Parameterize an airfoil with 6–10 parameters, sample about 50 cases with XFOIL or a RANS solver, fit a Kriging model and run Bayesian optimization. Compare against a genetic algorithm given the same evaluation budget.

# 5. Genetic algorithms for flow control

## The algorithm in brief

The genetic algorithm of Lecture 4, applied to controller parameters. Each individual is a parameter vector (actuation frequency, amplitude, position, …) and its fitness is the drag or separation measured in a simulation or experiment. Selection, crossover and mutation produce new generations. The method is robust, needs no gradient, and tolerates noisy or discontinuous objectives. One distinction worth teaching: much of the evolutionary flow-control literature uses genetic programming, which evolves the functional form of the control law rather than only its parameters.

### Starting papers and resources

- **Milano & Koumoutsakos (2002)**, *A clustering genetic algorithm for cylinder drag optimization*, J. Comput. Phys. 175.
- **Koumoutsakos, Freund & Parekh (2001)**, *Evolution strategies for automatic optimization of jet mixing*, AIAA J. 39.
- **Gautier et al. (2015)**, *Closed-loop separation control using machine learning*, J. Fluid Mech. 770. Genetic programming in an experiment.
- **Duriez, Brunton & Noack (2017)**, *Machine Learning Control: Taming Nonlinear Dynamics and Turbulence*, Springer (book).
- **Raibaudo, Zhong, Noack & Martinuzzi (2020)**, *Machine learning strategies applied to the control of a fluidic pinball*, Phys. Fluids 32.

> **Possible project:** Optimize the rotation rate or oscillation parameters of a cylinder to reduce drag at Re ≈ 100 on a coarse 2D mesh. Compare a genetic algorithm with CMA-ES under the same evaluation budget.

## 6. Reinforcement learning

### The algorithm in brief

An agent observes sensor data, chooses an actuation, and receives a reward such as reduced drag. Over many episodes it learns a policy — usually a neural network — mapping observations to actions. Compared with a genetic algorithm, which tunes fixed control parameters, reinforcement learning produces a feedback policy whose response depends on the instantaneous flow state; the cost is far more simulation episodes. The score-function (log-likelihood) gradient used by natural evolution strategies in Lecture 4 is closely related to the policy-gradient estimators used in RL.

### Starting papers and resources

- **Rabault, Kuchta, Jensen, Réglade & Cerardi (2019)**, *Artificial neural networks trained through deep reinforcement learning discover control strategies for active flow control*, J. Fluid Mech. 865. The standard starting point, with public code. Tang et al. (2020, Phys. Fluids 32) extends it to a range of Reynolds numbers.
- **Garnier et al. (2021)**, *A review on deep reinforcement learning for fluid mechanics*, Comput. Fluids 225.
- **Vignon, Rabault & Vinuesa (2023)**, *Recent advances in applying deep reinforcement learning for flow control*, Phys. Fluids 35.
- **Fan, Yang, Wang, Triantafyllou & Karniadakis (2020)**, *Reinforcement learning for bluff body active flow control in experiments and simulations*, Proc. Natl. Acad. Sci. 117.
- **Verma, Novati & Koumoutsakos (2018)**, *Efficient collective swimming by harnessing vortices through deep reinforcement learning*, Proc. Natl. Acad. Sci. 115.
- **Bucci et al. (2019)**, *Control of chaotic systems by deep reinforcement learning*, Proc. R. Soc. A 475.
- **Background and tools:** Sutton & Barto, Reinforcement Learning: An Introduction; the PPO algorithm (Schulman et al., 2017, arXiv); Stable-Baselines3.

> **Possible project:** Start on a cheap environment such as the Kuramoto–Sivashinsky equation or 1D Burgers, then move to the 2D cylinder wake.

# 7. Data assimilation for flows

## The algorithm in brief

Data assimilation combines a model with sparse, noisy measurements — probe signals or PIV — to reconstruct the full flow state or unknown parameters. Variational methods (4D-Var) minimize a misfit between model and data using an adjoint gradient. Sequential methods (Kalman filter, EnKF) update the state as data arrive. Topics 2 and 3 supply the algorithms; this topic is the application, and the unifying viewpoint is that assimilation is an optimization problem constrained by the flow equations.

## Starting papers and resources

- **Foures, Dovetta, Sipp & Schmid (2014)**, *A data-assimilation method for Reynolds-averaged Navier–Stokes-driven mean flow reconstruction*, J. Fluid Mech. 759.
- **Symon, Sipp & Schmid (2017)**, *Data assimilation of mean velocity from 2D PIV measurements of flow over an idealized airfoil*, Exp. Fluids 58. (verify)
- **Suzuki (2012)**, *Reduced-order Kalman-filtered hybrid simulation combining particle tracking velocimetry and direct numerical simulation*, J. Fluid Mech. 709.
- **Mons, Chassaing, Gomez & Sagaut (2016)**, *Reconstruction of unsteady viscous flows using data assimilation schemes*, J. Comput. Phys. 316.
- **Asch, Bocquet & Nodet (2016)**, *Data Assimilation: Methods, Algorithms, and Applications*, SIAM (book).
- **Law, Stuart & Zygalakis (2015)**, *Data Assimilation: A Mathematical Introduction*, Springer (book).

> **Possible project:** An identical-twin experiment: generate synthetic "measurements" from a known 2D cylinder wake, then try to recover a hidden parameter (the Reynolds number or an inflow profile) with 4D-Var and with an ensemble Kalman filter.

# Practical suggestions

## Shared test beds

Use 1D Burgers or the Kuramoto–Sivashinsky equation for fast prototyping, then a laminar cylinder wake at Re ≈ 40–100 for realism. If two groups attack the same flow with different algorithms, the final comparison becomes meaningful: same objective, same budget, different method.

## Suggested difficulty order, easiest first

Surrogates (4), genetic algorithms (5), EKF or EnKF on a reduced model (2), data assimilation (7), reinforcement learning (6), adjoint on a full solver (1), ensemble-variational (3).

## Natural pairings for comparison projects

- **Adjoint (1) versus ensemble (3):** same inverse problem, with and without adjoint code.
- **Genetic algorithm (5) versus reinforcement learning (6):** fixed control parameters versus a learned feedback policy.
- **Surrogate (4) versus genetic algorithm (5):** equal simulation budgets on the same design problem.

- **EKF (2) versus ensemble methods (3, 7):** accuracy and cost as the state dimension grows.

## Connection to the lecture material

| Topic | Related lecture content |
|---|---|
| **1 Adjoint** | Lecture 3, Part D: Lagrangian, state and adjoint equations, one-shot versus gradient-based optimization. Lectures 1–2: gradient extraction and descent methods. |
| **2 EKF** | Lectures 1–2: derivatives and Jacobians; Newton and quasi-Newton methods (the EKF update acts like a Gauss–Newton step). |
| **3 Ensemble variational** | Lecture 4: cross-entropy, natural evolution strategies and CMA-ES (ensemble statistics); Lecture 3: adjoints. |
| **4 Surrogates** | Textbook Chapters 14–16 (not covered in the lectures); Lecture 4 provides the global-search comparison. |
| **5 Genetic algorithms** | Lecture 4, Part B: genetic algorithms; Part A: CMA-ES as a competitor. |
| **6 Reinforcement learning** | Lecture 4: natural evolution strategies (the score-function gradient behind policy-gradient methods); Lectures 1–2: gradient-based training. |
| **7 Data assimilation** | Lecture 3: adjoint-based gradients (the basis of 4D-Var); Lecture 1: finite-difference versus adjoint gradients. |