Solving parameterised optimal control problems using machine learning

Kickoff Workshop of Working Group 2 “ML for CT”

Hendrik Kleikamp, IDea_Lab, University of Graz, Austria

December 5, 2025

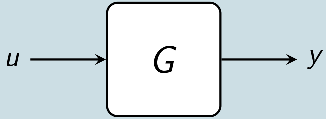
Funded by the European Union

Overview of the presentation

Parametrised optimal control problems

Optimal control problems with a parameter-dependence

Control systems

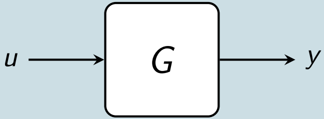


$$u^* = \arg\min_u \; \mathcal{J}(u, y)$$

$$\text{s.t.} \qquad y = G(u)$$

Optimal control problems with a parameter-dependence

Control systems



$$u^* = \arg\min_u \ \mathcal{J}(u, y)$$
$$\text{s.t.} \qquad y = G(u)$$

Here, $G(u)$ often involves solving a dynamical system, for instance

$$\frac{d}{dt}x(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t).$$

Optimal control problems with a parameter-dependence

Control systems

$$u \longrightarrow \boxed{G} \longrightarrow y$$

$$\begin{aligned} u^* = \arg\min_u \ & \mathcal{J}(u, y) \\ \text{s.t.} \quad & y = G(u) \end{aligned}$$

Control systems with parameter $\mu \in \mathcal{P}$

$$u \longrightarrow \boxed{G_\mu} \longrightarrow y_\mu$$

$$\begin{aligned} u_\mu^* = \arg\min_u \ & \mathcal{J}_\mu(u, y_\mu) \\ \text{s.t.} \quad & y_\mu = G_\mu(u) \end{aligned}$$

Here, $G(u)$ often involves solving a dynamical system, for instance

$$\begin{aligned} \frac{d}{dt}x(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t). \end{aligned}$$

Optimal control problems with a parameter-dependence

Control systems

$$u \longrightarrow \boxed{G} \longrightarrow y$$

$$u^* = \underset{u}{\arg\min}\ \mathcal{J}(u, y)$$
$$\text{s.t.} \qquad y = G(u)$$

Here, $G(u)$ often involves solving a dynamical system, for instance

$$\frac{d}{dt}x(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t).$$

Control systems with parameter $\mu \in \mathcal{P}$

$$u \longrightarrow \boxed{G_\mu} \longrightarrow y_\mu$$

$$u_\mu^* = \underset{u}{\arg\min}\ \mathcal{J}_\mu(u, y_\mu)$$
$$\text{s.t.} \qquad y_\mu = G_\mu(u)$$

$$(\mu_1, u_{\mu_1}^*) \qquad (\mu_2, u_{\mu_2}^*) \qquad (\mu_3, u_{\mu_3}^*) \qquad \dots$$

Multi-query and real-time scenarios

Multi-query context

- Solutions for many different parameter values required:
  - Parameter studies
  - Optimization
  - Uncertainty quantification

Real-time context

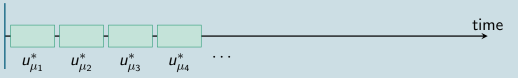
- Solution for specific parameter needed very quickly
- Possibly limited computational resources

Multi-query and real-time scenarios

Multi-query context

- Solutions for many different parameter values required:
  - Parameter studies
  - Optimization
  - Uncertainty quantification

Multi-query scenario

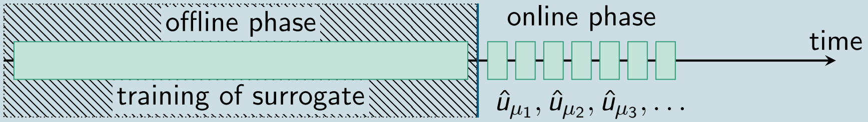


Multi-query context

- Solutions for many different parameter values required:
  - Parameter studies
  - Optimization
  - Uncertainty quantification

Multi-query scenario

Multi-query scenario using a surrogate



(Some of the) Challenges in parametric problems

- Construction of a suitable *surrogate model* is required:
  - We are not (primarily) interested in making the solution process of a single optimal control problem more efficient.
  - Need a surrogate that is much faster and tailored to the parametric structure.
  - How to train an efficient and accurate model without too much effort?

(Some of the) Challenges in parametric problems

- Construction of a suitable *surrogate model* is required:
  - We are not (primarily) interested in making the solution process of a single optimal control problem more efficient.
  - Need a surrogate that is much faster and tailored to the parametric structure.
  - How to train an efficient and accurate model without too much effort?

- Relatively *small amount of training data*:
  - Some high-fidelity solves for different parameters are fine to gather training data.
  - Not the amount typically available in machine learning.
  - Requires suitable ML techniques able to work well in the small data regime.

General strategies

Goal: Solve optimal control problems for many different parameter values, but **make use of their common structure**!

General strategies

Goal: Solve optimal control problems for many different parameter values, but **make use of their common structure**!

1. Speed up the solution process of parametric dynamical control systems.
   - Required as subproblems of the optimal control problem.
   - Independent of the OCP but helpful to accelerate the overall solution process.

General strategies

> *Goal:* Solve optimal control problems for many different parameter values, but **make use of their common structure**!

1. Speed up the solution process of parametric dynamical control systems.
   - Required as subproblems of the optimal control problem.
   - Independent of the OCP but helpful to accelerate the overall solution process.
2. Use the OCP's structure and design algorithms specifically to tackle parametric OCPs.
   - Typically focused on certain classes of problems.
   - Very efficient for the particular problems of interest.

General strategies

Goal: Solve optimal control problems for many different parameter values, but **make use of their common structure**!

1. Speed up the solution process of parametric dynamical control systems.
   - Required as subproblems of the optimal control problem.
   - Independent of the OCP but helpful to accelerate the overall solution process.
2. Use the OCP's structure and design algorithms specifically to tackle parametric OCPs.
   - Typically focused on certain classes of problems.
   - Very efficient for the particular problems of interest.
3. Hybrid strategies using specific properties of the OCP as well as general reductions.

Classical methods for parametric systems

Main idea: Solve in a low-dimensional subspace of the solution space.

Model order reduction

Projection-based reduced order models

Main idea: Solve in a low-dimensional subspace of the solution space.

For example: Replace the system

$$\frac{d}{dt}x_\mu(t) = A_\mu x_\mu(t) + B_\mu u(t),$$
$$y_\mu(t) = C_\mu x_\mu(t)$$

$$\tfrac{d}{dt}x_\mu(t) \quad = \quad \boxed{A_\mu} \quad x_\mu(t) \quad + \quad \boxed{B_\mu} \quad u(t)$$

$$y_\mu(t) \quad = \quad \boxed{C_\mu} \quad x_\mu(t)$$

Main idea: Solve in a low-dimensional subspace of the solution space.

For example: Replace the system

$$\frac{d}{dt}x_\mu(t) = A_\mu x_\mu(t) + B_\mu u(t),$$
$$y_\mu(t) = C_\mu x_\mu(t)$$

$$\tfrac{d}{dt}x_\mu(t) = A_\mu \; x_\mu(t) + B_\mu \; u(t)$$
$$y_\mu(t) = C_\mu \; x_\mu(t)$$

by a reduced system

$$\frac{d}{dt}\hat{x}_\mu(t) = \hat{A}_\mu \hat{x}_\mu(t) + \hat{B}_\mu u(t),$$
$$\hat{y}_\mu(t) = \hat{C}_\mu \hat{x}_\mu(t).$$

$$\tfrac{d}{dt}\hat{x}_\mu(t) = \hat{A}_\mu \; \hat{x}_\mu(t) + \hat{B}_\mu \; u(t)$$
$$\hat{y}_\mu(t) = \hat{C}_\mu \; \hat{x}_\mu(t)$$

Machine learning approaches

PINNs for parametric optimal control problems

Main idea: Apply a physics-informed neural network to approximate the state, the adjoint and the control as functions of the parameter.

An extended physics informed neural network for preliminary analysis of parametric optimal control problems (2023) *Nicola Demo, Maria Strazzullo and Gianluigi Rozza*

PINNs for parametric optimal control problems

Main idea: Apply a physics-informed neural network to approximate the state, the adjoint and the control as functions of the parameter.

Example: Poisson problem

$$\min_{u_\mu, y_\mu} \frac{1}{2}\|y_\mu - \mu_1\|^2_{L^2(\Omega)} + \frac{\mu_2}{2}\|u_\mu\|^2_{L^2(\Omega)}$$

$$\begin{aligned} \text{s.t.} \quad -\Delta y_\mu(x) &= u_\mu(x) \qquad && \text{in } \Omega, \\ y_\mu(x) &= 0 && \text{on } \partial\Omega. \end{aligned}$$

An extended physics informed neural network for preliminary analysis of parametric optimal control problems (2023) *Nicola Demo, Maria Strazzullo and Gianluigi Rozza*

Main idea: Apply a physics-informed neural network to approximate the state, the adjoint and the control as functions of the parameter.

Example: Poisson problem

$$
\begin{aligned}
-\Delta y_\mu(x) &= u_\mu(x) && \text{in } \Omega,\\
y_\mu(x) &= 0 && \text{on } \partial\Omega,\\
y_\mu(x) - \Delta z_\mu(x) &= \mu_1 && \text{in } \Omega,\\
z_\mu(x) &= 0 && \text{on } \partial\Omega,\\
\mu_2 u_\mu(x) &= z_\mu(x) && \text{in } \Omega.
\end{aligned}
$$

An extended physics informed neural network for preliminary analysis of parametric optimal control problems (2023) *Nicola Demo, Maria Strazzullo and Gianluigi Rozza*

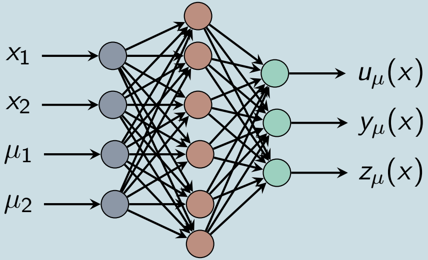
PINNs for parametric optimal control problems

Main idea: Apply a physics-informed neural network to approximate the state, the adjoint and the control as functions of the parameter.

Example: Poisson problem

$$
\begin{aligned}
-\Delta y_\mu(x) &= u_\mu(x) && \text{in } \Omega,\\
y_\mu(x) &= 0 && \text{on } \partial\Omega,\\
y_\mu(x) - \Delta z_\mu(x) &= \mu_1 && \text{in } \Omega,\\
z_\mu(x) &= 0 && \text{on } \partial\Omega,\\
\mu_2 u_\mu(x) &= z_\mu(x) && \text{in } \Omega.
\end{aligned}
$$

PINN architecture (simplification)



An extended physics informed neural network for preliminary analysis of parametric optimal control problems (2023) *Nicola Demo, Maria Strazzullo and Gianluigi Rozza*

Reinforcement learning for the control of parametric PDEs

- Solve an optimal control problem of the form

$$\min_{u} J_\mu(y_\mu, u) := \int_0^T L(t, y_\mu(t), u(t))\,\mathrm{d}t + F(y_\mu(T)),$$

$$\text{s.t. } \frac{d}{dt} y_\mu(t) = f_\mu(t, y_\mu(t)) + g_\mu(t, y_\mu(t))u(t).$$

Neural network approaches for parameterized optimal control (2025) *Deepanshu Verma, Nick Winovich, Lars Ruthotto and Bart van Bloemen Waanders*

Parametric PDE control with deep reinforcement learning and differentiable L_0-sparse polynomial policies (2024) *Nicolò Botteghi and Urban Fasel*

Reinforcement learning for the control of parametric PDEs

- Solve an optimal control problem of the form

$$\min_u J_\mu(y_\mu, u) := \int_0^T L(t, y_\mu(t), u(t))\,\mathrm{d}t + F(y_\mu(T)),$$

$$\text{s.t. } \frac{d}{dt} y_\mu(t) = f_\mu(t, y_\mu(t)) + g_\mu(t, y_\mu(t))u(t).$$

- Compute a policy/control (represented as a neural network $u(\theta)$ with parameters θ) that minimizes the cumulative reward (the cost function of the optimal control problem).

Neural network approaches for parameterized optimal control (2025) *Deepanshu Verma, Nick Winovich, Lars Ruthotto and Bart van Bloemen Waanders*

Parametric PDE control with deep reinforcement learning and differentiable L_0-sparse polynomial policies (2024) *Nicolò Botteghi and Urban Fasel*

Reinforcement learning for the control of parametric PDEs

- Solve an optimal control problem of the form

$$\min_u J_\mu(y_\mu, u) := \int_0^T L(t, y_\mu(t), u(t))\,\mathrm{d}t + F(y_\mu(T)),$$

$$\text{s.t. } \frac{d}{dt} y_\mu(t) = f_\mu(t, y_\mu(t)) + g_\mu(t, y_\mu(t))u(t).$$

- Compute a policy/control (represented as a neural network $u(\theta)$ with parameters θ) that minimizes the cumulative reward (the cost function of the optimal control problem).
- The parameter is incorporated by considering the expected value over all parameters:

$$\min_\theta \Psi(\theta) := \mathbb{E}_{\mu\sim\eta}\left[J_\mu(y_\mu, u(\theta))\right],$$

where y_μ solves the underlying dynamical system for parameter μ and control $u(\theta)$ and J is the cost function.

Neural network approaches for parameterized optimal control (2025) *Deepanshu Verma, Nick Winovich, Lars Ruthotto and Bart van Bloemen Waanders*

Parametric PDE control with deep reinforcement learning and differentiable L_0-sparse polynomial policies (2024) *Nicolò Botteghi and Urban Fasel*

Combination of model order reduction and machine learning

Main idea: Learn the parameter to reduced solution map and make use of a posteriori error estimates for the reduced model.

Be greedy and learn: efficient and certified algorithms for parametrized optimal control problems (2025) *H. K., Martin Lazar and Cesare Molinari*

Application of an adaptive model hierarchy to parametrized optimal control problems (2024) *H. K.*

Combination of model order reduction and machine learning

> *Main idea:* Learn the parameter to reduced solution map and make use of a posteriori error estimates for the reduced model.

- Approximate optimal final time adjoint φ_μ^* in a low-dimensional subspace as

$$\varphi_\mu^{\mathrm{RB}} = \sum_{i=1}^{N} \alpha_i(\mu) \cdot \varphi_i.$$

Be greedy and learn: efficient and certified algorithms for parametrized optimal control problems (2025) *H. K., Martin Lazar and Cesare Molinari*

Application of an adaptive model hierarchy to parametrized optimal control problems (2024) *H. K.*

Combination of model order reduction and machine learning

Main idea: Learn the parameter to reduced solution map and make use of a posteriori error estimates for the reduced model.

- Approximate optimal final time adjoint φ_μ^* in a low-dimensional subspace as
$$\varphi_\mu^{\mathrm{RB}} = \sum_{i=1}^{N} \alpha_i(\mu) \cdot \varphi_i.$$
- Approximate reduced coefficients by a surrogate
$$\varphi_\mu^{\mathrm{ML}} = \sum_{i=1}^{N} \alpha_i^{\mathrm{ML}}(\mu) \cdot \varphi_i \approx \varphi_\mu^{\mathrm{RB}}.$$

Be greedy and learn: efficient and certified algorithms for parametrized optimal control problems (2025) *H. K., Martin Lazar and Cesare Molinari*

Application of an adaptive model hierarchy to parametrized optimal control problems (2024) *H. K.*

Combination of model order reduction and machine learning

> *Main idea:* Learn the parameter to reduced solution map and make use of a posteriori error estimates for the reduced model.

- Approximate optimal final time adjoint φ_μ^* in a low-dimensional subspace as
$$\varphi_\mu^{\mathrm{RB}} = \sum\nolimits_{i=1}^{N} \alpha_i(\mu) \cdot \varphi_i.$$
- Approximate reduced coefficients by a surrogate
$$\varphi_\mu^{\mathrm{ML}} = \sum\nolimits_{i=1}^{N} \alpha_i^{\mathrm{ML}}(\mu) \cdot \varphi_i \approx \varphi_\mu^{\mathrm{RB}}.$$
- Use error estimator of RB-ROM also for ML-ROM.

Be greedy and learn: efficient and certified algorithms for parametrized optimal control problems (2025) *H. K., Martin Lazar and Cesare Molinari*

Application of an adaptive model hierarchy to parametrized optimal control problems (2024) *H. K.*

Combination of model order reduction and machine learning

Main idea: Learn the parameter to reduced solution map and make use of a posteriori error estimates for the reduced model.

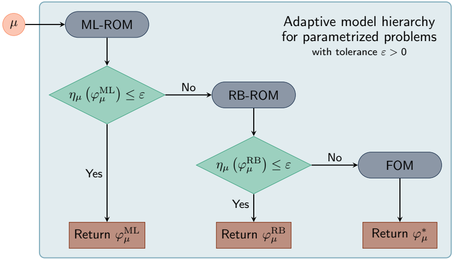
- Approximate optimal final time adjoint φ_μ^* in a low-dimensional subspace as

$$\varphi_\mu^{\mathrm{RB}} = \sum\nolimits_{i=1}^{N} \alpha_i(\mu) \cdot \varphi_i.$$

- Approximate reduced coefficients by a surrogate

$$\varphi_\mu^{\mathrm{ML}} = \sum\nolimits_{i=1}^{N} \alpha_i^{\mathrm{ML}}(\mu) \cdot \varphi_i \approx \varphi_\mu^{\mathrm{RB}}.$$

- Use error estimator of RB-ROM also for ML-ROM.



Be greedy and learn: efficient and certified algorithms for parametrized optimal control problems (2025) *H. K., Martin Lazar and Cesare Molinari*

Application of an adaptive model hierarchy to parametrized optimal control problems (2024) *H. K.*

Open problems

Nonlinear and transport-dominated problems

- *Goal:* Solve optimal control problems where the governing dynamics are parametric and nonlinear or transport-dominated, for instance

$$\frac{\partial y}{\partial t} + \mu \frac{\partial y^2}{\partial x} = u.$$

- *Contact:* **Martin Lazar**. Email: `mlazar@unidu.hr`
- *Required skills:* Good command in numerics of PDEs and in control theory
- *Some details/related questions:*
  - Nonlinear problems pose difficulties for traditional, linear approximation schemes.
  - General question for model order reduction.
  - What is possible in the context of (optimal) control of such systems?
  - Where can machine learning help to overcome these issues (nonlinear strategies such as autoencoders, etc.)?

Control of flow problems such as the Navier-Stokes equations

- *Goal:* Solve optimal control problems for flows governed by parametric Navier-Stokes equations, for instance

$$\begin{aligned}\frac{\partial y}{\partial t} - \mu\Delta y + (y \cdot \nabla)y + \nabla p = u,\\ \operatorname{div} y = 0.\end{aligned}$$

- *Contact:* **Maria Strazzullo**. Email: `maria.strazzullo@polito.it`
- *Required skills:* Good command in numerics of PDEs and in control theory
- *Some details/related questions:*
  - Navier-Stokes equations are an important model for (viscous) flow in real-world applications and are used for simulations of many different processes.
  - Depending on the Reynolds number (roughly the parameter μ in the formulation above), the solution behavior can change completely.
  - Can machine learning help in order to deal with the turbulent regime?

General convex objective functionals

- *Goal:* Induce sparsity in the control by solving a problem of the form

$$u_\mu^* = \arg\min_u \|u\|_{L^1([0,T];U)} + \frac{\alpha}{2}\|u\|^2_{L^2([0,T];U)} + h(x_\mu)$$

  where x_μ solves a dynamical system with parameter $\mu \in \mathcal{P}$ and control u.
- *Contact:* **Cesare Molinari**. Email: `cesare.molinari@edu.unige.it`
- *Required skills:* Good command in (convex) optimization; Python programming
- *Some details/related questions:*
  - Which algorithm is suited best to solve this OCP?
  - How to deal with the parameter dependence?
  - Can reduced order modeling be applied here in a suitable manner?
  - If so, how to combine it in a reasonable way with machine learning?

Optimization in the parameter space

- *Goal:* Solve problems of the form

$$\mu^* = \arg\min_{\mu \in \mathcal{P}} F(\mu; u_\mu)$$

  where u_μ solves an optimal control problem for the parameter $\mu \in \mathcal{P}$.
- *Contact:* **Hendrik Kleikamp**. Email: `hendrik.kleikamp@uni-graz.at`
- *Required skills:* Knowledge in optimization and control theory; Python programming
- *Some details/related questions:*
  - Optimal control problem for a fixed parameter as an “inner” problem.
  - Derivatives with respect to the parameter are typically required.
  - Optimizer usually moves outside ot the range of training data points.
    $\longrightarrow$ How to extrapolate properly?

Small data regime

- *Goal:* How to deal with relatively small amount of available data?
- *Contact:* **Hendrik Kleikamp**. Email: `hendrik.kleikamp@uni-graz.at`
- *Required skills:* Good command of machine learning and control theory
- *Some details/related questions:*
  - Training data (at least using the FOM) is costly to obtain.
  - Which quantities are easiest to learn when only a small amount of data is available?
    - Optimal control $\longrightarrow$ How to obtain performance guarantees?
    - Reduced quantities $\longrightarrow$ Combination with MOR techniques often allows to collect more training data and to make use of their error estimates.
    - Open loop vs. closed loop systems $\longrightarrow$ Feedback control requires different architectures and learning techniques.

Applications in uncertainty quantification

- *Goal:* Make use of the derived surrogates in multilevel Monte Carlo methods:

$$\mathbb{E}[M_L] = \mathbb{E}[M_0] + \sum_{\ell=0}^{L} \mathbb{E}[M_\ell - M_{\ell-1}].$$

- *Contact:* **Hendrik Kleikamp**. Email: `hendrik.kleikamp@uni-graz.at`
- *Required skills:* Machine learning and surrogate modeling; a bit of probability theory and statistics; Python programming
- *Some details/related questions:*
  - Consider different applications in which we want efficient estimates of unknown quantities.
  - Interactions of the different models?
  - Strategies to select the models and the number of evaluations on different levels?
  - Can we derive probabilistic guarantees that this works?

Short term scientific missions

If you are interested in working on one of these problems or on a similar project, *contact* the respective person and *discuss* in more detail about it. And . . .

Apply for STSMs related to parametrised optimal control problems!

We would be very happy to hear from you and to work with you on inspiring projects!

Thank you for your attention!