Imperial College London

Simultaneous Process Design and Control Optimization using Reinforcement Learning

S. Sachio, E.A. del Rio Chanona, **P. Petsagkourakis**

Email: p.petsagkourakis@ucl.ac.uk

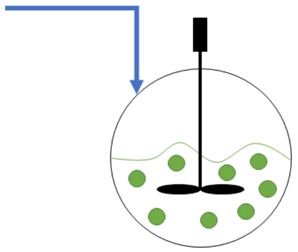
AICHE 2020

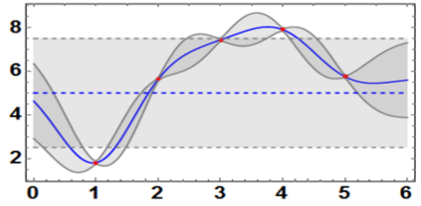
Centre for Process Systems Engineering

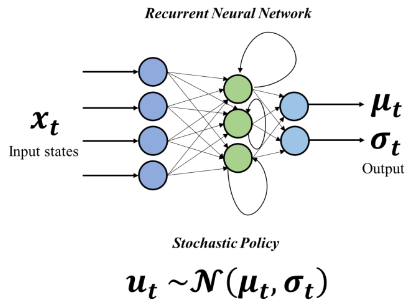
@Pnosapp

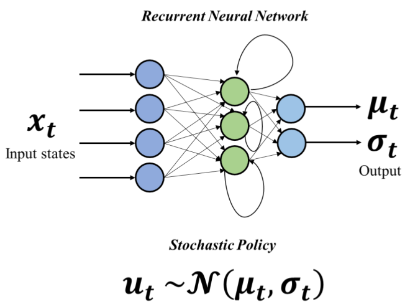
Outline

- **Introduction**
- **Integration of Design and Control**
- **Reinforcement Learning for Process Control**
- **Integration via RL**
- **Case study**
- **Conclusion and Future Work**

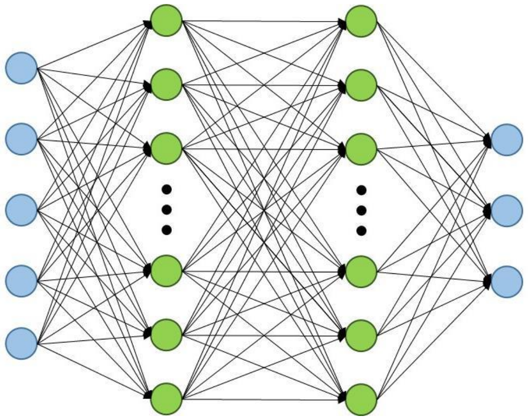


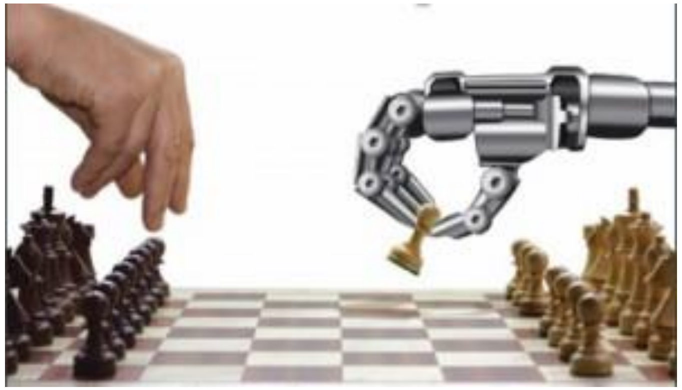




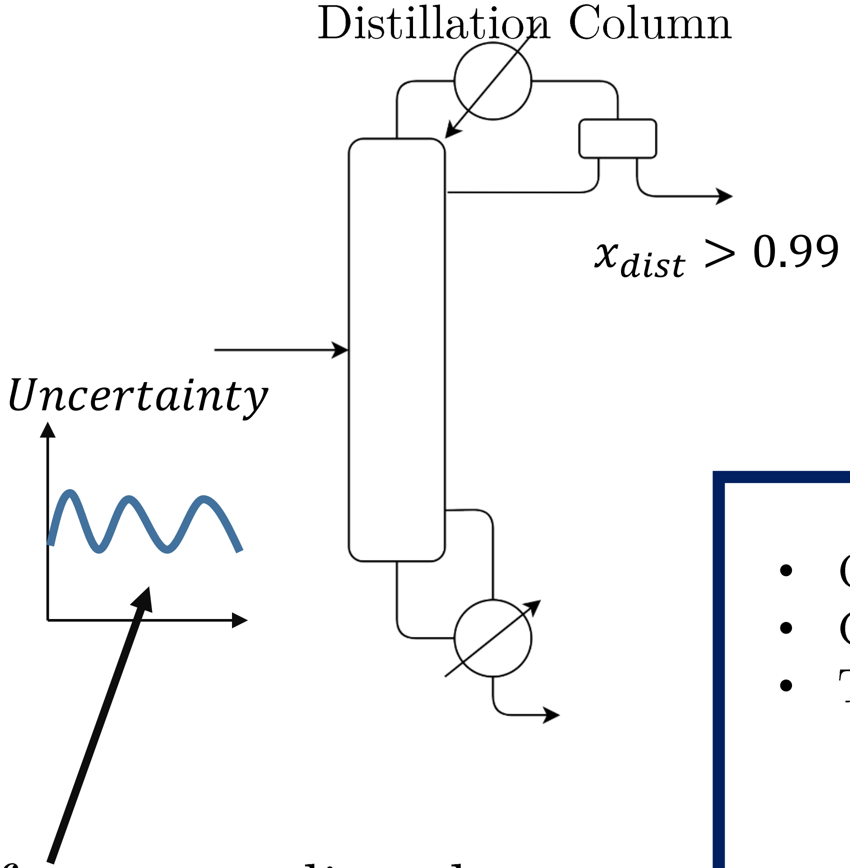


Introduction

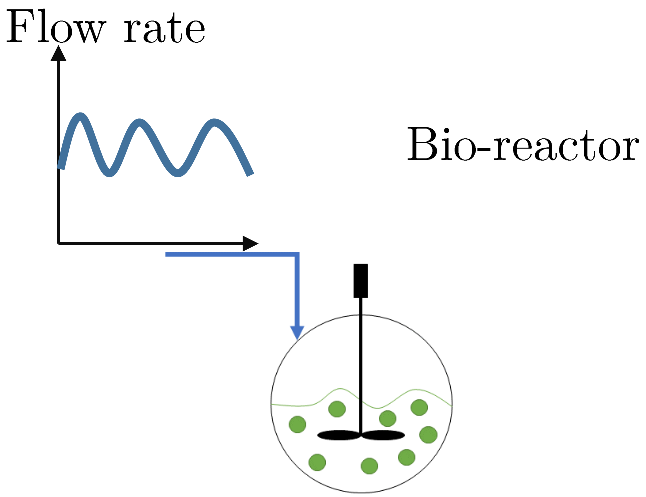




Perform Process Design and Control under uncertainty



High frequency disturbance

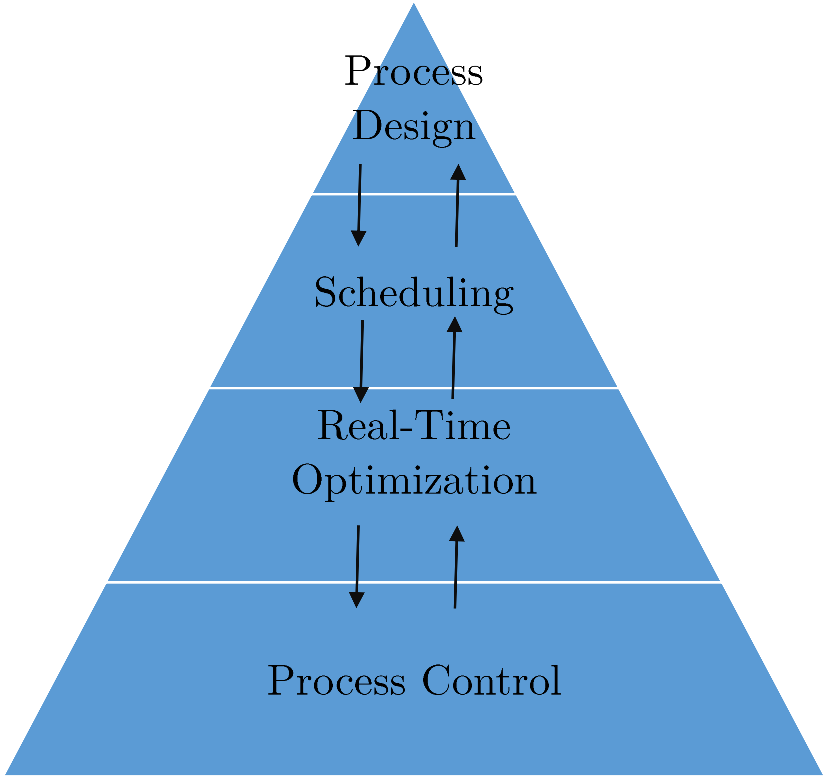


- Optimize the process under the presence of disturbance
- Control design depends on the design variables
- The process design depends on the control actions

Simultaneously Process and Control Design

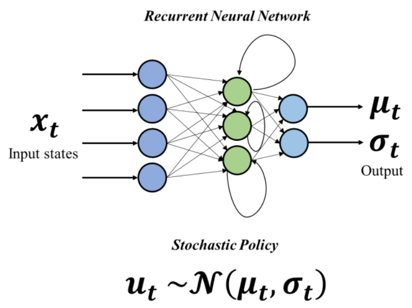
Decision Making in Process Engineering

Hierarchical Interaction

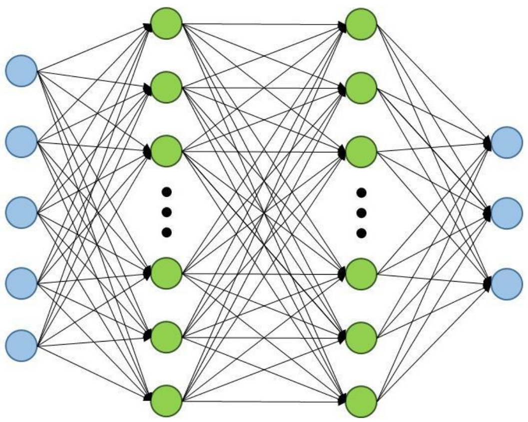


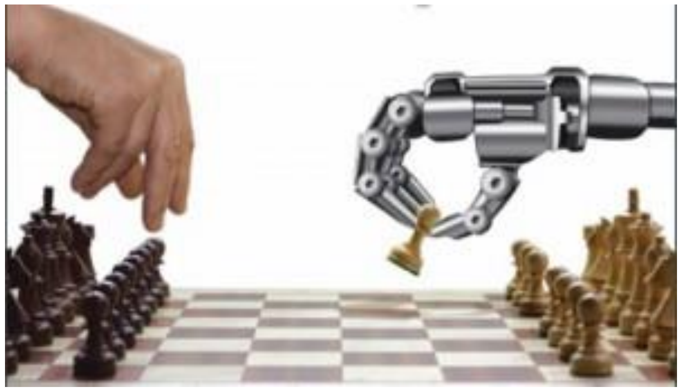
	Relevant Contributions
Flores-Tlacuahuac and Biegler (2008), Brengel and Seider (1992), Mehta and Ricardez-Sandoval (2016), Kookos and Perkins (2016), Li and Barton (2015)	Iterative MINLP formulation with stochastic back-off formulation for uncertainty
Pistikopoulos & co-workers (2000, 2002, 2003), Ricardez-Sandoval (2012)	Simultaneous/decomposition (MI)DO process and P-PI-PID control design
Diangelakis, N., et al (2017), Burnak and Pistikopoulos (2020), Washington and Swartz (2014), Ricardez-Sandoval (2012a)	Simultaneous/decomposition (MI)DO process and MPC design

B. Burnak et al., 2019, Comput. Chem. Eng., 125, 164-189
Diangelakis, N., et al, 2017. AICHE J., 63, 4827–4846



Integration of Design and Control





Bilevel Formulation of Integrated Problem

Control related cost

$$\min_{\boldsymbol{p}} \; C_p(\boldsymbol{p}) + \boxed{\kappa C_\kappa(\boldsymbol{x}(\tau), \boldsymbol{y}(\tau), \boldsymbol{u}(\tau), \boldsymbol{p}, \theta^P(\tau))}$$

$$s.t. \; \boxed{\begin{aligned} & f_p(\boldsymbol{p}, \theta^P(\tau)) = 0 \\ & g_p(\boldsymbol{p}, \theta^P(\tau)) \leq 0 \end{aligned}}$$

Constraints related to the process

$$\boxed{\begin{aligned} \min_{\boldsymbol{u}(\tau)} \; & C_u(\boldsymbol{x}(\tau), \boldsymbol{y}(\tau), \boldsymbol{u}(\tau), \boldsymbol{p}, \theta^P(\tau)) \\ s.t. \; & \dot{\boldsymbol{x}} = f_u(\boldsymbol{x}(\tau), \boldsymbol{y}(\tau), \boldsymbol{u}(\tau), \boldsymbol{p}, \theta^P(\tau)) \\ & g_u(\boldsymbol{x}(\tau), \boldsymbol{y}(\tau), \boldsymbol{u}(\tau), \boldsymbol{p}, \theta^P(\tau)) = 0 \\ & h_u(\boldsymbol{x}(\tau), \boldsymbol{y}(\tau), \boldsymbol{u}(\tau), \boldsymbol{p}, \theta^P(\tau)) \leq 0 \end{aligned}}$$

Optimal Controller

- Very hard problem to solve!
- Simplifications in the control problem have been applied

Bilevel Formulation of Integrated Problem

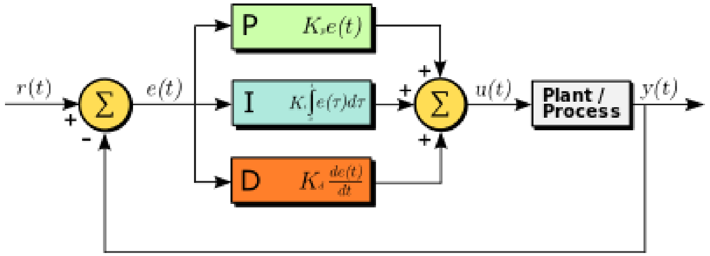
Control related cost

$$\min_{\boldsymbol{p}} \; C_p(\boldsymbol{p}) + \boxed{\kappa C_\kappa(\boldsymbol{x}(\tau), \boldsymbol{y}(\tau), \boldsymbol{u}(\tau), \boldsymbol{p}, \theta^P(\tau))}$$

$$s.t. \; \boxed{\begin{aligned} & f_p(\boldsymbol{p}, \theta^P(\tau)) = 0 \\ & g_p(\boldsymbol{p}, \theta^P(\tau)) \leq 0 \end{aligned}}$$

Constraints related to the process

Explicit expression via PID



Optimal Controller

- Very hard problem to solve!
- Simplifications in the control problem have been applied

Bilevel Formulation of Integrated Problem

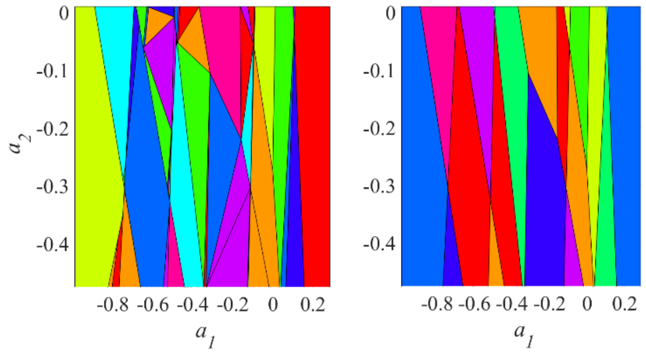
Control related cost

$$\min_{\boldsymbol{p}} \; C_p(\boldsymbol{p}) + \boxed{\kappa C_\kappa(\boldsymbol{x}(\tau), \boldsymbol{y}(\tau), \boldsymbol{u}(\tau), \boldsymbol{p}, \theta^P(\tau))}$$

$$s.t. \; \boxed{\begin{aligned} f_p(\boldsymbol{p}, \theta^P(\tau)) &= 0 \\ g_p(\boldsymbol{p}, \theta^P(\tau)) &\leq 0 \end{aligned}}$$

Constraints related to the process

Explicit expression via multiparametric MPC



This is optimal, but…

!Model approximations are still needed!

Optimal Controller

- Very hard problem to solve!
- Simplifications in the control problem have been applied

Bilevel Formulation of Integrated Problem

- Very hard problem to solve!
- Simplifications in the control problem have been applied
- Reinforcement learning to the rescue

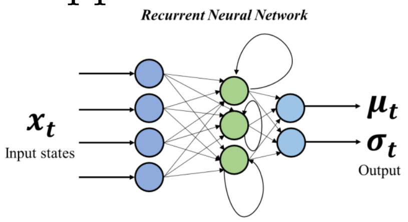
Control related cost

$$\min_{\boldsymbol{p}} \; C_p(\boldsymbol{p}) + \boxed{\kappa C_\kappa(\boldsymbol{x}(\tau), \boldsymbol{y}(\tau), \boldsymbol{u}(\tau), \boldsymbol{p}, \theta^P(\tau))}$$

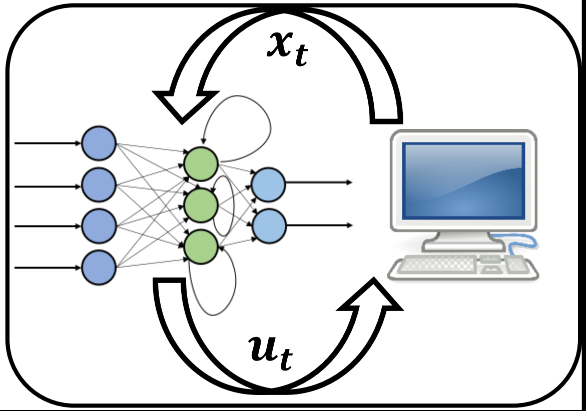
$$s.t. \; \boxed{\begin{aligned} f_p(\boldsymbol{p}, \theta^P(\tau)) &= 0 \\ g_p(\boldsymbol{p}, \theta^P(\tau)) &\leq 0 \end{aligned}}$$

Constraints related to the process

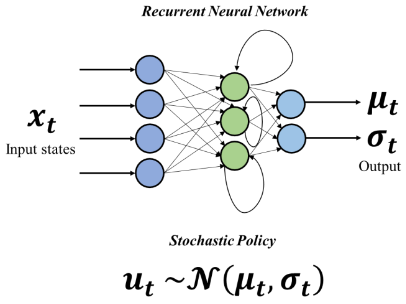
If only there was a way to be optimal without approximations…



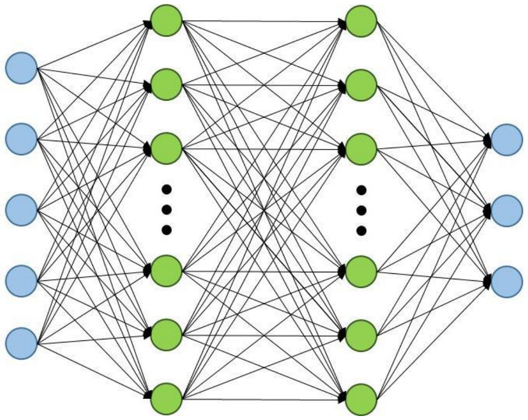
Reinforcement Learning

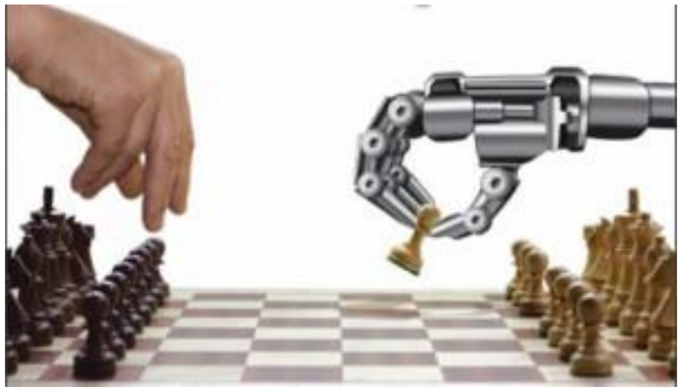


Optimal Controller

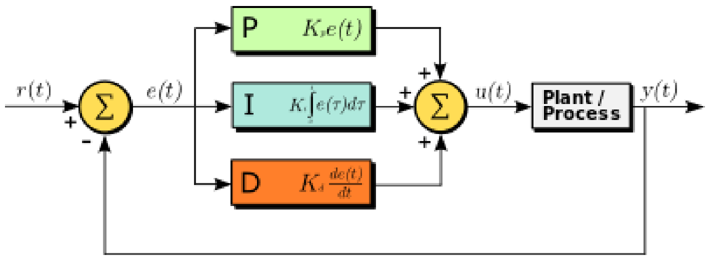


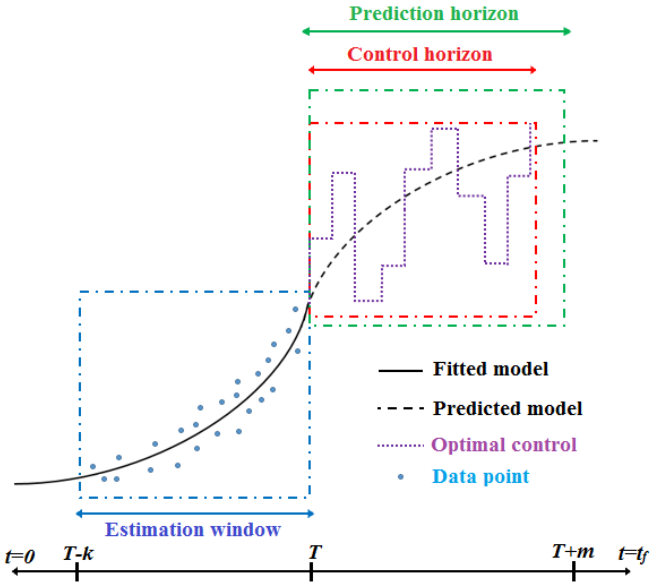
Reinforcement Learning for Process Control





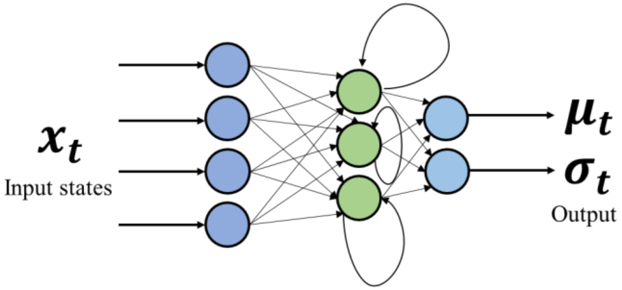
Reinforcement Learning in action





Explicit closed-loop control law

Recurrent Neural Network



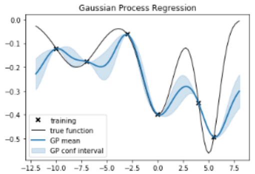
Stochastic Policy

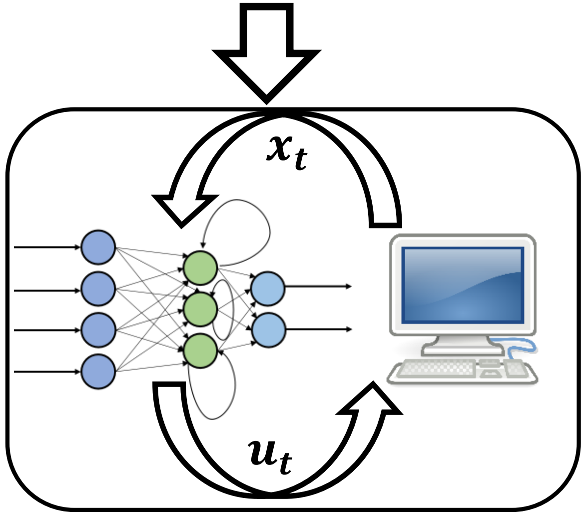
$$u_t \sim \mathcal{N}(\mu_t, \sigma_t)$$

Process model (in silico)

$$\frac{dx}{dt} = f(x)$$

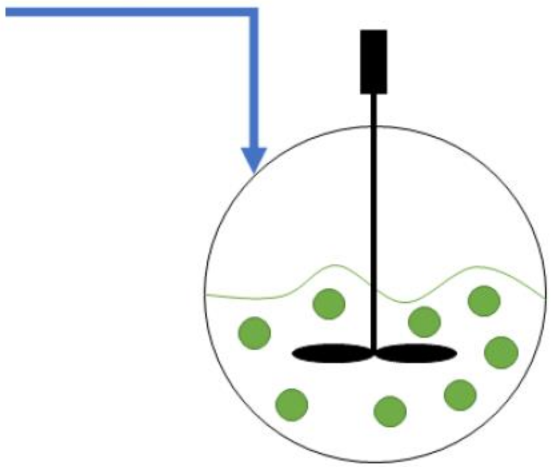
$$x(0) = x_0$$





Reinforcement Learning in silico

Problem Statement – Stochastic Optimal Control Problem



Process Optimization

- Difficult to optimize due to their unsteady-state operation
- Highly complex
  - Plant-model mismatch is often present

Real Physical System

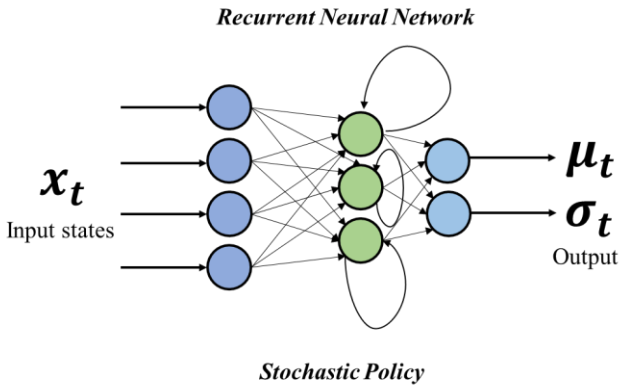
- $\mathbf{x}_{t+1} \sim p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{u}_t)$
- $\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t, \mathbf{d}_t)$

- Given an objective (economic criterion) J and the batch time T.
- Shrinking horizon.
- Find an optimal policy.

At each time $t_s \in \{0, \ldots, T-1\}$

$$\mathcal{P}(\pi(\cdot)) := \begin{cases} \max_{\pi(\cdot)} \mathbb{E}\{J(\mathbf{x}_t, \mathbf{u}_t)\} \\ \text{s.t.} \\ \mathbf{x}_0 \sim p(\mathbf{x}_0) \\ \mathbf{x}_{t+1} \sim p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{u}_t) \\ \mathbf{u}_t \sim p(\mathbf{u}_t|\mathbf{x}_t) = \pi(\mathbf{x}_t) \\ \mathbf{u}_t \in \mathbb{U} \\ \mathbb{P}(\bigcap_{i=0}^{T}\{\mathbf{x}_i \in \mathbb{X}_i\}) \geq 1-\alpha \\ \forall t \in \{0, \ldots, T-1\} \end{cases}$$

Policy Gradients – How do we build a policy network?



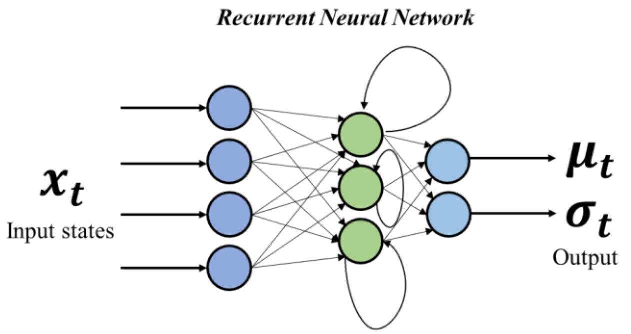
$$u_t \sim \mathcal{N}(\mu_t, \sigma_t)$$

- Given a (R)NN that takes states as inputs, and outputs a control actions (policy network).
- Could you conduct 'some sort' of steepest ascent/descent on the dynamic system?

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha \nabla_{\boldsymbol{\theta}} J(\boldsymbol{x}, \boldsymbol{u})$$

$$\mathcal{P}(\pi(\cdot)) := \begin{cases} \max\limits_{\pi(\cdot)} \mathbb{E}\{J(\mathbf{x}_t, \mathbf{u}_t)\} \\ \text{s.t.} \\ \mathbf{x}_0 \sim p(\mathbf{x}_0) \\ \mathbf{x}_{t+1} \sim p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{u}_t) \\ \mathbf{u}_t \sim p(\mathbf{u}_t|\mathbf{x}_t) = \pi(\mathbf{x}_t) \\ \mathbf{u}_t \in \mathbb{U} \\ \mathbb{P}(\bigcap\limits_{i=0}^{T}\{\mathbf{x}_i \in \mathbb{X}_i\}) \geq 1 - \alpha \\ \forall t \in \{0, ..., T-1\} \end{cases}$$

Policy Gradients – How do we build a policy network?



Stochastic Policy

$$\boldsymbol{u_t} \sim \mathcal{N}(\boldsymbol{\mu_t}, \boldsymbol{\sigma_t})$$

- Given a (R)NN that takes states as inputs, and outputs a control actions (policy network).
- Could you conduct 'some sort' of steepest ascent/descent on the dynamic system?

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha \, \nabla_{\boldsymbol{\theta}} J(\boldsymbol{x}, \boldsymbol{u})$$

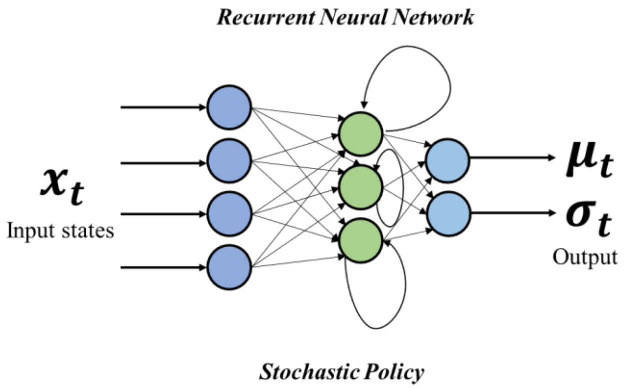
$$\max_{\pi(\cdot)} \mathbb{E}_{\boldsymbol{\tau} \sim p(\boldsymbol{\tau}|\theta)} [J(\boldsymbol{\tau})]$$

- $J(\boldsymbol{\tau}) = \sum_{t=0}^{T} \gamma^t R_t(\mathbf{u}_t, \mathbf{x}_t)$
- $\boldsymbol{\tau} = (\mathbf{x}_0, \mathbf{u}_0, R_0, \ldots, \mathbf{x}_{T-1}, \mathbf{u}_{T-1}, R_{T-1}, \mathbf{x}_T, R_T)$

Policy Gradient Theorem

$$\nabla_{\boldsymbol{\theta}} \hat{J}(\boldsymbol{\tau}) \coloneqq \nabla_{\theta} \mathbb{E}_{\tau \sim p(\boldsymbol{\tau}|\boldsymbol{\theta})}[J(\boldsymbol{\tau})] = \nabla_{\theta} \mathbb{E}_{\tau} \left[J(\boldsymbol{\tau}) \nabla_{\boldsymbol{\theta}} \sum_{t=0}^{T-1} \log\left(\overbrace{\pi(\boldsymbol{u}_t | \boldsymbol{x}_t, \boldsymbol{\theta})}^{\text{NN}} \right) \right]$$

Policy Gradients – How do we build a policy network?



$$u_t \sim \mathcal{N}(\mu_t, \sigma_t)$$

- Given a (R)NN that takes states as inputs, and outputs a control actions (policy network).
- Could you conduct 'some sort' of steepest ascent/descent on the dynamic system?

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha \nabla_{\boldsymbol{\theta}} J(\boldsymbol{x}, \boldsymbol{u})$$

$$\max_{\pi(\cdot)} \mathbb{E}_{\boldsymbol{\tau} \sim p(\boldsymbol{\tau}|\theta)}[J(\boldsymbol{\tau})]$$

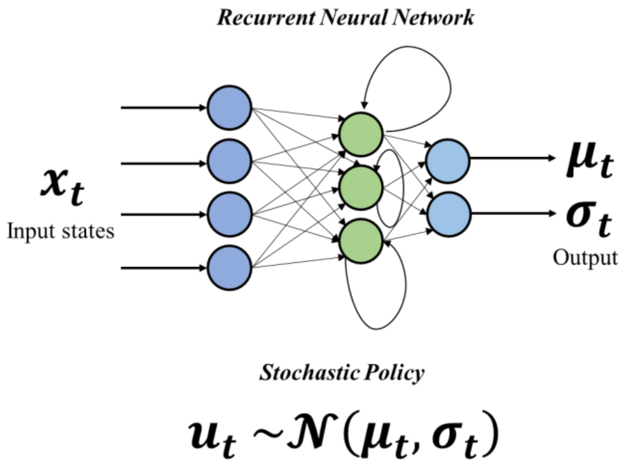
- $J(\boldsymbol{\tau}) = \sum_{t=0}^{T} \gamma^t R_t(\mathbf{u}_t, \mathbf{x}_t)$
- $\boldsymbol{\tau} = (\mathbf{x}_0, \mathbf{u}_0, R_0, \ldots, \mathbf{x}_{T-1}, \mathbf{u}_{T-1}, R_{T-1}, \mathbf{x}_T, R_T)$

This expectation can be calculated via Monte Carlo runs

Policy Gradient Theorem

$$\nabla_{\boldsymbol{\theta}} \hat{J}(\boldsymbol{\tau}) \coloneqq \nabla_\theta \mathbb{E}_{\tau \sim p(\boldsymbol{\tau}|\boldsymbol{\theta})}[J(\boldsymbol{\tau})] = \boxed{\nabla_\theta \mathbb{E}_\tau}\left[J(\boldsymbol{\tau}) \nabla_{\boldsymbol{\theta}} \sum_{t=0}^{T-1} \log\left(\pi(\boldsymbol{u}_t | \boldsymbol{x}_t, \boldsymbol{\theta})\right)\right]$$

Policy Gradients – How do we build a policy network?



- Given a (R)NN that takes states as inputs, and outputs a control actions (policy network).
- Could you conduct 'some sort' of steepest ascent/descent on the dynamic system?

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha \nabla_{\boldsymbol{\theta}} J(\boldsymbol{x}, \boldsymbol{u})$$

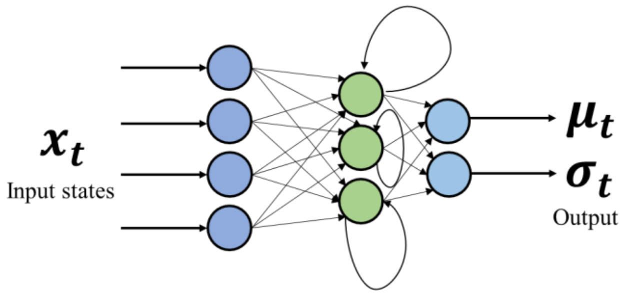
First presented in:

R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," Machine learning, vol. 8, no. 3-4, pp. 229–256, 1992.

For process engineering audience:

P. Petsagkourakis, et, al. "Reinforcement learning for batch bioprocess optimization," Computers & Chemical Engineering, vol. 133, p. 106649, 2020.

Policy Gradients – How do we build a policy network?



Stochastic Policy

$$\boldsymbol{u}_t \sim \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\sigma}_t)$$

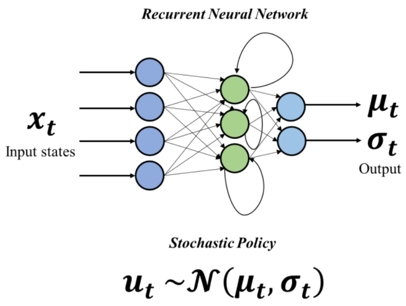
- Given a (R)NN that takes states as inputs, and outputs a control actions (policy network).
- Could you conduct 'some sort' of steepest ascent/descent on the dynamic system?

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha \nabla_{\boldsymbol{\theta}} J(\boldsymbol{x}, \boldsymbol{u})$$

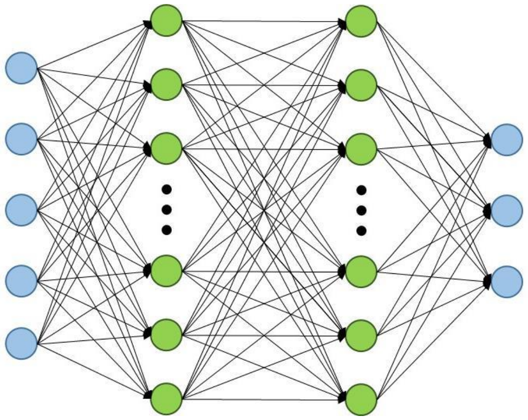
Algorithm: "Vanilla" policy gradient algorithm

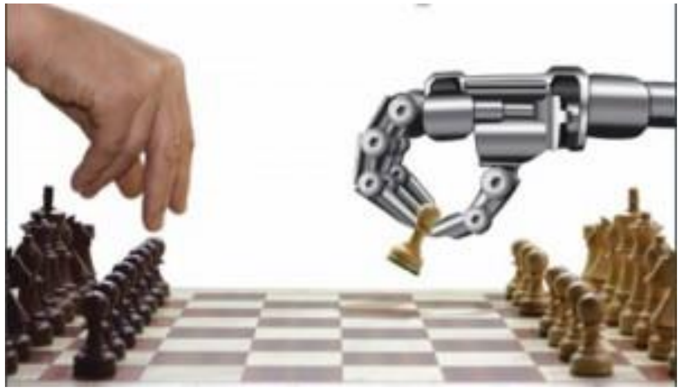
1. Initialize policy parameters $\boldsymbol{\theta}$
2. **for** $k \coloneqq 1, \dots$ **do**
   a) Collect N a set of trajectories by executing the current policy $\boldsymbol{\tau}_n = [\boldsymbol{x}_0^n, \boldsymbol{u}_0^n, \dots, \boldsymbol{u}_{T-1}^n, \boldsymbol{x}_T^n]$
   b) Compute $\nabla_{\boldsymbol{\theta}} \hat{J}(\boldsymbol{\tau}) \coloneqq \frac{1}{N} \sum_{n=0}^{N} \left[J(\boldsymbol{\tau}_n) \nabla_{\boldsymbol{\theta}} \sum_{t=0}^{T-1} \log\left(\pi(\boldsymbol{u}_t | \boldsymbol{x}_t, \boldsymbol{\theta})\right) \right]$
   c) Steepest ascent type step $\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha \nabla_{\boldsymbol{\theta}} \hat{J}(\boldsymbol{\tau})$

end for



Overall Framework





Overall Framework

Control related cost

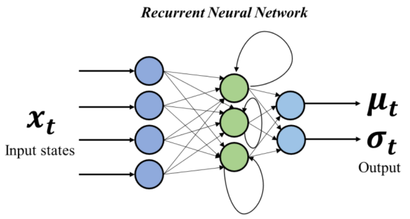
$$\min_{\boldsymbol{p}} \; C_p(\boldsymbol{p}) + \boxed{\kappa C_\kappa(\boldsymbol{x}(\tau), \boldsymbol{y}(\tau), \boldsymbol{u}(\tau), \boldsymbol{p}, \theta^P(\tau))}$$

$$s.t. \; \boxed{\begin{aligned} & f_p(\boldsymbol{p}, \theta^P(\tau)) = 0 \\ & g_p(\boldsymbol{p}, \theta^P(\tau)) \leq 0 \end{aligned}}$$

Constraints related to the process

- Very hard problem to solve!
- Simplifications in the control problem have been applied
- Reinforcement learning to the rescue

Reinforcement Learning



Stochastic Policy

$\boldsymbol{u_t} \sim \mathcal{N}(\boldsymbol{\mu_t}, \boldsymbol{\sigma_t})$

Trained policy via RL:
Here Policy Gradient

Optimal Controller

Overall Framework

State design-control simultaneous optimization

OCP as RL:

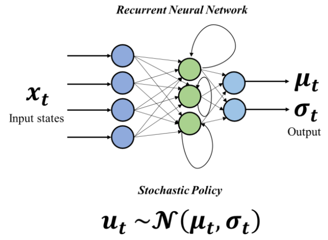
Pretrain using MPC or PID

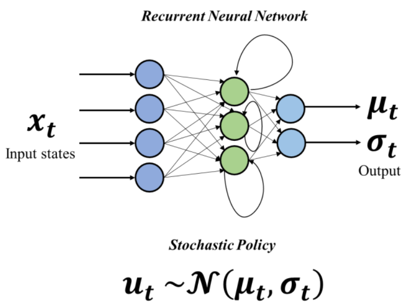
Policy learning via Policy gradient

Solve the new MIDO problem

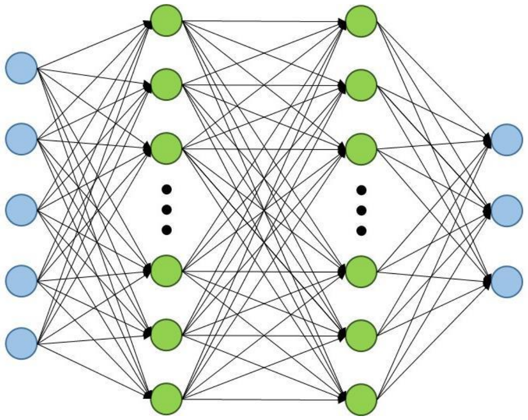
$$
\begin{aligned}
\min_{\boldsymbol{p}}\ & C_p(\boldsymbol{p}) + \kappa C_\kappa(\boldsymbol{x}(\tau), \boldsymbol{y}(\tau), \boldsymbol{u}(\tau), \boldsymbol{p}, \theta^P(\tau)) \\
s.t.\ & f_p(\boldsymbol{p}, \theta^P(\tau)) = 0 \\
& g_p(\boldsymbol{p}, \theta^P(\tau)) \le 0 \\
& \min_{\boldsymbol{u}(\tau)}\ C_u(\boldsymbol{x}(\tau), \boldsymbol{y}(\tau), \boldsymbol{u}(\tau), \boldsymbol{p}, \theta^P(\tau)) \\
& \quad s.t.\ \dot{\boldsymbol{x}} = f_u(\boldsymbol{x}(\tau), \boldsymbol{y}(\tau), \boldsymbol{u}(\tau), \boldsymbol{p}, \theta^P(\tau)) \\
& \qquad g_u(\boldsymbol{x}(\tau), \boldsymbol{y}(\tau), \boldsymbol{u}(\tau), \boldsymbol{p}, \theta^P(\tau)) = 0 \\
& \qquad h_u(\boldsymbol{x}(\tau), \boldsymbol{y}(\tau), \boldsymbol{u}(\tau), \boldsymbol{p}, \theta^P(\tau)) \le 0
\end{aligned}
$$

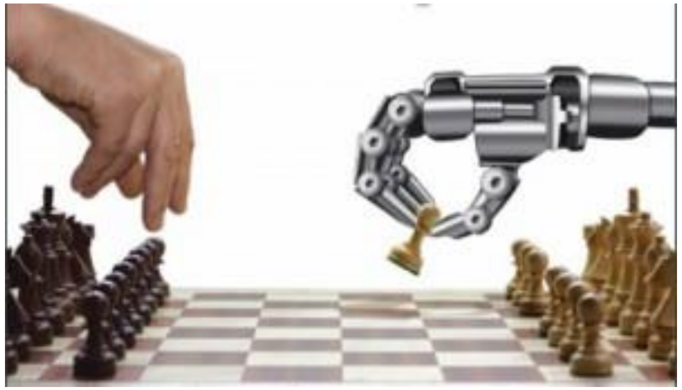
$$
\mathcal{P}(\pi(\cdot)) := \begin{cases}
\max_{\pi(\cdot)} \mathbb{E}\{J(\mathbf{x}_t, \mathbf{u}_t)\} \\
\text{s.t.} \\
\mathbf{x}_0 \sim p(\mathbf{x}_0) \\
\mathbf{x}_{t+1} \sim p(\mathbf{x}_{t+1} | \mathbf{x}_t, \mathbf{u}_t) \\
\mathbf{u}_t \sim p(\mathbf{u}_t | \mathbf{x}_t) = \pi(\mathbf{x}_t) \\
\mathbf{u}_t \in \mathbb{U} \\
\mathbb{P}\left(\bigcap_{i=0}^{T} \{\mathbf{x}_i \in \mathbb{X}_i\}\right) \ge 1 - \alpha \\
\forall t \in \{0, \ldots, T-1\}
\end{cases}
$$





Case Studies





Optimization and Control of a Tank[1]

- Control task: Find a closed-loop policy $\pi_\theta(\cdot)$ that steer the volume to a desired policy
- Optimization task: Find the maximum F_{dev}
- ***Note*** that in the case of the tank the set point for the volume is dynamically correlated with the maximum value of the inlet flow rate

$$\max_{V_{tank}, F_{dev}, F_{nom}, V(0)} \quad J_{SDC} = \int_0^1 F_{dev} d\tau$$

$$\begin{aligned}
s.t.\ \frac{dV(\tau)}{d\tau} &= F_{in}(\tau) - F_{out}(\tau) \\
F_{out}(\tau) &= \alpha_t V(\tau) \\
F_{in}(\tau) &= F_{nom} + F_{dev} \sin(\tau / freq) \\
freq &= \frac{1}{2\pi} \\
V_{SP} &= F_{nom} + F_{dev} \leq V_{tank} \\
err_{\pi_\theta} &= \int_0^1 \frac{||V(\tau) - V_{SP}||}{V_{SP}} d\tau
\end{aligned}$$

End-point constraints:

$$(1 - \varepsilon/100)V(0) \leq V(T_F) \leq (1 + \varepsilon/100)V(0)$$

$$err_{\pi_\theta} \leq \varepsilon/100$$

$$\frac{dV(\tau)}{d\tau} = F_{in}(\tau) - F_{out}(\tau)$$

$$F_{out}(\tau) = \boxed{\alpha_t} V(\tau)$$

Manipulated variable

$$F_{in}(\tau) = F_{nom} + F_{dev} \sin(\tau / freq)$$

$$freq = \frac{1}{2\pi}$$

$$V_{SP} = F_{nom} + F_{dev} \leq V_{tank}$$

Closed-loop policy via RL

$$\boxed{a_t = \pi_\theta(F_{in,t}, V_{SP}, V_t, V_{t-1})}\ \forall t \in \{0, ..., n_T - 1\}$$

[1] Diangelakis, N. A., Burnak, B., Katz, J., & Pistikopoulos, E. N. (2017), *AIChE J.*, *63*(11), 4827–4846.

Optimization and Control of a Tank[1]

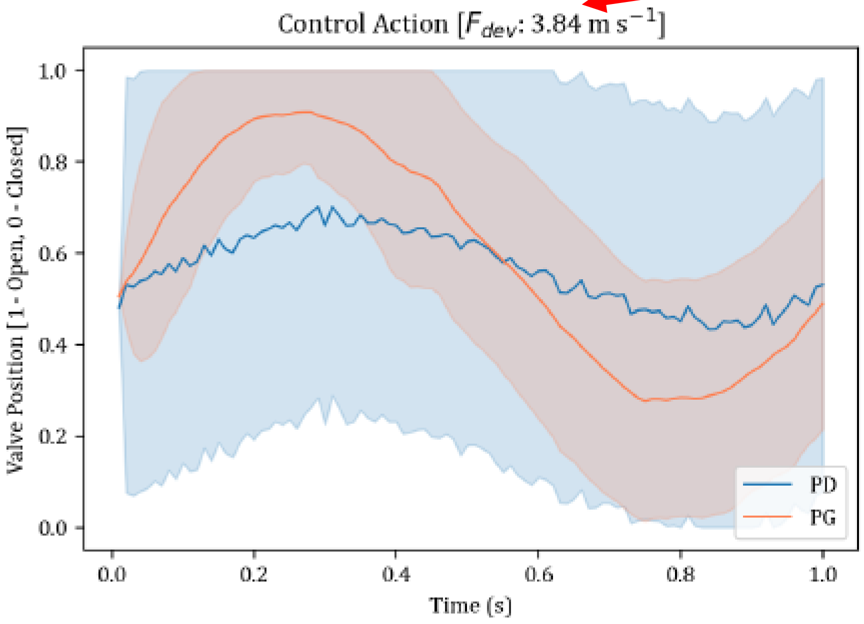
- 2 $\tanh(\cdot)$ hidden layers are used for the policy
- Only the mean is applied in the integrated part

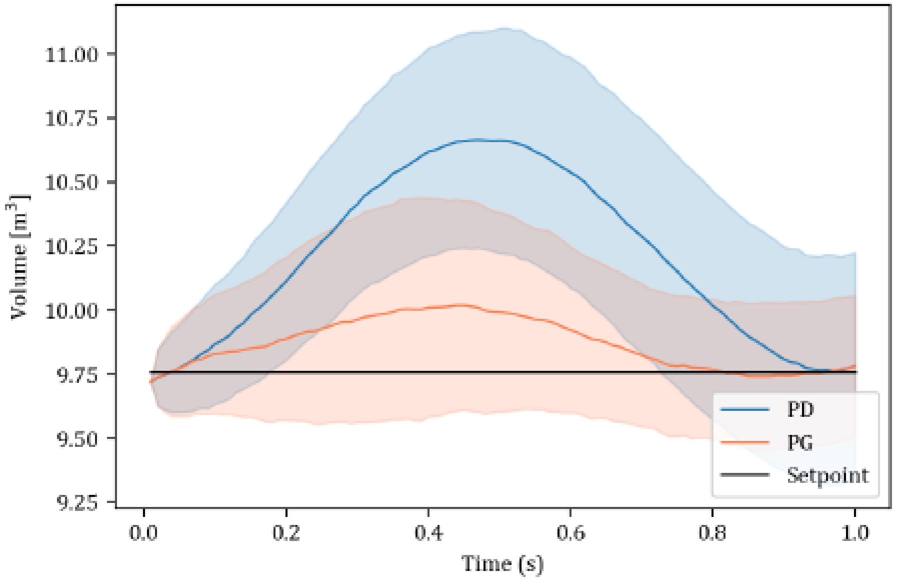
[1] Diangelakis, N. A., Burnak, B., Katz, J., & Pistikopoulos, E. N. (2017), *AIChE J.*, *63*(11), 4827–4846.

Optimization and Control of a Tank (Comparison with PD controller)

- Control task: Find a closed-loop policy $\pi_\theta(\cdot)$ that steer the volume to a desired policy
- Optimization task: Find the maximum F_{dev}
- ***Note*** that in the case of the tank the set point for the volume is dynamically correlated with the maximum value of the inlet flow rate

2.485 in [1]





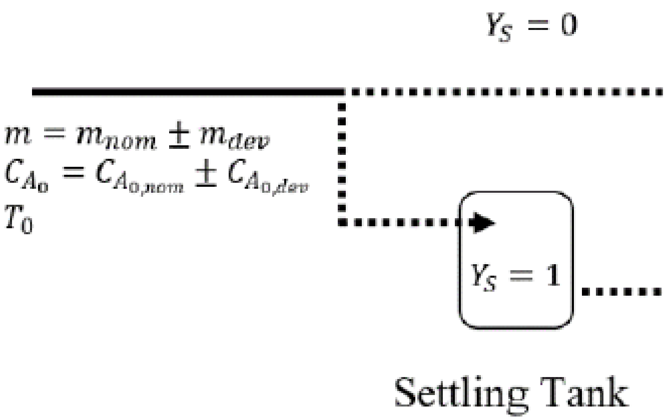
[1] Diangelakis, N. A., Burnak, B., Katz, J., & Pistikopoulos, E. N. (2017), *AIChE J.*, *63*(11), 4827–4846.

Optimization and Control of CSTR[1]

- First order endothermic reaction with heating jacket.
- Disturbance present on the inlet mass flow and the inlet concentration.
- Temperature of the CSTR should not be higher than 450 K.
- Temperature of heating jacket is the final control element.

Control task: Minimise concentration of A while satisfying the temperature constraint.

Optimization task: : Minimise a cost related objective function while satisfying the constraints.



$$\frac{dC_A(t)}{dt} = \frac{m(t)}{\rho V}\left(C_{A_0}(t) - C_A(t)\right) - k_0 C_A(t) e^{-\frac{E_A}{RT}}$$

$$\frac{dT(t)}{dt} = \frac{m(t)C_P\left(T_0 - T(t)\right) + V\Delta H_{rxn} k_0 C_A(t) e^{-\frac{E_A}{RT}} + UA\left(T_H - T(t)\right)}{V\rho C_P}$$

$$C_{A_0}(t) = C_{A_{0,nom}} + C_{A_{0,dev}} \sin(t/freq)$$

$$m(t) = m_{nom} + m_{dev} sw_{Y_S}$$

$$\frac{dsw_{Y_S}}{dt} = (1 - Y_S)(t/freq)\cos(t/freq)$$

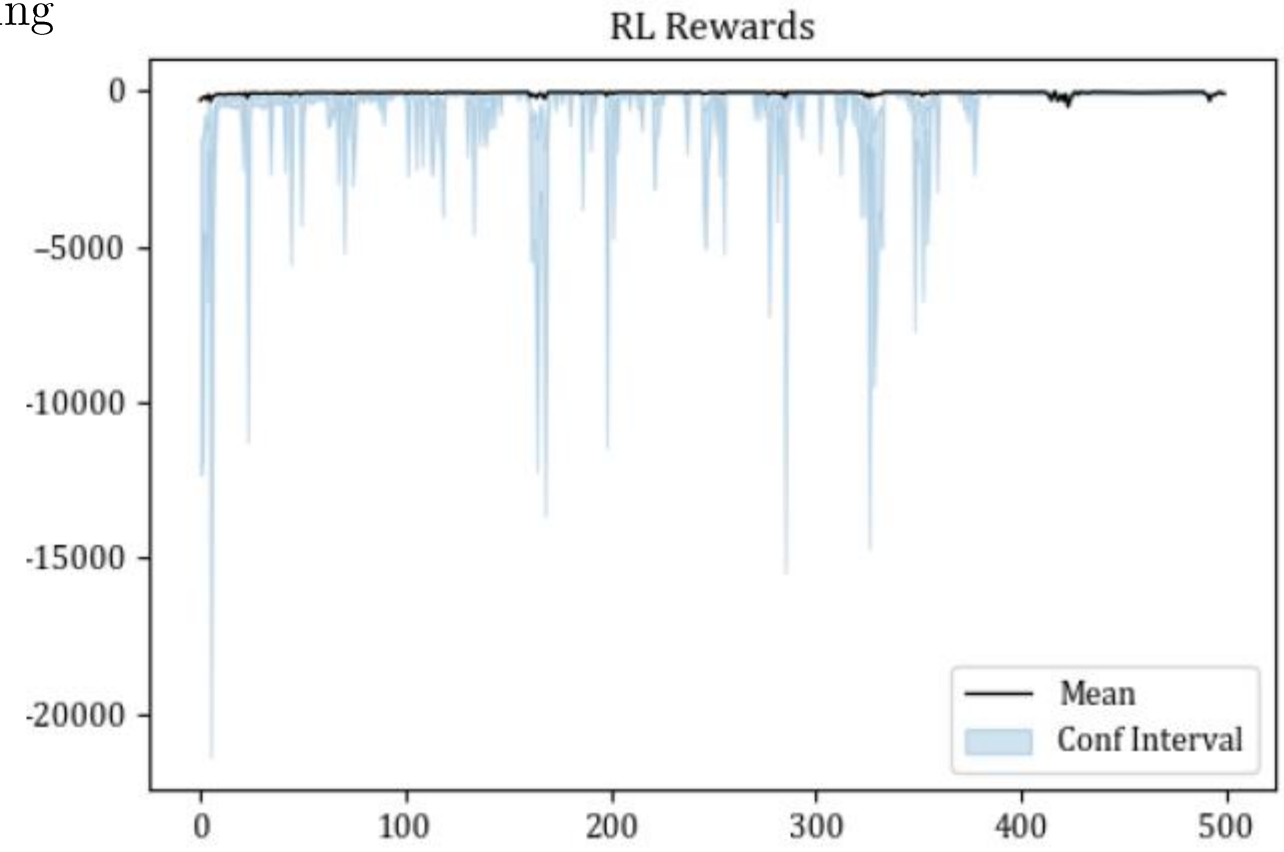
$$freq = 100/(2\pi)$$

[1] Diangelakis, N. A., Burnak, B., Katz, J., & Pistikopoulos, E. N. (2017), *AIChE J.*, *63*(11), 4827–4846.

Optimization and Control of CSTR[1]

- First order endothermic reaction with heating jacket.
- Disturbance present on the inlet mass flow and the inlet concentration.
- Temperature of the CSTR should not be higher than 450 K.
- Temperature of heating jacket is the final control element.

Results for training



[1] Diangelakis, N. A., Burnak, B., Katz, J., & Pistikopoulos, E. N. (2017), *AIChE J.*, *63*(11), 4827–4846.

$$\min_{V, m_{dev}, m_{nom}, C_{A,dev}, C_{A,nom}, Y_S} J_{SDC} = Cost_{Total}$$

$$s.t \quad \frac{dC_A(t)}{dt} = \frac{m(t)}{\rho V}\left(C_{A_0}(t) - C_A(t)\right) - k_0 C_A(t) e^{-\frac{E_A}{RT}}$$

$$\frac{dT(t)}{dt} = \frac{m(t)C_P(T_0 - T(t)) + V\Delta H_{rxn} k_0 C_A(t) e^{-\frac{E_A}{RT}} + UA\left(T_{H,t} - T(t)\right)}{V\rho C_P}$$

$$C_{A_0}(t) = C_{A_{0,nom}} + C_{A_{0,dev}} \sin(t/freq)$$

$$m(t) = m_{nom} + m_{dev} sw_{Y_S}$$

$$\frac{dsw_{Y_S}}{dt} = (1 - Y_S)(t/freq)\cos(t/freq)$$

$$freq = 100/(2\pi)$$

$$T_{raw,t} = \pi_\theta\left(C_{A,t}, C_{A,t-1}, T_t, V, C_{A_0,t}, C_A^{SP}\right) \ \forall t \in \{0, \dots, n_T - 1\}$$

$$err_{\pi_\theta} = \int_0^{T_F} C_A(t) - C_A^{SP} dt$$

$$C_A^{SP} = 0$$

Objective function:

$$Cost_{Total} = Cost_{Equipment} + Cost_{Operational}$$

$$Cost_{Equipment} = 10((V - 750)/\pi) + 1000 + 400 Y_{S,f}$$

$$\frac{dCost_{Operational}}{dt} = -m\left(C_{A_0}(t) - C_{A(t)}\right) - 4Y_S$$

Endpoint constraints:

$$err_{\pi_\theta} \leq 100$$

$$C_{A_{0,dev}} \leq C_{A_{0,nom}}$$

$$m_{dev} \leq m_{nom}$$

Interior point constraints: $\forall t \in \{0, \dots, n_T\}$

$$Y_{S,t} - Y_{S,f} \leq 0, \quad Y_{S,t} \in \{0,1\} \text{ and } Y_{S,f} \in \{0,1\}$$

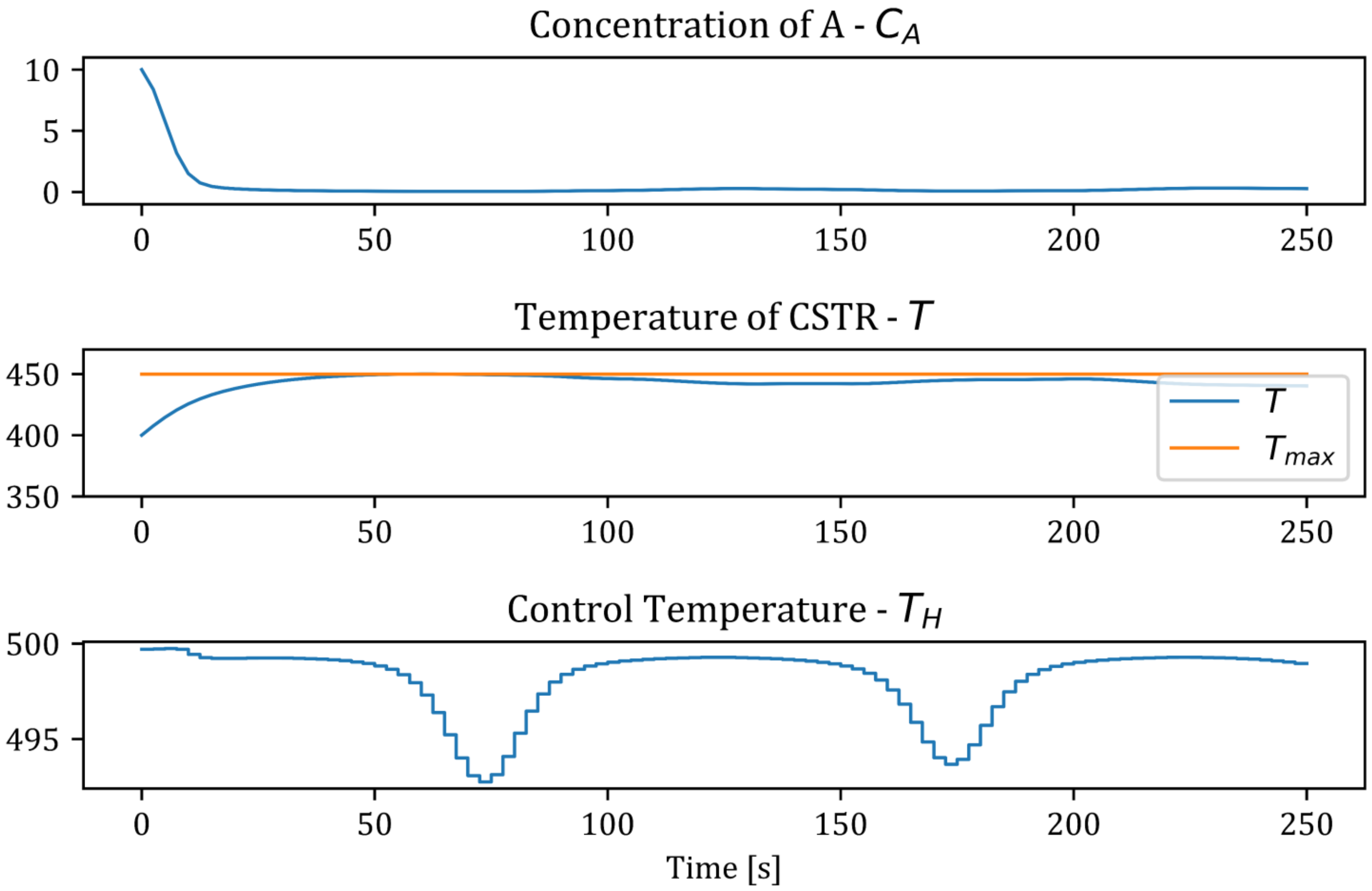
$$T_t \leq 450$$

Dynamic system

Closed-loop policy via RL

Optimization and Control of a Tank (Comparison with PD controller)

- Concentration of A was minimised with good control (very close to zero).
- Temperature of CSTR does not violate the constraint.
- Overall very good control performance of the PG controller for a wide variety of design



[1] Diangelakis, N. A., Burnak, B., Katz, J., & Pistikopoulos, E. N. (2017), *AIChE J.*, *63*(11), 4827–4846.

Conclusions and Future Work

Conclusions

- Policy gradient method shows promising results in both of the case studies.
- The method can handle constraints and measurement noise naturally which makes it very powerful for the MIDO problem.

Future work

- Integrate scheduling and planning via RL

Thanks!

Centre for
Process Systems Engineering

Sargent Centre for Process Systems Engineering

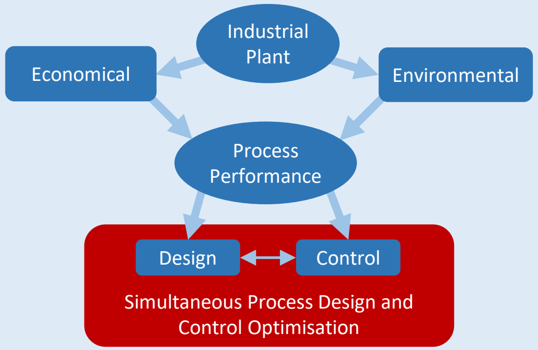
Simultaneous Process Design and Control Optimisation using Reinforcement Learning

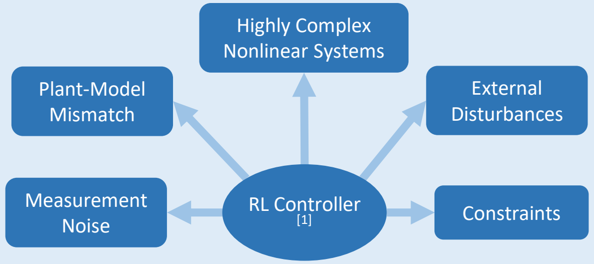
Steven Sachio[a], Antonio E. del-Rio Chanona[a], Panagiotis Petsagkourakis[b]

[a]Department of Chemical Engineering, Imperial College London, SW7 2AZ, London, UK

[b]Department of Chemical Engineering, University College London, WC1E 7JE, London, UK

IMPORTANCE





OBJECTIVES

1. Propose a new approach using reinforcement learning (**policy gradient**).

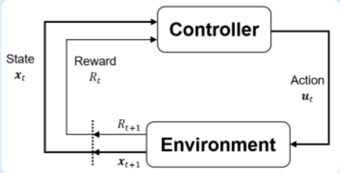
2. Showcase the control performance using two case studies from [2].

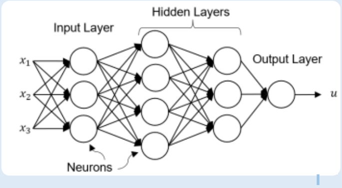
METHODOLOGY

Formulate bi-level problem

$$
\begin{aligned}
&\min_{\boldsymbol{p}} C_p(\boldsymbol{p}) + \kappa C_\kappa(\boldsymbol{x}(t), \boldsymbol{y}(t), \boldsymbol{u}(t), \boldsymbol{p}, \theta^P(t)) \\
&s.t.\ f_p(\boldsymbol{p}, \theta^P(t)) = 0 \\
&\quad g_p(\boldsymbol{p}, \theta^P(t)) \leq 0 \\
&\quad \min_{\boldsymbol{u}(\tau)} C_u(\boldsymbol{x}(t), \boldsymbol{y}(t), \boldsymbol{u}(t), \boldsymbol{p}, \theta^P(t)) \\
&\quad s.t.\ \dot{\boldsymbol{x}} = f_u(\boldsymbol{x}(t), \boldsymbol{y}(t), \boldsymbol{u}(t), \boldsymbol{p}, \theta^P(t)) \\
&\qquad g_u(\boldsymbol{x}(t), \boldsymbol{y}(t), \boldsymbol{u}(t), \boldsymbol{p}, \theta^P(t)) = 0 \\
&\qquad h_u(\boldsymbol{x}(t), \boldsymbol{y}(t), \boldsymbol{u}(t), \boldsymbol{p}, \theta^P(t)) \leq 0
\end{aligned}
$$

Prepare controller

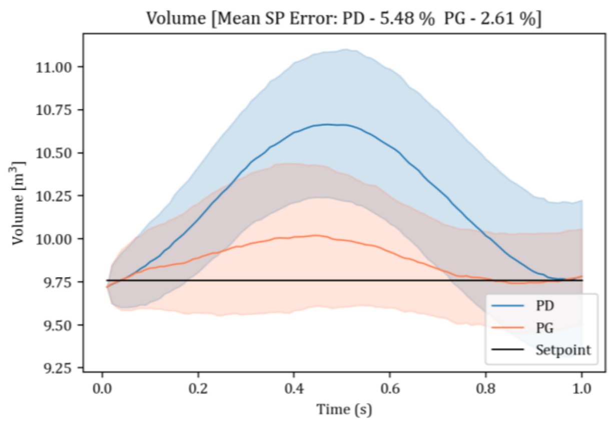




Solve the bi-level problem

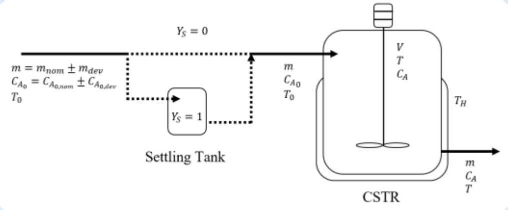
$$\boldsymbol{u}_t = \pi_\theta\left(\boldsymbol{x}_t^{obs}, \boldsymbol{x}_{t-1}^{obs}, \boldsymbol{y}_t^{obs}, \boldsymbol{y}_{t-1}^{obs}, \boldsymbol{p}^{obs}\right) \forall t$$

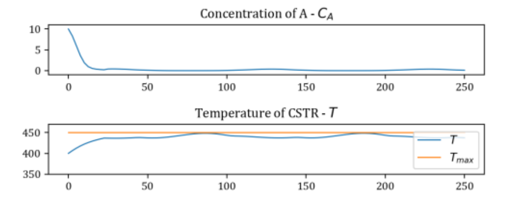
CASE STUDY 1: Tank



- Disturbance on inlet flow.
- **OCP**: Maintain **dynamic** setpoint.
- **Design**: Maximum flow **disturbance**.

CASE STUDY 2: CSTR





- Minimise reactant concentration.
- **OCP**: **Constrained** control.
- **Design**: Cost related optimisation with **binary decision** making.

CONCLUSIONS

Control Performance ✓ | Constraints ✓ | High Non-linearity [2] ✓ | Bi-linear Problem ✓

REFERENCES

[1] Petsagkourakis, P., et al. (2020) *Reinforcement learning for batch bioprocess optimization*. Computers & Chemical Engineering. [Online] 133, 106649. Available from: doi:10.1016/j.compchemeng.2019.106649.

[2] Diangelakis, N.A., et al. (2017) *Process design and control optimization: A simultaneous approach by multi-parametric programming*. AIChE Journal. [Online] 63 (11), 4827–4846. Available from: doi:10.1002/aic.15825.

Imperial College London