Mario R. Eden, Marianthi Ierapetritou and Gavin P. Towler (Editors) Proceedings of the 13th
International Symposium on Process Systems Engineering – PSE 2018
July 1-5, 2018, San Diego, California, USA © 2018 Elsevier B.V. All rights reserved.
https://doi.org/10.1016/B978-0-444-64241-7.50087-2

Reinforcement Learning Applied to Process Control: A Van der Vusse Reactor Case Study

G. O. Cassol[1], G. V. K. Campos[2], D. M. Thomaz[1], B. D. O. Capron[3*], A. R. Secchi[1]

[1]Universidade Federal do Rio de Janeiro, Programa de Engenharia Química, PEQ/COPPE/UFRJ, Caixa Postal 68.502, Rio de Janeiro, RJ 21941-914, Brazil
[2]University of California Davis, Department of Chemical Engineering, One Shields Avenue, Davis, CA 95616, United States
[3]Universidade Federal do Rio de Janeiro, Departamento de Engenharia Química, Escola de Química, Av. Horacio Macedo 30, CT-Bloco E, Rio de Janeiro, RJ 21949-900, Brazil
bruno@eq.ufrj.br

Abstract

With recent advances in industrial automation, data acquisition, and successful applications of Machine Learning methods to real-life problems, data-based methods can be expected to grow in use within the process control community in the near future. Model-based control methods rely on accurate models of the process to be effective. However, such models may be laborious to obtain and, even when available, the optimization problem underlying the online control problem may be too computationally demanding. Furthermore, the process degradation with time imposes that the model should be periodically updated to stay reliable. One way to address these drawbacks is through the merging of Reinforcement Learning (RL) techniques into the classical process control framework. In this work, a methodology to tackle the control of nonlinear chemical processes with RL techniques is proposed and tested on the well-known benchmark problem of the non-isothermal CSTR with the Van de Vusse reaction. The controller proposed herein is based on the implementation of a policy that associates each state of the process to a certain control action. This policy is directly deduced from a measure of the expected performance gain, given by a value function dependent on the states and actions. In other words, in a given state, the action that provides the highest expected performance gain is chosen and implemented. The value function is approximated by a neural network that can be trained with pre-simulated data and adapted online with the continuous inclusion of new process data through the implementation of an RL algorithm. The results show that the proposed adaptive RL-based controller successfully manages to control and optimize the Van de Vusse reactor against unmeasured disturbances.

Keywords: Reinforcement Learning; Model Predictive Control; Adaptive Control.

1. Introduction

In the past few decades the process industry has experienced a large increase in automation and data acquisition systems, and as acquiring good quality and reliable data becomes easier and cheaper, data-based methodologies become more susceptible to application. Machine Learning and Data Science methods have grown in importance outside of the process community, driven by successful applications in real-life problems. It is important for the process industry to explore the application of data-based methods, and incorporate the most successful ideas to the traditional process control framework.

Another incentive for the application of data-based methods arises from the drawbacks of classical model-based control. Most model predictive control (MPC) methods still rely on the assumption of linearity, which is not true for the majority of chemical processes. Nonlinear MPC (NMPC) methods may be applied, but they require a nonlinear process model that may be laborious to obtain. In addition, solving the NMPC optimization problem may be too computationally demanding for online applications.

In order to tackle some of the aforementioned problems, a control approach based on model-free learning may be used. In the process control community model-free control strategies have been proposed since the early 1960s, while in the artificial intelligence community the field of reinforcement learning (RL) was developed during the 1980s, focusing on the design of control algorithms that work solely based on data (Busoniu *et al.*, 2010). There is not yet a vast literature on the application of RL techniques to the control of chemical processes, as evidenced by Syafiie *et al.* (2008), but a few authors proposed general purpose model-free controllers in the past. Stenman (1999) proposed a partially model-free approach combining MPC with adaptive techniques that estimate linear models online, called model-on-demand. Lee and Wong (2010) proposed a modification to the stochastic MPC framework based on approximate dynamic programming (ADP), and explored different aspects of the problem such as parametric vs. nonparametric value function approximators, and batch vs. continuous learning. Morinelly and Ydstie (2016) formulated an adaptive optimal control algorithm for systems with uncertain dynamics, called RL dual MPC (RLdMPC), in which an exploratory component is embedded in the objective function of the MPC and the competing results of the optimal solution and the cost function approximation provide a balance in exploration and exploitation.

Focusing on the application of RL techniques to chemical processes, Hoskins and Himmelblau (1992) proposed a neural network (NN) based control, in which two different NN models are used to predict the control performance measure and the control action/policy. Martinez (2000) proposed a sampling strategy (sequential experiment design) based on RL for optimization of batch processes, where the RL solution is used to systematically shrink the region of interest for the optimization. Syafiie *et al.* (2008) proposed a model-free learning control method based directly on RL techniques, more specifically on Q-learning with finite-discrete action space. Although easy to implement, the discrete version of Q-learning may not be the most appropriate for chemical processes, in which states and actions are continuous and must be discretized to obtain a good representation. Shah and Gopal (2016) proposed a model-free predictive control, employing a fuzzy inference system to approximate the value function. While the fuzzy model may account for uncertainties, it is still subject to the curse of dimensionality and requires tuning of many localized parameters.

In the present work, an adaptive controller based on an RL algorithm with value function approximation via NN is proposed and tested for the control of a nonlinear non-isothermal CSTR with the Van de Vusse reaction. We aim to assess the application of the RL algorithm to a chemical process, while addressing aspects such as continuous representation of states, combined optimization and control, and comparison with linear MPC.

2. Methodology

The problem addressed by RL is that of decision-making. An agent (a controller in our case) receives information about the state x of the system (process) and implements a policy, a mapping from the states into actions, which defines which action the controller has to take based on the actual state of the process. This policy is derived from the so-called action-value function $Q_{(x,u)}$, computed from data by an RL algorithm, that assigns an expected performance gain or return value G over time to every possible state-action pair. In our work, a multilayer perceptron (MLP) NN is employed to approximate $Q_{(x,u)}$ in order to use a continuous representation of the states.

A policy iteration algorithm is used to compute the policy used by our controller. Iteratively, the action-value function for the actual policy is evaluated using the Monte Carlo policy evaluation and the policy is then improved using the well-known *ε-greedy* algorithm (Busoniu *et al.*, 2010), in which the action that leads to the maximum value function is chosen with a 1−*ε* probability. Otherwise, a random action is taken to help guaranteeing that all the actions are considered in the optimization of the return value in a given state. The NN is trained taking into account the states and actions as inputs and the expected return value as output.

Different configurations of the controller are tested in this work:

(i) Pre-trained non-adaptive RL-based Controller (RL-C): the NN is only trained with pre-simulated data corresponding to different initial steady-state conditions randomly selected.

(ii) Untrained adaptive RL-C: the NN is only trained with the process data acquired online. In this case, a step control is used to take into account the difference e_{k-1} between the predicted reward from the NN model and the actual reward obtained in the last step. The control action is limited when the NN model is not capable to correctly predict the outcome, as it is shown in Eq. (1), where a_k^P is the action predicted by the NN and a_k^R the action really taken by the controller.

$$a_k^R = a_k^P\left(1 - 0.95 \cdot 1/\left(1 + e^{|e_{k-1}|}\right)\right) \tag{1}$$

(iii) Pre-trained adaptive RL-C: the NN is pre-trained and continues to be trained online. Two hidden layers of seventy neurons are used by the NN model. The hidden layers use logistic functions and the output layer a linear one. The NN training is performed using MATLAB NN toolbox, which provides batch and online algorithms. 80% of the available data were used for training, 10% for validation and 10% for testing.

3. Results

3.1. Process description

The performance of the proposed controller is tested through the simulation of the control of a non-isothermal CSTR, with the Van de Vusse reaction, as described by Klatt and Engel (1998). The reaction consists in the synthesis of cyclopentenol (B) from cyclopentadiene (A) by acid-catalyzed electrophilic addition of water in a dilute solution. Due to the strong reactivity of both A and B, dicyclopentadiene (D) is produced by the Diels-Alder reaction as a side product, and cyclopentanediol (C) as a consecutive product by addition of another water molecule. It is usually desirable to maximize the production of component B, while minimizing the production of C and D. The reaction is presented in Eq. (2), and the mass and energy balances in Eq. (3) to (5), where $C_{i,in}$ and C_i are the inlet and outlet concentration of the i-th component, $k_{j(T)}$ is the reaction rate parameter for reaction j (according to Arrhenius law), ΔH_{R_j} is the constant heat of reaction j, T_k is the jacket temperature, and other parameters are as defined in classical notation. The parameters values can be found in Klatt and Engel (1998).

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C \qquad 2A \xrightarrow{k_3} D \tag{2}$$

$$\frac{dC_A}{dt} = \frac{F}{V}\left(C_{A,in} - C_A\right) - k_1(T)C_A - k_3(T)C_A^2 \tag{3}$$

$$\frac{dC_B}{dt} = \frac{F}{V}\left(C_{B,in} - C_B\right) + k_1(T)C_A - k_2(T)C_B \tag{4}$$

$$\begin{aligned}\frac{dT}{dt} = \frac{1}{\rho \cdot C_p}\left[k_1(T)C_A\left(-\Delta H_{R_1}\right) + k_2(T)C_B\left(-\Delta H_{R_2}\right) + k_3(T)C_A^2\left(-\Delta H_{R_3}\right)\right] + \\ \frac{F}{V}(T_{in} - T) + \frac{K_w\, A_R}{\rho\, C_p\, V}(T_k - T)\end{aligned} \tag{5}$$

This process was chosen due to its nonlinear characteristics, the most important of those being the gain inversion of the product concentration C_B with respect to inlet flow F. The location of the gain inversion depends on the reactor temperature T, which in turn is controlled by manipulating the cooling jacket temperature T_k.

3.2. Controller definition

The actions vector u includes the manipulated variables inlet flow F and cooling jacket temperature T_k increments. The state vector x includes the outlet concentrations C_A and C_B, the reactor temperature T, and the manipulated variables F and T_k. The control objective is to track a certain F_B ($F.C_B$) while maximizing C_B. The corresponding instantaneous reward R is defined in Eq. (6), where F_B^{SP} is the setpoint for F_B and the subscript k corresponds to the time instant. The proposed reward function prioritizes the set-point tracking of F_B (the sigmoidal function used goes to 1 when the difference between F_B and its set-point is large). Then, when F_B is close to the set-point, the reward starts to focus on the optimization of C_B. The weight w is used as a tuning factor. The return G, or accumulated reward, is simply defined as the discounted sum of rewards, as shown in Eq. (7), where $\gamma \in [0,1]$ is the discount factor, a measure of how “far-sighted” the controller is in considering its rewards.

$$\begin{aligned}R(k) = |F_B(k-1) - F_B^{SP}| - |F_B(k) - F_B^{SP}| \\ + w \cdot [C_B(k) - C_B(k-1)]\left\{1 - 1\Big/\left(1 + e^{\left|\frac{F_B(k) - F_B^{SP}}{F_B^{SP}}\right|}\right)\right\}\end{aligned} \tag{6}$$

$$G(\hat{k}) = \sum_{k=0}^{\hat{k}} \gamma^k R(k) \tag{7}$$

3.3. Initial Conditions

The initial conditions used for the NN model training in different episodes are random steady states for different values of F and T_k, with $C_{A,in} = 5.1$ mol/L, $C_{B,in} = 0$ mol/L and T_{in} = 110 °C. The initial steady state considered for the following results is defined by F = 1000 L/h and T_k = 100 °C, with the same conditions for $C_{A,in}$, $C_{B,in}$ and T_{in} as for training. The F_B set-point was set to 2000 mol/h. The time step used for all simulations was 0.01 h.

3.4. Process Control/Optimization without disturbances

In this section, the performance of the pre-trained non adaptive RL-C, the untrained adaptive RL-C, the linear MPC controller and the nonlinear MPC (NMPC) are compared when changes in the operating conditions take place and no disturbances enter the system. The model of the MPC is obtained from the linearization of the nonlinear model defined in Section 3.1 at the initial condition, when disturbance steps of ±10% are applied to the manipulated variables. The NMPC uses the complete model of the system.

The results in Figure 1 show that the linear MPC is not able to maintain F_B at the desired set-point. This system presents a gain inversion of C_B with respect to the manipulated variables, which means the available model given to the MPC is no longer accurate. The NMPC is the first to achieve the desired set-point of F_B, but takes more time than the pre-trained RL-C to reach the maximum of C_B. The pre-trained non adaptive RL-C, having been trained with data from different initial conditions, quickly manages to bring F_B to its set-point while maximizing C_B. Lastly, the untrained adaptive RL-C expresses a behavior that is initially similar to the linear MPC strategy, but then manages to change its course employing actions that are more similar to the pre-trained NN model. In addition, it is possible to verify that the untrained adaptive RL-C first takes actions that brings F_B to its set-point, and then starts to make changes in the system to optimize C_B (the control strategy "walks" in the F_B set-point curve).

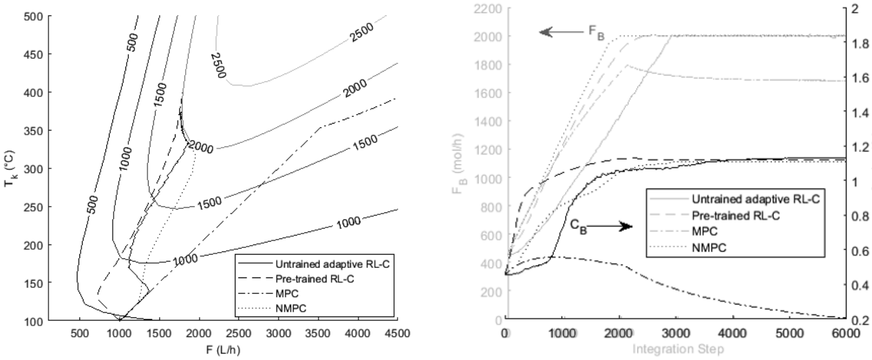

Figure 1. Simulation results without unmeasured disturbances

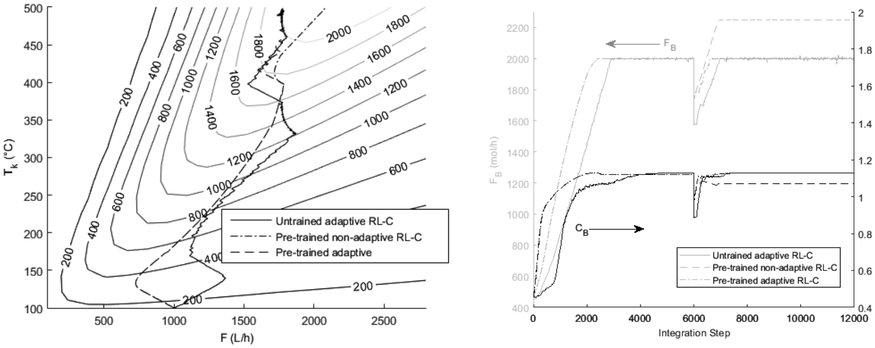

Figure 2. Simulation results with unmeasured disturbance

3.5. Process Control/Optimization with unmeasured disturbances

In this section, the performances of the three configurations of the RL-C defined in the methodology section are compared when an unmeasured disturbance occurs (the inlet temperature T_{in} changes from 110 ºC to 90 ºC at time step 6000). The results in Figure 2 show that both pre-trained strategies have the same behavior when the system is not disturbed. However, as expected, after the unmeasured disturbance occurs, only the pre-trained adaptive RL-C is able to bring the system back to the F_B set-point and to adapt to the new process conditions that are different from the ones used for training. In addition, the untrained adaptive RL-C also manages to bring F_B back to its set-point and to optimize C_B before and after the unmeasured occurs, although in a less efficient way, showing that the previous knowledge from the data can enhance its performance.

4. Conclusions

In this work a continuous RL algorithm was applied to the problem of adaptive control of chemical processes, in particular of chemical reactors with nonlinear response and gain inversion. A MLP neural network model was used to represent the action value function in terms of the continuous states and actions, which could be initially trained with a batch of pre-simulated process data. The results show that the adaptive RL-based controller successfully manages to control and optimize the Van de Vusse reactor against unmeasured disturbances and verify that the use of previous data from the system can help the NN model to adapt in a better way to changes in the system's behavior.

References

L. Busoniu, R. Babuska, B. De Schutter, D. Ernst, 2010, Reinforcement Learning and Dynamic Programming Using Function Approximators, CRC Press.

J. C. Hoskins, D. M. Himmelblau, 1992, Process Control via Artificial Neural Networks and Reinforcement Learning, Computers and Chemical Engineering, Vol. 16, No. 4, pp. 241-251.

K.-U. Klatt, S. Engell, 1998, Gain-scheduling trajectory control of a continuous stirred tank reactor, Computers & Chemical Engineering 22, Issues 4–5, pp. 491-502.

J. H. Lee, W. Wong, 2010, Approximate dynamic programming approach for process control, Journal of Process Control, Vol. 20, pp.1038– 1048.

E. C. Martinez, 2000, Batch Process Modeling for Optimization Using Reinforcement Learning, Computers and Chemical Engineering, Vol. 24, pp. 1187-1193.

J. E. Morinelly, B. E. Ydstie, 2016, Dual MPC with Reinforcement Learning, IFAC-PapersOnLine 49-7, pp. 266–271.

M. A. Mustafa, J. A. Wilson, 2012 Application of Reinforcement Learning to Batch Distillation, In: Proceedings of the Sixth Jordan International Chemical Engineering Conference, Amman, Jordan.

H. Shah, M Gopal, 2016, Model-free Predictive Control of Nonlinear Processes Based on Reinforcement Learning, IFAC-PapersOnLine 49-1 (2016) 089–094

A. Stenman, 1999, Model-free predictive control, In: Proceedings of the 38th IEEE Conference on Decision & Control, Phoenix, AZ, USA, pp.3712–3717.

S. Syafiie, F. Tadeo, E. Martinez, 2008, Model-Free Learning Control of Chemical Processes, Inside "Reinforcement Learning: Theory and Applications", Edited by C. Weber, M. Elshaw, N. M. Mayer, I-Tech Education and Publishing, pp.424.