Reinforcement Learning Based H Control for Oil & Gas De-Oiling System

Li, Shaobao; Durdevic, Petar; Yang, Zhenyu

Publication date:
2019

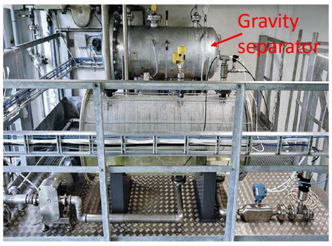
Reinforcement Learning Based H_∞ Control for Oil & Gas De-Oiling System

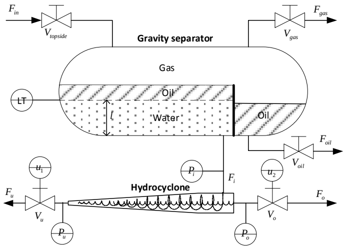
Department of Energy Technology, Aalborg University, Niels Bohrs Vej 8, 6700 Esbjerg, Denmark

Shaobao Li, Petar Durdevic and Zhenyu Yang

Introduction

A de-oiling system consisting of a set of gravity separators and hydrocyclones is used to separate oil from water in O&G production, to ensure low OiW concentration in the discharge. PID is currently used for de-oiling system control, but it is not always effective to guarantee separation efficiency. H_∞ control has been verified its effectiveness comparing with PID controllers in our previous works. However, the current H_∞ control is model-based, requiring a lot of work for system identification. Therefore, it is difficult to transfer the developed H_∞ control algorithms into different industrial facilities. In this work, we aim to develop an automatic control generation method such that the de-oiling control can automatically learn the optimal control policy from its behaviour in an online manner, i.e., learning from data without requiring system identification.

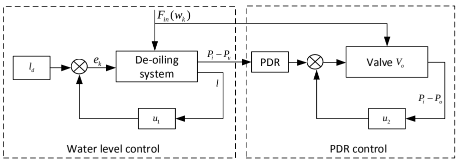




A de-oiling facility and its structure diagram

System Description

We consider the combined separator level control and hydrocyclone PDR control together, where PDR= $(P_i - P_o)/(P_i - P_u)$. From functional point of view, we formulate the control problem for a cascade system as shown in the following:



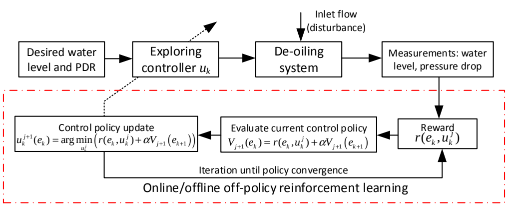
Two subsystems have similar control structure. The objective is to find a controller such that the influence of the disturbance w_k to the tracking error e_k can be attenuated within a desired bound governed by γ as follows

$$\frac{\sum_{k=0}^{\infty} \alpha^k (e_k^T Q e_k + u_k^T R u_k)}{\sum_{k=0}^{\infty} \alpha^k || w_k ||^2} \leq \gamma^2 \quad (1)$$

which is equivalent to finding a Nash equilibrium of the following cost function

$$V(e_k, u_k^*, w_k^*) = \min_{u_k} \max_{w_k} \sum_{i=k}^{\infty} \alpha^{i-k} (e_k^T Q e_k + u_k^T R u_k - \gamma^2 || w_k ||^2) \quad (2)$$

This is a zero-sum game problem that u_k and w_k are the two players want to maximize their own benefits. Reinforcement learning is applied to solve this problem because system dynamics are considered to be unknown.



Key idea of the RL algorithm

Acknowledgment

The author would like to thank Stefan Jespersen from AAU, for his technical support in algorithm testing, and also thank supports from the DHRTC CTR1 AAU projects.

Model-Free H_∞ Control via RL

An off-policy RL algorithm is developed for optimal control policy learning. The system is written into the following form for off-policy learning:

$$x_{k+1} = Ax_k + Bu_k^j + Ew_k^j + B\left(u_k - u_k^j\right) + E(w_k - w_k^j) \quad (3)$$

We use a fixed control policy u_k to generate data x_k under disturbance w_k. System matrices A, B and E are not required to be known. The data is used to learn the optimal control policy u_k^j and disturbance policy w_k^j iteratively via

Algorithm 1: State feedback control via RL

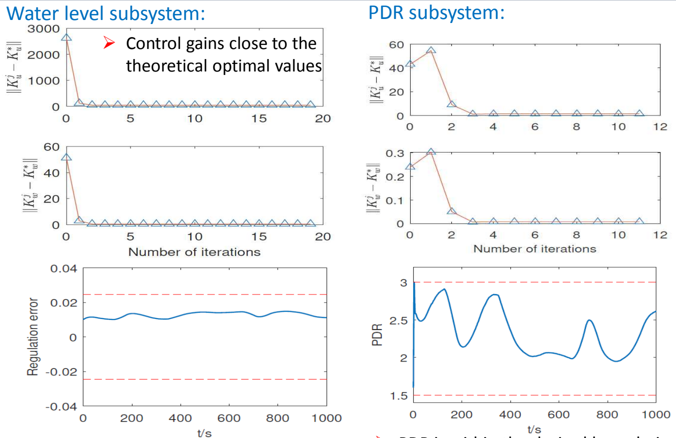
1. *Data generation:* give a fixed control policy u_k (e.g., PID) to system to collect data x_k, u_k, w_k.
2. *Initialization:* Give initial stable policies $u_k^0 = K_u^0 x_k$ and $w_k^0 = K_w^0 x_k$.
3. *Policy evaluation:* Solve for $V^j, \nabla V^{jT} B, \nabla^{jT} E$ simultaneously through
$$V^j(x_k) - V^j(x_{k+1}) = (e_k^T Q e_k + u_k^{jT} R u_k^j - \gamma^2 || w_k^j ||^2) + \nabla V^{jT} B\left(u_k - u_k^j\right) + \nabla V^{jT} E\left(w_k - w_k^j\right) \quad (4)$$
using a critic neural network
$$V^j(x_k) = W^T \varphi(x_k) \quad (5)$$
4. *Policy updating:*
$$u_k^{j+1} = -\frac{\alpha}{2} R^{-1} B^T \nabla V^j(x_k) \quad (6)$$
$$w_k^{j+1} = \frac{\alpha}{2\gamma^2} E^T \nabla V^j(x_k) \quad (7)$$
5. Go to step 3 until W reach convergence.

Algorithm 2: Output feedback control via RL (x_k is not measurable)

1. *Data generation:* give a fixed control policy u_k (e.g., PID) to system to collect historical data for state estimation
$$\zeta_k = (y_{k-1}, \cdots, y_{k-N}, u_{k-1}, \cdots, u_{k-N}, w_{k-1}, \cdots, w_{k-N}) \quad (8)$$
2. *Initialization:* Give initial stable policies $u_k^0 = K_u^0 \zeta_k$ and and $w_k^0 = K_w^0 \zeta_k$.
3. *Policy evaluation:* Solve for $V^j, \nabla V^{jT} B, \nabla^{jT} E$ simultaneously through
$$V^j(\zeta_k) - V^j(\zeta_{k+1}) = (e_k^T Q e_k + u_k^{jT} R u_k^j - \gamma^2 || w_k^j ||^2) + \nabla V^{jT} B\left(u_k - u_k^j\right) + \nabla V^{jT} E\left(w_k - w_k^j\right) \quad (9)$$
using a critic neural network
$$V^j(\zeta_k) = W^T \varphi(\zeta_k) \quad (10)$$
4. *Policy updating:*
$$u_k^{j+1} = -\frac{\alpha}{2} R^{-1} B^T \nabla V^j(\zeta_k) \quad (11)$$
$$w_k^{j+1} = \frac{\alpha}{2\gamma^2} E^T \nabla V^j(\zeta_k) \quad (12)$$
5. Go to step 3 until W reach convergence.

Simulation Results

Water level subsystem: PDR subsystem:



- Water level tracking error is within the boundaries given by $\pm \frac{\gamma w_{max}}{\sqrt{||Q||}}$.
- PDR is within the desired boundaries 1.5—3.

References

[1] P. Durdevic and Z. Yang, "Application of H_∞ robust control on a scaled offshore oil and gas de-oiling facility," *Energies*, 11(2): 287, 2018.
[2] P. Durdevic and Z. Yang, "Dynamic efficiency analysis of an offshore hydrocyclone system, subjected to a conventional PID-and robust-control solution," *Energies*, 11(9): 2379, 2018.
[3] B. Kiumarsi, F. L. Lewis, and Z.-P. Jiang, "H_∞ control of linear discrete-time systems: Off-policy reinforcement learning," *Automatica*, 78:144-152, 2017.
[4] F. L. Lewis and K. G. Vamvoudakis, "Reinforcement learning for partially observable dynamic processes: Adaptive dynamic programming using measured output data," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 41(1):14-25, 2011.
[5] F. L. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *IEEE circuits and systems magazine*, 9(3):32-50, 2009.
[6] H. Modares and F. L. Lewis, "Optimal tracking control of nonlinear partially-unknown constrained-input systems using integral reinforcement learning," *Automatica*, 50(7):1780-1792, 2014.