Proc. 15th European Meeting on Cybernetics and Systems Research (EMCSR 2000), Vienna, 2000, vol. 2, pp. 555-558,
Austrian Society for Cybernetic Studies, 2000

Reinforcement-Learning-Based Control Scheme for a Visually Guided Combustion Process *

V. Stephan†, H.-M. Gross†, K. Debes†, F. Wintrich‡, H. Wintrich‡

† Department of Neuroinformatics, Ilmenau Technical University
98684 Ilmenau, Germany
volker.stephan@informatik.tu-ilmenau.de

‡ ORFEUS Combustion Engineering GmbH, 45128 Essen, Germany
info@orfeus.de

Abstract

In this paper, we present a control scheme based on reinforcement-learning for a hard-coal combustion process in a power plant. Because of great demands on environmental protection, the plant operator is interested in a minimization of the nitrogen oxides emission on one hand, while other process parameters have to be kept within predefined limits. On the other hand, an increased efficiency factor is also desired. In order to meet these requirements, we extract visual features of the flame in addition to conventionally measured process data, and apply a reinforcement-based control scheme.
Keywords: Reinforcement-Learning, combustion process, visual flame observation

1 Introduction

Since the immediate object of a power plant is the production of energy, the plant operator is trying to maximize the efficiency factor. In parallel, both the system-constraints and great demands on environmental protection limit the workspace. Because of time varying plant properties caused by pollution, fair wear and tear, changing coal qualities, etc., a control system is sought, which autonomously tries to minimize a predefined cost function.

Reinforcement learning (RL) can be used to solve such problems. The main idea of RL consists in using experiences obtained through interaction with the process to progressively learn an optimal value function. This function predicts the best long-term outcome an agent can receive from a given state when it applies a specific action and follows the optimal policy thereafter [Sutton, 1988]. The agent can use a RL-algorithm such as SUTTON's $TD(\lambda)$ algorithm [Sutton, 1988], or WATKINS' Q-learning algorithm [Watkins and Dayan, 1992] to improve the long-term estimate of the value function associated with the current state and the selected action. However, in systems having continuous state and action spaces, the value function must operate with real-valued variables representing states and actions. Therefore, the value functions are typically represented by *function approximators*, which use finite resources to represent the value of continuous state-action pairs. Function approximators are useful because they can generalize the expected return of state-action pairs the agent actually experiences to other regions of the state-action-space. Thus, the agent can estimate the expected return of state-action pairs that it has never experienced before. Many classes of function approximators have been presented, each with advantages and disadvantages. The choice of a function approximator depends mainly on how accurate it is in generalizing the values for unexplored state-action pairs, and how expensive it is to store in memory.

To the best of our knowledge, this paper is first to present a reinforcement-learning approach to control the combustion process of a power plant. There exist alternative approaches to control such complex combustion processes, which extensively use process models [Baldini et al., 1999]. But these systems crucially require detailed information about the plant to build the model. Therefore, the quality of the process model limits the quality of the control-strategy and decreases the portability to other plants.

Finally let us shortly introduce the power plant "Tiefstack" we used for our experiments. It is owned by the "Hamburgische Elektrizaetswerke" (HEW) and is situated in the south of Hamburg. The subsystem to be controlled consists of 6 burners aligned in 2 columns at 3 levels and has a maximal output of 252MW (see figure 1). The burners at each level are supplied with coal by one coal mill. Although the distribution of inlet coal should be equal for each of the two supplied burners, due to varying flow dynamics or pollution this equilibrium is shifted to the benefit of one burner. Unfortunately, the exact amount of inlet coal can not be measured for each burner separately. In order to detect such asymmetries, we observe all 6 flames by camera systems and use this information to control the combustion process.

2 Architecture

Before describing our architecture, we first discuss the interface to the process: the reinforcement function,

*This work is supported by the German BMBF (grant number 032 6843 B)

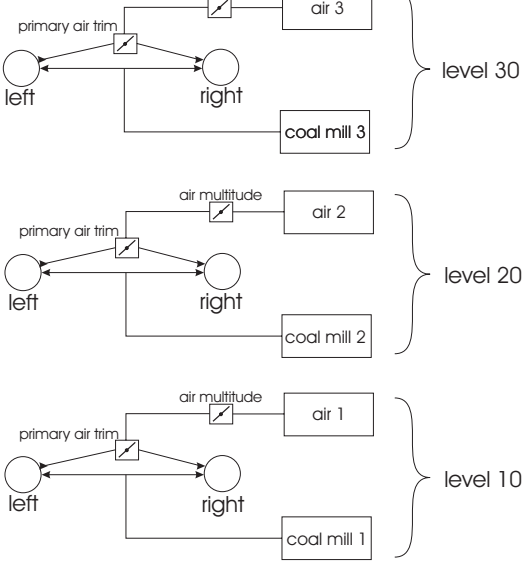

Figure 1: Schematical view of the combustion chamber with coal and air supply.

the controls, and last but not least, the measured values describing the combustion process.

2.1 Reinforcement-function

The reinforcement-function is the central part of any RL-system, since it defines the goal of the control system. It was defined in accordance with the plant operator as follows:

- the temperature of the steam entering the turbine must not fall below 540° C
- the waste gas temperature must not fall below of 340° C
- the nitrogen oxides (NO_x) concentration in exhaust fumes must not exceed $1200mg/m^3$ and has to be minimized
- the concentration of unused carbon must not exceed 5%
- the O_2 concentration in waste gas must not fall below 3%
- minimize the complete inlet air multitude of the combustion process in order to increase the efficiency factor

Equation 1 shows the mathematical description of the requirements stated above. The terms K_{NO_x} and K_λ allow to balance the importance of the NO_x concentration and the efficiency value. We used for our experiments $K_{NO_x} = K_\lambda = 0.5$.

$$r = \begin{cases} 0 : any\ threshold\ violated \\ 1 - K_\lambda \frac{air'}{coal'} - K_{NO_x} * NO'_x : else \end{cases} \quad (1)$$

r	:	scalar reinforcement signal
air'	:	normalized air consumption
$coal'$	:	normalized coal consumption
NO'_x	:	normalized NO_x concentration in exhaust fumes
K_{NO_x}	:	importance of NO_x concentration
K_λ	:	importance of efficiency factor

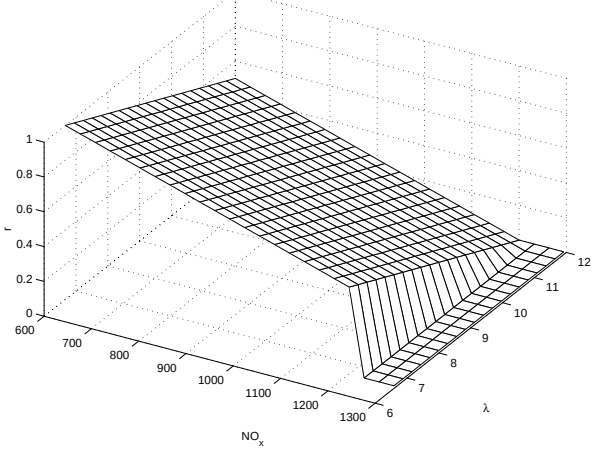

Figure 2: Reinforcement-function, if none of the thresholds described above is violated.

Figure 2 depicts the reinforcement-function in case none of the formulated thresholds is violated. As can be seen, the maximum reinforcement of 1.0 is only given, if both the air consumption and the NO_x concentration are minimal.

2.2 Controls

The plant operator has given us direct access to the following controls:

control	meaning
primary air trim at level 10	distribution of the primary air multitude on the lower burner level between the left and right burner
primary air trim at level 20	distribution of the primary air multitude on the middle burner level between the left and right burner
primary air trim at level 30	distribution of the primary air multitude on the upper burner level between the left and right burner
air multitude at level 10	air multitude on the lower burner level
air multitude at level 20	air multitude on the middle burner level
air multitude at level 30	air multitude on the upper burner level

Please hold in mind, that these controls (see also figure 1) only influence the air multitude and the distribution of air between these 6 burners, but neither multitude nor distribution of inlet hard-coal! To reduce the immense action space we use relative instead of absolute controls. That means, we define for each control only three actions: increase by 1%, remain unchanged or decrease by 1% (the use of absolute controls in 1% steps would take 21 actions, if we assume a control-range of 40%...60%, for instance for the primary air trims). Because all 6 controls are independent of each other, our control system has to cope with $3^6 = 729$ different actions.

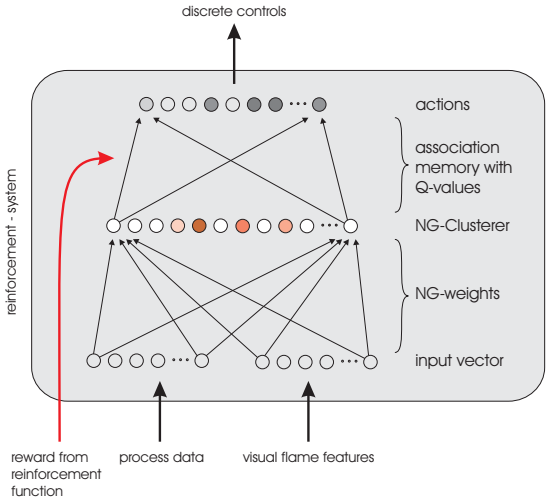

Figure 3: Neural control architecture for reinforcement-learning. The input-vector unifies process data and flame-describing features obtained from the camera systems observing the 6 flames of the combustion process. The neural clusterer maps the continuous and high dimensional input-space onto a discrete state-space, whereupon the Q-values of all executable actions are estimated.

2.3 Process describing data

After describing the reinforcement-function and the controls we have to select a subset from all measured values of the combustion process to describe the current process situation. This selection is very difficult, because to much information can be as paralyzing as insufficient information about the current process situation. If the control system has not enough information about the combustion process, the process is not sufficiently observable (POMDP) and therefore no stable outcomes of state-action pairs could be observed.

First of all, the control system needs all reinforcement-relevant process data. Furthermore information about the coal distribution between the 6 burners is essential, since our controls only distribute the air multitudes between these burners. Hence, the input-feature-space is determined by the following process data:

- currently inlet coal and air multitudes
- temperature of the steam entering the turbine
- waste gas temperature
- NO_x concentration in waste gas
- O_2 concentration in waste gas
- concentration of unused carbon
- mean flame intensity of all 6 burners

These process data constitute the input-vector consisting of 13 real-valued components.

2.4 Neural function approximator

As mentioned in section 1, our architecture has to select an action for each process situation, that is promising the highest future reward. In this paper, we present a very first and simple approach to a state-action function approximator that combines a neural vector quantization technique (Neural Gas [Martinetz and Schulten, 1991]) for optimal clustering of a high-dimensional, continuous input space [Gross et al., 1998] (equation 2) with a subsequent associative memory, to estimate the values of all 729 executable actions (see figure 3). Equation 2 shows the neural-gas weight $\underline{w}_i(t)$ updating rule for the neuron i, where $\eta^{NG}(t)$ is a learning rate, $s(i)$ is the index of neuron i in the list sorted by distance to the input $\underline{x}(t)$ and $h(t)$ is the learning radius.

$$\Delta\underline{w}_i(t) = \eta^{NG}(t) \cdot e^{-\frac{s(i)}{h(t)}} \cdot [\underline{x}(t) - \underline{w}_i(t)] \quad (2)$$

For action-value approximation, we utilize the Q-learning [Watkins and Dayan, 1992] variant of reinforcement-learning (equation 4).

$$\Delta Q(s^t, a^t) = \eta\Big\{r^t + \gamma V(s^{t+1}) - Q(s^t, a^t)\Big\} \quad (3)$$

with

$$V(s^{t+1}) = \max_a Q(s^{t+1}, a^{t+1}) \quad (4)$$

For our experiments we have 75 states (NG-neurons), a discount factor for the value of the subsequent state $\gamma = 0.5$, and a Q-learning-rate of $\eta = 0.2$.

3 Results

One of the most important properties of reinforcement-learning techniques is their ability to extract the necessary information about the consequences of their actions through interaction with the process itself. Neither detailed and system-specific information nor an accurate model of the underlying process are required. Nevertheless, we used a system, that we pretrained on past process data, to reduce the exploration time and space of the system. This explorative behavior is realized by addition of a noise term n_a^t to the estimated Q-values $Q(s^t, a^t)$ and the subsequent selection of the action a_e^t with the highest value (equation 5).

$$a_e^t = argmax_a \left[Q(s^t, a^t) + n_a^t\right] \quad (5)$$

Figure 4 depicts very first results of our reinforcement-based control architecture, obtained during the online exploration phase in the power plant "Tiefstack" Hamburg. The performance of the RL-system is reflected by the obtained reinforcement. The defined reinforcement function, described in section 2.1, rewards both low air consumptions and low NO_x emissions. As can be seen, our reinforcement-control-system is able to reduce the NO_x concentrations in the range of higher coal multitude compared to the control scheme applied up to now (figure 4, top). The previous control system defined the amount of inlet air by a characteristic curve depending on the inlet coal amount, where a fixed symmetrical distribution between the left and right burner was applied. Furthermore, also the air consumption could be reduced by the RL-system (figure 4, bottom). This is a very promising result.

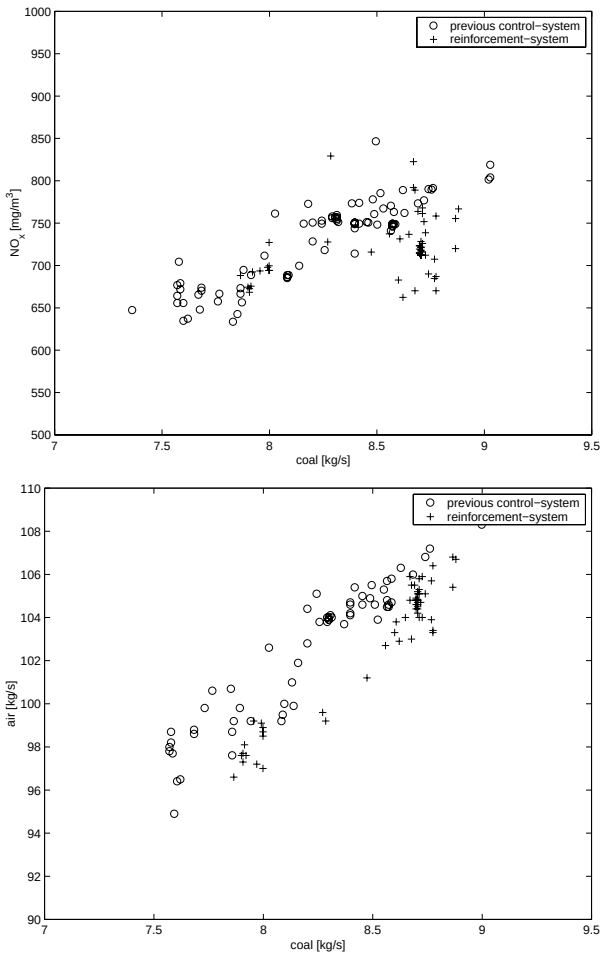

Figure 4: **top**: Comparison of nitrogen oxides emission for previously applied control scheme (circles) and RL-control system (crosses) over 6 days. **bottom**: Comparison of air consumption for previously applied control scheme (circles) and RL-control system (crosses) over 6 days.

4 Conclusions and Outlook

In this paper we presented a control scheme based on reinforcement-learning for a hard-coal combustion process in a power plant. The presented first approach uses a neural vector quantization technique to map the high-dimensional input space onto a discrete state space, whereupon a classical Q-learning approach estimates the values of the actions. This reinforcement-based control approach requires neither detailed informations about the plant nor an exact process model. The ability to explore autonomously the consequences of its own actions guarantees on one hand the plasticity under changing system properties, but on the other hand the system is forced to perform non-optimal actions. Nevertheless, the presented first results are very encouraging and demonstrate, that reinforcement-learning is also capable to cope with complex industrial control problems.

Further work should address a decomposition of the

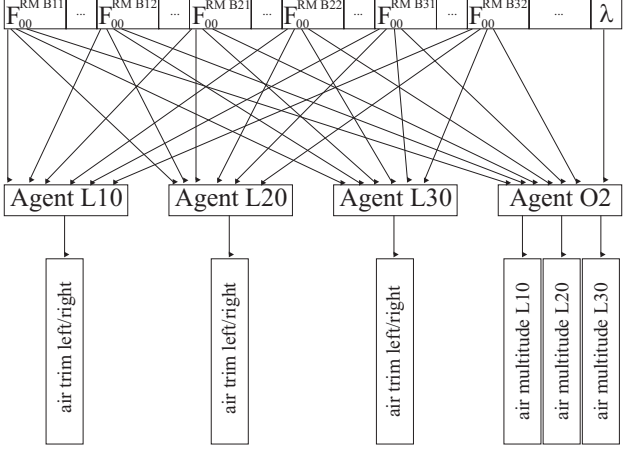

Figure 5: Decomposition of the control task into 4 agents with their inputs and their corresponding controls. Each agent observes the relevant part of the situation space and has access to an assigned subset of controls.

presented monolithic architecture into several agents controlling only a subset of all control variables. This will simplify the function-approximation-task, since each agent has to observe only a small part of the present high-dimensional input space. A first suggestion for this decomposition is shown in figure 5. Thus, AGENTL10, AGENTL20, and AGENTL30, control the air distribution at each case of one burner level. AGENTO2 controls the total amount of air consumption for each burner level. Unlike the monolithic approach with $3^6 = 729$ different actions the multi-agent-approach requires only $3+3+3+27 = 36$ actions! Hence, the number of possible control actions can be reduced substantially.

References

[Baldini et al., 1999] Baldini, G., Bittanti, S., Marco, A., Longhi, F., Poncia, G., Prandoni, W., and Vettorelo, D. (1999). A Dynamic Model of Moving Flames for the Analysis and Control of Combustion Instabilities. In *Proceedings of European Control Conference99, Karlsruhe, Germany*, page 585. VDI/VDE Gesellschaft Mess- und Automatisierungstechnik (GMA).

[Gross et al., 1998] Gross, H.-M., Stephan, V., and Krabbes, M. (1998). A Neural Field Approach to Topological Reinforcement Learning in Continuous Action Spaces. In *Proc. of WCCI-IJCNN'98, Anchorage*, pages 1992–1997. IEEE Press.

[Martinetz and Schulten, 1991] Martinetz, T. and Schulten, K. (1991). A "neural gas" network learns topologies. In Kohonen, T., Mäkisara, K., Simula, O., and Kangas, J., editors, *Artificial Neural Networks*, pages 397–402. Elsevier Amsterdam.

[Sutton, 1988] Sutton, R. (1988). Learning to predict by the methods of temporal differences. *Machine Learning*, 3:9–44.

[Watkins and Dayan, 1992] Watkins, C. and Dayan, P. (1992). Q-learning. *Machine Learning 8, 1992*, pages 279–292.