Proc. IEEE-INNS-ENNS Int. Joint Conference on Neural Networks (IJCNN 2000), Como (Italy), 2000, pp.VI 217-222

A Reinforcement Learning based Neural Multi-Agent-System for Control of a Combustion Process [1]

V. Stephan†, K. Debes†, H.-M. Gross†, 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

Abstract

In this paper, we present a control scheme based on reinforcement-learning for a industrial hard-coal combustion process in a power plant. To comply with the great demands on environmental protection, the plant operator is interested in a minimization of the nitrogen oxides emission, while other process parameters have to be kept within predefined limits. To cope with both the tremendous action and situation space of the power plant, we present a multiagent-reinforcement-system consisting of 4 agents, which are realized by relatively simple neural function approximators. We demonstrate, that our multiagent-system was able to significantly reduce the overall air consumption of the real combustion process of the power plant.

Keywords: Reinforcement-Learning, combustion process, visual flame observation

1 Introduction

Since the immediate objective of a power plant is the production of energy, the plant operator is trying to maximize the efficiency factor. Simultaneously, 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 [8]. The agent can use a RL-algorithm such as SUTTON's $TD(\lambda)$ algorithm [8], or WATKINS' Q-learning algorithm [9] to improve the long-term estimate of the value function associated with the current state and the selected action. However, in systems with 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 *neural 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.

2 State of the Art

To the best of our knowledge, this paper, presents for the first time, a reinforcement-learning based approach to control the combustion process of an industrial power plant. Traditional control approaches 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. For the solution of these problems a number of very different basic approaches were suggested and tested very intensively over the last years.

Large-scale combustion power plants are monitored by process control systems to solve the problems of visualization and alarm indications. The status quo in control for these processes is still the application of

[1] This work is supported by the German BMBF (grant number 032 6843 B)

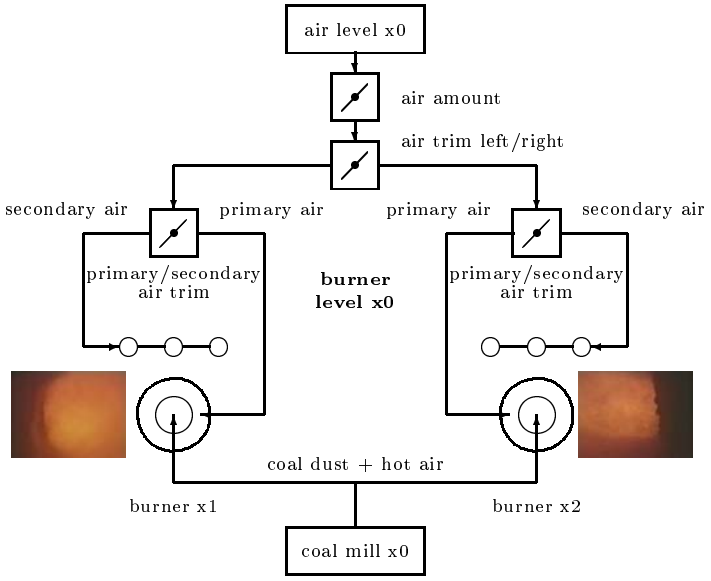

Fig. 1: Schematic view of the combustion chamber with coal and air supply for one out of three burner-levels.

standardized control-components. Additionally, there is a set of strategies leading to better results, but the main problem is the inflexibility of such solutions under varying working conditions.
Alternative approaches to control such complex combustion processes make extensive use of process models. Due to incomplete mathematical models and large computational demand, this way does not seem practicable [6, 10]. The second group of solutions for optimization are the knowledge based systems [3, 1]. But the key problem, the complex and difficult creation of a system-dependent knowledge base, limits both flexibility and portability of these solutions. For this reason, the authors suggest for the case of controlling a combustion process a control scheme, that requires no a priori knowledge and no mathematical model of the combustion process. The proposed Reinforcement-learning (RL) based control scheme allows an autonomous exploration of the state space of the combustion process, while predefined quality factors established by the plant operator have to be optimized.

3 The plant

The plant we used for our experiments 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 (10, 20, 30) and has a maximal output of 252 MW (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 may be shifted to benefit one burner. Unfortunately, the exact amount of inlet coal can not be measured for each burner separately. The plant operator has given us direct access to the following controls:

control	meaning
primary air trim at levels 10, 20, 30	distribution of the air amount between the left and right burner on the specified level
primary/secondary air trim at burners 11, 12, 21, 22, 31 and 32	distribution between primary and secondary air amount at the specified burner
air amount at levels 10, 20, 30	overall air amount on the specified level

Please bear in mind, that these 12 controls (see also figure 1) only influence the air amount and the distribution of air between these 6 burners, but neither amount nor distribution of inlet hard-coal! To reduce this immense action space, we use relative instead of absolute controls. That means, we define only three actions for each control: increase by 1%, remain unchanged or decrease by 1% (the use of absolute controls with only 10 quantization steps for each control would lead to an overall action space of 10^{12} different actions!!) Despite the usage of relative controls, our system has to cope with an enormous action space of $3^{12} = 531441$ actions.

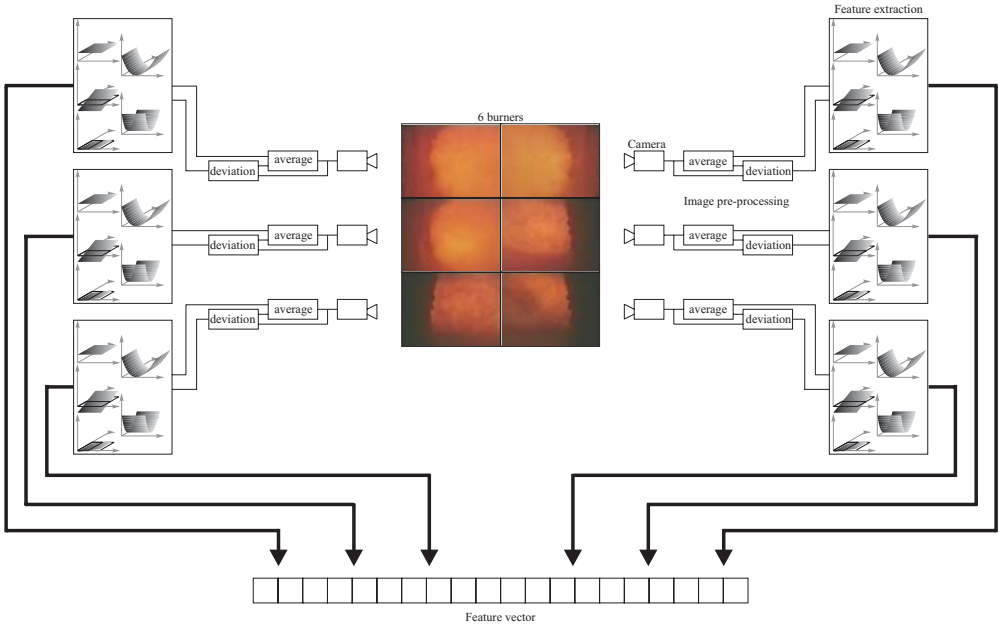

Fig. 2: Schematic view of the extraction of visual features describing the combustion process very closely. The obtained camera images are first filtered in the time domain, and thereafter we calculate fitvalues to describe the global shape of the flame.

After this description of the available controls, we will describe now, what kind of information we use to select appropriate controls to guide the process. Since all process data are measured outside the combustion chamber, for instance, in the waste gas, there is no direct information available about the combustion process itself. But to meet the requirements, we need exactly that direct information, for instance, about the distribution of coal between the burners of a level, the temperature and shape of the flame, etc. Therefore we observe each of the 6 flames by a special color camera system, provided by the Orfeus Combustion Engineering GmbH and use these data to control the process. Figure 2 depicts the extraction of visual features describing the combustion process.

4 Architecture

As mentioned in section 1, our architecture has to select that action for each process situation, which is promising the highest future reward. However, the key problem is the tremendous action space in combination with the very large input space. A monolithic architecture to estimate all action-outcomes in all situations is neither practicable nor explorable in finite time and therefore not practicable for our control task.

4.1 Problem decomposition

Consequently, we designed several agents, each observing only a relevant subset of the situation space and using only a subset of the available controls. Figure 3 (left) depicts the decomposition into 4 agents with their inputs and their corresponding controls. 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. The dimensions of the 4 agents are the following

	AgentL10	AgentL20	AgentL30	AgentO2
Inputs	3	3	3	7
NG-neurons	20	20	20	50
actions	27	27	27	27

The introduction of a scheduling of the 4 agents at this point is very important, since the reinforcement-approach assumes, that each agent is able to directly observe the consequences of its own actions. If two agents would perform their actions together, the consequences of their actions (e.g. NO_x concentration) would interfere and the resulting cross-talk between the agents would prevent to correct acquisition of the real outcomes of the respective actions. Hence, in this first approach, we defined, that all 4 agents operate sequentially in time intervals of 10 minutes.

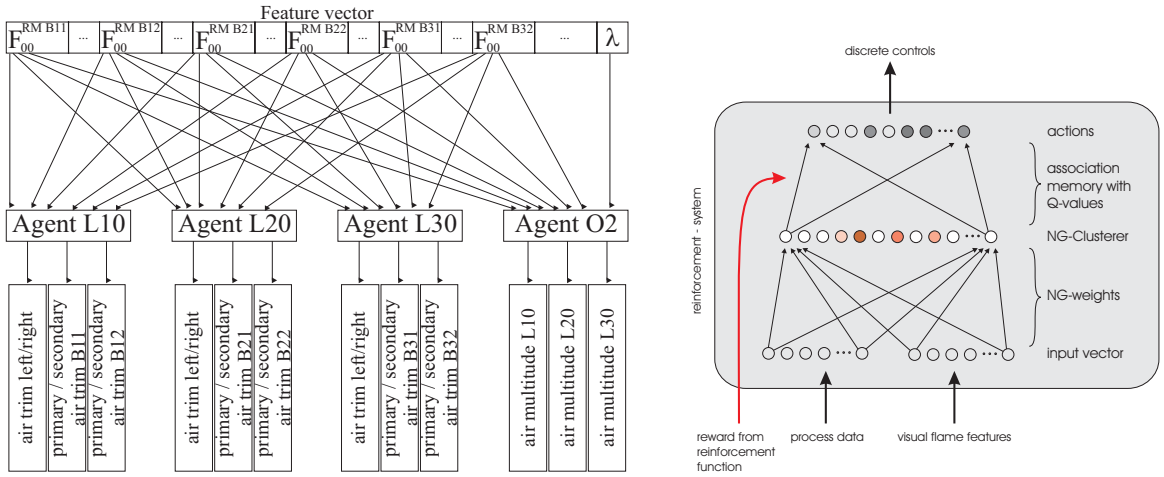

Fig. 3: (Left) Decomposition of the control task into 4 agents with their inputs and their corresponding controls. Each agent observes the relevant part of the situation space and has access to an assigned subset of controls. (Right) neural control architecture for reinforcement-learning of a single agent. The input-vector consists of visual 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 assigned actions are estimated.

4.2 Neural function approximator

Each of these 4 agents contains a neural function approximator. In this paper, we present a very first and simple approach to this state-action function approximator that combines a neural vector quantization technique (Neural Gas [7]) for optimal clustering of the high-dimensional, continuous input space [4] (equation 1) with a subsequent associative memory, to estimate the values of the assigned actions (see figure 3).
Equation 1 shows the neural-gas weight $\underline{w}_k(t)$ updating rule for the neuron k, where $\eta^{NG}(t)$ is a learning rate, $i(k)$ is the index of neuron k in the list sorted by distance to the input $\underline{x}(t)$ and $h(t)$ is the learning radius.

$$\Delta\underline{w}_k(t) \quad = \quad \eta^{NG}(t) \cdot e^{\frac{i(k)}{h(t)}} \cdot \left[\underline{x}(t) - \underline{w}_k(t)\right] \tag{1}$$

For action-value approximation Q for state s^t and action a^t, we utilize the Q-learning [9] variant of reinforcement-learning (equation 2).

$$\Delta Q(s^t, a^t) \quad = \quad \eta\Big\{r^t + \gamma V(s^{t+1}) - Q(s^t, a^t)\Big\} \qquad \text{with} \tag{2}$$

$$V(s^{t+1}) \quad = \quad \max_a Q(s^{t+1}, a^{t+1}) \tag{3}$$

For our experiments, we use a discount factor for the value of the subsequent state of $\gamma = 0.5$, and a Q-learning-rate of $\eta = 0.2$. The reinforcement r is the result of an agent-specific reinforcement function: Agents AGENTL10, AGENTL20 and AGENTL30 are rewarded, if the NO_x or the O_2 concentrations decrease and punished, if these concentrations increase (equation 4). The reinforcement depends on the O_2 concentration, since these agents can only change the distribution of the air, and a reduction of unused oxygen implies, that this redistribution caused a more complete combustion of the coal. Agent AGENTO2 is also rewarded, if the NO_x concentration or the total amount of used air decreases (equation 5). Any violation of various thresholds for process data, that are defined by safety precautions of the plant, result in a very strong punishment (equations 4 and 5). 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 $K_{NO_x} = K_\lambda = 0.5$.

$$r^{AgentLXX} = \begin{cases} -10.0 & : \ any\ threshold\ violated \\ K_{NO_x} \cdot \Delta NO_x + K_{O_2} \cdot \Delta O_2 & : \ else \end{cases} \tag{4}$$

$$r^{AgentO2} = \begin{cases} -10.0 & : \ any\ threshold\ violated \\ K_{NO_x} \cdot \Delta NO_x + K_{air} \cdot \Delta Air & : \ else \end{cases} \tag{5}$$

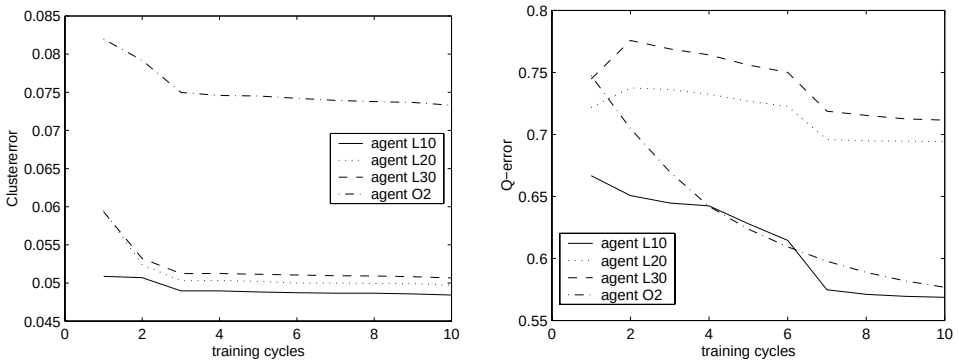

Fig. 4: Development of the cluster errors (left) and the Q–prediction-errors over 10 training cycles on passed process data of the 4 agents. For details see text.

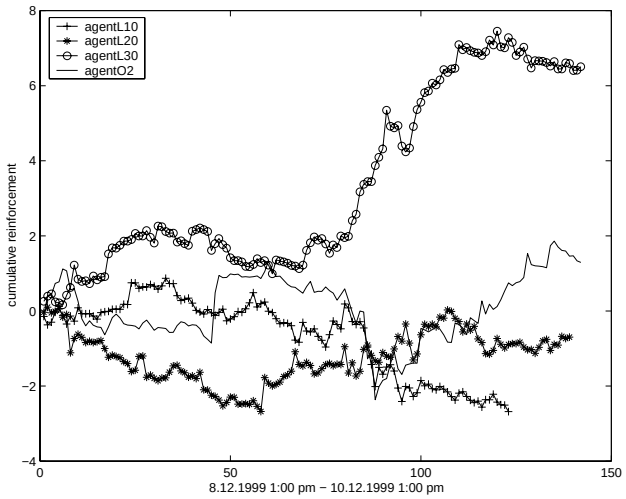

Fig. 5: Development of the cumulative online-reinforcements of the 4 agents during 2 days after pre-training with a relative high exploration performance. As can be seen, despite the still present exploration performance, especially *AgentL30* and *AgentO2* mostly receive positive reinforcements. This is plausible, since the upper burner level has the strongest influence on the waste gas concentrations, as a consequence of the flow characteristics inside the combustion chamber.

5 Results

To reduce the exploration time for the plant, we pre-trained our multiagent-approach on past process data. This is a kind of supervised reinforcement learning. Figure 4 shows the development of the cluster-errors (left) and of the Q-prediction-error (right). The decreasing cluster error documents the adaptation of the neural gas towards the distribution of process situations in the input space. The Q-error shows the convergence of our function approximator minimizing the Q-prediction error. After pre-training, we installed the neural multiagent-reinforcement-system on the plant. The exploration behavior of our system was not frozen at this time, instead we introduced a noise term on all estimated Q-values, which decreases over time to a fixed minimum greater than zero. Thus, we guarantee a fading exploration behavior in favor of the exploitational behavior, whereby always a certain exploration performance remains. Figure 5 depicts the evolving cumulative reinforcements of the 4 agents after the pre-training of the networks. In Figure 6 a comparison of the conventional control scheme with fixed air distributions used up to now and our multiagent-reinforcement-system is given. As can be seen, the amount of used air could be reduced significantly by the reinforcement-system (left). In contrast, the NO_x waste-gas concentration remained at the same level in these first investigations, whereby we have to remark, that the potential for NO_x reduction vanishes with increasing load factors of the power plant.

6 Conclusions and Outlook

In this paper we presented a reinforcement-based multiagent approach to control a complex industrial combustion process. To cope with both the tremendous action and situation space of the power plant, we decomposed the complex system into several agents. The proposed multiagent-reinforcement-system consists of 4 agents, which are realized by relatively simple neural function approximators. Neural function approximators are very useful, because they can generalize the expected return of state-action pairs the

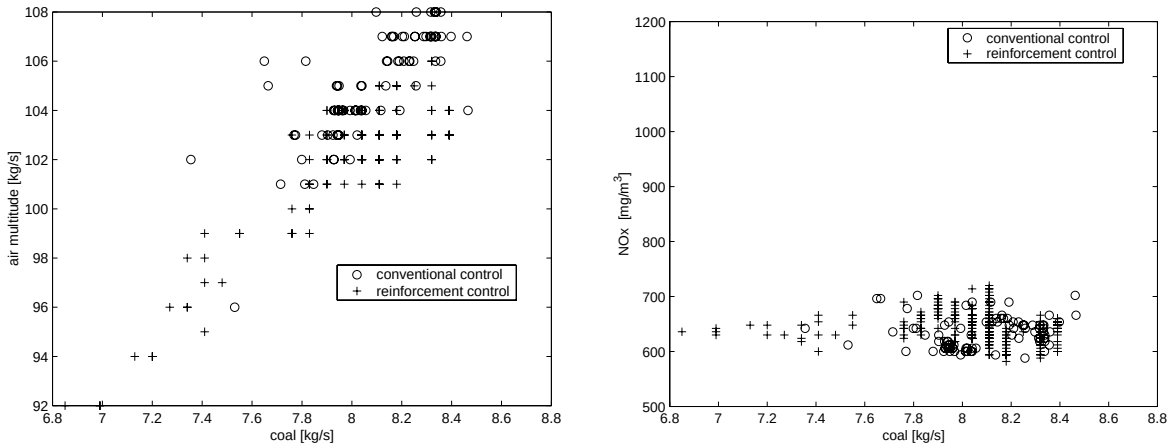

Fig. 6: Comparison of the conventional and our reinforcement-based control scheme by means of consumed air (left) and NO_x concentrations in the waste gas for a time period of about 2 days. At this time, the power plant worked with a load of about 90%.

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.
Future work should address the development of more powerful function approximators than our very simple approach utilizes, because this kind of network tries to approximate the probability density distribution of the input data in the feature space. This data driven statistical learning seems to be insufficient. For this reason, incremental neural networks are very promising alternatives, for example, the Growing Neural Gas Approach of Fritzke [2] or the life-long learning approach of Hamker [5]. For a faster learning, we also plan to investigate function approximators on the basis of the Adaptive Resonance Theory, e.g., the Fuzzy-Art approach. In this context, the stability-plasticity-dilemma has to be addressed, since changing coal qualities, wear and tear, etc. result in time varying process properties and a powerful system has to consider and solve these problems for the use in an industrial process. Additionally, we have to think the sequential scheduling regime of the agents over.
Nevertheless, the first results are very promising, but the application of RL-methods to this pretentious control problem is a great challenge.

References

[1] P. Eklund and F. Klawonn. Neural Fuzzy Logic Programming. *IEEE Trans. on Neural Networks*, 3(5), 1992.

[2] B. Fritzke. A Growing Neural Gas Network Learns Topologies. *Advances in Neural Information Processing Systems 7, MIT-Press, Cambridge MA*, 1995.

[3] S. Gehlen, M. Hormel, and J. Kopecz. Einsatz neuronaler Netze zur Kontrolle komplexer industrieller Prozesse. *Automatisierungstechnik*, 2, 1995.

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

[5] F. Hamker and H.-M. Gross. A lifelong learning approach for incremental neural networks. In *Proc. of EM-CSR'98, Vienna*, pages 599–604, 1998.

[6] H. Maier. *Experimentelle Untersuchungen der Kohlenstaubverbrennung unter Berueecksichtigung der Brennstoff-faufbreitung.* PhD thesis, Universitaet Stuttgart, Fakultaet Energietechnik, 1997.

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

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

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

[10] S. Wirtz. *Mathematische Modellierung der Kohlenstaubverbrennung.* PhD thesis, Ruhr-Universitaet Bochum, Fakultaet fuer Maschinenbau, 1989.