Adv Syst Sci Appl 2017; **3**:58–66
Published online at http://ijassa.ipu.ru/ojs/ijassa/article/view/499

Regression Tree Control of Multidimensional Static Object

Ekaterina Mangalova[1]*

[1] *System Analysis and Control Department, Siberian State Aerospace University, Krasnoyarsk, Russia*

Abstract: In this paper the control problem of a static system under incomplete information is discussed. A nonparametric control algorithm based on Classification and Regression Tree is proposed and evaluated for different task formulations: controlled inputs – one-dimensional output, controlled and observed uncontrolled inputs – one-dimensional output, controlled and observed uncontrolled inputs – multi-dimensional outputs.

Keywords: control, regression tree, machine learning, optimization

1. INTRODUCTION

The modeling of technical systems and processes is the essentially important tool used in engineering to design, improve, optimize and control systems. Using simulation modeling is generally cheaper, safer and faster than conducting real-world experiments. This allows to use modeling for different alternatives analyses.

There are two different ways to create simulation models:

- Manually built models. A specialist builds the simulation model manually based on his theoretical knowledge and technical experience. This method is the most useful and imminent for systems that do not yet exist.
- Statistical modeling. Statistical modeling is based on generating models by observations. This method can be useful for complicated systems when input-output relationships are not evident or are unknown, internals are not essential for the technical problem. Statistical models are used to detect a deviation of the normal behavior of a system and to control.

Model Predictive Control is a family of controllers in which there is a direct use of an explicit and separately identifiable model. Control algorithms based on Model Predictive Control concept have found wide acceptance in practical applications and have been studied by researchers. In [1] one of the first successful applications of Model Predictive Heuristic Control is described. There are two commonly used and well studied approaches to Model Predictive Control: based on Artificial Neural Networks and based on Fuzzy Logic. The use of artificial neural networks in model based control, both as process models and as controllers, is investigated by D. C. Psichogios and L. H. Ungar [2]. A. Draeger, S. Engell, H. Ranke [3] implemented a feed-forward neural network as the nonlinear prediction model in an extended DMC-algorithm to control. H. Sarimveis and G. Bafas [4] introduceed the method based on a dynamic fuzzy model of the process to be controlled, which is used for predicting the future behavior of the output variables. In the paper [5] J. M da Costa Sousa and U. Kaymak investigated the use of fuzzy decision making in model predictive control.

*Corresponding author: e.s.mangalova@hotmail.com

In contrast to neural network and fuzzy logic approaches, in [6] a control algorithm based on the Nadaraya-Watson estimator [7] (as a predictive model) and their sequence is proposed. However, there are major problems with this approach in the case of a multidimensional control task. On the one hand, it is connected with observations distribution in a high-dimensional feature space (especially in the case of small number of observations). On the other hand, the Nadaraya-Watson estimator has the high computational complexity. The bigger feature space dimension, the harder to optimize the vector of bandwidths. Decision trees are used for solving such regression tasks [8]. To avoid these problems it is suggested to use a decision tree instead of the Nadaraya-Watson estimator. In this paper an approach based on Classification and Regression Tree (CaRT) is proposed to solve control tasks.

The paper is organized as follows: in the second section, statements of the modeling and control problems are introduced; in the third section, the nonparametric modeling and control algorithm based on the Nadaraya-Watson estimator is presented; the fourth section is devoted to Classification and Regression Tree (CaRT) and the control algorithm based on CaRT with its variants for different control task statements.

2. MODELING AND CONTROL TASKS

The block scheme of the control process is shown in Figure 1. The following designations are taken: $\bar{x}$ is the vector of output variables, $\bar{x}^*$ is the vector of the desired output $\bar{x}$ values, $\bar{u}$ is the vector of controlled inputs, $\bar{v}$ is the vector of observed uncontrolled inputs, ξ is the unobserved input (noise).

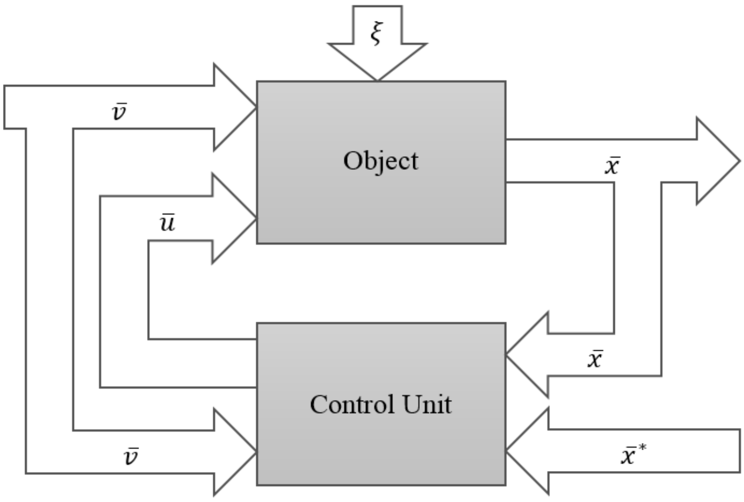

Fig.1 The block scheme of the control process

It is evident from Figure 1 that the output variables $\bar{x}$ depend on the inputs $\bar{u}$, $\bar{v}$, ξ. The control task is to build a control unit which generates $\bar{u}$ such that $E\left(\bar{x}, \bar{x}^*\right)$ is minimized, where E is some error measurement.

The modeling (regression) and control tasks are adjacent. If a system reaction to an input (prediction) is known, it is possible to get the desired outputs by identifying the controlled inputs [9].

The forward model (regression model) $\hat{x}\left(\bar{u}, \bar{v}\right)$ is fit using the training set $\left(\bar{u}_i, \bar{v}_i, \bar{x}_i, i = 1, ..., n\right)$, where n is the observations number (Figure 2).

Using the forward model $\hat{x}\left(\bar{u}, \bar{v}\right)$ the reactions to a given input can be predicted. For control task solving, it is necessary to invert the model $\hat{x}\left(\bar{u}, \bar{v}\right)$. It means that we should find controlled inputs $\bar{u}^*$ that could result in the desired outputs $\bar{x}^*$ in the case of estimated

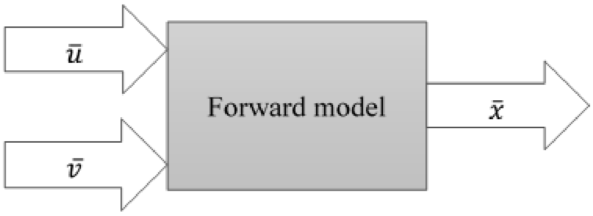

Fig.2 Forward model scheme (regression task)

uncontrolled inputs $\bar{v}^{predicted}$ (Figure 3). In contrast to the forward model, the backward model $\hat{u}\left(\bar{x}^{*}, \bar{v}^{predicted}\right)$ is fit using the training set $(\bar{u}_i, \bar{v}_i, \bar{x}_i, i = 1, ..., n)$ such that the outputs are given, uncontrolled inputs are predicted and we need to determine an appropriate controlled input.

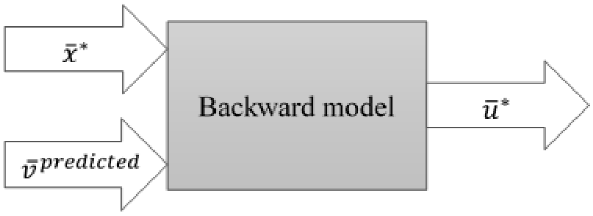

Fig.3 Backward model scheme (control task)

3. NADARAYA-WATSON ESTIMATOR APPROACH

The Nadaraya-Watson estimator approach was proposed to solve control tasks in such forward-backward models statement [6].

3.1. Modeling

The Nadaraya-Watson estimator has been widely applied for the nonparametric regression, using the weighted average observations output in the neighborhood around $(\bar{u}, \bar{v})$:

$$\hat{x}^k\left(\bar{u}, \bar{v}\right) = \frac{\sum_{i=1}^{n} x_i \prod_{j=1}^{m_u} K\left(u^j, u_i^j, c_u^j\right) \prod_{j=1}^{m_v} K\left(v^j, v_i^j, c_v^j\right)}{\sum_{i=1}^{n} \prod_{j=1}^{m_u} K\left(u^j, u_i^j, c_u^j\right) \prod_{j=1}^{m_v} K\left(v^j, v_i^j, c_v^j\right)},\ k = 1, ..., m_x, \tag{3.1}$$

where the kernel function K is a non-negative function that integrates to one and has zero mean, $\bar{c}_u$, $\bar{c}_v$ are the vectors of bandwidths, m_x is the number of output variables.

To identify the forward models $\hat{x}^k\left(\bar{u}, \bar{v}\right)$, the bandwidths $\bar{c}_u$, $\bar{c}_v$ should be optimized according to the selected accuracy measurement.

3.2. Control

Backward models of Nadaraya-Watson estimators have the following form:

$$\hat{u}^m\left(u^1,...,u^{m-1},\bar{x},\bar{v}\right)=$$
$$\frac{\sum_{i=1}^{n}u_i^m\prod_{j=1}^{m-1}K\left(u^j,u_i^j,c_u^j\right)\prod_{j=1}^{m_v}K\left(v^j,v_i^j,c_v^j\right)\prod_{j=1}^{m_x}K\left(x^{j*},x_i^j,c_x^j\right)}{\sum_{i=1}^{n}\prod_{j=1}^{m-1}K\left(u^j,u_i^j,c_u^j\right)\prod_{j=1}^{m_v}K\left(v^j,v_i^j,c_v^j\right)\prod_{j=1}^{m_x}K\left(x^{j*},x_i^j,c_x^j\right)},$$
$$m=1,...,m_u, \quad (3.2)$$

the bandwidths $\bar{c}_u$, $\bar{c}_v$ in the equations (3.1) and (3.2) are not identical.

There are some important limitations on the Nadaraya-Watson algorithm implementation:

- *Ranking input features.* The controlled input variables $\bar{u}$ should be ranked by their importance. The process (3.2) starts from the most important controlled input feature and continues with less and less important ones.
- *Bijective relationship.* The forward models (the true relationships between input features and output features) should be bijective functions. Suppose there are two significantly different controlled inputs $\bar{u}_1^*$ and $\bar{u}_2^*$ that produce the desired output. The result of the algorithm is a controlled input values between $\bar{u}_1^*$ and $\bar{u}_2^*$ (this control does not lead to the desired output) instead of choosing one of them.
- *Curse of dimensionality.* It is connected with observations distribution in a high dimensional feature space (especially in the case of a small number of observations). Suppose, there are $n = 1000$ points uniformly distributed over the ten dimensional unit cube $[0,1]^{10}$. An average over the neighborhood of diameter 0.25 (in each coordinate) results in the volume of $0.25^{10} \approx 0.00000095$ for the corresponding ten-dimensional cube. Hence, the expected number of observations in this cube will be 0.00095 and any averaging can not be expected. If we fix the count $k = 1$ of observations over which to average, the diameter of the typical neighborhood will be larger than 0.5. It means that the average is calculated over at least one-half of the range along each coordinate [10].

4. DECISION TREE APPROACH

Decision trees are commonly used for solving regression tasks in multidimensional cases and allow preventing the Nadaraya-Watson algorithm limitations described in section 3.

4.1. Modeling

Decision tree is a piecewise constant nonparametric model. The most popular decision tree model is Classification and Regression Tree (CaRT) [8]. CaRT is a binary tree where each root node represents an input variable u^{j_r} and a split point b_r. Let us aggregate controlled and observed uncontrolled inputs to make the description easier: $\bar{u}=\{u^1,...,u^{m_u},v^1,...,v^{m_v}\}$. Each leaf node contains values of the output variable x which is used to make a prediction.

CaRT fitting involves input variables and split points selection until a suitable tree is constructed.

Input variables and split points are chosen using a greedy algorithm to minimize the:

$$\min_j \min_b \left(\sum_{i:u_i^j<b} L\left(x_i,\hat{x}^-\left(\bar{u}_i,j,b\right)\right)+\sum_{i:u_i^j\geqslant b} L\left(x_i,\hat{x}^+\left(\bar{u}_i,j,b\right)\right)\right) \quad (4.3)$$

where $\hat{x}^-\left(\bar{u}_i,j,b\right)$ or $\hat{x}^+\left(\bar{u}_i,j,b\right)$ is the prediction in the point $\bar{u}_i$ based on the subspaces after splitting:

$$\hat{x}^{-}\left(\bar{u}_i, j, b\right) = \frac{\sum_{i:u_i^j<b} x_i}{\sum_{i:u_i^j<b} 1}, \quad \hat{x}^{+}\left(\bar{u}_i, j, b\right) = \frac{\sum_{i:u_i^j\geqslant b} x_i}{\sum_{i:u_i^j\geqslant b} 1}. \tag{4.4}$$

The tree construction ends using a predefined stopping criterion, such as the minimum number of observations assigned to each leaf node of the tree or the maximum depth of the tree.

Each node in the tree corresponds to a rectangular region of the predictor space S^k, a subset of the observations lying in the region S^k, a constant $\hat{x}_k$ which is the average response of the observations in k-th rectangular region. Thus, the binary tree model can be formalized as follows:

$$\hat{x}\left(\bar{u}\right) = \left\{\hat{x}_k : \bar{u} \in S^k, k = 1, 2, ..., K\right\}. \tag{4.5}$$

4.2. Control

Control tasks can be solved using the regression trees in different formulations depending on the number of input and output values, presence of observed uncontrolled input variables. Consider the basic formulation of the problem.

4.2.1. Controlled inputs - one-dimensional output. Assume there are only controlled variables $\bar{u}$ and one output variable x. Decision tree is fit using the training dataset containing simultaneous observations of $\bar{u}$ and x.

The control algorithm:

1. Set the control target x^*.
2. Search for such leaf node that

$$k^* = \arg\min_k D^1\left(\hat{x}_k, x^*\right), \tag{4.6}$$

   where D^1 is a one-dimensional distance measurement. Call k^*-th node "target node".
3. Find control $\bar{u}^*$ contained in the region S^{k^*}. There are different ways to choose this control:
   - The center of the rectangle guaranties the most accurate forward model prediction (decision tree prediction).
   - Points on the rectangle boundary allow to explore object and collect more information.
   - Also the distance between the previous control and the current control can be minimized to reduce interference in the system.

The experimental result. Suppose the object is represented by the equation: $x = \sin\left(2\pi u^1\right) + \sqrt{u^2} - u^3 + \xi$. This equation is used in the computational experiment to simulate a real process. Input variables u^1, u^2, u^3, u^4 are bounded by $[0, 1]$. The initial dataset contains 10 points distributed uniformly. Control is selected as the center of the target node. Figure 4 illustrates the desired outputs and the reactions to a generated control. Figure 5 shows the initial dataset and the observations obtained during the simulation process.

4.2.2. Controlled and observed uncontrolled inputs - one-dimensional output. Assume there are controlled and observed uncontrolled variables $\bar{u}$ and one output variable x. The decision tree is fit using the training dataset containing simultaneous observations of $\bar{u}$ and x.

The control algorithm:

1. Set the control target x^*.

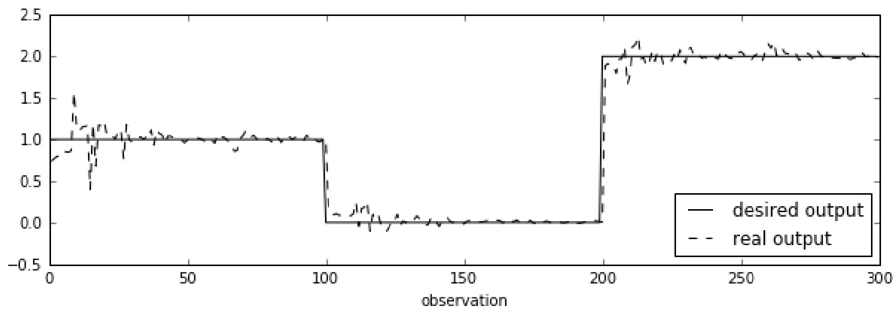

Fig.4 Simulation of a process. Desired outputs and reactions to a generated control

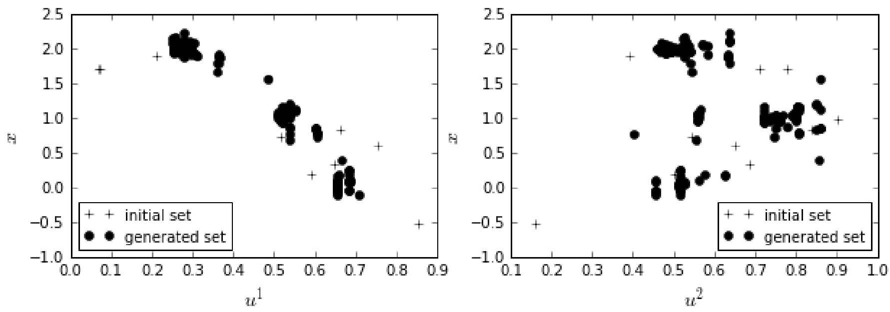

Fig.5 Simulation of a process. Initial dataset and generated dataset

2. Predict the uncontrolled input variables $u^{j,predicted}, j = m_u + 1, ..., m_u + m_v$.
3. Cut nodes which can not be achieved with the predicted uncontrolled input variables.
4. Search for a leaf node such that

$$k^* = \arg \min_{k, \bar{u}' \in S^k} D^1 (\hat{x}_k, x^*), \tag{4.7}$$

   where $\bar{u}'$ is the input values such that u'^j,$j = 1, ..., m_u$ can take any value and $u'^j = u^{j,predicted}$,$j = m_u + 1, ..., m_u + m_v$.
5. Find a control $\bar{u}^*$ contained in the region S^{k^*}.

The experimental result. Suppose now that u^2 is uncontrolled and changed as follows $u_i^2 = \sin(\pi i/25)$. The previous value of u_{i-1}^2 is used as a prediction for the next step $u^{2,prediction}$. Figure 6 illustrates the desired outputs and the reactions to a generated control. Figure 7 shows the initial dataset and the observations obtained during the simulation process.

4.3. Multi-dimensional output, controlled and observed uncontrolled inputs

Assume there are controlled and observed uncontrolled variables $\bar{u}$ and the vector of output variables $\bar{x}$. Decision trees are fit using the training dataset containing simultaneous observations of $\bar{u}$ and $\bar{x}$.

The control algorithm:

1. Set the control targets $\bar{x}^*$.
2. Predict the uncontrolled input variables $u^{j,predicted}, j = m_u + 1, ..., m_u + m_v$.
3. Find intersections of nodes which contain the predicted uncontrolled variables $S^k_{1,...,m_u}$, $k = 1, ..., K'$ and the output values associated with these intersections $\hat{x}^1_k$,..., $\hat{x}^{m_x}_k$.

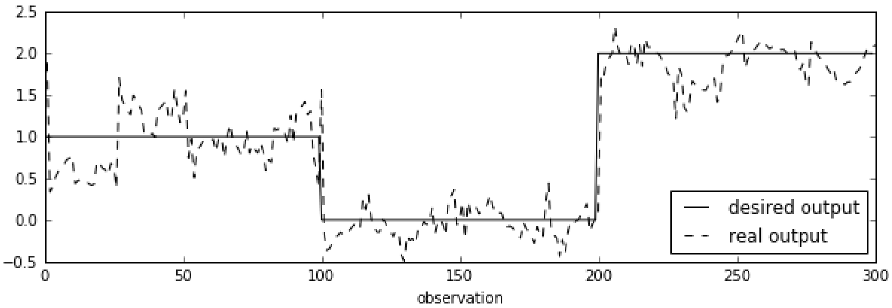

Fig.6 Simulation of a process with uncontrolled input. The desired outputs and the reactions to the generated control

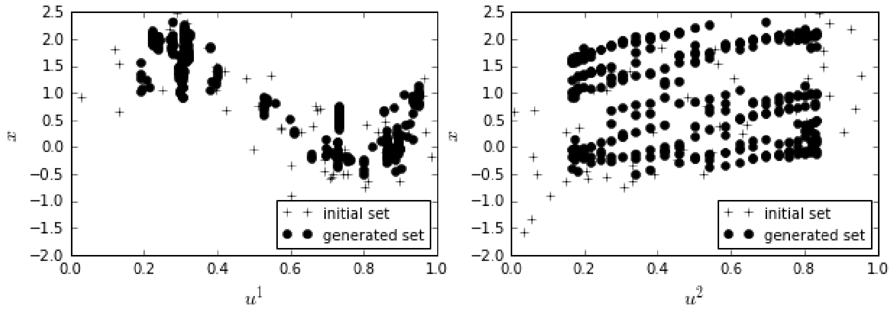

Fig.7 Simulation of a process with uncontrolled input. The initial dataset and the generated dataset

4. Search for a leaf node such that

$$k^* = \arg\min_k D^{m_x}\left(\hat{x}_k^1, ...\hat{x}_k^{m_x}, \bar{x}^*\right), \tag{4.8}$$

where D^{m_x} is a multi-dimensional distance measurement.
5. Find a control $\bar{u}^*$ contained in the region $S_{1,...,m_u}^{k^*}$.

The experimental result. Suppose now there are two outputs $x^1 = \sin(2\pi u^1) + \sqrt{u^2} - u^3 + \xi$ and $x^2 = \sqrt{u^2} - u^3 + \xi$. Figure 8 illustrates the desired outputs and the reactions to a generated control. The desired outputs for x^1 and x^2 are the same. Figure 9 shows the initial dataset and the observations obtained during the simulation process.

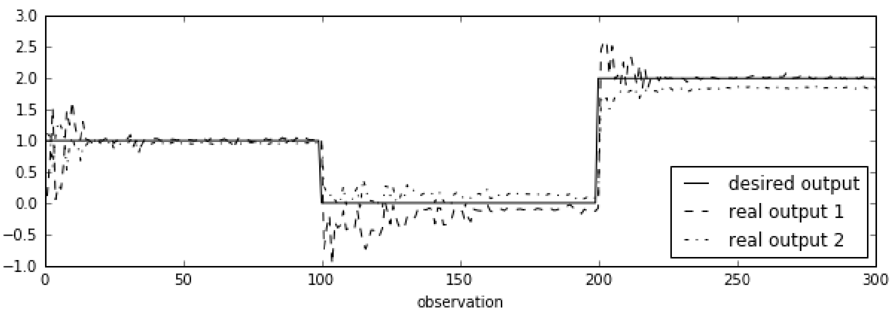

Fig.8 Simulation of a multidimensional process. The desired outputs and the reactions to a generated control

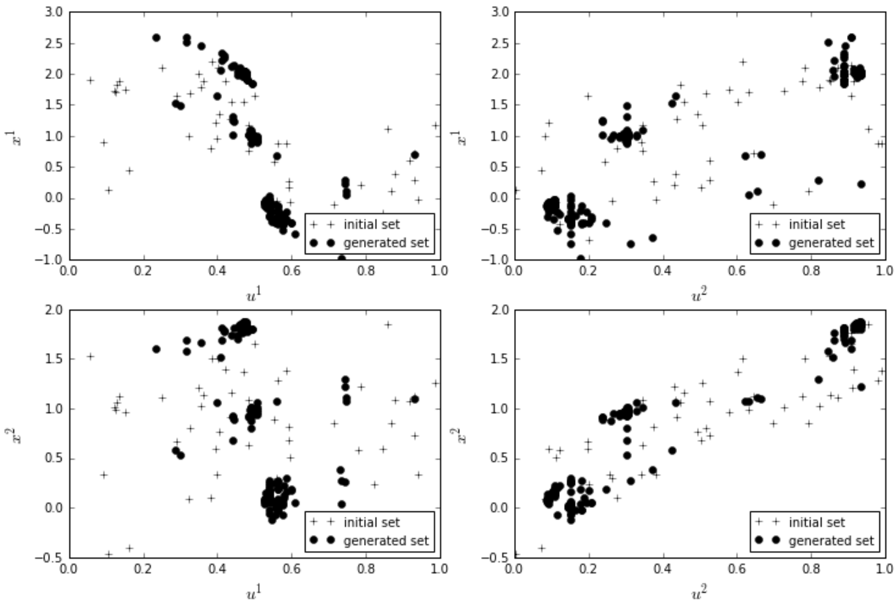

Fig.9 Simulation of a multidimensional process. The initial dataset and the generated dataset)

CONCLUSION

In this paper a control task of a multidimensional static system is discussed. The paper presents the advantages and disadvantages of the nonparametric algorithm based on the Nadaraya-Watson estimator and their sequence. In order to avoid the disadvantages of this nonparametric algorithm, the modification with piecewise constant approximation of the nonparametric estimator is put forward. In the proposed algorithm Classification and Regression trees are used as a predictive model instead of the Nadaraya-Watson estimator. The control algorithm based on CaRT is tested for different task formulations: controlled inputs and one-dimensional output, controlled and observed uncontrolled inputs and one-dimensional output, controlled and observed uncontrolled inputs and multi-dimensional outputs.

REFERENCES

1. Richalet J., Rault A., Testud J.L. & Papon J. (1978). Model predictive heuristic control: Applications to industrial processes, *Automatica*, **14** (5), 413–428.
2. Psichogios D. C. & Ungar L. H. (1991). Direct and indirect model based control using artificial neural networks, *Industrial & engineering chemistry research*, **30** (12), 2564–2573.
3. Draeger A., Engell S. & Ranke H. (1995). Model predictive control using neural networks, *IEEE Control Systems Magazine*, **15** (5), 61–66.
4. Sarimveis H. & Bafas G. (2003). Fuzzy model predictive control of non-linear processes using genetic algorithms, *Fuzzy sets and systems*, **139** (1), 59–80.
5. da Costa Sousa J.M. & Kaymak U. (2001). Model predictive control using fuzzy decision functions, *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, **31** (1), 54–65.
6. Medvedev A.V., & Raskina A.V. (2017). On the Nonparametric Identification and Dual Adaptive Control of Dynamic Processes, *Journal of Siberian Federal University. Mathematics & Physics*, **10** (1), 96–107.

7. Nadaraya E. A. (1964). On estimating regression, *Theory of Probability and Its Applications*, **9**, 141–142.
8. Breiman L., Friedman J., Stone C. J. & Olshen, R. A. (1984). *Classification and regression trees.* CRC press.
9. Kunh S. & Guhmann C. (2008) Modeling and Control with Local Linearizing Nadaraya Watson Regression, *arXiv:0809.3690*.
10. Härdle W. (1990) *Applied nonparametric regression.* Cambridge University Press.

