Synthesized optimal control based on machine learning

Askhat Diveev and Elizaveta Shmalko
Federal Research Center "Computer Science and Control" of the Russian academy of sciences
E-mail: aidiveev@mail.ru, e.shmalko@gmail.com

Abstract. The article discusses symbolic regression methods as a machine learning technology. The technique is tested on a complex problem of control systems synthesis. A new type of control based on changing the position of a stable equilibrium point is proposed. The implementation of such control requires the construction of a double feedback loop. The inner contour ensures the stability of the control object relative to some point in the state space. The outer contour provides optimal control of the stable equilibrium point position. To implement control, symbolic regression methods are used as machine learning technologies. It is shown that such a control is the least sensitive to external disturbances and model uncertainties.

1. Introduction

Machine learning is one of the areas of artificial intelligence associated with solving problems based on algorithms that can learn or gradually improve the performance of a given task. The term "machine learning" itself was coined in 1959 by Arthur Samuel, and defined it as the process by which computers are able to show behavior that was not explicitly programmed into them. Machine learning is based on the idea that computing systems can detect patterns and make decisions on their own.

Many people associate machine learning today with neural network technologies. This conception formed by the widespread use of neural networks [1–3] in machine learning, in contrast to other technologies, such as, for example, symbolic regression methods [4].

At its core, a neural network is a function with a specific structure and a large number of unknown parameters. Learning, or, more precisely, training, neural networks is finding such parameter values so that some quality functional has an optimal value. Such construction has its drawbacks. First, the choice of the type and structure of the neural network, determination of the size and number of layers an so on is carried out intuitively, at the discretion of the developer, based on his experience and knowledge. Another big drawback of neural networks is the lack of interpretability of the object model, since in fact the model in this approach is a black box, which gives few opportunities for its understanding and analysis [5]. Moreover, approaches based on deep neural networks often suffer from insufficient reproducibility, caused largely by non-determinism in the learning process [6]. And finally, an important limitation of neural networks is the large amount of data required to train them. In many problems, this amount of training data simply does not exist.

With this in mind, more and more work appears on the study of other artificial intelligence methods for machine learning. One of these areas is the methods of symbolic regression [7–9],...
or symbolic reasoning [5].

Symbolic regression methods allow to search for the desired mathematical expression in coded form. The problem of obtaining a mathematical expression arises in various situations: approximation of experimental data to determine a physical law or a trend model; efficiently analyze and predict variables or indicators based on previous observations; identification of a mathematical model of a process or a dynamic object; generalization of the control law based on the current state of the control object. Previously, computer search for mathematical expressions meant that a researcher first determined a mathematical expression accurate to parameters, and then the computer looked for the optimal values of these parameters in accordance with some given criterion. Now all analytical problems, where solutions must be obtained in the form of mathematical expressions, can be solved by computers. Nonlinear and differential equations, integrals, inverse functions and other problems can now be solved by numerical symbolic regression methods.

These tasks also include the task of control system synthesis. The challenge is to find a mathematical expression for the control function. In the general case, this control function must have a special property: when it is inserted into the differential equations of the mathematical model of the control object, then a special stable equilibrium point appears in the space of solutions of these differential equations. All solutions of differential equations from a certain region of initial conditions will tend to this singular point of attraction. It seems that neural networks can also be used for these purposes. But a neural network is a function of a given structure, in which it is necessary to find the optimal values of a large number of parameters. Unlike neural networks, symbolic regression methods allow finding the optimal structure of the desired function together with the optimal parameter values. In this sense, this approach is more flexible. Moreover, the received function is described in the way understandable for a researcher, not a black box as in neural networks, that is also very important. And we can assume that symbolic regression methods are a further development of neural networks, because any neural network can be encoded using symbolic regression.

In this article, we consider the problem of synthesizing a control system in which it is necessary to ensure the movement of the control object to a given state from a given range of initial values. The problem of control synthesis in the overwhelming case does not have analytical solutions, but recently machine learning methods have been effectively used to solve it numerically [10].

One of the features of the problem is that the control system in applied devices requires adjustment. There is a large gap between real control systems and analytical solutions for control systems. For the applied implementation of analytical solutions, it is necessary to construct additional stabilization systems, which significantly complicate the system as a whole, changing it. As a result, analytical trajectories are no longer more optimal.

The paper proposes a double synthesis approach. Initially, it is proposed to make the object stable. For this, machine learning technologies based on symbolic regression are used. After the first stage, we get an object that is stabilized relative to some point, the position of which can be changed. At the second stage, the problem of optimal movement of the equilibrium point is solved to achieve the optimal value of the quality indicator.

2. Statement of Problem

Given a model of control object

$$\dot{x} = f(x, u),$$

where $x \in \mathbb{R}^n$, $u \in U \subseteq \mathbb{R}^m$, $U$ is a compact set, $m \leq n$.

Given a domain of initial conditions

$$X_0 \subseteq \mathbb{R}^n.$$
Given the terminal position \( x_f \in \mathbb{R}^n \).

Given the integral
\[
J = \int_0^{t_f} f_0(x(t), u(t))dt,
\]
where
\[
t_f = \begin{cases} 
t, \text{ if } t < t^+ \text{ and } \|x_f - x(t)\| < \varepsilon, \\
t^+, \text{ otherwise} \end{cases}
\]
and \( t^+ \) and \( \varepsilon \) are given positive numbers \( f_0(x, u) : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^1 \).

It is necessary to find function \( u = h(x) \),

for any initial conditions from the domain (2) \( \forall x^0 \in X_0 \), \( x(t, x^0) \) is optimal onto criterion (4)
\[
\int_0^{t_f(x^0)} f_0(x(t, x^0), h(x(t, x^0)))dt = \min_{\forall u \in U} \left\{ \int_0^{t_f} f_0(x(t), u(t))dt \right\}
\]
C) The system (7) has s contraction property
\[
\|x(t, x^{0,\alpha}) - x(t, x^{0,\beta})\| \leq \|\bar{x}(t + \delta t, x^{0,\alpha}), x(t + \delta t, x^{0,\beta})\|,
\]
\( \forall x^{0,\alpha} \in X_0, \forall x^{0,\beta} \in X_0, \delta t > 0. \)

Property C determines real feasibility of the control system \( h(x) \) on a board of the control object.

To solve this problem numerical methods of symbolic regression are used. On the first stage the function is found in the form
\[
u = g(x^* - x),
\]
where \( x^* \) is a constant vector, \( x^* = [x^*_1 \ldots x^*_n]^T \). The function is such that if it is inserted in right side of differential equation (1) there is an a stable equilibrium point in the state space
\[
\tilde{x}(\tilde{x}^*) \in \mathbb{R}^n
\]
\[
f(\tilde{x}(\tilde{x}^*), g(\tilde{x} - x^*)) = 0,
\]
\[
det(\lambda E - A) = \lambda^n + b_{n-1}\lambda^{n-1} + \ldots + b_1 \lambda + b_0 = \prod_{j=1}^{n} (\lambda - \lambda_j) = 0,
\]
where
\[
\lambda_j = \alpha_j + i\beta_j, \quad j = 1, \ldots, n,
\]
\[
\alpha_j < 0, \quad j = 1, \ldots, n,
\]
\[
i = \sqrt{-1},
\]
\[
A = \frac{\partial f(\tilde{x}(x^*), g(x^* - \tilde{x}(x^*))))}{\partial x},
\]
E is $n \times n$ unit matrix.

In the second stage the synthesis problem with the functional (4) is solved. It is necessary to find the vector $x^*$ as a function of space vector

$$x^* = s(x),$$  \hfill (17)

where $s(x) : \mathbb{R}^n \rightarrow \mathbb{R}^n$.

As a result the initial problem (1)–(6) is solved

$$u = g(x^* - x) = g(s(x) - x) = h(x).$$  \hfill (18)

3. Symbolic Regression

In the considering problem it is necessary to find multidimensional functions with vector arguments $g(x^* - x)$ and $s(x)$. In both cases an approach based on machine learning of symbolic regression is used.

Symbolic regression can look for mathematical expression on the space of formulas. This space of formulas is defined as follows.

The set of elementary functions is set. These functions have one, two or three arguments.

$$F = \{a_1 = f_{1,1}(z), \ldots, a_w = f_{1,W}(z)a_{w+1} = f_{2,1}(z_1, z_2), \ldots, a_{W+V} = f_{2,V}(z_1, z_2), a_{W+V+1} = f_{3,1}(z_1, z_2, z_3), \ldots, a_{W+V+Q} = f_{3,Q}(z_1, z_2, z_3)\}. \hfill (19)$$

Here (19) the first index specifies on a number of arguments, and the second index is the function number. There are $W$ functions with one argument, $V$ functions with two arguments and $Q$ functions with three arguments.

The set of arguments is defined

$$F_0 = y_1 = f_{0,1}, \ldots, y_N = f_{0,N}. \hfill (20)$$

Arguments are considered as functions without arguments.

The space of formulas are compositions of some elementary functions from the set of functions (19) and some elements from the set of arguments

$$S_0 = s_1 \circ \ldots \circ s_d, \hfill (21)$$

where $s_i \in F \cup F_0$, $i = 1, \ldots, d$.

The sets (19) and (20) are called alphabet. Composition (21) is called word.

To write a word it is necessary to consider a rule of records.

Change elements in the word (21) onto functions with indexes without arguments

$$S = f_{i_1,j_1} \circ \ldots \circ f_{i_d,j_d}. \hfill (22)$$

For any position $k$ in the record (22) on the right from it must be

$$R = 1 + \sum_{i=1}^{k} (i_l - 1) \hfill (23)$$

elements from the set of arguments (20).

Symbolic regression codes words of formulas in the specific form, and it looks for optimal mathematical expression according to the given goal function.

There are more than a dozen symbolic regression methods methods: grammatical evolution, analytical programming, Cartesian genetic programming, inductive genetic programming, the network operator method, binary genetic programming, and others. Here we use one of them, namely the Cartesian genetic programming [11]. The Cartesian genetic programming was chosen among other methods because it does not change the length of a code after crossover operation, unlike the most popular genetic programming method.
4. Cartesian Genetic Programming

The Cartesian genetic programming codes every call of function as an integer vector with four components

\[ \mathbf{v} = [v_1 \ldots v_4]^T, \]

where \( v_1 \) is a number of element in the set of functions (19), \( v_2, v_3, v_4 \) are the numbers of elements from the set of arguments (20). If the function \( v_1 \) has arguments less than three, then others elements are not considered. Result of function computation is included in the set of arguments.

Consider an example of coding a mathematical expression by the Cartesian genetic programming. Let it be given a mathematical expression

\[ w = \exp(-c_1 x_1) \cos(c_2 x_2 + c_3). \]

The set of functions is

\[ F = \{ a_1 = f_{1,1}(z) = z, a_2 = f_{1,2}(z) = -z, a_3 = f_{1,3}(z) = \exp(z), a_4 = f_{1,4}(z) = \cos(z), f_2, a_5 = f_{2,1}(z_1, z_2) = z_1 + z_2, a_6 = f_{2,2}(z_1, z_2) = z_1 z_2 \} \]

The set of arguments is

\[ F_0 = \{ y_1 = x_1, y_2 = x_2, y_3 = c_1, y_4 = c_2, y_5 = c_3 \} \]

The list of function calls is

\[
\begin{align*}
  w_1 &= c_1 x_1 = f_{2,2}(c_1, x_1) \rightarrow a_6 \circ y_3 \circ y_1 \circ y_1 \rightarrow \mathbf{v}^1 = [6 \ 3 \ 1 \ 1]^T, \\
  w_2 &= -w_1 = f_{1,2}(w_1) \rightarrow a_2 \circ y_7 \rightarrow \mathbf{v}^2 = [2 \ 7 \ 2 \ 3]^T, \\
  w_3 &= \exp(w_2) = f_{1,4}(w_2) \rightarrow a_3 \circ y_8 \rightarrow \mathbf{v}^3 = [3 \ 8 \ 4 \ 5]^T, \\
  w_4 &= c_2 x_2 = f_{2,2}(c_2, x_2) \rightarrow a_6 \circ y_4 \circ y_2 \rightarrow \mathbf{v}^4 = [6 \ 2 \ 4 \ 6]^T, \\
  w_5 &= w_4 + c_3 = f_{2,1}(w_4, c_3) \rightarrow a_5 \circ y_{10} \circ y_5 \rightarrow \mathbf{v}^5 = [5 \ 10 \ 5 \ 7]^T, \\
  w_6 &= \cos(w_5) = f_{1,3}(w_5) \rightarrow a_3 \circ y_{11} \rightarrow \mathbf{v}^6 = [3 \ 11 \ 8 \ 9]^T, \\
  w_7 &= w_3 w_6 = f_{2,2}(w_3, w_6) \rightarrow a_6 \circ y_9 \circ y_{12} \rightarrow \mathbf{v}^7 = [6 \ 9 \ 12 \ 10]^T.
\end{align*}
\]

As the result the following Cartesian genetic programming code is received

\[ C = \begin{bmatrix} 6 & 2 & 3 & 6 & 5 & 3 & 6 \\ 3 & 7 & 8 & 2 & 10 & 11 & 9 \\ 1 & 2 & 4 & 4 & 5 & 8 & 12 \\ 1 & 3 & 5 & 6 & 7 & 9 & 7 \end{bmatrix}. \]

5. Computational example

As an example, consider the problem of optimal control for one mobile robot moving from some initial state to the terminal position in the complex environment with multiple state constraints. We have received the optimal control by two different approaches (the synthesized and the direct ones) and then we studied the behavior of the model of the object with the obtained control in the presence of inaccuracies in the model and in the initial conditions, which were added in the form of noise.

The mathematical model of a robot has the following form

\[
\begin{align*}
  \dot{x} &= 0.5(u_1 + u_2) \cos(\theta), \\
  \dot{y} &= 0.5(u_1 + u_2) \sin(\theta), \\
  \dot{\theta} &= 0.5(u_1 - u_2),
\end{align*}
\]
where \( \mathbf{x} = [x \ y \ \theta]^T \) is a vector of state space, \( \mathbf{u} = [u_1 \ u_2]^T \) is a control vector.

Restrictions on control are given
\[
u_i^- \leq u_i \leq u_i^+, \ i = 1, 2.
\] (31)

Initial conditions are set
\[
x(0) = x^0, \ y(0) = y^0, \ \theta(0) = \theta^0.
\] (32)

Terminal conditions are set
\[
x(t_f) = x^f, \ y(t_f) = y^f, \ \theta(t_f) = \theta^f.
\] (33)

A quality criterion is given
\[
J = t_f + \sum_{i=1}^{r} \left( a_1 \sqrt{(x_i^f - x(t_f))^2 + (y_i^f - y(t_f))^2 + (\theta_i^f - \theta(t_f))^2} + a_2 \int_{t_0}^{t_f} \vartheta(\rho_i^2 - (\bar{x}_i - x(t))^2) - (\bar{y}_i - y(t))^2) \, dt \right) \to \min,
\] (34)
where \((\bar{x}_i, \bar{y}_i)\) are coordinates of the given phase constraints, \(i = 1, \ldots, r\), \(r\) is a number of phase constraints, \(a_1, a_2\) are weight coefficients, \(\vartheta(a)\) is a step Heaviside function
\[
\vartheta(a) = \begin{cases} 
1, & \text{if } a > 0 \\
0, & \text{otherwise}
\end{cases}.
\]

Firstly, the stated problem was solved by the synthesized optimal control approach.

In the experiment the following values of parameters and constants were used: \(u_i^- = -10, \ u_i^+ = 10, \ i = 1, 2\), \(x(0) = x_0 = 0, \ y(0) = y_0 = 0, \ \theta(0) = \theta_0 = 0, \ x^f = 10, \ y^f = 10, \ \theta^f = 0, \ t^f = 2.5\ s, \ \varepsilon = 0.01, \ x^- = -1, \ y^- = -1, \ \theta^- = -1.57, \ x^+ = 12, \ y^+ = 12, \ \theta^+ = 1.57, \ a_1 = 1, \ a_2 = 3, \ \delta t = 0.625, \ K = 3\). Constraints were set as following \(\bar{x}_1 = 8, \ \bar{y}_1 = 8, \ r_1 = 1.5, \ \bar{x}_2 = 2, \ \bar{y}_2 = 2, \ r_2 = 1.5, \ \bar{x}_3 = 2.5, \ \bar{y}_3 = 7.5, \ r_3 = 2.5, \ \bar{x}_4 = 7.5, \ \bar{y}_4 = 2.5, \ r_4 = 2.5\).

The following optimal solution was found \(\mathbf{x}^{*1} = [11.8264 \ 4.3850 \ 0.0280]^T, \ \mathbf{x}^{*2} = [11.1745 \ 6.6811 \ 0.3700]^T, \ \mathbf{x}^{*3} = [12.8290 \ 0.1496]^T\).

Fig.1 shows the moving trajectory of the robot with the obtained synthesized optimal control. The functional was \(J = 2.4735\).

![Figure 1. Optimal trajectory of the synthesized approach](image)
For comparison, the same optimal control problem was also solved by the direct approach. The control was approximated by a piece-wise linear function in each interval \((k-1)\delta t \leq t \leq k\delta t\)

\[
\begin{align*}
\tilde{u}_1(t) & = q_j + \frac{q_{j+1} - q_j}{\delta t} (t - (j-1)\delta t), \\
\tilde{u}_1(t) & = q_{j+S} + \frac{q_{j+S+1} - q_{j+S}}{\delta t} (t - (j-1)\delta t),
\end{align*}
\]

where \(j = 1, \ldots, S\), \(\delta t\) is a value of time interval, \(S\) is a number of intervals \(S = \lfloor t^+ / \delta t \rfloor\).

In the experiment \(\delta t = 0.25\), \(S = 10\).

\[\text{Figure 2. Optimal trajectory of the direct approach}\]

Fig.2 shows the moving trajectory of the robot with the optimal control obtained by the direct approach. The functional was \(J = 1.9190\).

Then we studied the obtained controls in the presence of uncertainties in model and initial conditions. For this purpose perturbations were inserted into the mathematical model of the control object (30)

\[
\begin{align*}
\dot{x} & = 0.5(u_1 + u_2) \cos(\theta) + \beta \xi(t), \\
\dot{y} & = 0.5(u_1 + u_2) \sin(\theta) + \beta \xi(t), \\
\dot{\theta} & = 0.5(u_1 - u_2) + \beta \xi(t),
\end{align*}
\]

where \(\beta\) is a constant positive parameter, \(\xi\) is a random function that takes values from \(-1\) to \(1\).

The results of comparative computational experiments are presented in Table 1. Values of the functional were calculated as mean values of 10 experiments.

| \(\beta\) | Synthesized control | Direct control |
|----------|---------------------|--------------|
| 0        | 2.4865              | 2.5835       |
| 5        | 2.6742              | 6.7678       |
| 10       | 4.3449              | 7.4849       |

The influence of perturbations of the initial conditions was also studied by adding random noise

\[
\begin{align*}
x(0) & = x_0 + \gamma \xi(0), \\
y(0) & = y_0 + \gamma \xi(0), \\
\theta(0) & = \theta_0 + \gamma \xi(0),
\end{align*}
\]

where \(\gamma\) is a constant positive value.
Table 2. Values of the quality criterion in the presence of perturbations in initial conditions

| γ | Synthesized control | Direct control |
|---|---------------------|----------------|
| 0.1 | 3.5309 | 5.2930 |
| 0.5 | 5.7881 | 9.9566 |
| 1  | 7.1876 | 11.4535 |

The results are presented in Table 2.

The results of all experiments show an advantage of the synthesized optimal control under the direct approach despite the fact that without perturbations direct optimal control could provide slight better value of the functional (34) than synthesized control. But the synthesized control occurs less sensitive to the uncertainties whatever are they.

6. Conclusions

The paper presents an approach to control system synthesis based on machine learning. Machine learning is considered in terms of symbolic regression that allows not only tune the parameters of some given structure like in neural networks but also searches for the needed structure of functions.

References

[1] Sutton R S and Barto A G 1998 Reinforcement Learning: An Introduction MIT Press Cambridge MA
[2] Martens J 2010 Deep learning via hessian-free optimization Proc of the 27th International Conference on Machine Learning Haifa Israel
[3] Witten I H and Frank E 2005 Data mining: practical machine learning tools and techniques Morgan Kaufman - Elsevier, 2nd edition
[4] Derner E, Kubalík J and Babuška R 2018 Reinforcement Learning with Symbolic Input–Output Models Proceedings 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) Madrid Spain pp 3004—3009
[5] Jiayuan M, Chuang G, Pushmeet K, Joshua T and Jiajun W 2019 The Neuro-Symbolic Concept Learner: Interpreting Scenes, Words, and Sentences from Natural Supervision International Conference on Learning Representations
[6] Nagarajan P and Warnell G 2018 The impact of nondeterminism on reproducibility in deep reinforcement learning 2nd Reproducibility in Machine Learning Workshop at ICML Stockholm Sweden
[7] Duriez T, Brunton S L and Noack B R 2017 Machine Learning Control – Tuning Nonlinear Dynamics and Turbulence Springer Cham
[8] Freitas A 2002 Data Mining and Knowledge Discovery with Evolutionary Algorithms Springer-Verlag
[9] Winkler S 2008 Evolutionary System Identification - Modern Concepts and Practical Applications Reihe C - Technik und Naturwissenschaften Trauner Verlag, Linz
[10] Diveev A I 2020 Numerical Method of Synthesized Control for Solution of the Optimal Control Problem Intelligent Computing SAI 2020 Advances in Intelligent Systems and Computing, vol 1228. Springer Cham
[11] Miller J F 2011 Cartesian Genetic Programming Springer Berlin Heidelberg 342 p