Basic approaches to programming by demonstration for an anthropomorphic robot

V Alchakov, V Kramar* and A Larionenko

Sevastopol State University, 299053, Universitetskaya, 33, Sevastopol, Russia

*kramarv@mail.ru

Abstract. The article is devoted to the consideration of the issue of building intelligent control of manipulators of an anthropomorphic robot. The paper considers the approach of transmitting data when building management through training with a teacher using a copying suit. The structure of the knowledge base for building management has been considered. The proposed approach is an effective alternative to building control through solving complex systems of differential and finite-difference equations of a high order, which do not always take into account all the features of the functioning of manipulators in various environments.

1. Introduction

Mathematical models of the complex robotic systems are approximate and have significant limitations. Also, the existing mathematical models of the complex robotic systems are systems of nonlinear equations [1, 2], because of the complexity of solving them in real time is almost impossible. Also, often complex robotic systems are subject to disturbances that are difficult to foresee when creating a control system. Therefore, the design of modern complex robotic systems operating in conditions of uncertainty is currently primarily determined by the level of intelligence of management.

To make a decision in the uncertainty conditions, it is necessary to analyze large amounts of information with significant time limits for assessing the situation and making a decision. In such cases, decisions are most often made based on previous experience.

This ideology is inherent in one of the current areas of development of control algorithms - the development of control algorithms based on learning the system according to operator actions or Programming by Demonstration (PbD) [3].

The idea of PbD is to set problems solutions or to describe knowledge in general, not by explicitly programming every detail, but by demonstrating a specific set of examples of solving a problem or a set of problems. In fig. 1 shows a functional diagram of a PbD system developed for the Puma robot [3].

In general, the idea of PbD is that the operator, according to the task that the robotic complex needs to perform, performs specific manipulations with the help of the copying suit. The data taken from the copying suit is transmitted to the robotic complex controlling system. The data is analyzed and decomposed into elementary operations, as a result of which the robotic complex knowledge base is formed [4]. Based on the generated knowledge base, the control of the complex robotic system is
developed, either by interpolating the data or by regressing it, for example, using approaches based on
the use of neural networks.

![Functional Diagram](image1)

**Figure 1.** Example of the functional diagram for the PbD system for the Puma robot.

2. **Description of the control object**
The object of research is the anthropomorphic manipulators - the prototype of the underwater
manipulator under development (Fig. 2) [5].

![Manipulator Image](image2)

**Figure 2.** Anthropomorphic manipulator.

The manipulator is characterized by the fact that the weight of the manipulator is 14.8 kg; capture
weight — 1.2 kg; DOF manipulator — 12; manipulator length — 780 mm; rotational speed of articulated
drives — 170 degrees / s; rotational speed of gripping drives — 110 degrees / s; the accuracy of fixation
of the positions of the joints is 12 bits.

Manipulator and gripping modules serve as the main actuating mechanism of the complex robotic
systems and are of most interest from the point of view of building a PbD-based control system.

3. **Interaction with the robotic complex**
The transfer of control commands to the complex robotic system is carried out using specialized
software. In this case, several modes of operation are allowed — as mentioned above, a mode based on
the construction of a mathematical model of the complex (classical approach) and a control mode
based on the knowledge base of typical operating modes of the complex robotic system (PbD
approach). The implementation of the classical approach, in view of the above limitations, can cause certain difficulties.

The effectiveness of the implementation of the second approach depends on the completeness of the knowledge base, the so-called exemplary or typical movements of manipulators and seizures of the complex robotic system for solving a particular task. One of the options for filling the knowledge base is to record the movements that the operator performs with the help of the copying suit (Fig. 3).

*Figure 3.* The copying suit for the training of robotic complex.

The copying suit is a system of sensors, each of which corresponds to one of the executive devices of the manipulator system and the gripping of the robotic complex. Each sensor is equipped with an individual analog-to-digital converter (ADC), the signals from the ADC, in turn, form data packets that are transmitted to control programs and then transmitted to the complex robotic system actuators (Fig. 4).

The interaction with the control complex of the complex robotic system (Fig. 5) is carried out via the TCP network protocol. The use of this protocol involves the transmission of command codes and the receipt of response codes using a single-byte character encoding ASCII. The transfer of the numerical values of the parameters of the actuators is carried out by transmitting a sequence of characters corresponding to real numbers without a mantissa.
At the same time, the transmission of control signals from the copying suit to the control program occurs via the UDP protocol, which makes it possible to use the data stream by several client programs at the same time.

4. «Supervised learning» to build a manipulator control system.

The “Supervised learning” approach can be attributed to direct methods for constructing a control system for the complex robotic system’s manipulator. The idea of applying the “Supervised learning” is that the operator in a copying suit performs some sequence of exemplary actions that are digitized by the ADC of the suit sensors and transmitted via the network protocol to the knowledge base. Further, the control program selects from the knowledge base the necessary set of typical movements, forms on their basis a control program that transmits control commands via a network protocol to actuators of the robotic complex.

The enlarged block diagram of building a control system based on the “Supervised learning” approach is presented in Fig. 5.

![Figure 4. Data collection system from copying suit.](image1)

![Figure 5. Structural diagram of interaction with PbD-based robotic complex.](image2)

The implementation of the proposed approach is possible when building a client-server software solution, which implies the presence of a centralized control server with a knowledge base and a set of client programs, for interaction between the hardware parts of the suit and the complex robotic system. The client of the copying suit listens to a specialized port (port 10003 is used by default) using UDP protocol and the subsequent registration of digitized signals from the suit sensors in the knowledge base.

A data packet from a copying suit is an array of bytes, the structure of which is defined in the program settings file and may vary depending on the task solved. Each suit sensor corresponds to a section of the configuration file in XML format. An example of such a configuration block is:

```xml
<joint name="L.ElbowR" scaler="0.094"/>
```
The configuration block defines the attributes that describe the parameters of the sensors and their values. In the given example of the configuration block, the characteristics of the sensor of rotation of the elbow joint of the manipulator represented. Here, "L.ElbowR" (the name attribute in the configuration block) defines the identifier of the angle-of-rotation sensor of the left elbow joint of the manipulator.

Possible values of the name attribute are listed in Table 1.

Table 1. Name attribute values

| Name                  | Description                     |
|-----------------------|---------------------------------|
| R.ShoulderF L.ShoulderF | Movement of the shoulder joint forward |
| R.ShoulderS L.ShoulderS | Movement of the shoulder joint to the side |
| R.Elbow L.Elbow        | Elbow bend                      |
| R.ElbowR L.ElbowR      | Elbow rotation                   |
| R.WristF L.WristF      | Move the brush forward           |
| R.WristS L.WristS      | Brush to side                   |
| R.WristR L.WristR      | Brush rotation                   |
| R.Clavicle L.Clavicle  | Clavicle movement               |
| R.Finger.Thumb L.Finger.Thumb | Thumb fold                |
| R.Finger.ThumbS L.Finger.ThumbS | Rotation of the thumb to the side |
| R.Finger.Index L.Finger.Index | Index finger fold               |

For the rotation sensor of the elbow joint of the “L.ElbowR” manipulator, the “scaler” scaling factor is used, the value of which is 0.094. Next, in the configuration block, the <raw> nested tag is applied. This tag determines the position of the data from the specified sensor in the information frame and the type of this data. In this case, it is assumed that the information from the sensor begins with 258 bytes of the information frame and occupies 2 bytes (the word length for _int16).

Table of data types and the corresponding size in bytes are shown in Table 2.

Table 2. Table of data types and the corresponding size in bytes

| Data type | Number of bytes | Datatype | Number of bytes |
|-----------|-----------------|----------|-----------------|
| _byte     | 1               | _int16   | 2               |
| _int32    | 4               | _float   | 4               |

For the real value of the rotation angle of the control, the node uses the ratio

\[
\text{Rotation} = (\text{raw} + \text{offset}) \times \text{scaler}
\]

The parameters “offset” and “scaler” are described in the corresponding configuration block and determined at the device calibration stage. This ratio can be used to obtain all the information from the sensors of actuators listed in Table 1.

For the real value of the rotation angle of the control, the node uses the ratio

\[
\text{Rotation} = (\text{raw} + \text{offset}) \times \text{scaler}
\]
5. 4. The format of the knowledge base.

Typical robotic complex movements are stored in the form of data tables recorded with a resolution of 100 ms in CSV format (Fig. 6).

![Figure 6. Data from copying suit.](image)

For processing the data received from the copying suit, the authors have developed a specialized library for reading the knowledge base and the subsequent transfer of commands to the complex robotic system actuators. C# and the Net platform were chosen as the programming language. As an example, below are descriptions of the corresponding classes and methods.

To read the knowledge base, use the RecorderCommand class. The class header is

```csharp
public class RecorderCommand
{
    public TimeSpan Duration { get; set; }
    public List<CostumeJoint> Joints { get; set; }
}
```

The list of objects of the CostumeJoint type defines a set of angles of rotation of individual nodes of the actuators, and the Duration parameter defines the time interval for which the specified state should be reached. As a rule, the value of this parameter coincides with the discrete step used when recording the reference motion and is 100 ms.

Typical function for controlling the robot has a header is:

```csharp
public RobotAnswer ExecuteCommand(
    List<CostumeJoint> joints,
    TimeSpan time)
```

The joints parameter defines the list of actuators of the robotic complex actuators with final values for turn corners, and the time parameter defines the time for the execution of a given command. Data arrays are converted into an information frame and are transmitted via TCP to real-time complex robotic system devices.

The structure of the information frame, in general, has a syntax template like this

"robot: MOTORS:{Node names}:GO:{End positions}:{Time}"

An example of command with real parameters

robot: MOTORS:L_ShoulderF:GO:15.84:2.5
The specified command translates the complex robotic system manipulator forward with a final inclination angle of 15.84 degrees for an interval of 2.5 seconds.

6. Conclusion.
The article describes a general approach to building the control of an anthropomorphic manipulator based on the PbD approach and “Supervised learning”. The proposed approach is the basis for building a common method of developing a control system for the complex robotic system based on a learning algorithm. The data obtained in the process of learning a complex robotic system can be used to construct various models of the processes under study using regression methods. It also becomes possible to study the dynamics of transient processes of the complex robotic system, which can later be used to improve the quality of control algorithms and comply with the requirements for the characteristics of transients in control devices of the complex robotic system.

Acknowledgments
Studies were supported by the Ministry of Science and Higher Education of the Russian Federation (agreement № 14.578.21.0264, project RFMEFI57818X0264)

References
[1] Peters J, Lee D, Kober J, Nguyen-Tuong D, Bagnell J A, Schaal S 2016 Chapter 15. Robot learning, Handbook of robotics Ed B Siciliano, O Khatib (Springer-Verlag Berlin Heidelberg), p 2227
[2] Hersch M, Guenter F, Calinon S, Billard A 2009 IEEE Transactions on Robotics 24(6) pp1463 - 1467
[3] Munch S, Kreuziger J, Kaiser M, Dillmann R. 1994 Proceedings of the 24th International Symposium on Industrial Robots
[4] Billard A, Calinon S, Dillmann R, Schaal S 2008 Chapter 59: Robot Programming by Demonstration, Handbook of Robotics // Ed B Siciliano, O Khatib (Springer-Verlag Berlin Heidelberg) p.1591
[5] Zhidenko I G, Kutlubayev I M, Bogdanov A A, Sychkov V B 2017 Reshetnev’s readings 17 p.278-280