Article

Cooperative Control of Underwater Vehicle–Manipulator Systems Based on the SDC Method

Aleksey Kabanov *, Vadim Kramar, Ivan Lipko and Kirill Dementiev

Robotics and Intelligent Control Systems Laboratory, Sevastopol State University, 299053 Sevastopol, Russia; kramarv@mail.ru (V.K.); ivanlipko@yandex.ru (I.L.); iuts@sevsu.ru (K.D.)
* Correspondence: kabanov@sevsu.ru

Abstract: The paper considers the problem of cooperative control synthesis for a complex of N underwater vehicle–manipulator systems (UVMS) to perform the work of moving a cargo along a given trajectory. Here, we used the approach based on the representation of nonlinear dynamics models in the form of state space with state-dependent coefficients (SDC-form). That allowed us to apply methods of suboptimal control with feedback based on the state-dependent differential Riccati equation (SDDRE) solution at a finite time interval, providing the change in control intensity with the transient effect of the system matrices in SDC form. The paper reveals two approaches to system implementation: a general controller for the whole system and a set of N independent subcontrollers for UVMSs. The results of both approaches are similar; however, for the systems with a small number of manipulators, the common structure is recommended, and for the systems with a large number of manipulators, the approach with independent subcontrollers may be more acceptable. The proposed method of cooperative control was tested on the task of cooperative control for two UVMSs with six-link manipulators Orion 7R. The simulation results are presented in the article and show the effectiveness of the proposed method.

Keywords: underwater vehicle–manipulator system; cooperative control; state-dependent coefficient; SDDRE-method; mathematical modeling

1. Introduction

The cooperation of robotic systems has many applications in the industry that require coordinated control, such as repair and welding work, space transportation of objects, and others [1,2]. Most of them do not exclude the possibility of operating in an underwater environment, and this fact significantly complicates the implementation of control methods for each case.

Underwater vehicles are divided into two common type classes [1–5]: remotely operated underwater vehicles (ROVs) and unmanned underwater vehicles (UUVs) [6–10]. The first type of UV requires operator control from an on-surface sea vessel. The second type of UV is equipped with all necessary hardware and can move autonomously.

Autonomous underwater vehicles (AUVs) have been widely used in recent years as an alternative to extremely costly, time-consuming, and risky human underwater operations. Most of the tasks performed by these vehicles are for data-gathering applications. However, there is an increasing interest for their use in inspection, maintenance, and repair operations that require manipulation and interaction with objects in the underwater environment. Nowadays, this is mainly performed by remotely operated vehicles (ROVs) [11] and one of the strongest research trends regards the use of Underwater Vehicle–Manipulator Systems (UVMS) [12,13]. Underwater vehicle–manipulator systems (UVMS) can fulfill underwater sampling, grabbing, operation, and other tasks in addition to observation. It is widely used in marine scientific investigations, marine engineering, and other fields [14].

Most underwater manipulator tasks can be performed more efficiently when multiple manipulators are used cooperatively [15]. On the other hand, the tasks of coordinated

control of several underwater robots with manipulators are quite complex, with the main difficulties in implementation related to the communication limitations of the aquatic environment. The use of a single base robot carrier with multiple manipulators overcomes such limitations. This research focuses on the development of coordinated control algorithms for underwater robot manipulators that hold an object and move it along a specified trajectory.

It is important to consider the following question: why is using a robot with several manipulators with complex dynamics and limitations more effective than using a robot with one manipulator of higher power and carrying capacity?

No additional rationale is needed to move objects with large geometries relative to the size of the robot, as the interaction of multiple arms increases motion stability and payload capacity. The problem is to justify the use of two or more manipulators to move a concentrated mass (a load of insignificant size relative to the robot and manipulators). Some studies have shown that the use of robots with coordinated (cooperative) control increases the carrying capacity even for concentrated mass loads [15–18]. Therefore, the present study considers the problem of a coordinated operation of several underwater robotic arms when transporting an object with a concentrated mass in a limited working space. The control scheme implies different ways to distribute the load between the manipulators.

Because the underwater manipulator is a complex system, highly non-linear, and vulnerable to perturbations caused by underwater influences, reliable control strategies must be developed for the manipulator to obtain a reliable system.

There are several methods to synthesize control for a complex of robots; for example, for the sliding mode, [19,20] suggests using a control and position controller, and [21] proposes an additional verification of the results by Lyapunov functions. The PID control law is considered in [22]. However, these methods are closer to the classical techniques and are not always effective in these circumstances because the quality of manipulator control is easily affected in the underwater working environment. In [23], an intelligent controller based on a fuzzy neural network RBF (radial basis function) method is proposed for accurate and stable control of an underwater robot. Similar methods using neural networks and fuzzy logic are presented in [24–26]. For underwater robots, the technique of control synthesis using a nonlinear behavior prediction model is also acceptable [27].

As the basic method of synthesizing controls in this study, we propose to use the representation of nonlinear models of the manipulators' dynamics in the linear form of the state-space with state-dependent parameters (state-dependent coefficients techniques). That makes it possible to build simple (close to linear) trajectory controllers for manipulator grasping modules in the form of feedback using the so-called state-dependent differential Riccati equation (SDDRE) method.

Finally, it should be noted that the proposed control scheme can be implemented as a centralized control algorithm for the whole system, as well as in a decentralized form, because the algorithm involves the possibility of calculating control signals for each manipulator separately without exchanging data with each other based on information obtained exclusively from their sensors (for example, position and speed measurements). This significantly increases the reliability of the coordinated control system.

The rest of the paper is organized as follows. Section 2 discusses the features of the proposed methods to solve the problem statement, while Section 3 presents applications of the above methods, as well as a test case. In Section 4, the results and analysis are presented. The discussion is presented in Section 5. Finally, the conclusions are given in Section 6.

2. Materials and Methods

In the present paper, a cooperative control approach based on the solution of the state-dependent differential Riccati equation is proposed. It should be noted that it is not the only way to obtain control for a cooperative system, but it is optimal, which highlights its advantages among fuzzy logic methods or classical techniques [1,15,28].

Furthermore, the paper applies the well-known algorithm for solving the inverse kinematics problem of robot manipulators, which is based on the pseudo-inversion of the geometric Jacobian [29].

One of the main aspects of solving the problem of mathematical modeling and, in particular, the verification of control laws is the synthesis of the system dynamics model.

The majority of the known scientific research in the field of underwater vehicle modeling is based on the publications of T. Fossen [30]. They include models in 3, 4, and 6 degrees of freedom (DoF), description of external disturbances, control systems, and general calculation procedures. The mathematical model is a fundamental element in path planning, control, and navigation; therefore, this problem is extremely important and complicated in solving the problems identified in the article.

Besides identifying dynamic parameters in control synthesis, it is necessary to solve the problem of inverse dynamics, which allows obtaining the desired forces from the side of manipulating robots. The paper considers the approach of equal load distribution between manipulators, which is described in detail in [31,32].

A substantial contribution to the paper was made by [33], which describes in detail the dynamics and kinematics of the UVMS, the refinements of their design, as well as the specifics of the hydrodynamic parameters' calculation technique. It is necessary to note that the well-known parametrization approach, presented in [15,34] and which is applicable to the task set in the paper, was used for the composition of the dynamic parameters of the system.

Collectively, all the methods described above form the base for the development of a tracking control system for the cooperative complex of underwater manipulation robots during the transfer of an object with a concentrated mass, which is one of the essential goals of this work.

3. Theoretical Part

3.1. General Problem Statement

The proposed approach is based on the development of a nonlinear tracking optimal control scheme for the underwater robot–manipulator system UVMS and the object. We assume that all UVMS are informed about both the desired configuration of the object and the position of obstacles in the workspace. Therefore, the control task is to route the entire formation to the target configuration avoiding collisions with static obstacles in the workspace.

Let us consider N underwater robots with manipulators that have m degrees of freedom and rigidly capture an object of mass m_O in a limited workspace. Figure 1 shows an example of a cooperative system of two robots with manipulators. Assume that each manipulator is fully driven concerning the coordinate system associated with the working tool (gripper) and the manipulators are equipped with appropriate sensors to measure the position and speed of the manipulator links. In addition, the geometric parameters of both robots with manipulators and the object of manipulation are known.

Motion, trajectory, and coordinates of the manipulator joints are measured relative to the main reference frame $\{X_0, Y_0, Z_0\}$. The center of mass (CoM) coordinates of the manipulated object is denoted as x_O, y_O, z_O, the center coordinates of the manipulator working tools (grippers)—as x_i, y_i, z_i when $i = 1, 2, \ldots, N$. The load is considered as a concentrated mass; therefore, the trajectories of the load and the working bodies are similar, i.e., $(x_O, y_O, z_O) = (x_i, y_i, z_i)$. In the case of a load with a concentrated mass, therefore, the knowledge of the gripper orientation is not required. As a result, the minimum degree of freedom of the manipulators for designing the trajectory of planar motion is two, and for spatial motion it is three.

The task is to design distributed control protocols for each of the N manipulators to safely transport the object along the desired trajectory, considering the desired configuration for the manipulated object movement.
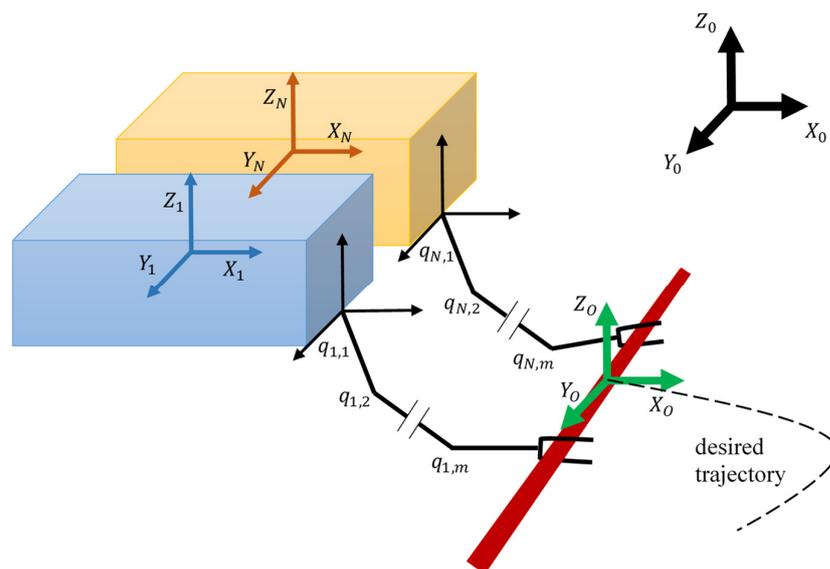
Figure 1. Schematic representation of two underwater robots with manipulators as a cooperative system.

3.2. Mathematical Model of the System

3.2.1. The Dynamics Model of the Manipulated Object

Without loss of generality, consider the following model of dynamics for an underwater object, which can be derived on the basis of the Newton–Euler formulations [31,33]:

$$or \dot{p}_O = J_O(\eta_{2,O})^{-1} v_O,$$
$$M_O(p_O)\dot{v}_O + C_O(v_O, p_O)v_O + D_O(v_O, p_O)v_O + g_O = -f_e, \tag{1}$$

where $p_O = \left[\eta_{1,O}^T, \eta_{2,O}^T\right]$, $\eta_{1,O}^T = [x_O, y_O, z_O]^T$ and $\eta_{2,O}^T = [\varphi_O, \theta_O, \psi_O]^T$, M_O—is a positively defined inertia matrix of the object, C_O—is the Coriolis matrix, and D_O models dissipative effects. The gravity and buoyancy vector is denoted as g_O. The matrix $J_O(\eta_{2,O})$—is the Jacobian of the object, which converts the derivative of the Euler angles to the angular velocity ω_O (angular velocity in the system coupled to the object); it can be defined as:

$$J_O(\eta_{2,O}) = \begin{pmatrix} I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & J_O^I(\eta_{2,O}) \end{pmatrix},$$
$$J_O^I(\eta_{2,O}) = \begin{pmatrix} 1 & 0 & -sin(\varphi_O) \\ 0 & cos(\varphi_O) & cos(\theta_O)sin(\varphi_O) \\ 0 & -sin(\varphi_O) & cos(\theta_O)cos(\varphi_O) \end{pmatrix}. \tag{2}$$

The external force $f_e \in \mathbb{R}^6$, acting on the object must be distributed among the manipulators $f_e = \sum_{i=1}^{N} f_{e,i}$, in order to isolate the effect $f_{e,i}$ from each manipulator.

3.2.2. Kinematic Model of a Robot with Manipulators

We define the coordinates of the working organ of each manipulator as $p_i = \left[\eta_{1,p_i}^T, \eta_{2,p_i}^T\right]$, where $\eta_{1,p_i}^T = [x_{p_i}, y_{p_i}, z_{p_i}]^T$ and $\eta_{2,p_i}^T = [\varphi_{p_i}, \theta_{p_i}, \psi_{p_i}]^T$ denote the position and orientation expressed in Euler angles relative to the inertial reference system $\{X_0, Y_0, Z_0\}$. Let us also determine the generalized velocities of the manipulator's working tool through $v_i = [\dot{\eta}_{1,i}^T, \omega_i^T]^T$, where $\dot{\eta}_{1,i}^T$—is the linear velocity, ω_i—is the angular velocity. Suppose $q_i \in \mathbb{R}^m$ describes the position and orientation of the carrier robot, $q_i \in \mathbb{R}^m$ is a vector of generalized coordinates of manipulator joints positions.

The position and orientation of the manipulator grip are given by the direct kinematics as follows:

$$p_i = F_i(q_i), \quad i = 1, 2 \dots, N. \tag{3}$$

For the differential kinematics of each manipulator, we have

$$v_i = J_i(q_i)\dot{q}_i, i = 1, 2 \dots, N, \\ \dot{q}_i = [\dot{q}_{Vi}^T, \dot{q}_{Mi}^T]^T, \tag{4}$$

where $\dot{q}_i$ is the velocity vector of the manipulator joint coordinates and $J_i(q_i)$—is the geometric Jacobian [28,31]. Note that the matrix $J_i(q_i)$ becomes singular at the points of kinematic specifics (singularities) defined by the set

$$S_i = \left\{ q_i \in R^{n_i} : det(J_i(q_i)J_i\left(q_i\right)^T) = 0 \right\}, \quad i = 1, 2 \dots, N. \tag{5}$$

The task of identifying the geometric Jacobian can be varied, but the well-known approach is based on vector multiplication and the application of transformation matrices [35].

The number of rows in the Jacobian matrix equals the number of states in the velocity vector, while the number of columns is dependent on the number of joints in the UVMS:

$$J_i(q_i) = \begin{bmatrix} \mathcal{J}_1(q_i) & \cdots & \mathcal{J}_{m+6}(q_i) \end{bmatrix}. \tag{6}$$

The components of angular and linear velocities are defined by the following expressions

$$\mathcal{J}_i(q) = \begin{bmatrix} \mathcal{J}_{v_i}(q) \\ \mathcal{J}_{\omega_i}(q) \end{bmatrix} = \begin{bmatrix} z_{i-1} \times (o_{m+6} - o_{i-1}) \\ z_{i-1} \end{bmatrix} \text{ if } i \text{ link is rotational,} \\ \mathcal{J}_i(q) = \begin{bmatrix} \mathcal{J}_{v_i}(q) \\ \mathcal{J}_{\omega_i}(q) \end{bmatrix} = \begin{bmatrix} z_{i-1} \\ 0_{3 \times n} \end{bmatrix} \text{ if } i \text{ link is prismatic,} \tag{7}$$

where o_{i-1}—the first three elements of the fourth column of the matrix $T_{i-1}(q)$, z_{i-1}—the first three elements of the third column of the matrix $T_{i-1}(q)$.

3.2.3. Dynamics Model of an Underwater Robot with Manipulators

The dynamics of underwater robot manipulators can be written as [31,33]:

$$M(q)\ddot{q} + C(\dot{q}, q)\dot{q} + D(\dot{q}, q)\dot{q} + g(q) = u + J^T(q)\overline{f}_e, \tag{8}$$

where $q = [q_1^T, \dots, q_N^T]^T \in R^{Nm}$, $u = [u_1^T, \dots, u_N^T]^T \in R^{Nm}$, $M = diag\{M_i\} \in R^{Nm \times Nm}$, $C = diag\{C_i\} \in R^{Nm \times Nm}$, $D = diag\{D_i\} \in R^{Nm \times Nm}$, $g = [g_1^T, \dots, g_N^T]^T \in R^{Nm}$, $\overline{f}_e = [f_{e,1}^T, f_{e,2}^T, \dots, f_{e,N}^T]$.

The trajectory of the manipulators is predetermined by the desired motion of the object; therefore, Equation (1) is a known equation, but the values of $f_{e,i}$, $i = 1, 2, \dots, N$ are undefined. To solve this problem and divide $f_{e,i}$ between the robots, one can apply the optimal load distribution method [32].

Assume that the robotic arms are rigidly gripping the object. Due to the rigid grip of the object, the following equations are satisfied:

$$p_i = p_O + \begin{bmatrix} {}^I R_O l_i \\ \alpha_i \end{bmatrix}, i = 1, 2, \dots, N, \tag{9}$$

where vectors $l_i = [l_{ix}, l_{iy}, l_{iz}]^T$ and $\alpha_i = [\alpha_{ix}, \alpha_{iy}, \alpha_{iz}]^T$ represent the constant relative position and orientation of the manipulator working tool relative to the object, expressed in the object frame and ${}^I R_O$ denotes the rotation matrix, which describes the orientation of the object, expressed in the inertial reference frame $\{I\}$. Therefore, using (9) for each manipulator,

it is possible to calculate the position of the object relative to the inertial reference frame $\{I\}$, because the geometric parameters of the object are considered to be known.

In addition, the kinetostatic duality along with the gripping rigidity assumes that the force f_e, acting on the object's center of mass and the generalized forces $f_{e,i}$, acting on the manipulators at the gripping points are related through [32].

$$
f_e = \begin{bmatrix} J_{O1}^T & J_{O2}^T & \cdots & J_{ON}^T \end{bmatrix} \begin{bmatrix} f_{e,1}^T & f_{e,2}^T & \cdots & f_{e,N}^T \end{bmatrix}^T = \\
= J_{O1}^T f_{e,1} + J_{O2}^T f_{e,2} + \ldots + J_{ON}^T f_{e,N} = \sum_{i=1}^{N} J_{Oi}^T f_{e,i}, \; J_{Oi} = J_{iO}^{-1}.
\tag{10}
$$

Now, consider the constants k_i, $i = 1, 2, \ldots, N$ satisfying the $k_1 + k_2 + \ldots + k_N = 1$ condition, which we enter as load distribution coefficients between manipulators.

Given $k_1 + k_2 + \ldots + k_N = 1$, the dynamics of the object (1) can be rewritten as (the dependence in the dynamics matrices on the number is omitted to simplify the notation):

$$
\sum_{i=1}^{N} k_i \{ M_O \dot{v}_O + C_O v_O + D_O v_O + g_O \} = - \sum_{i=1}^{N} J_{Oi}^T f_{e,i}.
\tag{11}
$$

From the last relation, we can calculate the forces for each of the manipulators

$$
f_{e,i} = -k_i J_{iO}^T f_e.
\tag{12}
$$

Once you have the load distribution scheme between the manipulators, you can develop a controller to control the entire system (8).

3.3. Control Algorithm

Figure 2 demonstrates a scheme of the UVMS tracking control, which clearly shows the separation of the applied methods described in the second section. Each block has its own task, and together they form a complete architecture of a cooperative control system. The used methodology, encapsulated in each block, is described in the second section of the paper. Therefore:
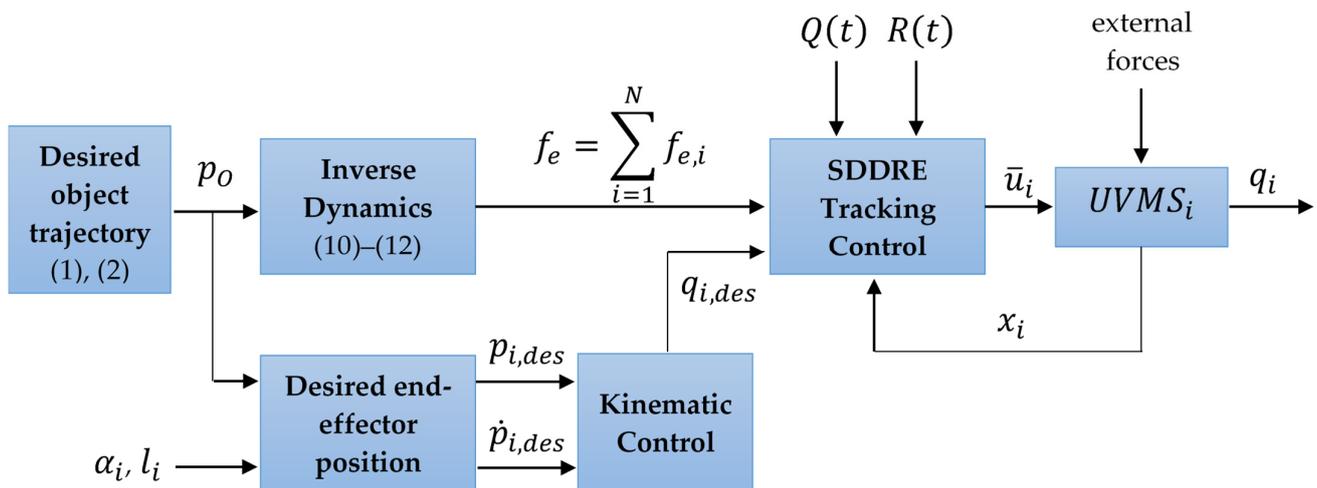

Figure 2. UVMS's tracking control diagram.

1. «Desired object trajectory» generates the desired trajectory of the object's center of mass p_O;
2. «Inverse Dynamics» calculates the desired forces $f_{e,i}$, applied to the manipulation robots;
3. «SDDRE Tracking Control» produces optimal control itself;
4. «$UVMS_i$» is a dynamics model of the underwater vehicle–manipulator system;

5. «Desired end-effector position» calculates the desired position of the UVMS tool according to a given trajectory and relative end-effector position;
6. «Kinematic Control» transposes from the end-effector's Cartesian coordinate system $p_{i,des}$ into the space of the manipulator's generalized coordinates $q_{i,des}$.

The control synthesis algorithm can be written as a set of the following steps:

1. Setting the desired trajectory of the object $p_O = \left[\eta_{1,O}^T, \eta_{2,O}^T \right]$;
2. The inverse dynamics problem is solved on the basis of the motion model of the object (1). Knowing the motion, determine which force $f_e = \sum\limits_{i=1}^{N} f_{e,i}$ should be applied to the object to provide the desired movement. After that, this force is distributed between manipulators in proportion according to (11) and (12);
3. Based on the desired motion of the object, the problem of determining the desired motion of the working tools of manipulators is solved through (9). It is necessary to set the parameters of the relative position of the object and the manipulator grips, i.e., to set vectors for each manipulator $l_i = [l_{ix}, l_{iy}, l_{iz}]^T$ and $\alpha_i = [\alpha_{ix}, \alpha_{iy}, \alpha_{iz}]^T$;
4. Knowing the required motions of the manipulator working tools, the problem of kinematic control is solved using the given motion of the working tool of the manipulator to determine the required value of the generalized coordinates of the robot–manipulator system, i.e., a kinematic regulator is constructed (Figure 3);
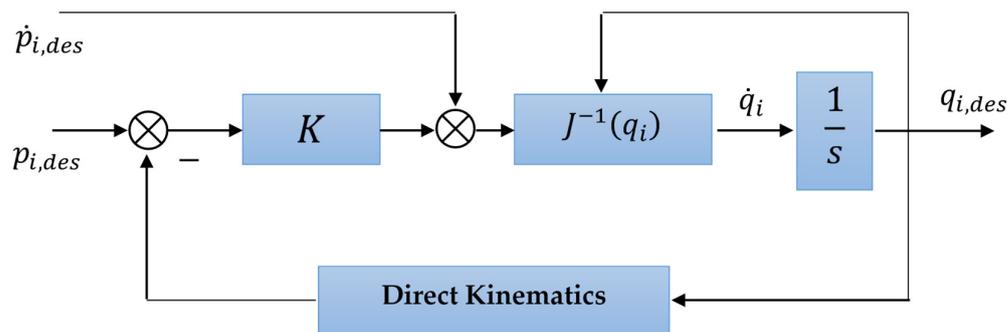

Figure 3. Schematic representation of the kinematic control.

5. Knowing the desired value of the generalized coordinates and the required force for each manipulator, the problem of synthesizing a tracking SDDRE controller is solved.

There are a number of methods to find the inverse Jacobian for solving the inverse kinematics problem; however, in this paper, we will consider the so-called damped least-squares (DLS) inverse method proposed in [29]:

$$J^{-1}(q_i) = J^T(q_i) \left[J(q_i) \cdot J^T(q_i) + k^2 I \right]^{-1}, \tag{13}$$

where k is a damping factor that renders the inversion better conditioned from a numerical viewpoint.

3.4. Synthesis of an SDDRE Tracking Controller

To synthesize the controller while solving the problem of tracking the desired value of the manipulator's generalized coordinates, we use the SDDRE method, which has proven itself in many applications, including the control of manipulators [36].

We denote $x = \begin{bmatrix} q^T & \dot{q}^T \end{bmatrix}^T$ and rewrite the nonlinear system (8) with new notations:

$$\dot{x} = \begin{pmatrix} \dot{q} \\ M(q)^{-1} \left(u + J^T(q)\overline{f}_e - C(\dot{q},q)\dot{q} - D(\dot{q},q)\dot{q} - g(q) \right) \end{pmatrix}. \tag{14}$$

The latest equation can be represented in the form of state-space equations with state-dependent coefficients (SDC form)

$$\frac{dx}{dt} = A(x,t)x + B(x,t)\overline{u}, x(t_0) = x_o,$$

$$A(x,t)x = \begin{pmatrix} 0_{Nm \times Nm} & I_{Nm \times Nm} \\ 0_{Nm \times Nm} & -M(x)^{-1}(C(x) + D(x)) \end{pmatrix}, B(x,t) = \begin{pmatrix} 0_{Nm \times Nm} \\ M(x)^{-1} \end{pmatrix}, \quad (15)$$

$$\overline{u} = u + u_{add}(x), u_{add} = J^T(x)\overline{f}_e - g(x).$$

Note that representation (15) is not unique, and there are different possibilities for the representation of the system matrices. The problem of finding the optimal representation for the system matrices is called the parameterization problem (or factorization), and there are different approaches to its solution. For example, it is possible to select the structure and parameters of the matrices based on preserving the controllability requirements of the system. The parametrization problem itself is rather complicated, so these aspects will not be addressed here.

Consequently, for the obtained system (15), we formulate the problem of the criterion minimization (here and further, we will omit the dependence on time where it is not essential for shortening the notation)

$$S_{t_0 t_f}(u) = \frac{1}{2}e^T\left(t_f\right)Fe\left(t_f\right) + \frac{1}{2}\int_{t_0}^{t_f}\left[e^T Q(x)e + u^T R(x)u\right]dt, \quad (16)$$

$$Q(x) \geq 0, R(x) > 0, F > 0.$$

where $e = x - x_{ref}$, x_{ref}—is the desired value of the system state vector (14), which is determined by the desired value of the generalized coordinates.

If the pair of matrices $(A(x,t), B(x,t))$ is pointwise controllable for all $x(t)$ for $t \in \left[t_0, t_f\right]$, the solution of problem (16) in the form of feedback can be obtained using the SDDRE method.

3.5. Centralized Control Scheme

The solution of problem (16) for feedback control can be written in the same way as for the linear case [34]:

$$u = -R^{-1}(x)B^T(x,t)(K(x)x + g(x,t)), \quad (17)$$

where $\dot{K}(x) = -A(x)^T K(x) - K(x)A(x) - Q(x) + K(x)B(x)R(x)^{-1}B^T(x)K(x), K\left(t_f\right) = F,$
$\dot{g} = -(A(x) - B(x)R(x)^{-1}B^T(x)K(x))^T g + Q(x)r, g\left(t_f\right) = Fx_{ref}\left(t_f\right).$

Two approaches to the implementation of this solution are possible: solving N-times SDDRE type control (17) for each manipulator independently and solving SDDRE control once for the whole system. These approaches were proposed in [17].

This numerical solution requires integration of Equation (17) in the opposite direction from time t_f to t, which is not realizable because the state x of the system (15) at future time moments is not defined and, therefore, the state-dependent parameters of matrices $A(x,t)$, $B(x,t)$ are also unknown.

A Lyapunov-based approach solves the problem in one forward round and was recommended for a tracking problem [34]. Let time tend to infinity, then K in infinity for (17) results as a constant gain, and the steady-state form of K is reformed to the state-dependent Riccati equation (SDRE). As a result, we have a Lyapunov equation with state and control nonlinearities with the final boundary condition at the right time. The solution of this equation results in the suboptimal gain as

$$K(x) = K_{ss}(x) + P^{-1}(x), \quad (18)$$

where $P(x) = E(x) + e^{A_{cl}(x)(t-t_f)}\left(P\left(t_f\right) - E(x)\right)e^{A_{cl}^T(x)(t-t_f)}$, A_{cl} is a closed-loop system matrix, K_{ss} is a negative root of SDRE.

The closed-form solution for g for (17) is presented as follows. This equation is rewritten as

$$\dot{g} = -A(x)g + Q(x)r. \tag{19}$$

and the solution has the form of

$$g = A^{-1}(x)\left(Q(x)r + e^{A(x)(t_f - t)}A(x)g\left(t_f\right) - Q(x)r\right). \tag{20}$$

The proposed closed-form answer works perfectly. Nevertheless, when the end of the trajectory is not zero, oscillation and an unacceptable jump occur. The cause of this problem is an open question.

3.6. Decentralized Control Scheme

In the first case, we need to calculate the control law for each UVMS

$$\bar{u}_i = -R_i^{-1}(x_i)B_i^T(x_i, t)K_i(x_i, t)x_i + u_{i,add}, \tag{21}$$

where $x_i(t)$—are the states of the i manipulator. Matrix $K_i(x_i, t)$ is obtained from the above algorithm, where instead of block matrices $A(x)$, $B(x)$, $R(x)$, and $Q(x)$, the corresponding matrices for each manipulator are used separately, and the model of manipulator dynamics in SDC form is

$$\frac{dx_i}{dt} = A_i(x_i, t)x + B_i(x_i, t)\bar{u}_i, \; x_i(t_0) = x_{i,o},$$
$$A_i(x_i, t)x_i = \begin{pmatrix} 0_{m\times m} & I_{m\times m} \\ 0_{m\times m} & -M_i^{-1}(x)(C_i(x) + D_i(x)) \end{pmatrix}, \; B_i(x_i, t) = \begin{pmatrix} 0_{m\times m} \\ M_i^{-1}(x_i) \end{pmatrix}, \tag{22}$$
$$\bar{u}_i = u_i + u_{i,add}(x_i), \; u_{i,add} = J_i^T(x_i)f_{i,e} - g_i(x_i), \; i = 1, 2, \ldots, N.$$

The matrix $K(x, t)$ of the controller (17) when solving the control problems separately for each vehicle manipulator by analogy with [37] can be composed of elements $K_i(x_i, t)$ in the form

$$K(x, t) = \begin{pmatrix} K_1(x_1, t) & 0 & 0 & 0 & 0 \\ 0 & K_2(x_2, t) & 0 & \vdots & 0 \\ 0 & 0 & \ddots & 0 & \vdots \\ \vdots & \vdots & 0 & K_{N-1}(x_{N-1}, t) & 0 \\ 0 & 0 & \cdots & 0 & K_N(x_N, t) \end{pmatrix}. \tag{23}$$

In this case, the system has N independent subcontrollers for each vehicle manipulator separately.

The second approach uses parameterized matrices for the entire system (15), requires solving SDDRE only once, and implements the control law (17). This approach provides optimal control of the complete system.

3.7. Test Case
3.7.1. Object Dynamics

In order to verify the method, consider a specific test case. As an object of manipulation, a body in the form of a cylindrical rod (Figure 4) of mass $m_O = 23$ kg, length $l_O = 2.5$ m, and radius $r_O = 0.05$ m is used.
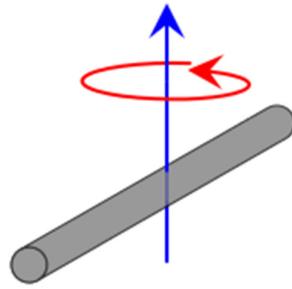
Figure 4. Cylindrical rod object.

The tensor of inertia of the handled object has the following definition

$$I_O = \begin{pmatrix} \frac{1}{12} m_O l_O^2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{1}{12} m_O l_O^2 \end{pmatrix}. \tag{24}$$

Since the working environment is underwater, it is necessary to take into account the hydrodynamic effects. The added masses of the rod, according to [5], are defined as

$$\begin{aligned} X_{\dot{u}} &= -0.1 m_O, \\ Y_{\dot{v}} &= -\pi \rho r_O^2 l_O, \\ Z_{\dot{w}} &= -\pi \rho r_O^2 l_O, \\ K_{\dot{p}} &= 0, \\ M_{\dot{q}} &= -\frac{1}{12} \pi \rho r_O^2 l_O^3, \\ N_{\dot{r}} &= -\frac{1}{12} \pi \rho r_O^2 l_O^3, \end{aligned} \tag{25}$$

where $\rho = 1025$ and represents the density of the fluid environment (salted water).

3.7.2. UVMSs Dynamics

As an example of the method's operability, consider the situation depicted in Figure 1. The work [38] presents the dynamic parameters of the Sf-30k underwater vehicle (Figure 5), which we use to simulate the mobile manipulator bases.
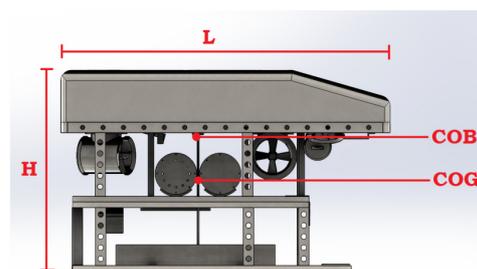
Figure 5. Underwater Robot Sf-30k.

Dynamics matrices for two identical Sf-30k with consideration of hydrodynamic effects are given below.

$$M_1 = \begin{bmatrix} 2798 & 0 & 0 & 0 & 872.9 & 0 \\ 0 & 3314 & 0 & -872.9 & 0 & 0 \\ 0 & 0 & 4831 & 0 & 0 & 0 \\ 0 & -872.9 & 0 & 942.1 & 0 & 0 \\ 872.9 & 0 & 0 & 0 & 1997 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1889 \end{bmatrix}; \tag{26}$$

$$D_1(v) = D_{LIN} + D_{QUAD}(v);$$

$$D_{QUAD}(v) = \begin{bmatrix} 773.6|u| & 0 & 0 & 0 & 0 & 0 \\ 0 & 1067|v| & 0 & 0 & 0 & 0 \\ 0 & 0 & 2277|w| & 0 & 0 & 0 \\ 0 & 0 & 0 & 302.5|p| & 0 & 0 \\ 0 & 0 & 0 & 0 & 1078|q| & 0 \\ 0 & 0 & 0 & 0 & 0 & 558.1|r| \end{bmatrix};$$

$$D_{LIN} = \begin{bmatrix} 123.8 & 0 & 0 & 0 & 0 & 0 \\ 0 & 170.7 & 0 & 0 & 0 & 0 \\ 0 & 0 & 364.3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 129.1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 188.0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 97.36 \end{bmatrix};$$

(27)

$$C_1(v) = \begin{bmatrix} 0 & 0 & 0 & 0 & 4831w & -3314v \\ 0 & 0 & 0 & -4831w & 0 & 2798u \\ 0 & 0 & 0 & 3314v & -2798u & 0 \\ 0 & 4831w & -3314v & 0 & 0 & 0 \\ -4831w & 0 & 2798u & -18 & 0 & 0 \\ 3314v & -2798u & 0 & 0 & 0 & 0 \end{bmatrix};$$

(28)

$$g_1(\eta) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 7081\cos\theta\sin\phi \\ 7081\sin\theta \\ 0 \end{bmatrix}.$$

(29)

Matrices (26)–(29) already contain hydrodynamic effects calculated according to the strip theory [38].

The six-link robot (Figure 6) with cylindrical rotating links as manipulative machines [35] will be considered below. Table 1 shows the parameters of both devices.

Figure 6. Six-link robot «Orion 7R» kinematic scheme [35].

Table 1. Manipulator link parameters.

Link i	Link Mass m_i, (kg)	Buoyancy B_i (N)	Radius r_i (m)	Length l_i (mm)
1	1.50	0.4	0.1	64.9
2	1.25	0.2	0.1	957.1
3	1.25	0.4	0.1	125
4	2.1	0	0.1	580
5	1.6	0	0.1	259
6	1.05	0.15	0.1	0

Workclass ROVs are often equipped with one or two manipulators, depending on the type of operation the ROV is designed for. The main reason for equipping ROVs

with manipulators is to design them for intervention tasks. By mounting the ROV with manipulators, additional hydrodynamic, hydrostatic, and rigid body inertia forces and moments will affect the vehicle [5,32]. If the ROV is large enough, these effects are negligible. However, these effects usually have to be modelled and taken into account when designing workclass ROVs. Here, the manipulator is of considerable size, and the presence of the manipulator will create forces and moments on the ROV that affects the response of the system. In addition, when the manipulators perform intervention tasks, they come in contact with the environment, which creates contact forces that also will affect the ROV. No matter the size difference between the ROV and the manipulator, the contact forces can be of considerable size, which needs to be taken into account when modelling the system.

The joint motion is often a very simple one degree of freedom linear or rotational motion. Manipulator systems therefore consist of either revolute or prismatic joints or a combination of these. The manipulator system considered in this paper, shown in Figure 6, is built up by revolute joints. This gives very favorable characteristics to the robot's workspace, and it makes the modelling a lot easier, as the homogeneous transformation matrices can be written in a very simple form using D-H parameters in Table 2.

Table 2. D-H parameters.

Link i	a_i (mm)	α_i (rad)	d_i (mm)	q_i (rad)
1	64.9	$\pi/2$	0	q_1
2	957.1	0	0	$q_2 + \pi/2$
3	125	$\pi/2$	0	$q_3 - \pi/2$
4	0	$-\pi/2$	580	q_4
5	259	$-\pi/2$	0	q_5
6	0	$-\pi/2$	0	q_6

The resulting equations are quite cumbersome, but the method of calculating the manipulator's dynamic parameters based on the Euler–Lagrange method is redundantly described in papers [35,37,38]. The source [33] provides formulas to calculate the hydrodynamic effects for the cylindrical shape, which were used to calculate the resulting parameters.

It is necessary to take into account the offset of the manipulator's base relative to the center of mass of the vehicle. The resulting required transformation matrix, which provides all the necessary basic kinematic transformations, is shown below

$$T_0 = \begin{bmatrix} 1 & 0 & 0 & 0.75 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -0.75 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{30}$$

From this, one can see the desired offset in the last column of the matrix (30).

3.7.3. Controller Parameters

One of the most important tasks in the synthesis of SDDRE control is the selection of penalty matrices Q and R. There are no clearly formulated rules in selecting the numerical values for each of them. This is a creative challenge that requires data calibration, for example, in numerical simulations.

In the context of this paper, the penalty matrices for the first and the second UVMS were matched as follows:

$$Q = \begin{pmatrix} 10^4 \cdot I_{12 \times 12} & 0_{12 \times 12} \\ 0_{12 \times 12} & 0_{12 \times 12} \end{pmatrix}, \ R = 10^{-4} \cdot I_{12 \times 12}. \tag{31}$$

In this way, parameters (31) restrict the states and provide more freedom to the system control.

4. Results

The following section will present the Simulink model applied to simulate the system, and all the relevant blocks will be reviewed. The Simulink model for the underwater vehicle–manipulator system is presented in Figure 7.
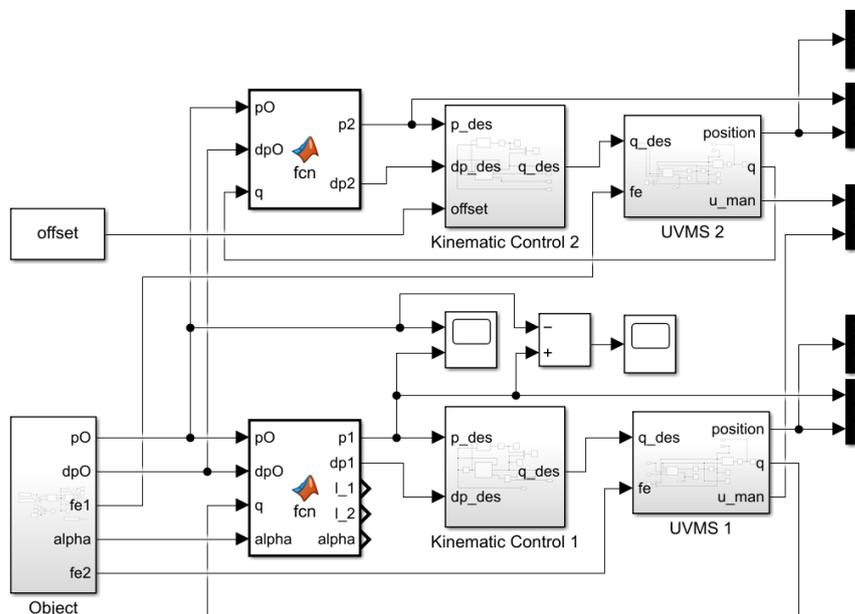

Figure 7. Simulink model for the two UVMSs.

The Simulink model consists of kinematic and dynamic modeling blocks for each link. Reference block "Object" define reference values and constants.

Figure 8 shows the subsystem for calculating the generalized coordinates and position of the working tool using Lagrange dynamics and solving the Jacobian velocity conversion task. The outputs of the block are position and vector q.
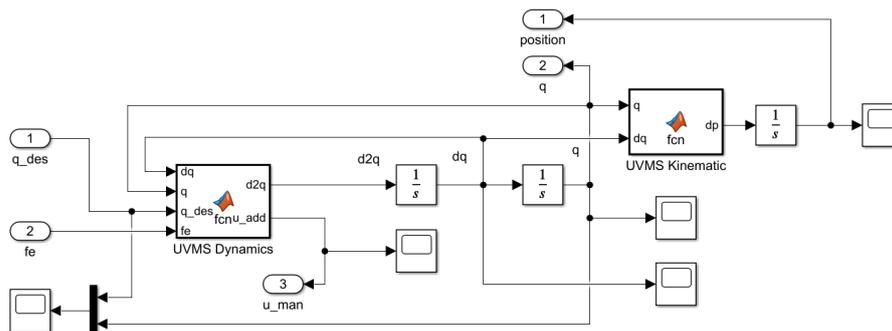

Figure 8. Simulink submodel of UVMS.

Figure 9 shows the module for calculating the desired generalized coordinates based on the method proposed in [29] using Equation (13).
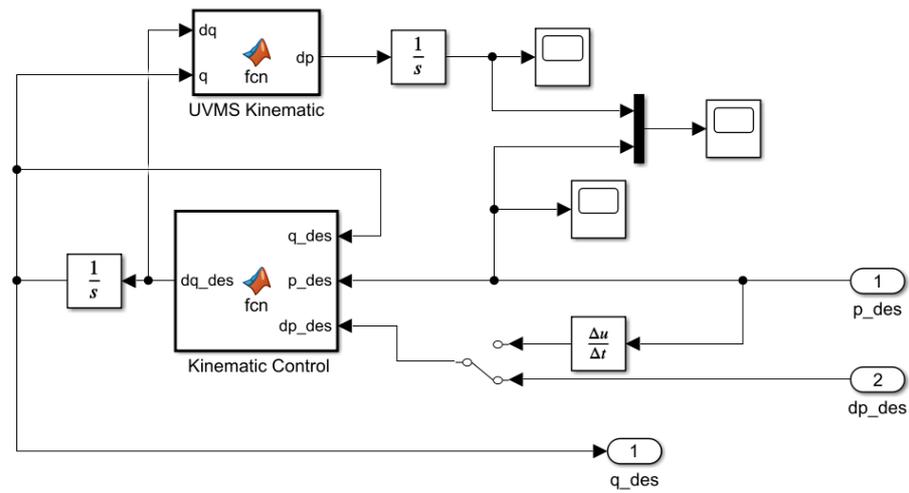
Figure 9. Simulink submodel of kinematic control.

　　The desired trajectory of the controlled object is shown in Figure 10. Here, you can see the desired angular and translational changes in the position of the object's center of mass. The object should move along a circular trajectory. The result of the first UVMS movement according to the desired trajectory is shown in Figure 11.
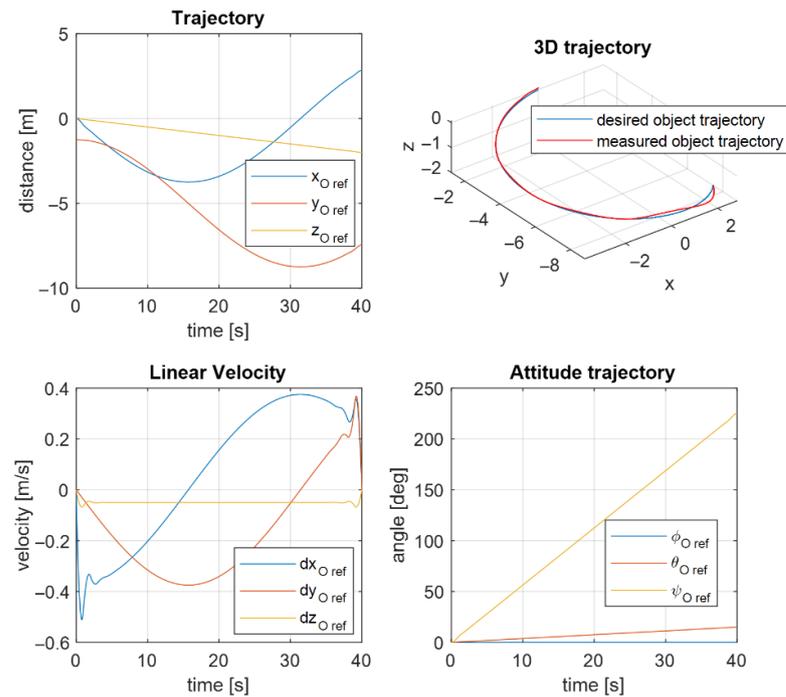

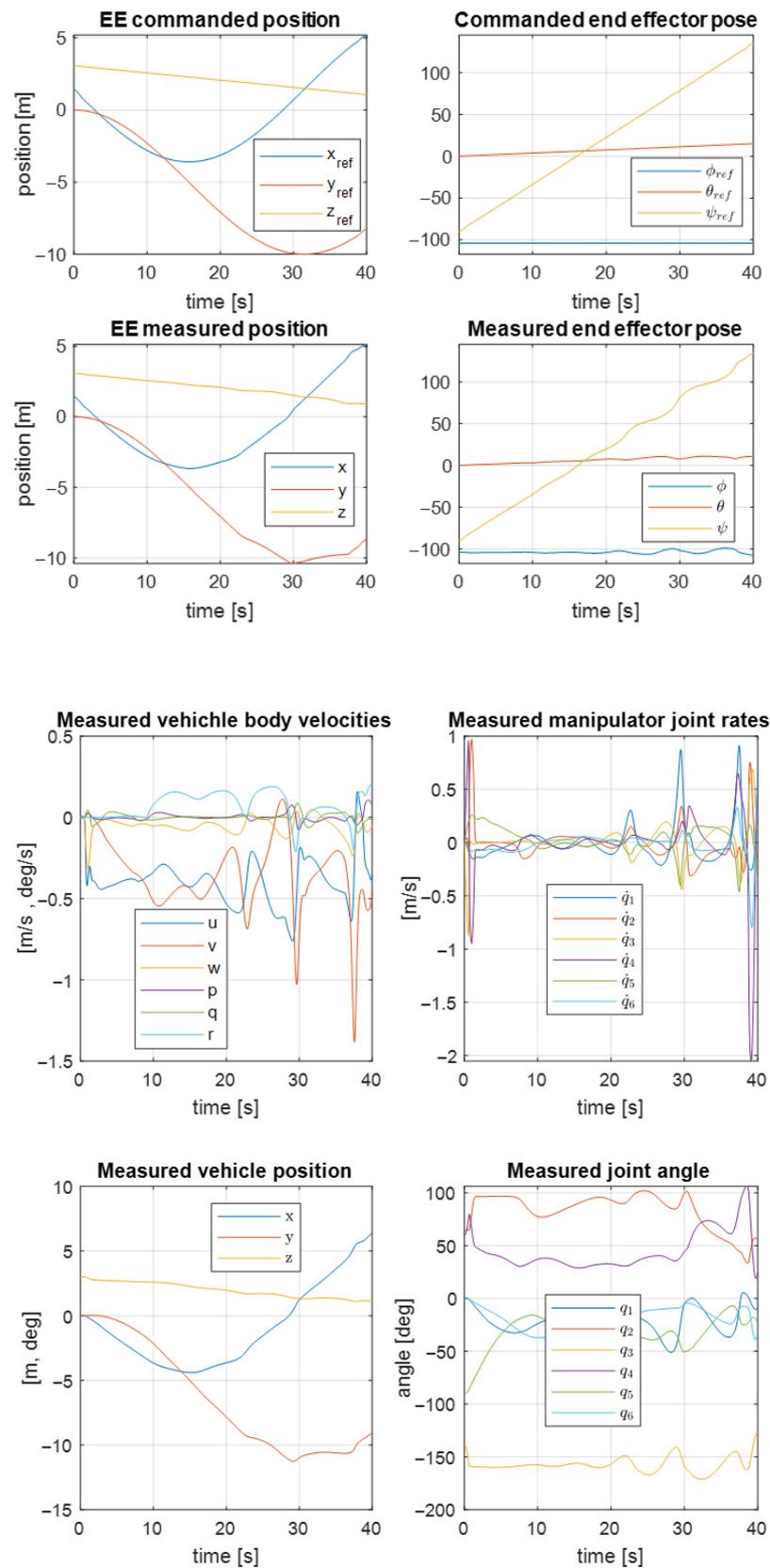**Figure 10.** Desired object trajectory.

Figure 11. Measured and commanded positions of the first UVMS.

Analogous results for the second UVMS are shown in Figure 12. As a result, the trajectory controller works correctly according to the system parameters.
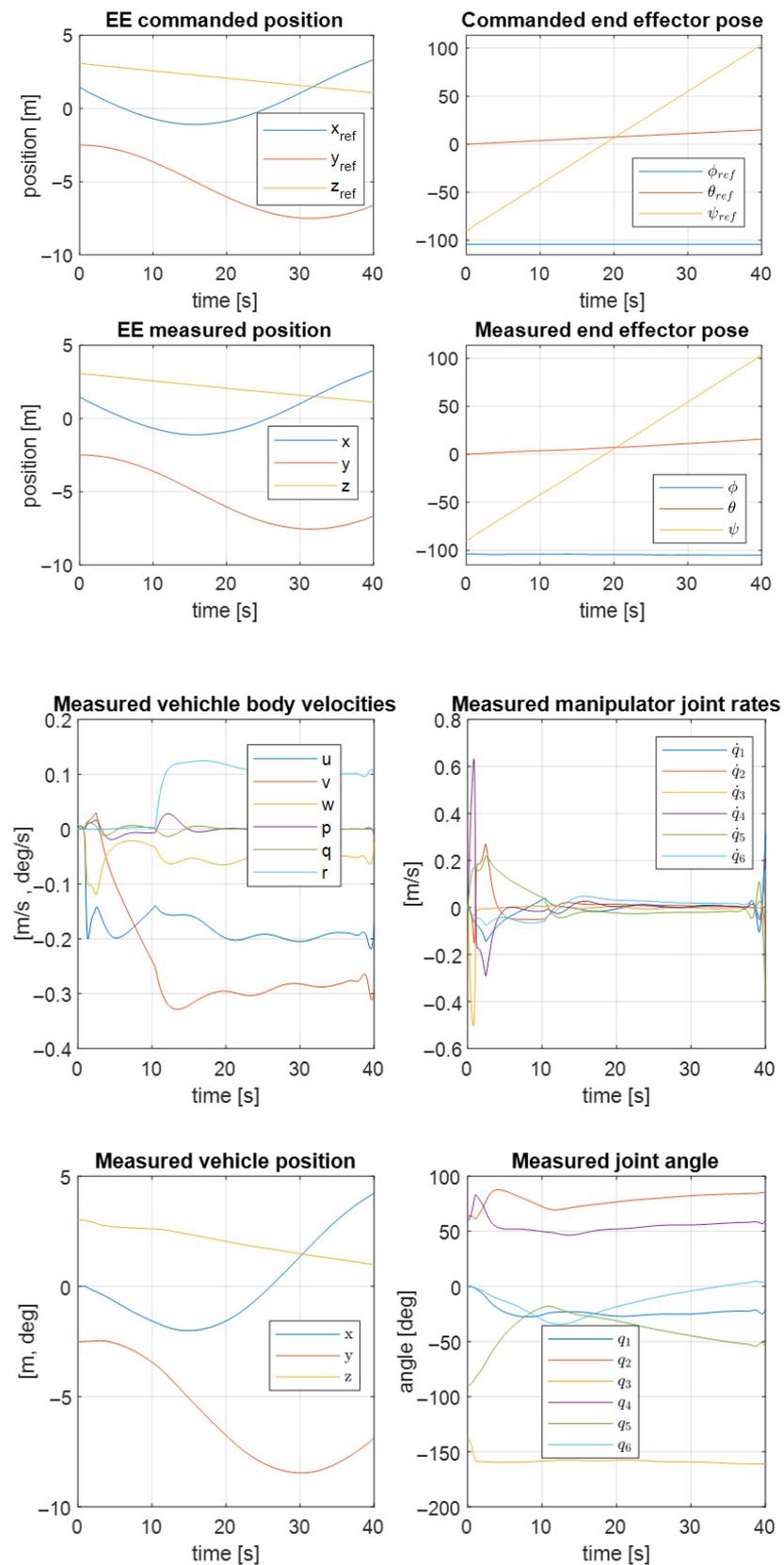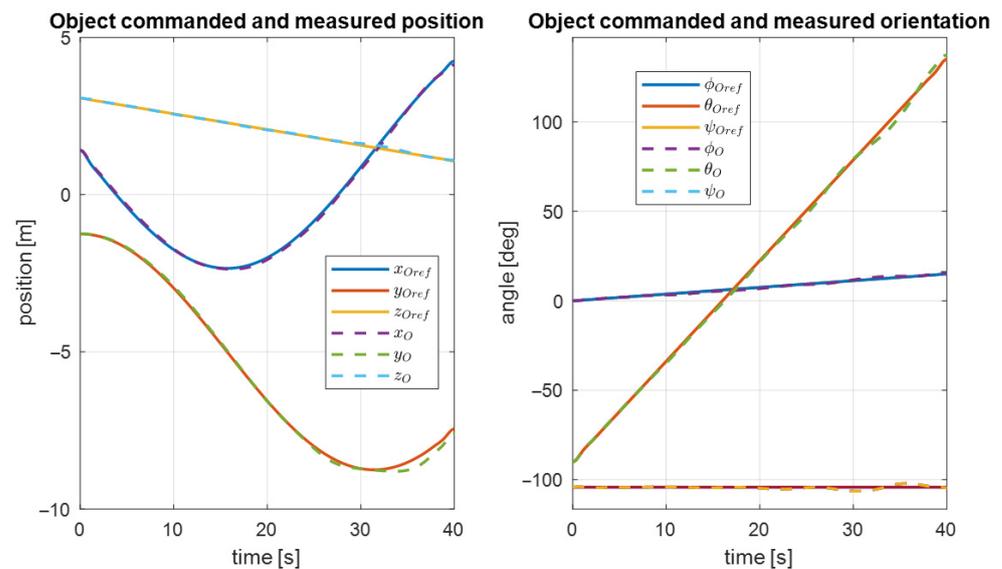
Figure 12. Measured and commanded positions of the second UVMS.

Figure 13 shows that the tracking control of the object occurs with a slight error; however, the endpoint of the trajectory is obtained.

Figure 13. Measured and commanded positions of the object.

The magnitude of the object's deviation from the desired trajectory is determined by the RMSE and MAE metrics, which are summarized in Table 3 for each degree of freedom.

Table 3. RMSE and MAE metrics of object position/orientation.

DOF	RMSE	MAE
x	0.0221	0.0481
y	0.0357	0.0664
z	0.0106	0.0166
ϕ	0.2515	0.6233
θ	0.0749	0.4101
ψ	0.6386	1.2101

5. Discussion

The results obtained in the simulation can be ambiguous for the considered method, but the synthesized control is optimal, which is an indisputable advantage in comparison with the classical methods of control systems.

Undoubtedly, the derived calculation methodology is valid, which is proved by the results of the third section of the paper. Practical applications are quite urgent, because there is a tendency for underwater robotics development in the world. In this case, it is necessary to synthesize a control law, which shall satisfy the optimality requirements. The work results can be used as a calculation method for controllers of robotic complexes in the underwater environment.

A further development vector is directed towards the task of comparing centralized and decentralized control systems. Additionally, it is necessary to consider the more complex case of a control system with many robots in a constrained environment with obstacles.

6. Conclusions

In this research, an algorithm for coordinated control of a robotic complex with N manipulators while performing the work of moving a concentrated mass cargo along a given trajectory was developed. To synthesize the control, we used an approach based on the representation of nonlinear manipulator dynamics models in the linear state-space form with state-dependent parameters (SDC-form). That allowed us to apply suboptimal control methods with the feedback based on the solution of the SDDRE. It should be noted that the SDDRE method provides the construction of control on a finite time interval and provides

the change in control intensity taking into account the transient effect of the system matrices in SDC form.

The research highlights two approaches of system implementation: a common controller for the complete system and a set of *N* independent subcontrollers for RTC manipulators.

The results of both approaches are similar, but for the systems with little interaction between manipulators, a common structure is recommended, and for the systems with a large number of manipulators, the approach with independent subcontrollers may be more acceptable.

In practical implementation, the failure of one subcontroller does not ruin the performance of the entire system. Therefore, it is recommended to use an independent design for experimental operations in order to maintain greater system safety.

Author Contributions: Conceptualization, A.K.; methodology, A.K. and V.K.; software, K.D.; validation, A.K., K.D. and I.L.; formal analysis, A.K.; investigation, A.K. and K.D.; resources, A.K.; data curation, I.L.; writing—original draft preparation, A.K. and K.D.; writing—review and editing, A.K.; visualization, K.D.; supervision, V.K.; project administration, A.K.; funding acquisition, A.K. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available upon request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kim, T.W.; Marani, G.; Yuh, J. Underwater Vehicle Manipulators. In *Springer Handbook of Ocean Engineering*; Dhanak, M.R., Xiros, N.I., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 407–422. ISBN 978-3-319-16648-3.
2. Sivčev, S.; Coleman, J.; Omerdić, E.; Dooly, G.; Toal, D. Underwater Manipulators: A Review. *Ocean Eng.* **2018**, *163*, 431–450. [CrossRef]
3. Whitcomb, L.L. Underwater Robotics: Out of the Research Laboratory and into the Field. In Proceedings of the Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065), San Francisco, CA, USA, 24–28 April 2000; Volume 1, pp. 709–716.
4. Neira, J.; Sequeiros, C.; Huamani, R.; Machaca, E.; Fonseca, P.; Nina, W. Review on Unmanned Underwater Robotics, Structure Designs, Materials, Sensors, Actuators, and Navigation Control. *J. Robot.* **2021**, *2021*, 5542920. [CrossRef]
5. Antonelli, G. *Underwater Robots*; Springer Tracts in Advanced Robotics; Springer International Publishing: Cham, Switzerland, 2018; Volume 123, ISBN 978-3-319-77898-3.
6. Specht, M.; Stateczny, A.; Specht, C.; Widźgowski, S.; Lewicka, O.; Wiśniewska, M. Concept of an Innovative Autonomous Unmanned System for Bathymetric Monitoring of Shallow Waterbodies (INNOBAT System). *Energies* **2021**, *14*, 5370. [CrossRef]
7. Krasuski, K.; Wierzbicki, D.; Bakuła, M. Improvement of UAV Positioning Performance Based on EGNOS + SDCM Solution. *Remote Sens.* **2021**, *13*, 2597. [CrossRef]
8. He, J.; Lin, J.; Ma, M.; Liao, X. Mapping Topo-Bathymetry of Transparent Tufa Lakes Using UAV-Based Photogrammetry and RGB Imagery. *Geomorphology* **2021**, *389*, 107832. [CrossRef]
9. Khawaja, W.; Semkin, V.; Ratyal, N.I.; Yaqoob, Q.; Gul, J.; Guvenc, I. Threats from and Countermeasures for Unmanned Aerial and Underwater Vehicles. *Sensors* **2022**, *22*, 3896. [CrossRef]
10. Kabanov, A.; Kramar, V.; Ermakov, I. Design and Modeling of an Experimental ROV with Six Degrees of Freedom. *Drones* **2021**, *5*, 113. [CrossRef]
11. González-García, J.; Gómez-Espinosa, A.; García-Valdovinos, L.G.; Salgado-Jiménez, T.; Cuan-Urquizo, E.; Cabello, J.A.E. Model-Free High-Order Sliding Mode Controller for Station-Keeping of an Autonomous Underwater Vehicle in Manipulation Task: Simulations and Experimental Validation. *Sensors* **2022**, *22*, 4347. [CrossRef]
12. Santhakumar, M. Investigation into the Dynamics and Control of an Underwater Vehicle–Manipulator System. *Model. Simul. Eng.* **2013**, *2013*, 17. [CrossRef]
13. Chang, Z.; Zhang, Y.; Zheng, Z.; Zhao, L.; Shen, K. Dynamics Simulation of Grasping Process of Underwater Vehicle–Manipulator System. *JMSE* **2021**, *9*, 1131. [CrossRef]

14. Zongyu, C.; Yang, Z.; Fangyuan, Z.; Zhongqiang, Z.; Jiliang, W. Research Progress of Underwater Vehicle–Manipulator Systems: Configuration, Modeling and Control. *J. Mech. Eng.* **2020**, *56*, 53. [CrossRef]

15. Korayem, M.H.; Nekoo, S.R. Controller Design of Cooperative Manipulators Using State-Dependent Riccati Equation. *Robotica* **2018**, *36*, 484–515. [CrossRef]

16. Ghasemi, A.; Keshmiri, M. Performance Assessment of a Decentralized Controller for Cooperative Manipulators; Numerical and Experimental Study. In Proceedings of the 2009 IEEE 6th International Symposium on Mechatronics and Its Applications, Sharjah, United Arab Emirates, 23–26 March 2009; pp. 1–6.

17. Tavasoli, A.; Eghtesad, M.; Jafarian, H. Two-Time Scale Control and Observer Design for Trajectory Tracking of Two Cooperating Robot Manipulators Moving a Flexible Beam. *Robot. Auton. Syst.* **2009**, *57*, 212–221. [CrossRef]

18. Yagiz, N.; Hacioglu, Y.; Arslan, Y.Z. Load Transportation by Dual Arm Robot Using Sliding Mode Control. *J. Mech. Sci. Technol.* **2010**, *24*, 1177–1184. [CrossRef]

19. Barbalata, C.; Dunnigan, M.; Petillot, Y. Coupled and Decoupled Force/Motion Controllers for an Underwater Vehicle–Manipulator System. *J. Mar. Sci. Eng.* **2018**, *6*, 96. [CrossRef]

20. Barbalata, C.; Dunnigan, M.W.; Petillot, Y. Position/Force Operational Space Control for Underwater Manipulation. *Robot. Auton. Syst.* **2018**, *100*, 150–159. [CrossRef]

21. Sun, Y.C.; Cheah, C.C. Coordinated Control of Multiple Cooperative Underwater Vehicle–Manipulator Systems Holding a Common Load. In Proceedings of the IEEE Oceans '04 MTS/IEEE Techno-Ocean '04 (IEEE Cat. No.04CH37600), Kobe, Japan, 9–12 November 2004; Volume 3, pp. 1542–1547.

22. Soylu, S.; Buckham, B.J.; Podhorodeski, R.P. IEEE Development of a Coordinated Controller for Underwater Vehicle–Manipulator Systems. In Proceedings of the IEEE OCEANS 2008, Quebec City, QC, Canada, 15–18 September 2008; pp. 1–9.

23. Zhu, Z.; Xu, X.; Zhu, Y.; Rodic, A.; Petrovic, P.B. Research on Fuzzy Adaptive and PD-Type Iterative Learning Control for Robot Manipulator. In Proceedings of the 2021 6th IEEE International Conference on Advanced Robotics and Mechatronics (ICARM), Chongqing, China, 3 July 2021; pp. 487–491.

24. Li, S.; Zhang, Y. *Neural Networks for Cooperative Control of Multiple Robot Arms*, 1st ed.; SpringerBriefs in Computational Intelligence; Springer: Singapore, 2018; ISBN 978-981-10-7037-2.

25. Zhang, S.; Lei, M.; Dong, Y.; He, W. Adaptive Neural Network Control of Coordinated Robotic Manipulators with Output Constraint. *IET Control Theory Appl.* **2016**, *10*, 2271–2278. [CrossRef]

26. Han, H.; Wei, Y.; Ye, X.; Liu, W. Motion Planning and Coordinated Control of Underwater Vehicle–Manipulator Systems with Inertial Delay Control and Fuzzy Compensator. *Appl. Sci.* **2020**, *10*, 3944. [CrossRef]

27. Heshmati-alamdari, S.; Karras, G.C.; Kyriakopoulos, K.J. A Distributed Predictive Control Approach for Cooperative Manipulation of Multiple Underwater Vehicle Manipulator Systems. In Proceedings of the 2019 IEEE International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 4626–4632.

28. Antonelli, G.; Fossen, T.I.; Yoerger, D.R. Modeling and Control of Underwater Robots. In *Springer Handbook of Robotics*; Siciliano, B., Khatib, O., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 1285–1306. ISBN 978-3-319-32550-7.

29. Siciliano, B. *Robotics: Modelling, Planning and Control*; Siciliano, B., Ed.; Advanced Textbooks in Control and Signal Processing; Springer: London, UK, 2009; ISBN 978-1-84628-641-4.

30. Fossen, T.I. *Guidance and Control of Ocean Vehicles*; Wiley: Chichester, UK; New York, NY, USA, 1994; ISBN 978-0-471-94113-2.

31. Heshmati-Alamdari, S.; Bechlioulis, C.P.; Karras, G.C.; Kyriakopoulos, K.J. Cooperative Impedance Control for Multiple Underwater Vehicle Manipulator Systems Under Lean Communication. *IEEE J. Ocean. Eng.* **2021**, *46*, 447–465. [CrossRef]

32. Heshmati-alamdari, S.; Bechlioulis, C.P.; Karras, G.C.; Kyriakopoulos, K.J. Decentralized Impedance Control for Cooperative Manipulation of Multiple Underwater Vehicle Manipulator Systems under Lean Communication. In Proceedings of the 2018 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV), Porto, Portugal, 6–9 November 2018.

33. Antonelli, G.; Pascal, M. Underwater Robots: Motion and Force Control of Vehicle–Manipulator Systems. Springer Tracts in Advanced Robotics, Vol. 2. *Appl. Mech. Rev.* **2003**, *56*, B81. [CrossRef]

34. Korayem, M.H.; Nekoo, S.R. State-Dependent Differential Riccati Equation to Track Control of Time-Varying Systems with State and Control Nonlinearities. *ISA Trans.* **2015**, *57*, 117–135. [CrossRef] [PubMed]

35. Spong, M.W.; Hutchinson, S.; Vidyasagar, M. Robot Modeling and Control. *Ind. Robot. Int. J.* **2006**, *33*, 403. [CrossRef]

36. Kabanov, A.A.; Kramar, V.A. Cooperative Control of Underwater Manipulators Based on the SDRE Method. In Proceedings of the 2021 IEEE International Russian Automation Conference (RusAutoCon), Sochi, Russia, 5–11 September 2021; pp. 515–520. [CrossRef]

37. Heydari, A.; Balakrishnan, S.N. Closed-Form Solution to Finite-Horizon Suboptimal Control of Nonlinear Systems: Closed-form solution to finite-horizon suboptimal control. *Int. J. Robust Nonlinear Control* **2015**, *25*, 2687–2704. [CrossRef]

38. Eidsvik, O.A.; Schjølberg, I. Determination of Hydrodynamic Parameters for Remotely Operated Vehicles. In Proceedings of the International Conference on Offshore Mechanics and Arctic Engineering, Busan, Korea, 19 June 2016; p. V007T06A025.