processes

MDPI

Article

Advanced Dynamics Processes Applied to an Articulated Robot

Florian Ion Tiberiu Petrescu

"Theory of Mechanisms and Robots" Department, Faculty of Industrial Engineering and Robotics, University POLITEHNICA of Bucharest, Splaiul Independentei Street 313, 060042 Bucharest, Romania; fitpetrescu@gmail.com; Tel.: +40-724040348

Abstract: The paper presents the dynamics of a 2R planar articulated robot, developed by two original methods. One is the classical "Lagrangian" adapted by the author, and the second method is absolutely original. The dynamics of the robot are based in both cases on the variation of the inertial forces in the mechanism, or practically on the influence of the masses of the moving elements of the robot. The influence of external loads, weights and the load to be transported is also taken into account. Another original element of the work is the choice of speeds in such a way that they correspond to an optimum necessary for the inverse kinematics imposed on the robot. For this reason, the dynamic operation will be quiet and without large variations or vibrations. If the speeds of the two electric motors (preferably stepper motors) areadapted to those recommended by the author, the controller (PID) used will have a very light load. It is even possible to eliminate it if the adjustment of the two stepper motors (actuators) is performed according to the speeds indicated by the author of the paper. The kinematic motion imposed by the indicated optimal speeds is dynamically and successfully checked by both methods used.

Keywords: dynamics; kinematics; articulated robot; Lagrange equation; two stepper actuators

check for updates

Citation: Petrescu, F.I.T. Advanced Dynamics Processes Applied to an Articulated Robot. *Processes* **2022**, *10*, 640. https://doi.org/10.3390/pr10040640

Academic Editor: Ján Piteľ

Received: 25 February 2022
Accepted: 22 March 2022
Published: 24 March 2022

1. Introduction

A definition of the physics or mechanics of dynamics says that it is a branch of mechanics (see mechanical meaning 1) that deals with forces and their relationship primarily with motion, but sometimes with the balance of bodies [1]. Dynamics is the discipline that deals with the real movement in any field, with it being the most important movement, a fact for which the dynamic study in mechanics, as well as in other fields, is very important. However, today, dynamics cover a multitude of aspects, starting from the forces in the systems and going to the new technologies and technological processes. A methodology for the flexible implementation of collaborative robots in intelligent production systems is presented in paper [2]. A method of optimizing the robot arm design by using a kinematic redundancy resolution technique is presented in paper [3]. Control of the trajectory of industrial robots using multilayer neural networks driven by iterative control of learning can be found in paper [4]. The dynamic and friction parameters of an industrial robot with the identification, comparison, and analysis of repetitiveness, represent other important aspects of the dynamic and robotic processes in the industry [5]. The impact of gravity compensation on learning-by-consolidation in goal-setting tasks for robotic manipulators is a relatively new issue in dynamic disciplines [6]. Another new dynamic aspect is the mechatronic redesign of a manual assembly workstation in collaboration with wiring assemblies [7], which can be directly associated with the new technological processes.

Another aspect of the dynamic process appears in paper [8] by developing a high-speed, low-latency remote-controlled robot hand system. The educational resources accessible for teaching and learning robotics [9] are also a dynamic aspect, but different from the physical–mechanical one that we are particularly interested in within this paper. Dynamic identification of the parameters of an indication mechanism taking into account the articulated play [10] is a basic dynamic process. The impact of cycle time and payload of an industrial robot on resource efficiency [11] is also an important aspect of dynamic processes. Today, adaptive control of the position (or force) of a robotic manipulator [12,13], as well as the trajectory control [14–18], are important dynamic processes.

There is no need to discuss the importance of the articulated robots studied in this paper because today they represent 90% of the industrial robots used almost everywhere, these articulated robots havethe task of making practically all the components of a car, of moving and maneuvering them, and then assembling them. Industrial robots are automatic, programmable machines with multiple axes of motion that can move to perform a task. An axis of motion is a joint in the body of the robot in which a segment can move. For example, a three-axis robot can rotate at the base, move its arm up and down, and rotate its grip at the end of the arm. They impress with their versatility whichallows them to use new types of application areas. Whether on the floor, ceiling, or wall—thanks to the integrated power supply and the compact control system—such a well-established system offers maximum accuracy in the smallest spaces. Safe-Robot functionality enables innovative automation concepts. Whether it is suitable for rooms with a controlled atmosphere, for areas with a risk of explosion, with a hygienic or splash-resistant design, it is always precise and fast in every model and movement. Whether it is dusty, humid, or sterile environments, such a robot achieves top performance in any environment production. Robots have penetrated everywhere, including the medical area, in the operating rooms [19]. A special issue in the dynamic processes of robots [20–23] is the study of their kinematics, closely related to dynamics. Another important dynamic aspect [24,25] is the balancing technological processes. The influence of forces [26] is in fact that which determines the dynamic, real operation of all mechanical processes, including robots. In this way, it is possible to control the trajectory of robots and/or aerospace ships (or drones) [27–37]. Mechanical transmissions actively participate in the performance dynamics of robotic technological dynamic processes [38,39]. An important aspect in the study of dynamical systems is the forces of inertia, which can be taken into account by the inertial masses of the system [40–43].

The main part of a three-axis articulated robot is the two planar movable arms operated by two actors. This simplified system will be taken into account in this paper [19–23]. The optimal trajectories of a robot can be synthesized and controlled only by a predictive, dynamic design, taking into account the dynamic kinematics of the robot [44–53]. No matter how many degrees of mobility an articulated robotic system has, it will be initially operated by a rotating platform that orients it quickly in space, which is also its spine, then two arms resembling a human hand, i.e., an arm and a forearm, in order to then directly insert the defector, or several additional modules with several degrees of mobility. For this reason, the main part of the system is the arm and forearm, as will be studied in this paper.

The paper will present two original methods of dynamic study of an articulated system with two mobilities, with two actuators (preferably stepper actuators). The working speed proposed by the author for the two actuators will be taken as variable so that all the positions planned within the reverse kinematics trajectory of the articulated robot can be reached. In this way, the dynamic operation will be very close to the designed kinematic one.

The importance of robot dynamics, which is the real workings of robots, has been reflected in many specialized works since the beginnings of modern robotics. In this area of mechanism dynamics, the robotics community has focused particularly on the issue of computational efficiency, with the obvious goal of facilitating dynamic computational methods with the help of computational ones [54–59]. Thus, many of the most efficient algorithms in dynamics, which are applicable to a wide class of mechanisms, have also been developed by researchers in robotics [60]. Computational efficiency continues to be important for the simulation and control of mechanisms (that are increasingly complex and that operate at higher speeds), but also other aspects of the problem of contour dynamics that are becoming important today. In this sense, the algorithms must be clearly formulated through a compact set of equations in order to facilitate their development and implementation. In addition, there must be a clear relationship between these equations and the recursive set from which the highest computational efficiency is obtained, which is why the utilization of spatial notations, as well as the algebra of the spatial operator, has increased the efficiency of dynamical systems and especially robots. It is still important to further develop algorithms with a more general character, valid in the dynamics of mechanisms as well as in robots, which have applicability to robotic mechanisms with general geometries and joint structures, which is why, based on the study of such classical systems [44–46], the author of this paper wish to present for the first time two new general algorithms for dynamic calculation of mechanisms, with applicability to articulated industrial robots, the most frequently used today in world robotics (in a proportion of over 90% of total industrial robots).

Between the years 2000 and 2008, the novelties in dynamics remained the same as those of the 20th century, using: (1) the Newton–Euler algorithm (NEA); (2) the Lagrange equations, kind I or II; (3) Laplace transformations, or Fourier transformations [42–44]. However, there are still some transformations and novelties, such as: (a) the recursive Newton–Euler algorithm (RNEA). This algorithm solves the inverse dynamics problem; (b) the articulated-body algorithm (ABA). This algorithm solves the forward dynamics problem; (c) the composite-rigid-body algorithm (CRBA). This algorithm calculates the joint-space inertia matrix, H. When combined with the RNEA, to calculate c, and a linear equation solver, to solve $Hq'' = \tau - c$ for q'', the result is an algorithm to solve the forward dynamics problem [48–51]. For a long time now, the problem of the dynamics and control of the end effector has been raised [52], as it is the most important problem for a robot. In this paper, the new methods presented will refer to the entire dynamics of the robot, studied on all its mobile elements, at any point or coupling of its connection, so including the end effector point, one of the most important. One considers the robot's kinematics starting from the end effector, so the inverse kinematics imposed by the useful trajectory of the robot. One solves the trajectory imposed simply and originally with the help of logical functions (see Appendix A).

The Mathcad software helps a lot in the correct, precise, simple, direct, and fast solution of all these algorithms and programs (Appendix A). More and more dynamics problems are starting to be solved with the help of specialized software, the most used lately being Matlab with its Simulink subprograms (that help to simulate the theory in real-time). When it comes to simulating the robot commands and control, the Simulink is most desirable, and possibly its subset, "SimMechanics" [53]. Even in these situations, the dynamic methods used remain basic, so this paper has its role well-determined in the realization of methods to study the dynamics of the machines that are more user-friendly and easier to understand and apply. Matrix methods are the most common, including in the calculations of space robots [60].

Frictional contact multibody simulation has been a challenging research topic for the past 30 years. Rigid body hypotheses are commonly used to approximate contact physics and, together with Coulomb friction, lead to difficult-to-solve nonlinear complementarity (NCP) problems. On the other hand, clamping robots often introduce significant compliance. Conforming contact, combined with regular friction, can be modeled entirely with ODE, avoiding NCP solutions. Unfortunately, regular friction introduces rigid high-frequency dynamics and even the default methods struggle with these systems, especially during slip-stick transitions. In order to improve the performance of the default integration for these systems, a transition line search (TALS) can be introduced, which greatly improves the convergence of Newton–Raphson iterations performed by the default integrators. This demonstrates that TALS works best with semi-implicit integration, but that explicit treatment of normal compliance can be problematic. To address this, a TAMSI (Transition-Aware Modified Semi-Default) integrator is being developed that has a similar computational cost to the semi-default methods, but implicitly couples compliant contact forces, leading to a more robust method. The robustness, accuracy, and performance of TAMSI are assessed and the approach is demonstrated along with relevant sim–real handling tasks [60]. In addition to Lagrangian or Hamiltonian mechanics, based today on the (transformed) Laplace or Fourier function, software and dynamic algorithms have increasingly been developed for the command and control of robots and/or aircraft, using modern Neural, Fuzzy, Genetic Algorithms methods, predictive, regressive, robust, closed-loop, or feedforward methods. Controllers today play a key role in command, controlling, and automating machine dynamics, and among other issues they raise are the dynamic programs and algorithms of the machine that need to be programmed, commanded, and controlled [47,52].

Reinforcement learning (RL) is promising for complicated nonlinear stochastic control problems. Without using a mathematical model, an optimal controller can be learned from data evaluated according to certain performance criteria by trial and error. However, the data-driven learning approach is renowned for not guaranteeing stability, which is the most fundamental property of any control system. For these reasons, Lyapunov's classical method of system stability is further studied and explored to analyze the uniform final limit stability (UUB) exclusively on the basis of data, without using a mathematical model. In this way RUB with a UUB guarantee can be applied for the control of dynamic systems with safety constraints. As a result, optimal controllers can be learned to guarantee the UUB of the closed-loop system both at convergence and during learning. The proposed algorithms are evaluated on a series of robotic tasks of continuous control with safety constraints. Compared to existing RL algorithms, the proposed method [61] can achieve superior performance in terms of maintaining safety. The actuators implemented on the respective system also have an important role in achieving the dynamic stability of a robot [62].

The dynamic stability of the robotic system also depends to a large extent on the way the actuator speeds are chosen. The present paper aims to bring superior stability to robotic systems based on the imposition of controlled speeds (see Section 5 of the paper). In this sense, two new methods of imposing actuator speeds are adapted, regardless of whether they are linear or rotational. The calculations in both situations (variants) are based (like the methods) on the numerical derivation by developing the functions in a Taylorseries. The closed-loop control that is usedtoday in all robotic systems brings major advantages in the dynamics of that system, and due to the change of actuators parameters based on real-time data provided by a lot of sensors (including visual and proximity) mounted on the robotic system [62]. Even so, with the use of servo motors, or the use of stepper actuators, which bring high precision and permanent active control, the use of advanced software in robotic systems based on simple and efficient dynamic algorithms has greater advantages in stability, dynamics of the system, and in its natural behavior, achieving an optimal function without vibrations and noises, with high offset speeds, and with high precision, following exactly the trajectory imposed by the inverse kinematics.

2. Methods

2.1. First Dynamic Proposed Method

The first original dynamic proposed method presented in the paper is a classic, improved one. The articulated robot mechanism with two degrees of mobility that is studied can be observed in Figure 1. The two movable elements 1 and 2, with a flat motion of the articulated robot, are actuated by two stepper actuators 1 and 2, respectively.

In the approximate classical dynamics, the variable angular velocities and the variable angular accelerations are determined; the variation is due to the forces of inertia or the variation of the masses of the moving elements (Figure 2).

The calculation relationships are further presented in system (1). The calculations were performed in Mathcad and can be traced more closely in Appendix A. J_A^{1*} represents the mass moment of inertia of the entire robot reduced (*) to element 1 in point A (kinematic coupling A). It takes the place of the rotating mass of the entire robot around the coupling A on element 1 and can represent the dynamic model of the entire robot reduced to element 1 in the active coupling A, so that it can generate the dynamic parameters of this element 1 (such as: w1, eps1). Its equation is obtained simply by conserving the kinetic energy of the entire machine (robot). Similarly, the expression of the rotating mass (J_B^{2*}) of the machine (robot) considered (reduced, *) at element 2 in the kinematic couple (motors) B is deduced. This simple, original, and efficient dynamic system is deduced by conserving the kinetic energy of the entire machine (robot) and behaves somewhat similar to that achieved by using the Lagrange dynamic equation of the second kind. It is applied separately to each mobile element, and not only to the driving elements. Thus, in order to determine the dynamic parameters of element 1, after determining the variable rotating mass J_A^{1*} of the entire system (machine, robot) reduced to element 1, the variable (dynamic, real, angular) velocity of element 1 (ω_1^D), is then determined. The quick method presented (Equation (1)) is an original one [38,39]. Another completely new method is presented in Section 2.2 of this paper, with slight advantages in terms of accuracy, but with major disadvantages in terms of its generality.

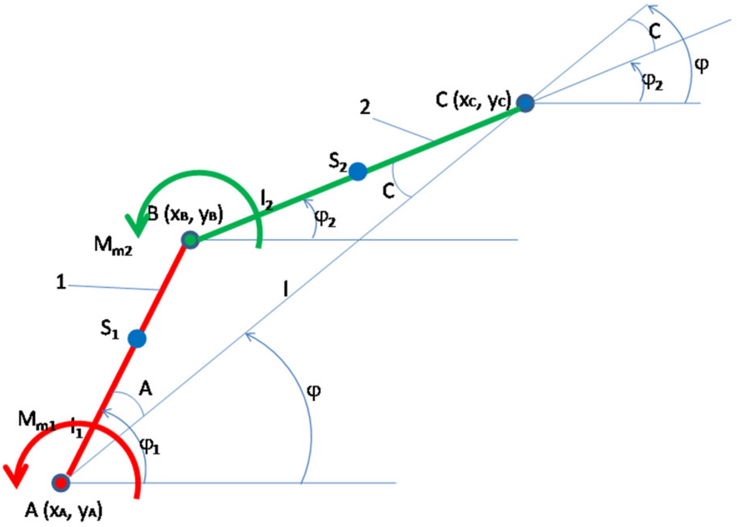

Figure 1. The articulated robot mechanism with two degrees of mobility.

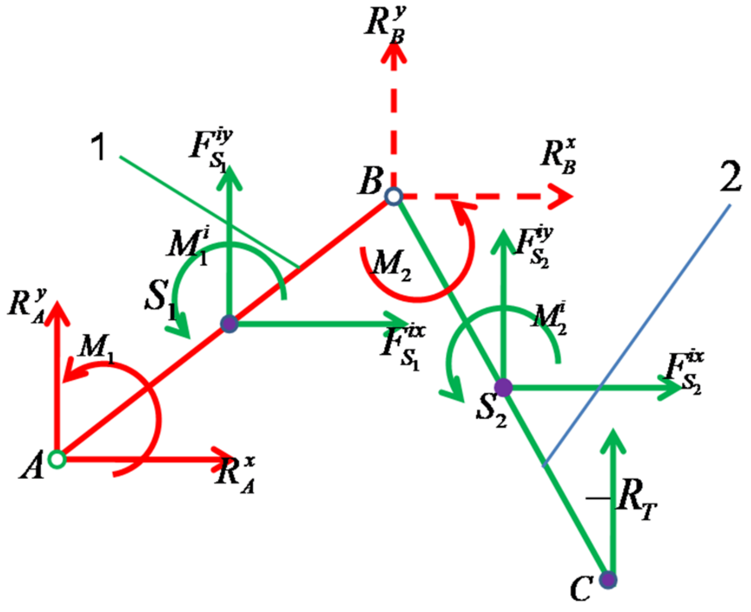

Figure 2. In the approximate classical dynamics, the variable angular velocities and the variable angular accelerations are determined; the variation is due to the forces of inertia.

Observation: All units of measurement used are those given by the international system.

$$\begin{cases} J_A^{1*} = J_{S_1} + \frac{\left(\dot{x}_{S_1}^2 + \dot{y}_{S_1}^2\right) \cdot m_1 + J_{S_2} \cdot \omega_2^2 + \left(\dot{x}_{S_2}^2 + \dot{y}_{S_2}^2\right) \cdot m_2}{\omega_{1m}^2} \\ J_B^{2*} = J_{S_2} + \frac{\left(\dot{x}_{S_1}^2 + \dot{y}_{S_1}^2\right) \cdot m_1 + J_{S_1} \cdot \omega_1^2 + \left(\dot{x}_{S_2}^2 + \dot{y}_{S_2}^2\right) \cdot m_2}{\omega_{2m}^2} \\ \omega_1^D = \sqrt{J_{Amed}^{1*}} \cdot \frac{\omega_{1m}}{\sqrt{J_A^{1*}}} \\ \omega_2^D = \sqrt{J_{Bmed}^{2*}} \cdot \frac{\omega_{2m}}{\sqrt{J_B^{2*}}} \end{cases} \tag{1}$$

J_{S1} is the rotational mass of element 1 determined in its center of mass S_1 about an axis perpendicular to the plane of the mechanism at point S_1. It is generally denoted by the mass (or mechanical) moment of inertia of element 1 determined in its center of mass S_1. J_{S2} is the rotational mass of element 2 determined in its center of mass S_2 about an axis perpendicular to the plane of the mechanism at point S_2. It is generally denoted by the mass (or mechanical) moment of inertia of element 2 determined in its center of mass S_2. m_1 represents the mass (classical in translation) of element 1, while m_2 is the mass of element 2. $\omega_{1\mathrm{m}}$ is the average rotational speed of element 1, and similarly, $\omega_{2\mathrm{m}}$ is the average angular velocity of element 2; ω_1 is the rotational speed of element 1, and similarly, ω_2 is the angular velocity of element 2; ${\omega_1}^D$ is the dynamic (real) rotational speed of element 1, and similarly, ${\omega_2}^D$ is the dynamic angular velocity of element 2. $\dot{x}_{S_1}$ and $\dot{y}_{S_1}$ represent the scalar components along the x and y axes, respectively, of the linear velocity of element 1 at point S_1 (center of mass of the element). Where $\dot{x}_{S_2}$ and $\dot{y}_{S_2}$ represent the scalar components along the x and y axes, respectively, of the linear velocity of element 1 at point S_2 (center of mass of the element).

There are two original basic aspects in the presented method: (1) the generalized way in which the moments of mass inertia of the whole mechanism are written, reduced to any of the desired elements of the machine, using the kinetic energy conservation equation of the whole mechanism; (2) the generalized way in which the angular velocities of each element of the mechanism are determined by using an original relation already presented in previous work [38,39], now repeatedly for each element of the machine, in this case of the simple articulated robot. For a better understanding of this original dynamic method with wide use, the author presents it in the paper and applies it to another type of robot in Section 4. For element 1, the moment of reduced mechanical inertia in joint A is determined, and for element 2, the moment of reduced mechanical inertia in joint B is determined. The classical dynamic angular velocity of each actuator (w_1, w_2) is determined according to the kinematic angular velocity and the variation of the respective moment of inertia (1, or 2) reduced to its joint (A or B). The method is simple, straightforward, and original [38,39]. Dynamic angular accelerations are determined directly from the angular velocities by approximate numerical derivation (using the Taylor series development). The solution ofsolving Lagrange's equations of the second kind (here being two degrees of mobility, so two independent variables) generates larger errors, and for this reason, we resorted to the method of finding direct angular accelerations based on dynamic, real angular velocity values [38,39].

Figures 3 and 4 show the variations of mass (mechanical) moments of inertia ($J_{A\mathrm{red}}$, or $J_{B\mathrm{red}}$ [kg·m^2]) reduced to a coupling A (element 1), and B (element 2), depending on the independent variable (k) initially chosen in the program, which in turn may depend on the rotation angle of elements 1 and 2, respectively, or directly on the time t (see the program written in Mathcad, in Appendix A).

Figure 5 shows the variation of the dynamic angular velocity 1 (w_{1D} [s^{-1}]), depending on the independent parameter k.

Figure 6 shows the variation of the dynamic angular velocity 2 (w_{2D} [s^{-1}]), depending on the independent parameter k.

Figure 7 shows the difference between the kinematic speed and the classical dynamic speedof the mobile element 1 of the robot.

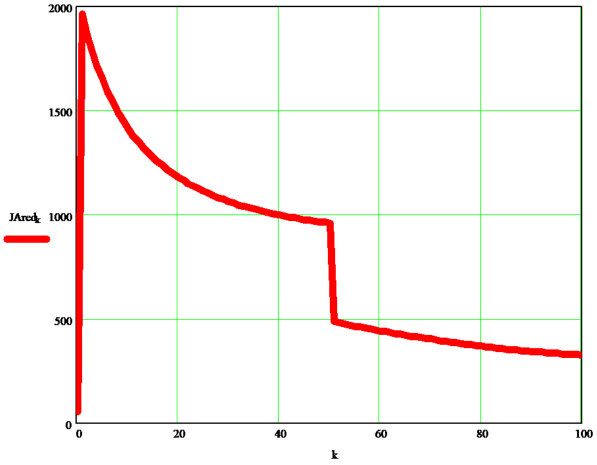

Figure 3. Variations of mass (mechanical) moments of inertia reduced to a coupling A (element 1).

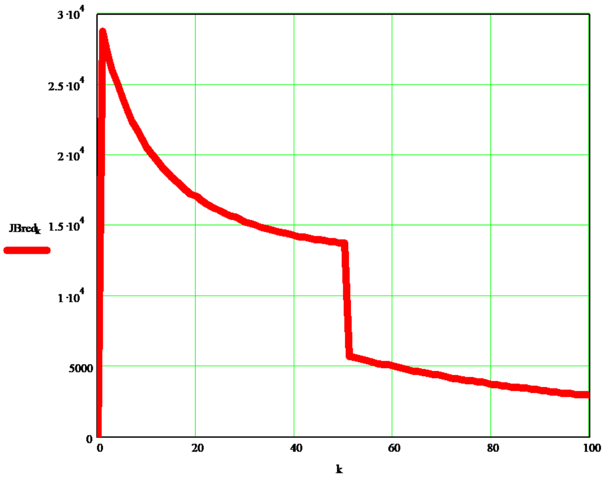

Figure 4. Variations of mass (mechanical) moments of inertia reduced to a coupling B (element 2).

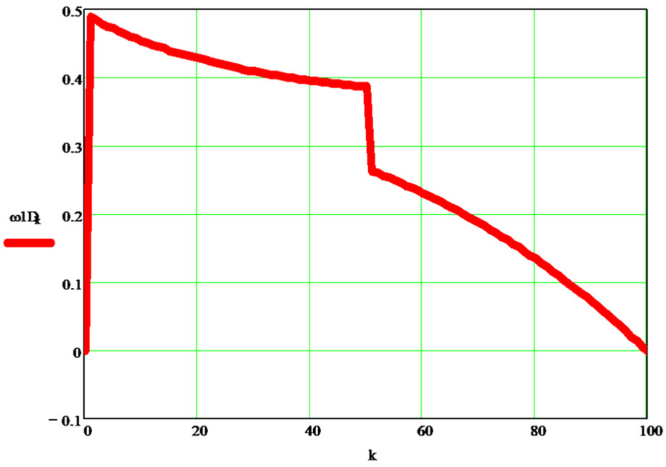

Figure 5. Variation of the dynamic angular velocity 1, depending on the independent parameter *k*.

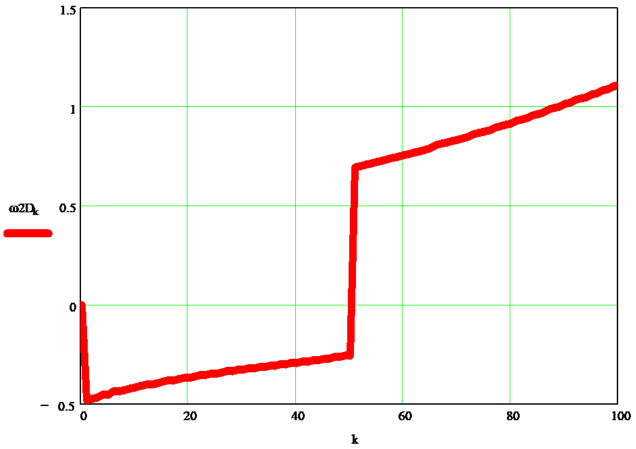

Figure 6. Variation of the dynamic angular velocity 2, depending on the independent parameter *k*.

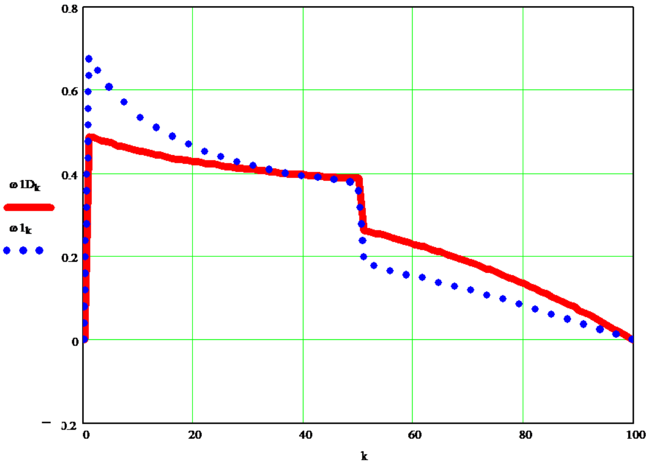

Figure 7. Difference between the kinematic speed and the classical dynamic speedof the mobile element 1 of the robot.

Figure 8 shows the difference between the kinematic speed and the classical dynamic speedof the mobile element 2 of the robot.

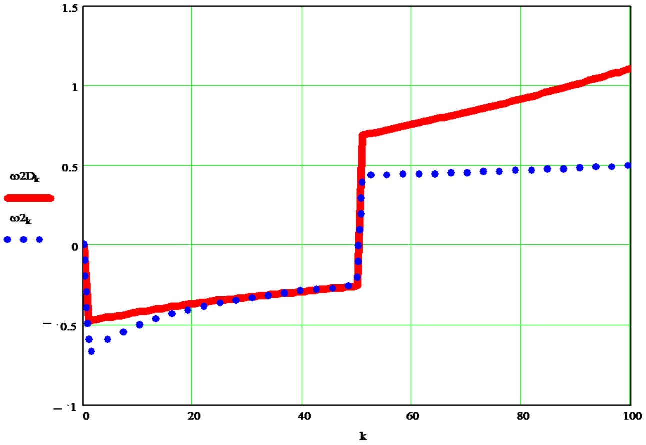

Figure 8. Difference between the kinematic speed and the classical dynamic speedof the mobile element 2 of the robot.

2.2. Second Original Dynamic Proposed Method

The second original dynamic method presented in the paper uses the same Figures 1 and 2. The basic idea is a simple one, namely, if the dynamic variations of the angular velocity belonging to a moving element compared to the desired kinematic velocity actually depend on the variation of the forces in the robot mechanism, then they will be determined based on the forces of the whole mechanism considered, without the moving elements being separated from each other or by the external input couplings, i.e., without loosening the connections (Figure 2). One starts from the simple calculation relations (2):

$$\begin{cases} \sum F_x^{1,2} = 0 \Rightarrow F_{S_1}^{ix} + F_{S_2}^{ix} = 0 \\ \sum F_y^{1,2} = 0 \Rightarrow F_{S_1}^{iy} + F_{S_2}^{iy} - R_T = 0 \end{cases} \tag{2}$$

In the first relation, whichrepresents the sum of the forces acting in the machine only on the abscissa axis (x-axis), we meet the inertia forces in the system $F_{S_1}^{ix}$, $F_{S_2}^{ix}$ concentrated in the centers of mass S_1 and S_2, respectively, oriented on the x-axis. In the second relation, whichrepresents the sum of the forces acting in the machine only on the y-axis, we meet the inertia forces in the system $F_{S_1}^{iy}, F_{S_2}^{iy}$ concentrated in the centers of mass S_1 and S_2, respectively, oriented on the y-axis, and the technological endurance R_T, the force which the machine must bear or overcome during the active working period; m_1 represents the mass of element 1, and m_2 that of element 2; ω_1 represents the angular velocity of element 1; ω_2 represents the angular velocity of element 2; ε_1 represents the angular acceleration of element 1; ε_2 represents the angular acceleration of element 2; $S_1 = AS_1$ and $S_2 = BS_2$ represent the distances to the centers of mass of elements 1 and 2, respectively; $l_1 = AB$ and $l_2 = BC$ represent the lengths of the two elements 1 and 2, respectively.

The system in (2) develops and takes the shape of (3):

$$\begin{cases} m_1 S_1 \cos\varphi_1 \omega_1^2 + m_1 S_1 \sin\varphi_1 \varepsilon_1 + m_2 l_1 \cos\varphi_1 \omega_1^2 + m_2 S_2 \cos\varphi_2 \omega_2^2 + m_2 l_1 \sin\varphi_1 \varepsilon_1 + m_2 S_2 \sin\varphi_2 \varepsilon_2 = 0 \\ m_1 S_1 \sin\varphi_1 \omega_1^2 - m_1 S_1 \cos\varphi_1 \varepsilon_1 + m_2 l_1 \sin\varphi_1 \omega_1^2 + m_2 S_2 \sin\varphi_2 \omega_2^2 - m_2 l_1 \cos\varphi_1 \varepsilon_1 - m_2 S_2 \cos\varphi_2 \varepsilon_2 = (m_1 + m_2) g + R_T \end{cases} \tag{3}$$

$$\begin{cases} \omega_{1d} = \omega_1 + d\omega_1; \varepsilon_{1d} = \frac{d\omega_1}{d\varphi_1} \cdot \omega_1 \\ \omega_{2d} = \omega_2 + d\omega_2; \varepsilon_{2d} = \frac{d\omega_2}{d\varphi_2} \cdot \omega_2 \end{cases} \tag{4}$$

The system of two equations in (3) is rewritten using the notations in (4) so that one arrives at a linear differential system of two equations with two unknowns, dw_1 and dw_2, which are then solved simply (see Equation (5) written in Mathcad); it is basically a finite difference method. The angles (φ_1 and φ_2) position elements 1 and 2, respectively, with respect to the x-axis (abscissa axis; Figure 1); g is the gravitational constant; all sizes used are those of the international standard. In order to be able to express the very long equations, the notations $N1$ and $N2$ were used for the numerator of the two fractions, and the notations n1 and n2 for the denominator of each of the two fractions (in the equations system (5)). The new equations used (in the second method presented) are with finite differences and have the great advantage of transforming the unknown ω^D into a sum ω + $d\omega$, in which ω is the precisely known angular velocity, in most cases it being a constant average value that depends on the speed of the respective actuator, which operates the element in question, while the new unknown obtained by the simple transformation with finite elements is $d\omega$, and its value is easily obtained from the corresponding equation, which in this way is linearized. In this way, the problem of the nonlinear dynamical system is solved directly, which is otherwise solved by using a Laplace or Fourier transform twice, or by using the Lagrange equations of the second kind. The advantages of the method presented and proposed in the paper are obvious. The real, nonlinear dynamic system is directly linearized, by using this new method with finite differences, the proposed equations are simple and direct, very precise, and easy to learn from a didactic point of view, but also simple to use by researchers. The proposed new systems (both new methods presented in the paper) have a high utility and high applicability, but the second method (with finite differences) is simpler and more direct, while the first proposed method has the great advantage of high generality so that it can be applied to any machine, robot, system, without prior preparation, where cg can take only the value 1 (system with the influence of gravitational forces) or 0 (system without the influence of gravitational forces);

$$\left\{\begin{array}{l}\left\{\begin{array}{l}N1=\omega_1^2\cdot(m_1\cdot S_1+m_2\cdot AB)\cdot[c_1\cdot(2\cdot\omega_2\cdot s_2-FV\cdot c_2)-s_1\cdot(2\cdot\omega_2\cdot c_2+FV\cdot s_2)]\\+m_2\cdot S_2\cdot\omega_2^2\cdot[c_2\cdot(2\cdot\omega_2\cdot s_2-FV\cdot c_2)-s_2\cdot(2\cdot\omega_2\cdot c_2+FV\cdot s_2)]\\+[(m_1+m_2)\cdot g\cdot cg+R_T]\cdot(2\cdot\omega_2\cdot c_2+FV\cdot s_2)\\ \\n1=(m_1\cdot S_1+m_2\cdot AB)\cdot[(2\cdot\omega_1\cdot c_1+FV\cdot s_1)\cdot(FV\cdot c_2-2\cdot\omega_2\cdot s_2)\\+(2\cdot\omega_1\cdot s_1-FV\cdot c_1)\cdot(2\cdot\omega_2\cdot c_2+FV\cdot s_2)]\\d\omega_1=\frac{N1}{n1}\\\omega_1^D=\omega_{1m}+d\omega_1\end{array}\right.\\ \left\{\begin{array}{l}N2=\omega_1^2\cdot(m_1\cdot S_1+m_2\cdot AB)\cdot[c_1\cdot(2\cdot\omega_1\cdot s_1-FV\cdot c_1)-s_1\cdot(2\cdot\omega_1\cdot c_1+FV\cdot s_1)]\\+m_2\cdot S_2\cdot\omega_2^2\cdot[c_2\cdot(2\cdot\omega_1\cdot s_1-FV\cdot c_1)-s_2\cdot(2\cdot\omega_1\cdot c_1+FV\cdot s_1)]\\+[(m_1+m_2)\cdot g\cdot cg+R_T]\cdot(2\cdot\omega_1\cdot c_1+FV\cdot s_1)\\ \\n2=m_2\cdot S_2\cdot[(2\cdot\omega_2\cdot c_2+FV\cdot s_2)\cdot(FV\cdot c_1-2\cdot\omega_1\cdot s_1)\\+(2\cdot\omega_2\cdot s_2-FV\cdot c_2)\cdot(2\cdot\omega_1\cdot c_1+FV\cdot s_1)]\\d\omega_2=\frac{N2}{n2}\\\omega_2^D=\omega_{2m}+d\omega_2\end{array}\right.\end{array}\right. \tag{5}$$

where: $s_1 = \sin\varphi_1;\ c_1 = \cos\varphi_1;\ s_2 = \sin\varphi_2;\ c_2 = \cos\varphi_2;\ S_1 = AS_1;\ S_2 = BS_2.$

Note: If one wants to eliminate the effect of gravitational forces then to the cg coefficient is assigned the value 0 instead of 1 (for example, when the robot, instead of operating in a vertical plane, works in a horizontal plane). FV is a velocity factor (see Section 5). Asthe second method presented in the paper, which has all the new elements, being an absolute novelty, has a lower generality, it can be applied to any important system by adapting equations (rewriting Equations (3) and (5)).

3. Results and Discussions

The original dynamic angular velocity w_{1d} [s^{-1}] can be seen in the graph in Figure 9, and the differences between it and the kinematic velocity w_1 [s^{-1}] are visible in the diagrams in Figure 10.

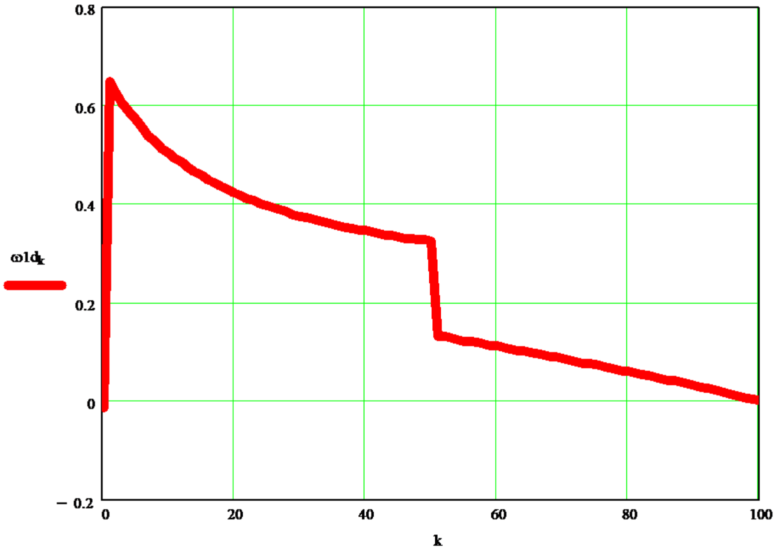

Figure 9. The original dynamic angular velocity w_{1d}.

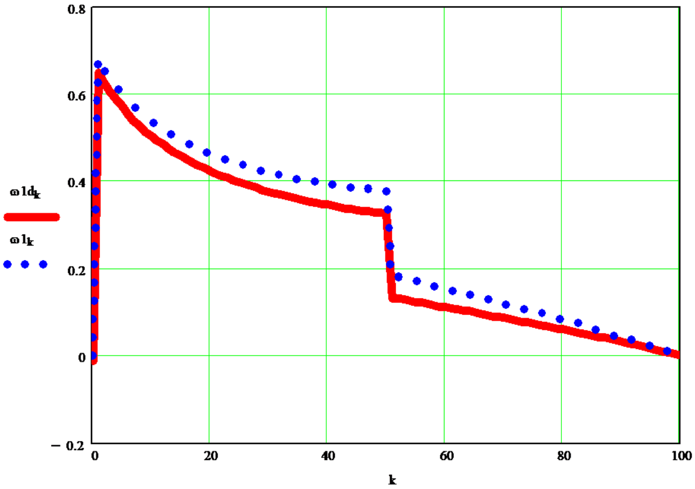

Figure 10. The differences between angular dynamic velocity w_{1d} and the angular kinematic velocity w_1.

The original dynamic angular velocity w_{2d} [s^{-1}] can be seen in the graph in Figure 11, and the differences between it and the kinematic velocity w_2 [s^{-1}] are visible in the diagrams in Figure 12.

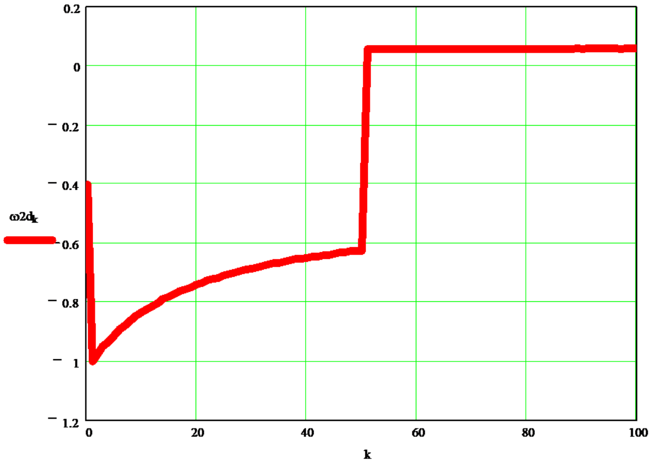

Figure 11. The original dynamic angular velocity w_{2d}.

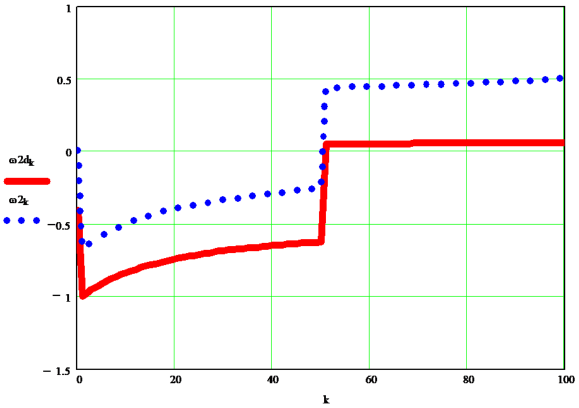

Figure 12. The differences between angular dynamic velocity w_{2d} and the angular kinematic velocity w_2.

For a comparison between the dynamic speed w_d obtained by the original method presented and the dynamic speed w_D obtained through the classically improved method, the diagrams in Figures 13 and 14 will be followed, for elements 1 and 2, respectively.

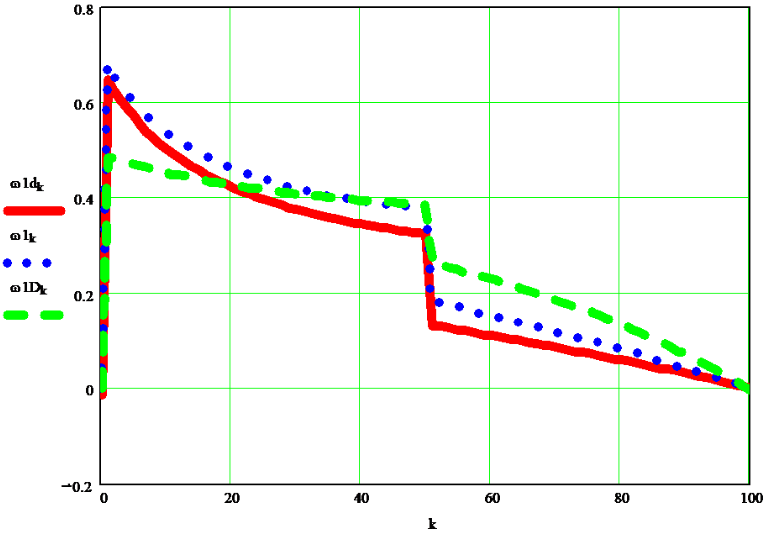

Figure 13. A comparison between the dynamic speed w_{1d} [s^{-1}] obtained by the original method presented, and the dynamic speed w_{1D} [s^{-1}] obtained through the classically improved method.

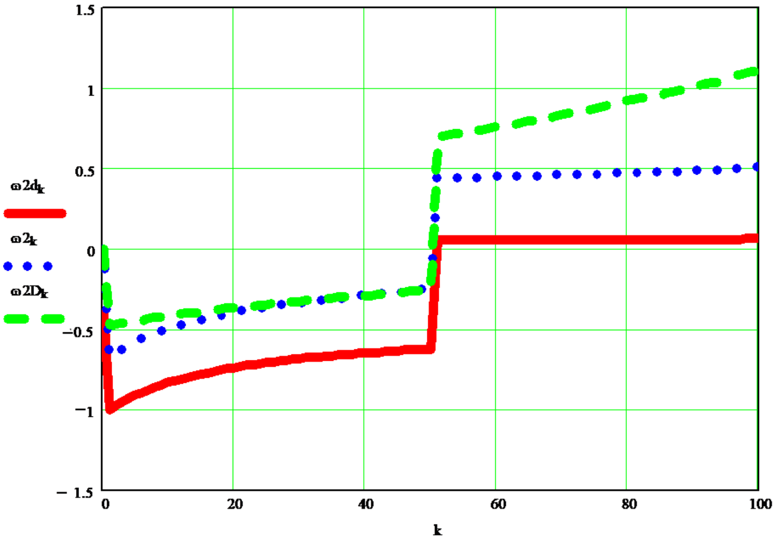

Figure 14. A comparison between the dynamic speed w_{2d} [s^{-1}] obtained by the original method presented, and the dynamic speed w_{2D} [s^{-1}] obtained through the classically improved method.

The classical method modified by the author uses the moments of mechanical or mass inertia reduced to the mobile element at which the calculation of the dynamic speed is made, reduced even in the torque that acts on the respective element. An older original calculation system was also adopted, for obtaining directly approximate dynamic angular velocities. A very exact method can be used here, even if it is approximate, presented by the author in the papers [38,39], but in this case, the calculations become extremely difficult and cumbersome, with the final differences compared to the first method presented in this article being small. All three results are close enough, even if they are obtained by three totally different calculation methods, and this specifies that the dynamics of the system is a normal, quiet one, based on the fact that the kinematic speeds imposed on the system were chosen by an original method designed to improve the operation of the robot; a method that aims to ensure that the kinematic parameters imposed (in the inverse kinematics) to the robot are all tracked and achieved in operation. If the kinematic speeds change, for example by adopting someone's constants, the dynamics obtained by both presented methods will be broken, and the real operation of the system will worsen. The fact that both methods presented give similar results indicates the correctness of both methods used in system dynamics.

4. First Original Proposed Dynamic Model Applied to a 2T6R Robot

The articulated robot from Figure 15 will be briefly presented, which in this situation is operated with two linear actuators (elements of variable length 3 and 4, respectively) instead of the two classic rotary actuators. In this way, the robot mechanism will have six moving elements instead of two; however, to facilitate the study the two linear actuators, formed each of two elements of constant length, will be considered as a single moving element of variable length, so that the six mobile elements of this robot will be reduced in the study to only four mobile elements.

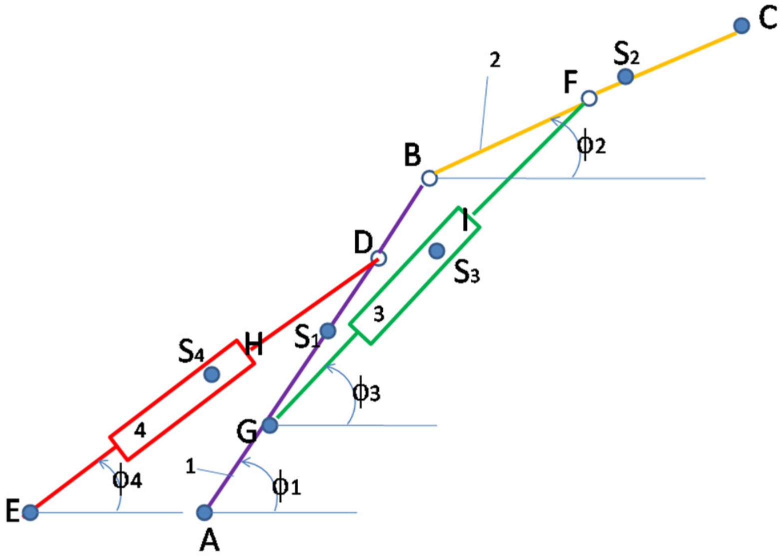

Figure 15. A simple articulated robot, operated with two linear actuators (a 6R2T robot).

The kinetic energy of the whole mechanism is conserved (Equation(6)) [38,39]:

$$\begin{cases} 2 \cdot E_{kinetic}^{(1,2,3,4)} = cons. = J_{S_1} \cdot \omega_1^2 + m_1 \cdot \left(\dot{x}_{S_1}^2 + \dot{y}_{S_1}^2\right) + J_{S_2} \cdot \omega_2^2 + m_2 \cdot \left(\dot{x}_{S_2}^2 + \dot{y}_{S_2}^2\right) + \\ J_{S_3} \cdot \omega_3^2 + m_3 \cdot \left(\dot{x}_{S_3}^2 + \dot{y}_{S_3}^2\right) + J_{S_4} \cdot \omega_4^2 + m_4 \cdot \left(\dot{x}_{S_4}^2 + \dot{y}_{S_4}^2\right) \end{cases} \quad (6)$$

$E_{kinetic}^{(1,2,3,4)}$ represents the kinetic energy of the whole machine, an element that is preserved; m_1 = mass of element 1; m_2 = mass of element 2; m_3 = mass of element 3; m_4 = mass of element 4; J_{S_1} is inertia mass of element 1 in the mass center; J_{S_2} is inertia mass of element 2 in the mass center; J_{S_3} is inertia mass of element 3 in the mass center; J_{S_4} is inertia mass of element 4 in the mass center; $\omega_1, \omega_2, \omega_3, \omega_4$ represents the rotational speed of the respective element (1, 2, 3, 4); $\dot{x}_{S_1}$ the velocity of the center of mass of element 1 (S_1) projected on the x-axis; $\dot{y}_{S_1}$ the velocity of the center of mass of element 1 (S_1) projected on the y-axis; $\dot{x}_{S_2}$ the velocity of the center of mass of element 2 (S_2) projected on the x-axis; $\dot{y}_{S_2}$ the velocity of the center of mass of element 2 (S_2) projected on the y-axis; $\dot{x}_{S_3}$ the velocity of the center of mass of element 3 (S_3) projected on the x-axis; $\dot{y}_{S_3}$ the velocity of the center of mass of element 3 (S_3) projected on the y-axis; $\dot{x}_{S_4}$ the velocity of the center of mass of element 4 (S_4) projected on the x-axis; $\dot{y}_{S_4}$ the velocity of the center of mass of element 4 (S_4) projected on the y-axis.

The second robot presented in the paper is in fact the first robot, operated differently, instead of rotary actuators, it has two linear actuators, which introduce into the robot system four more moving elements, two for each linear actuator, so that the configuration of the robot changes radically, and it is now being studied as if it were another system. Asthe first new method presented in the paper has a more general character, we applyit here to this second robot, it being practically similar to the one already written for the first robot but generalized by extension to several mobile elements. Instead of two mobile elements, the new robot has six, so as not to introduce so many extra mobile elements, we chose the situation in which each actuator is considered only as one additional mobile element, but of variable length, so that the two linear actuators introduce only two additional moving elements and the presented robotic system has a total of four moving elements. For this reason, the new equations presented (6)–(14) in "Dynamic Method 1" will be extended from two to four moving elements.

This is the first new dynamic method of the paper, which we will explain in detail here, and one starts with the kinetic energy of the entire robot (machine; $E_{kinetic}$) which is known to be conserved, but the equation used (6) is in fact that of twice the kinetic energy of the machine ($2 \cdot E_{kinetic}$); obviously if it is conserved (kinetic energy) thenits double is a constant throughout the operation of the machine, regardless of the position of the angles that position each moving element, and regardless of the actuators working speed (in this case being two linear actuators).

The originality of the dynamic method 1 proposed in the paper is how to determine the moments of mass (mechanical) inertia for each element, in order to reduce the mechanism to a single element (any of them) so that a dynamic model of the machine with an only one element can represent (replace) the dynamics of the whole machine (robot). Using successively Relation (6) (four times), for each element of the robot mechanism, one obtains the four equations that generate the moments of mechanical inertia (mass) reduced successively to each of the four elements (Equations (7)–(10)).

$$\begin{cases} 2 \cdot E_{kinetic}^{(1,2,3,4)} = J_A^{*(1)} \cdot \omega_1^2 \Rightarrow J_A^{*(1)} = J_{S_1} + m_1 \cdot \frac{\left(\dot{x}_{S_1}^2 + \dot{y}_{S_1}^2\right)}{\omega_1^2} + J_{S_2} \cdot \frac{\omega_2^2}{\omega_1^2} + m_2 \cdot \frac{\left(\dot{x}_{S_2}^2 + \dot{y}_{S_2}^2\right)}{\omega_1^2} + \\ J_{S_3} \cdot \frac{\omega_3^2}{\omega_1^2} + m_3 \cdot \frac{\left(\dot{x}_{S_3}^2 + \dot{y}_{S_3}^2\right)}{\omega_1^2} + J_{S_4} \cdot \frac{\omega_4^2}{\omega_1^2} + m_4 \cdot \frac{\left(\dot{x}_{S_4}^2 + \dot{y}_{S_4}^2\right)}{\omega_1^2} \end{cases} \tag{7}$$

$$\begin{cases} 2 \cdot E_{kinetic}^{(1,2,3,4)} = J_B^{*(2)} \cdot \omega_2^2 \Rightarrow J_B^{*(2)} = J_{S_1} \cdot \frac{\omega_1^2}{\omega_2^2} + m_1 \cdot \frac{\left(\dot{x}_{S_1}^2 + \dot{y}_{S_1}^2\right)}{\omega_2^2} + J_{S_2} + m_2 \cdot \frac{\left(\dot{x}_{S_2}^2 + \dot{y}_{S_2}^2\right)}{\omega_2^2} + \\ J_{S_3} \cdot \frac{\omega_3^2}{\omega_2^2} + m_3 \cdot \frac{\left(\dot{x}_{S_3}^2 + \dot{y}_{S_3}^2\right)}{\omega_2^2} + J_{S_4} \cdot \frac{\omega_4^2}{\omega_2^2} + m_4 \cdot \frac{\left(\dot{x}_{S_4}^2 + \dot{y}_{S_4}^2\right)}{\omega_2^2} \end{cases} \tag{8}$$

$$\begin{cases} 2 \cdot E_{kinetic}^{(1,2,3,4)} = J_G^{*(3)} \cdot \omega_3^2 \Rightarrow J_G^{*(3)} = J_{S_1} \cdot \frac{\omega_1^2}{\omega_3^2} + m_1 \cdot \frac{\left(\dot{x}_{S_1}^2 + \dot{y}_{S_1}^2\right)}{\omega_3^2} + J_{S_2} \cdot \frac{\omega_2^2}{\omega_3^2} + m_2 \cdot \frac{\left(\dot{x}_{S_2}^2 + \dot{y}_{S_2}^2\right)}{\omega_3^2} + \\ J_{S_3} + m_3 \cdot \frac{\left(\dot{x}_{S_3}^2 + \dot{y}_{S_3}^2\right)}{\omega_3^2} + J_{S_4} \cdot \frac{\omega_4^2}{\omega_3^2} + m_4 \cdot \frac{\left(\dot{x}_{S_4}^2 + \dot{y}_{S_4}^2\right)}{\omega_3^2} \end{cases} \tag{9}$$

$$\begin{cases} 2 \cdot E_{kinetic}^{(1,2,3,4)} = J_E^{*(4)} \cdot \omega_4^2 \Rightarrow J_E^{*(4)} = J_{S_1} \cdot \frac{\omega_1^2}{\omega_4^2} + m_1 \cdot \frac{\left(\dot{x}_{S_1}^2 + \dot{y}_{S_1}^2\right)}{\omega_4^2} + J_{S_2} \cdot \frac{\omega_2^2}{\omega_4^2} + m_2 \cdot \frac{\left(\dot{x}_{S_2}^2 + \dot{y}_{S_2}^2\right)}{\omega_4^2} + \\ J_{S_3} \cdot \frac{\omega_3^2}{\omega_4^2} + m_3 \cdot \frac{\left(\dot{x}_{S_3}^2 + \dot{y}_{S_3}^2\right)}{\omega_4^2} + J_{S_4} + m_4 \cdot \frac{\left(\dot{x}_{S_4}^2 + \dot{y}_{S_4}^2\right)}{\omega_4^2} \end{cases} \tag{10}$$

This method has long been known and often applied in the Romanian school of mechanisms [45,46], but only for a single mobile element, namely the motor, and only to machines with a single degree of mobility, after which was used the Lagrange differential equation of kind one for solving the nonlinear, dynamic system, with a single input with two unknowns, ω, and ε. The proposed new method has generalized the determination of a dynamic inertia mass of the whole mechanism for (reduced to) each mobile element of it, regardless of whether it is a direct motorized one or not, in order to obtain, at each element, a basic dynamic parameter on the basis of which the dynamics of that element of the machine (robot)can later be described. In the case of the robot presented with four moving elements, it results in four rotating masses ($J_A^{*(1)}$, $J_B^{*(2)}$, $J_G^{*(3)}$,$J_E^{*(4)}$), each reduced to one of the four moving elements of the robot (Equations (7)–(10)). In order to further solve the dynamics of each moving element of the machine, different methods can be used. A classic method would be to use the Lagrange dynamic equation of kind one, which the author has already described and applied to other types of mechanisms [39], with an original solution by the finite element method, somewhat similar to the one proposed in the second presented dynamic method of the paper. At this stage, it was decided to use a simpler method, presented for other machines in other works [38,39], which directly solves the dynamics of the machine without using differential equations. This method applies simply, directly, to each moving element (Equations (11)–(14)), to solve the dynamic of any machine mobile element. Each rotating mass of the whole machine reduced to a certain mobile element carries in it the dynamic characteristic of the whole machine reduced to that element and depends both on the positions occupied by the machine during operation and on the speeds of the actuators.

Below are the dynamic values of the angular velocities of each element in relation to Equations (11)–(14), which is the second original element of the dynamic method 1 proposed in the paper [38,39]:

$$\omega_1^D = \sqrt{J_{Amed}^*} \cdot \frac{\omega_1}{\sqrt{J_A^*}} \tag{11}$$

$$\omega_2^D = \sqrt{J_{Bmed}^*} \cdot \frac{\omega_2}{\sqrt{J_B^*}} \tag{12}$$

$$\omega_3^D = \sqrt{J_{Gmed}^*} \cdot \frac{\omega_3}{\sqrt{J_G^*}} \tag{13}$$

$$\omega_4^D = \sqrt{J_{Emed}^*} \cdot \frac{\omega_4}{\sqrt{J_E^*}} \tag{14}$$

This fast and accurate, but especially simple, method of determining the dynamics of any machine is based on a classic idea of conserving the kinetic energy of the whole machine applied twice, in two steps (through two passes).

For the second step (applied here in particular to the robot with two linear actuators), the conservation of the kinetic energy of the machine, written in the form of Equation (15), is also used.

$$\begin{cases} 2 \cdot E_{kinetic} = const. = J^* \cdot \omega^{*2} = J_{Max}^* \cdot \omega_{\min}^2 = J_{\min}^* \cdot \omega_{Max}^2 = J_{med}^* \cdot \omega_{med}^2 \Rightarrow \\ \Rightarrow J^* \cdot \omega^{*2} = J_{med}^* \cdot \omega_{med}^2 \Rightarrow \omega^{*2} = \frac{J_{med}^*}{J^*} \cdot \omega_{med}^2; \ or \ \omega^D \equiv \omega^* = \sqrt{\frac{J_{med}^*}{J^*}} \cdot \omega_{med} \end{cases} \tag{15}$$

The absolutely original Equation (15) is easily applied to any dynamic system, the machine (robot) if it is known for any desired element of the machine, but which also has rotational motion, its average kinematic rotational speed ω_m, and the rotational mass of the whole machine reduced to the respective element J^*. Thus, the reduced angular velocity (dynamics) ω^* = ω^D of the respective element is easily determined, provided that the average value of J^*, i.e., J_{med}^* is also found.

The proposed new method has a strong general character (the one most desired aspect), a high degree of applicability, ease of understanding and development, simplicity, efficiency, and directness. With its help, the real (dynamic) angular velocities of the machine can be determined with great precision, for any of its elements, in the case proposed for any of the 4 elements of the 6R2T robot presented (in Figure 16). The dynamic speeds of the discussed 2T6R robot obtained with the help of the new proposed Equations (11)–(14), will be presented in Figures 16–19, (in each of them the dynamic angular speed obtained iscompared with the static one given).

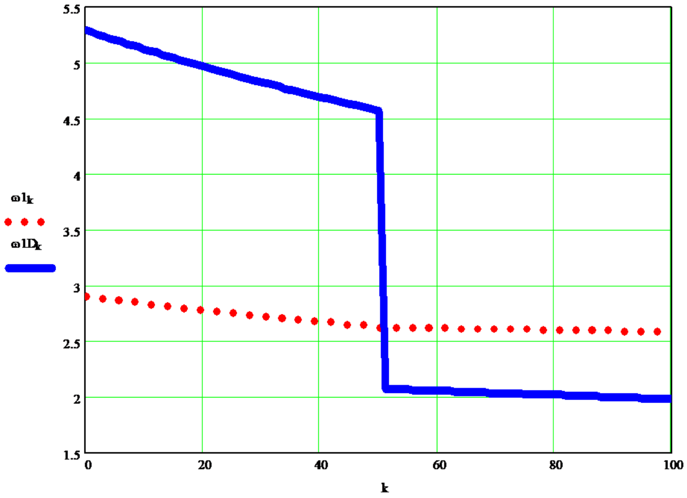

Figure 16. The dynamic angular speed of element 1 of the articulated robot, compared to the input (kinematics) considered speed (to a 6R2T robot).

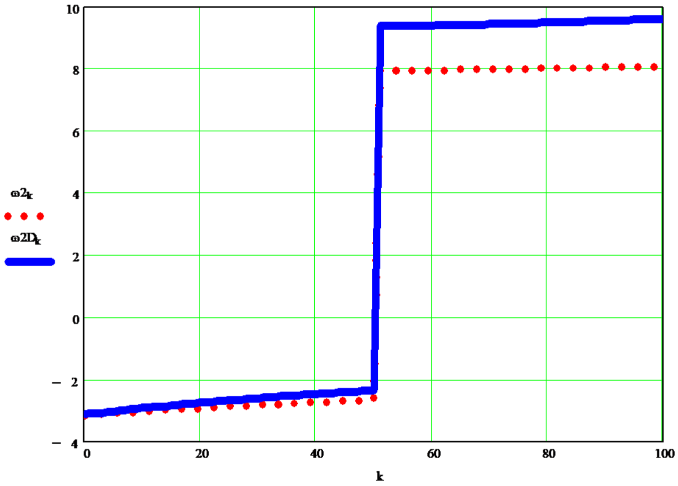

Figure 17. The dynamic angular speed of element 2 of the articulated robot, compared to the input (kinematics) considered speed (to a 6R2T robot).

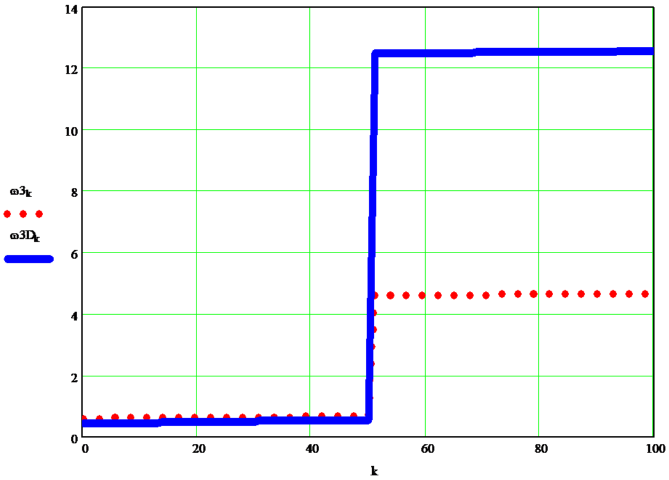

Figure 18. The dynamic angular speed of element 3 of the articulated robot, compared to the input (kinematics) considered speed (to a 6R2T robot).

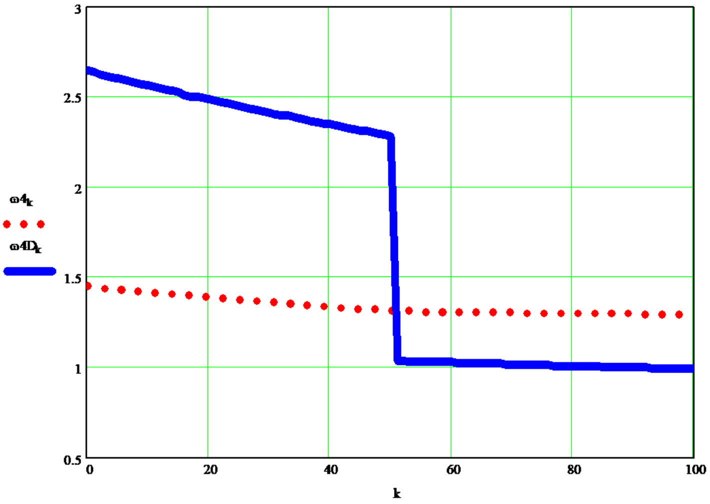

Figure 19. The dynamic angular speed of element 4 of the articulated robot, compared to the input (kinematics) considered speed (to a 6R2T robot).

The second original dynamic method proposed in the paper and presented briefly has the advantage of taking into account all the forces acting in the machine, but it is a method with a weak general character, requiring particular study for each case considered so that the calculation relations presented in a premiere have a character limited only to the mechanism of the simple articulated robot with two moving elements actuated with two rotary actuators. For another drive or another type of robot or machine, it is necessary to restore the calculation relations, so the method is difficult and has great importance only when you want to determine very precisely the dynamic parameters of an important machine, such as the articulated robot considered because it is being implemented more and more in today's industries.

5. Speed Command and Control

The command and control of speeds are achievedby the author by two different methods, both with real advantages. The first method of command and control of speeds refers to keeping the engine speeds constant so that their accelerations are zero or very low, and the movement of the elements of the machine, the robot, areperformedquickly, with good dynamics, and are easy to control. As the motor speeds of a robotic machine, or other machines, even if they are generally constant, must change the sign depending on how the actuator element moves (for both linear and circular displacement), the constant engine speeds are taken but with a changeable sign, the sign being given by the positions occupied successively by the mechanism. In this way, the intelligent command and control of the speeds are realized, with their automation. The simple sequence used to automate angular velocities in a robot with two motors can be traced in Relations (16), it iswritten using the "if logic" function.

$$\begin{cases} w1 := 5 \\ w2 := 3 \\ \omega 1_m := if[m < 1, 0, w1 \cdot sign(\phi 1_m - \phi 1_{m-1})] \\ \omega 2_m := if[m < 1, 0, w2 \cdot sign(\phi 2_m - \phi 2_{m-1})] \end{cases} \tag{16}$$

The modulus of a speed (w) is constant, and its sign is given by the finite displacements already known (in inverse kinematics these displacements areimposed by the necessary position of the endeffector, they aresolved most easily with the help of "if logic" functions).

The second method of command and control of speeds refers to the application of permanently variable speeds to the actuators, these speeds being those obtained from the approximate numerical derivation of the already known robot positions imposed (by the inverse kinematics). For the simplest numerical derivation, the finite element known method obtained by developing the position functions in the Taylor series was considered (Equation (17)).

$$\begin{cases} FV := 50 \\ \omega 1_k := if[k < 1, 0, FV \cdot (\phi 1_k - \phi 1_{k-1})] \\ \omega 2_k := if[k < 1, 0, FV \cdot (\phi 2_k - \phi 2_{k-1})] \end{cases} \tag{17}$$

The reduced speeds of the actuators are obtained based on the approximate numerical derivation of the positions (by developing the Taylor functions), they are variable in size and sign depending on the imposed and known positions (from inverse kinematics), and the absolute speeds are obtained by amplifying the reduced ones with a constant "velocity factor" (FV), chosen according to the time required to complete the respective race. Even though the two methods have similar sequences, they are completely different. In the first method, the finite positional differences impose only their sign and the speed of the respective actuator, the speed of the module being a chosen constant w, and in the second situation (method 2 presented), the finite differences of the positions impose both their sign and their value absolute of the respective actuator, which will result in this variable mode as well as the module, and which will be obtained from the reduced speed given by the finite positions by the amplification with a factor of speed (or time), FV.

6. Future Works

The dynamic model (Sections 2.1 and 4) can be easily resumed and presented on various other machines, including robotic systems, due to its generality, and simple and direct applicability. The author intends, in future works, to analyze and customize the second new dynamic method proposed in the paper (Section 2.2) for other types of machines and or robots, depending on their industrial importance. The author also would also like to continue to try to use the Matlab and Simulink programs instead of (or with) Mathcad simulation, in order to directly determine the command and dynamic control of mobile mechanical systems, with their automation.

7. Conclusions

The stability of moving systems today is due to the increasingly high-performance controllers, based on micro-, and recently, nanoprocessors, in the command and control software imposed on the system, usually a discrete or even continuous, robust PID controller, which makes a closed-loop with the help of the software and the constants determined based on the dynamics of the respective system, but more and more it is switched to mechanical systems with mixed control, in a closed-loop plus feedforward system. For this reason (for the very precise establishment of the operating constants and for the adjustment of the PID controller), it is necessary to know the dynamic parameters of the machine in question, a fact for which the present work is extremely important for determining the dynamic parameters of any mobile element of the respective machine, by using both methods presented in the paper.

The two new methods presented in the paper, both theoretically and simulated in Mathcad, are original, easy to understand and use, the first having a very strong generalized character, with the second, having the possibility of generating high precision dynamic solutions, being a method customized to the mechanism, which takes into account all the forces acting on it (but such a conclusion cannot be verified without experiment). The author wants to further implement these methods in other machines and robots and simulate them through Matlab software and its Simulink extension.

A special problem of these dynamic aspects, stability, command, control, and automation, is the command and control of speeds, depending on whether or not the system acquires intrinsic stability, increasing its ability to follow the trajectory imposed on the endeffector by inverse kinematics. The paper presented in Section 5 two new original methods of speed control of a mobile mechanical system, from the beginning, by imposing on all actuators certain speeds, constants, or variables according to intrinsic laws, deduced with the help of Taylor series decompositions of the speed control functions of the machine actuators.

It should be noted here that the novelties of the dynamic system can facilitate the work of a programmer of a controller, for example, a PID, by the fact that the new dynamics imposed by method 1 presented in the paper, can directly solve nonlinear systems, thus eliminating the classic methods where: first, make a linearization, work with the new linear function, and finally delimit the results obtained. The use of Fuzzy methods required an initial fusion and, finally, the defuzzification of all output data, similar toother neuro methods, or genetic algorithms. Method 1 presented in the paper automates this process, making the programmer's job easier. On the other hand, the method presented allows the use of models of a continuous PI-PID controller, in which all parameters are easy to find, while in classical dynamic methods strongly nonlinear systems of this kind involve the use of a discreet PID controller which is difficult to apply and quantify.

The paper studies two mechanisms corresponding to the same articulated robot with two moving elements, operated in two different situations: (1) with two rotary actuators; (2) with two linear actuators.

Funding: This research has not yet received external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The author declares no conflict of interest.

Nomenclature

m is the mass of an element in linear motion and is measured in kg; J is the rotational mass of an element (known as the moment of mass or mechanical inertia) and is measured in kg·m^2; ϕ or φ is a position parameter that defines the position angle of an element relative to the abscissa axis and is measured in rad; ω is the angular velocity with which a certain element rotates and is measured in Hertz or s^{-1}; J_{S_3} is the mass or mechanical inertia of the mobile element 3 determined around the axis of rotation passing through the center of mass noted with S_3 (kg.m^2); J^* is the mass or mechanical inertia of machine (of all mechanism, or robot) reduced to one single element (an element which must also have a rotational motion in the considered plane) [kg·m^2]; $\dot{x}_{S_3}$ the velocity of the center of mass of element 3 (S_3) projected on the x-axis [m/s]; $\dot{y}_{S_3}$ the velocity of the center of mass of the mobile element 3 (S_3) projected on the y-axis [m/s].

Appendix A

Reverse kinematics on a 2R planar robot.

(1) Establishing the desired coordinates of the trajectory of the endeffector C

$$\begin{aligned}
&AB := 6 \;\; BC := 5 \;\; XA := 0 \;\; AS1 := \tfrac{AB}{2} \;\; BS2 := \tfrac{BC}{2} \\
&X0 := 10 \;\; Y0 := 3 \;\; k := 0..100 \;\; c := 0.05 \;\; S1 := AS1 \;\; S2 := BS2 \\
&XC_k := if(k \leq 50, X0 - c \cdot k, X0 - c \cdot 50) \\
&YC := if[k \leq 50, Y0, Y0 + c \cdot (k - 50)]
\end{aligned}$$

Reverse kinematics on a planar 2R robot.

(2) Determining the scalar parameters of the kinematic coupling B

$$\begin{aligned}
&XB := 1 \; YB := 1 \; Given \\
&(XA - XB)^2 + (YA - YB)^2 = AB^2 \\
&(XB - XC_k)^2 + (YB - YC_k)^2 = BC^2 \\
&sol_k := Find(XB, YB) \\
&\begin{pmatrix} XB_k \\ YB_k \end{pmatrix} := sol_k
\end{aligned}$$

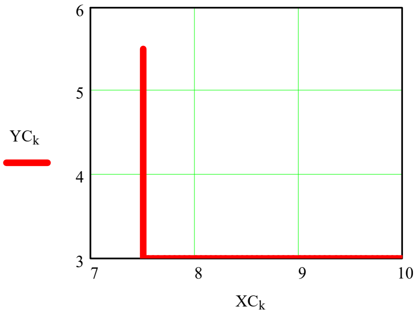

Figure A1. The hodograph of the end effector point C is the trajectory described by the end effector C.

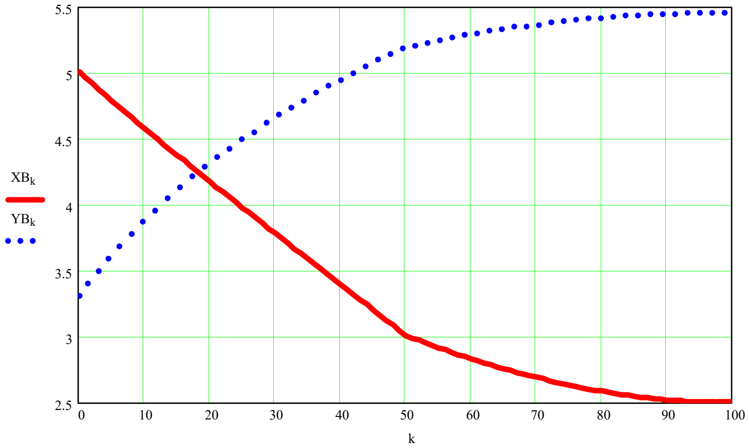

Figure A2. Scalar coordinates of kinematic coupling B as a function of the independent variable k.

Reverse kinematics on a 2R plane robot.

(3) Determining the angles Fi1 and Fi2

$$c1_k := \frac{XB_k - XA}{AB} \quad s1_k := \frac{YB_k - YA}{AB} \quad \phi1_k := sign(s1_k) \cdot a\cos(c1_k) \quad \phi10_k := \phi1_k \cdot \frac{180}{\pi}$$
$$c2_k := \frac{XC_k - XB_k}{BC} \quad s2_k := \frac{YC_k - YB_k}{BC} \quad \phi2_k := sign(s2_k) \cdot a\cos(c2_k) \quad \phi20_k := \phi2_k \cdot \frac{180}{\pi}$$

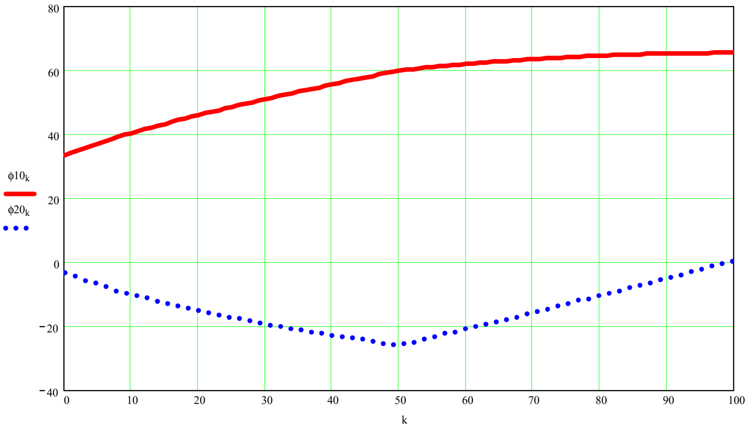

Figure A3. Values of angles ϕ1 and ϕ2 (in hexadecimal degrees) as a function of the independent variable k.

Reverse kinematics on a 2R plane robot.

(4) Determination of linear and angular velocities and accelerations

$$\begin{cases} FV := 50 \\ \omega1_k := if[k < 1, 0, FV \cdot (\phi1_k - \phi1_{k-1})] \\ \omega2_k := if[k < 1, 0, FV \cdot (\phi2_k - \phi2_{k-1})] \end{cases}$$

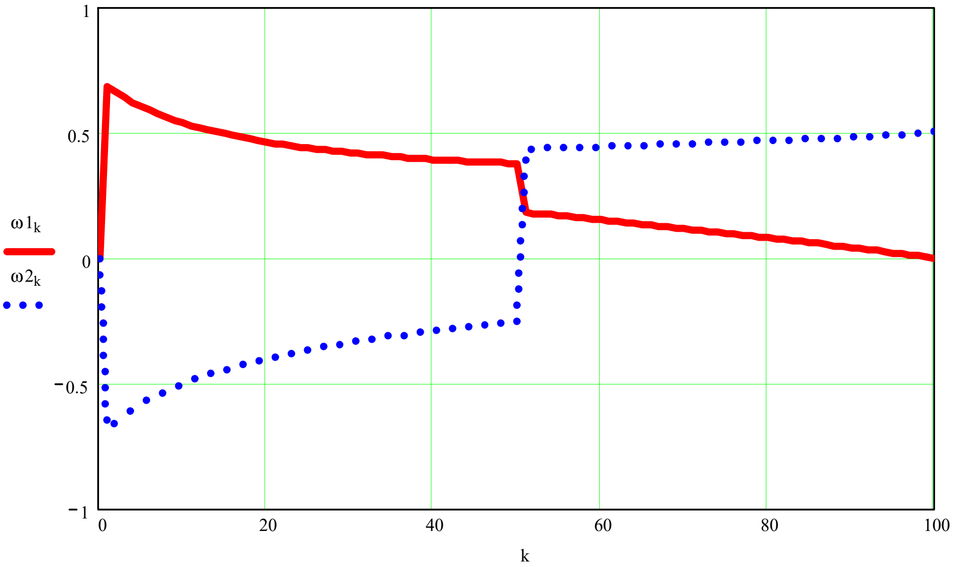

Figure A4. Values of angular velocities $\omega1$ and $\omega2$ (in hertz) as a function of the independent variable k.

$$\begin{cases} \varepsilon1_k := if[k < 1, 0, FV \cdot (\omega1_k - \omega1_{k-1})] \\ \varepsilon2_k := if[k < 1, 0, FV \cdot (\omega2_k - \omega2_{k-1})] \end{cases}$$

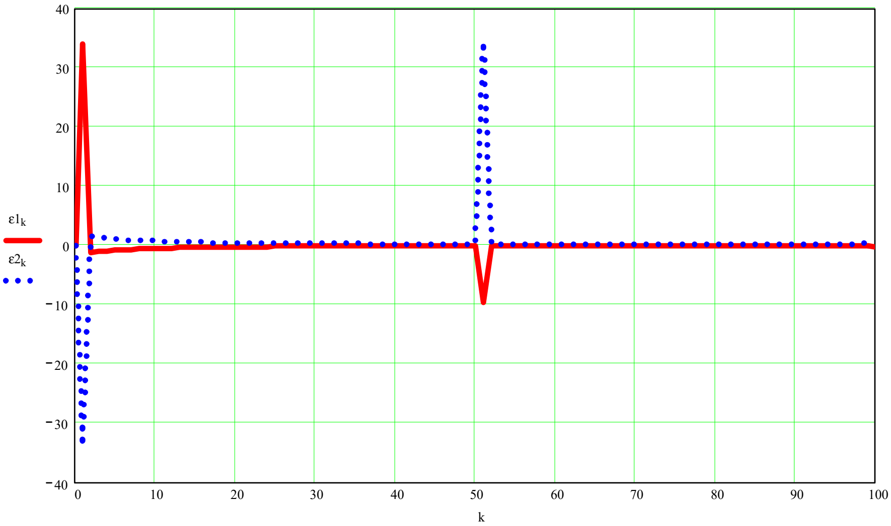

Figure A5. Values of angular accelerations $\varepsilon1$ and $\varepsilon2$ (in hertz^2) as a function of the independent variable k.

$$XB1_k := -AB \cdot \sin(\phi1_k) \cdot \omega1_k \;\; YB1_k := AB \cdot \cos(\phi1_k) \cdot \omega1_k \;\; B1_k := \sqrt{(XB1_k)^2 + (YB1_k)^2}$$
$$XB2_k := -AB \cdot \cos(\phi1_k) \cdot (\omega1_k)^2 - AB \cdot \sin(\phi1_k) \cdot \varepsilon1_k$$
$$YB2_k := -AB \cdot \sin(\phi1_k) \cdot (\omega1_k)^2 + AB \cdot \cos(\phi1_k) \cdot \varepsilon1_k \;\; B2_k := \sqrt{(XB2_k)^2 + (YB2_k)^2}$$

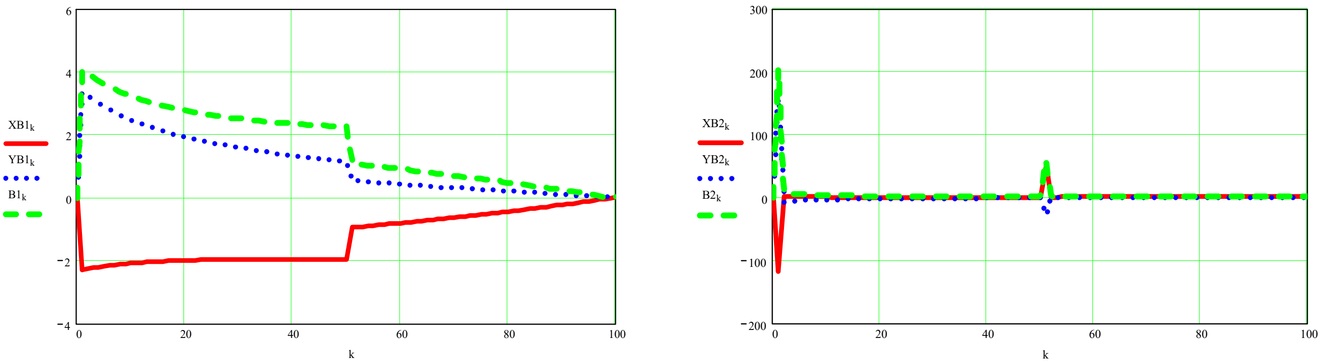

Figure A6. On the left side are the scalar coordinates of the speed of point B (red and blue) and its absolute speed (green); on the right the graphs of the accelerations of the same point B.

$$XC1_k := XB1_k - BC \cdot \sin(\phi 2_k) \cdot \omega 2_k \;\; YC1_k := YB1_k + BC \cdot \cos(\phi 2_k) \cdot \omega 2_k \;\; C1_k := \sqrt{(XC1_k)^2 + (YC1_k)^2}$$
$$XC2_k := XB2_k - BC \cdot \cos(\phi 2_k) \cdot (\omega 2_k)^2 - BC \cdot \sin(\phi 2_k) \cdot \varepsilon 2_k$$
$$YC2_k := YB2_k - BC \cdot \sin(\phi 2_k) \cdot (\omega 2_k)^2 + BC \cdot \cos(\phi 2_k) \cdot \varepsilon 2_k \;\; C2_k := \sqrt{(XC2_k)^2 + (YC2_k)^2}$$

Reverse kinematics on a 2R plane robot.

(5) Determining the positions, speeds, and accelerations of the centers of mass S1 and S2

$$XS1_k := \frac{AB}{2} \cdot \cos(\phi 1_k) \;\; YS1_k := \frac{AB}{2} \cdot \sin(\phi 1_k)$$
$$XS11_k := \frac{-AB}{2} \cdot \sin(\phi 1_k) \cdot \omega 1_k \;\; YS11_k := \frac{AB}{2} \cdot \cos(\phi 1_k) \cdot \omega 1_k \;\; S11_k := \sqrt{(XS11_k)^2 + (YS11_k)^2}$$
$$XS12_k := \frac{-AB}{2} \cdot \cos(\phi 1_k) \cdot (\omega 1_k)^2 - \frac{AB}{2} \cdot \sin(\phi 1_k) \cdot \varepsilon 1_k \;\; YS12_k := \frac{-AB}{2} \cdot \sin(\phi 1_k) \cdot (\omega 1_k)^2 + \frac{AB}{2} \cdot \cos(\phi 1_k) \cdot \varepsilon 1_k$$
$$S12_k := \sqrt{(XS12_k)^2 + (YS12_k)^2}$$
$$XS2_k := XB_k + \frac{BC}{2} \cdot \cos(\phi 2_k) \;\; YS2_k := YB_k + \frac{BC}{2} \cdot \sin(\phi 2_k)$$
$$XS21_k := XB1_k - \frac{BC}{2} \cdot \sin(\phi 2_k) \cdot \omega 2_k \;\; YS21_k := YB1_k + \frac{BC}{2} \cdot \cos(\phi 2_k) \cdot \omega 2 \;\; S21_k := \sqrt{(XS21_k)^2 + (YS21_k)^2}$$
$$XS22_k := XB2_k - \frac{BC}{2} \cdot \cos(\phi 2_k) \cdot (\omega 2_k)^2 - \frac{BC}{2} \cdot \sin(\phi 2_k) \cdot \varepsilon 2_k$$
$$YS22_k := YB2_k - \frac{BC}{2} \cdot \sin(\phi 2_k) \cdot (\omega 2_k)^2 + \frac{BC}{2} \cdot \cos(\phi 2_k) \cdot \varepsilon 2_k \;\; S22_k := \sqrt{(XS22_k)^2 + (YS22_k)^2}$$

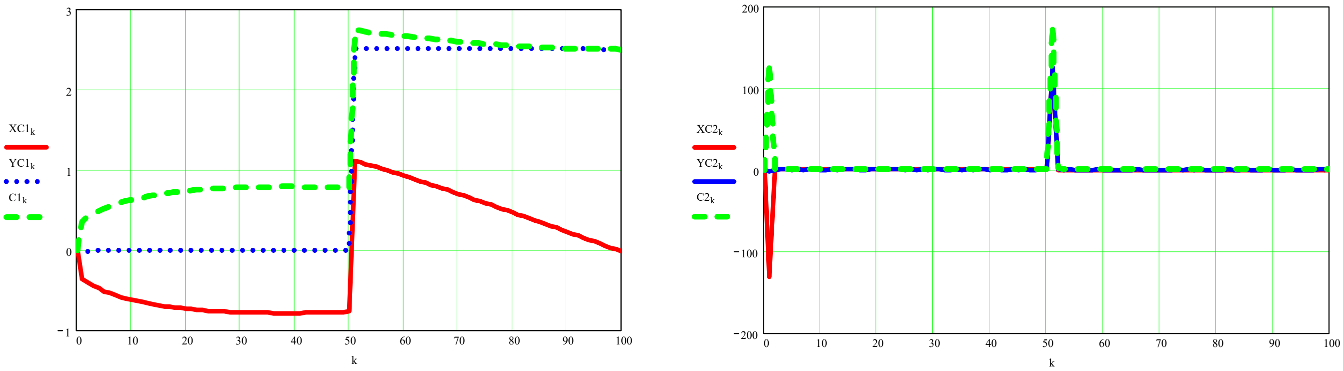

Figure A7. On the left side are the scalar coordinates of the speed of point C (red and blue) and its absolute speed (green); on the right the graphs of the accelerations of the same point C.

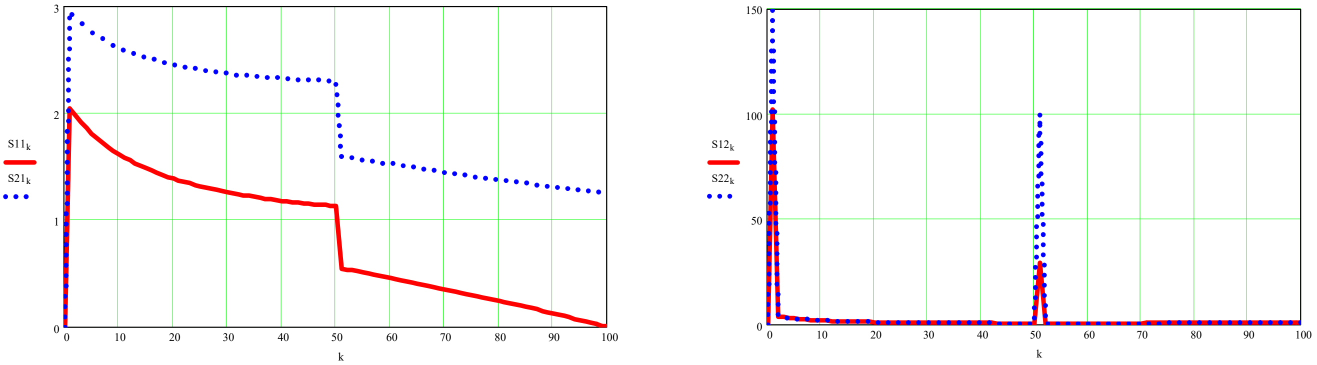

Figure A8. On the left are the absolute velocities of the center of mass of element 1 (red) and that of element 2 (blue); on the right the graphs of the absolute accelerations of the same points S1 and S2 respectively.

(6) Inverse kinematics in a 2R robot plane (6) Kinematics (forces and motor moments in the mechanism are determined)

$$RT := 500\ \ m1 := 3\cdot AB\ \ m2 := 3\cdot BC\ \ JS1 := m1\cdot\tfrac{AB}{12}\ \ JS2 := m2\cdot\tfrac{BC}{12}\ \ g := 9.81\ \ cg := 1$$
$$FixS1_k := -m1\cdot XS12_k\ \ FiyS1_k := -m1\cdot YS12_k - m1\cdot g\cdot cg\ \ Mi1_k := -JS1\cdot\varepsilon 1_k$$
$$FixS2_k := -m2\cdot XS22_k\ \ FiyS2_k := -m2\cdot YS22_k - m2\cdot g\cdot cg\ \ Mi2_k := -JS2\cdot\varepsilon 2_k$$
$$FiS1_k := \sqrt{(FixS1_k)^2 + (FiyS1_k)^2}\ \ FiS2_k := \sqrt{(FixS2_k)^2 + (FiyS2_k)^2}$$
$$M2_k := FixS2_k\cdot(YS2_k - YB_k) + FiyS2_k\cdot(XB_k - XS2_k) - Mi2_k + RT\cdot(XC_k - XB_k)$$
$$RBX_k := -FixS2_k\ RBY_k := RT - FiyS2_k\ \ RB_k := \sqrt{(RBX_k)^2 + (RBY_k)^2}$$
$$M1_k := FiS1_k\cdot(YS1_k - YA) + FiyS1_k\cdot(XA - XS1_k) - Mi1_k + RBX_k\cdot(YA - YB_k) + RBY_k\cdot(XB_k - XA)$$
$$RAX_k := RBX_k - FixS1_k\ \ RAY_k := RBY_k - FiyS1_k\ \ RA_k := \sqrt{(RAX_k)^2 + (RAY_k)^2}$$
$$Mm1_k := M1_k\ \ Mm2_k := M2_k$$

(7) Approximate classical dynamics (variable angular velocities and variable angular accelerations are determined; variation is due to inertial forces or variation in the masses of moving elements), Section 2.1.

$$med(\omega 1) := \frac{\max(\omega 1)+\min(\omega 1)}{2}$$
$$JAred_k := JS1 + \frac{m1\cdot[(XS11_k)^2+(YS11_k)^2]+JS2\cdot(\omega 2_k)^2+m2\cdot[(XS21_k)^2+(YS21_k)^2]}{(med(\omega 1))^2}$$
$$JAredm := \frac{\max(JAred)+\min(JAred)}{2}$$
$$\omega 1D_k := \sqrt{JAredm}\cdot\frac{\omega 1_k}{\sqrt{JAred_k}}\ \varepsilon 1D_k := if[k<1, 0, (\omega 1D_k - \omega 1D_{k-1})\cdot FV]$$

$$med(\omega 2) := \frac{\max(\omega 2)+\min(\omega 2)}{2}$$
$$JBred_k := JS2 + \frac{m1\cdot[(XS11_k)^2+(YS11_k)^2]+JS1\cdot(\omega 1_k)^2+m2\cdot[(XS21_k)^2+(YS21_k)^2]}{(med(\omega 2))^2}$$
$$JBredm := \frac{\max(JBred)+\min(JBred)}{2}$$
$$\omega 2D_k := \sqrt{JBredm}\cdot\frac{\omega 2_k}{\sqrt{JBred_k}}\ \varepsilon 2D_k := if[k<1, 0, (\omega 2D_k - \omega 2D_{k-1})\cdot FV]$$

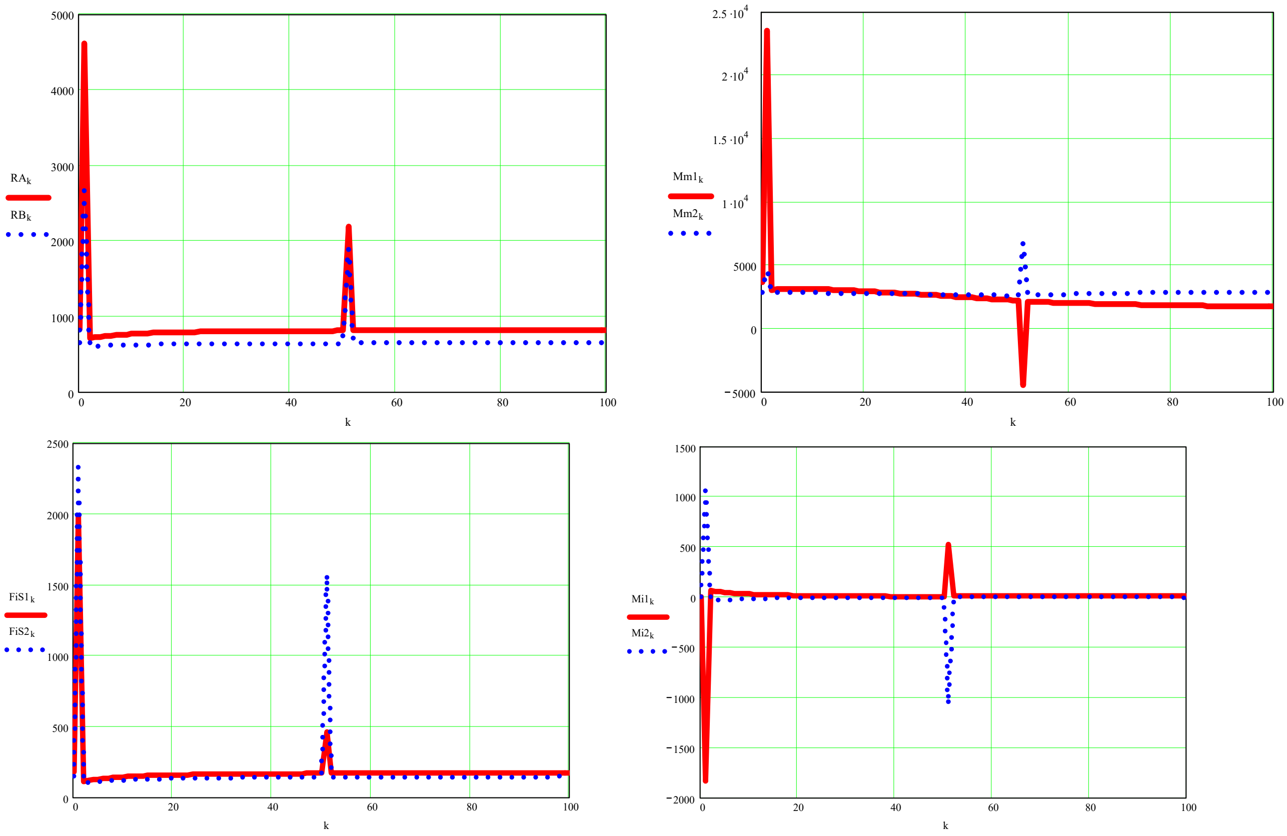

Figure A9. On the left are represented above the forces in the kinematic couples A (red) and B (blue) and below the inertial forces on element 1 (red) and 2 (blue), and on the right are the graphs of the motor moments at the top (red for element 1 and blue for element 2) and those of moments of inertia at the bottom.

$$XB1D_k := -AB\cdot\sin(\phi1_k)\cdot\omega1D_k\ YB1D_k := AB\cdot\cos(\phi1_k)\cdot\omega1D_k\ \ B1D_k := \sqrt{(XB1D_k)^2+(YB1D_k)^2}$$
$$XB2D_k := -AB\cdot\cos(\phi1_k)\cdot(\omega1D_k)^2 - AB\cdot\sin(\phi1_k)\cdot\varepsilon1D_k$$
$$YB2D_k := -AB\cdot\sin(\phi1_k)\cdot(\omega1D_k)^2 + AB\cdot\cos(\phi1_k)\cdot\varepsilon1D_k\ \ B2D_k := \sqrt{(XB2D_k)^2+(YB2D_k)^2}$$

$$XC1D_k := XB1D_k - BC\cdot\sin(\phi2_k)\cdot\omega2D_k\ \ YC1D_k := YB1D_k + BC\cdot\cos(\phi2_k)\cdot\omega2D_k\ \ C1D_k := \sqrt{(XC1D_k)^2+(YC1D_k)^2}$$
$$XC2D_k := XB2D_k - BC\cdot\cos(\phi2_k)\cdot(\omega2D_k)^2 - BC\cdot\sin(\phi2_k)\cdot\varepsilon2D_k$$
$$YC2D_k := YB2D_k - BC\cdot\sin(\phi2_k)\cdot(\omega2D_k)^2 + BC\cdot\cos(\phi2_k)\cdot\varepsilon2D_k\ \ C2D_k := \sqrt{(XC2D_k)^2+(YC2D_k)^2}$$

$$XS11D_k := \frac{-AB}{2}\cdot\sin(\phi1_k)\cdot\omega1D_k\ \ YS11D_k := \frac{AB}{2}\cdot\cos(\phi1_k)\cdot\omega1D_k\ \ S11D_k := \sqrt{(XS11D_k)^2+(YS11D_k)^2}$$
$$XS12D_k := \frac{-AB}{2}\cdot\cos(\phi1_k)\cdot(\omega1D_k)^2 - \frac{AB}{2}\cdot\sin(\phi1_k)\cdot\varepsilon1D_k\ \ YS12D_k := \frac{-AB}{2}\cdot\sin(\phi1_k)\cdot(\omega1D_k)^2 + \frac{AB}{2}\cdot\cos(\phi1_k)\cdot\varepsilon1D_k$$
$$S12D_k := \sqrt{(XS12D_k)^2+(YS12D_k)^2}$$
$$XS21D_k := XB1D_k - \frac{BC}{2}\cdot\sin(\phi2_k)\cdot\omega2D_k\ \ YS21D_k := YB1D_k + \frac{BC}{2}\cdot\cos(\phi2_k)\cdot\omega2D\ \ S21D_k := \sqrt{(XS21D_k)^2+(YS21D_k)^2}$$
$$XS22D_k := XB2D_k - \frac{BC}{2}\cdot\cos(\phi2_k)\cdot(\omega2D_k)^2 - \frac{BC}{2}\cdot\sin(\phi2_k)\cdot\varepsilon2D_k$$
$$YS22D_k := YB2D_k - \frac{BC}{2}\cdot\sin(\phi2_k)\cdot(\omega2D_k)^2 + \frac{BC}{2}\cdot\cos(\phi2_k)\cdot\varepsilon2D_k\ \ S22D_k := \sqrt{(XS22D_k)^2+(YS22D_k)^2}$$

(8) Original new dynamics (variable angular velocities are determined); Section 2.2.

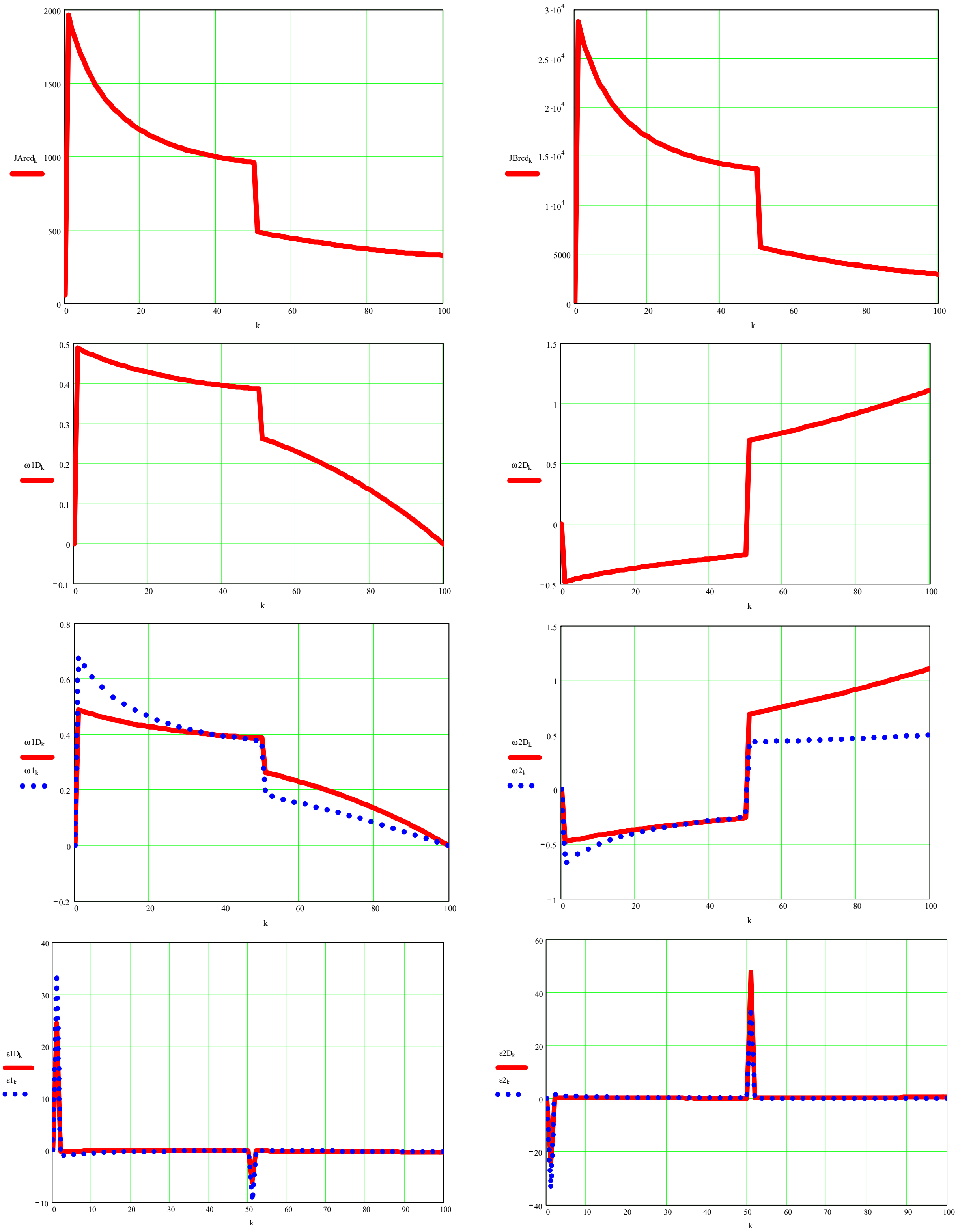

Figure A10. In the figure you can see above the moment of reduced mass inertia in the coupling A element 1 (**left**) and in the coupling B element 2 (**right**), then immediately below are represented the real, dynamic angular velocity of the element 1 (on the **left**) and of the element 2 (**right**), immediately below you can see the comparative graphs between the dynamic and kinematic angular velocity for element 1 (**left**) and element 2 (**right**), and below you can finally see the comparative graphs of the dynamic angular accelerations and kinematics for element 1 (**left**) and for element 2 (**right**).

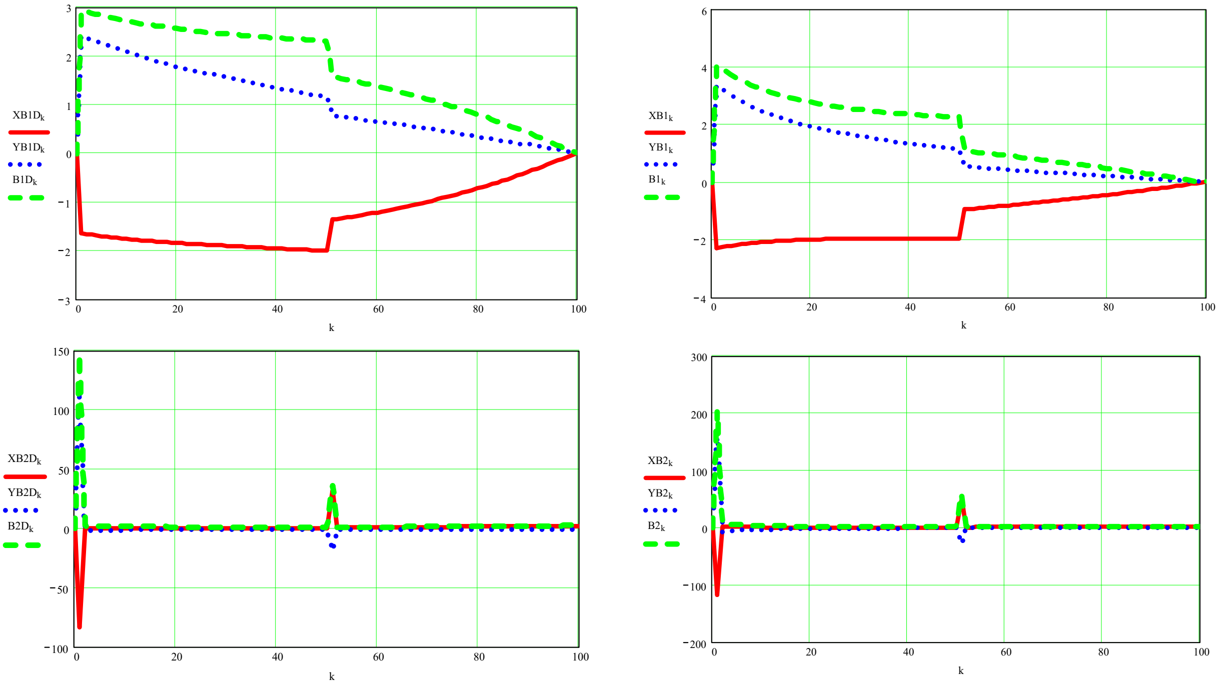

Figure A11. Here are represented the velocities of point B in their scalar coordinates (with red for the x-axis and with blue for the y-axis) and in absolute value (with green), dynamic on the left and kinematic on the right, at the top, and below are similarly represented the accelerations corresponding to the same point B.

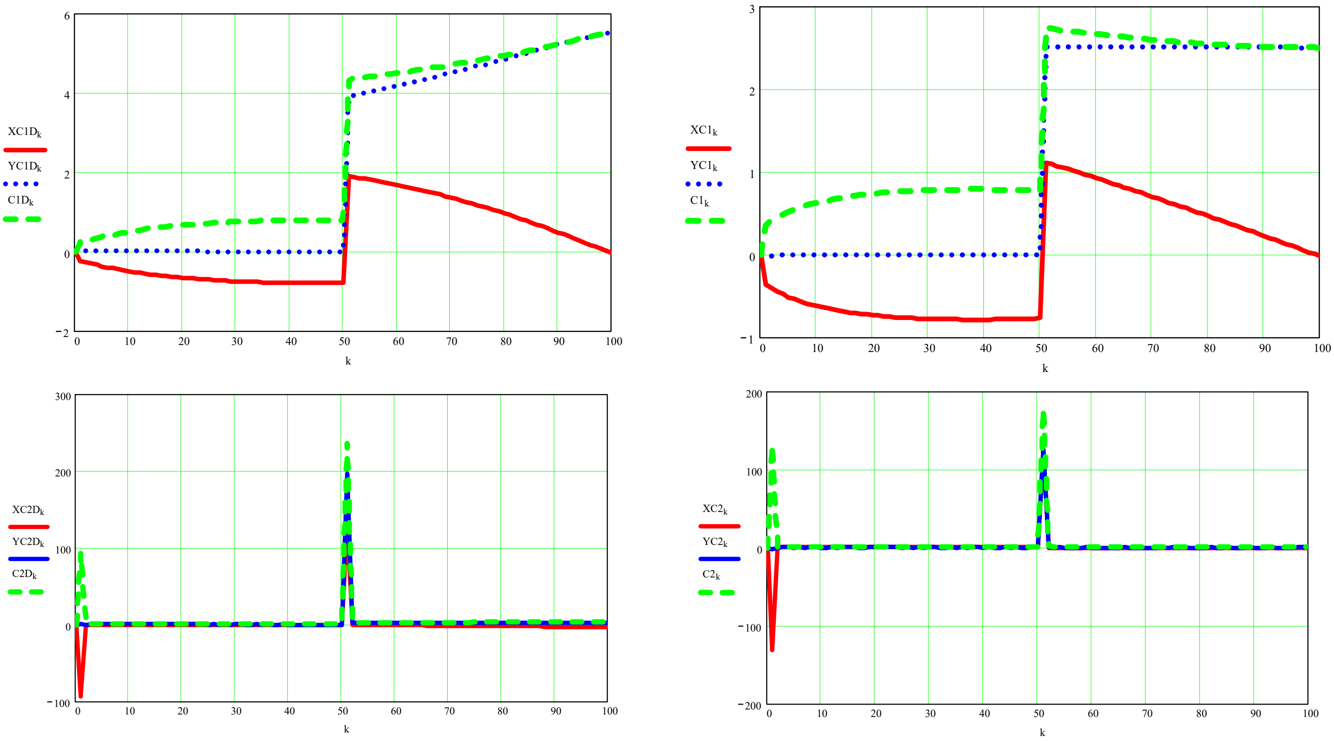

Figure A12. Here are represented the velocities of point C in their scalar coordinates (with red for the x-axis and with blue for the *y*-axis) and in absolute value (with green), dynamic on the left and kinematic on the right, at the top, and below are similarly represented the accelerations corresponding to the same point C.

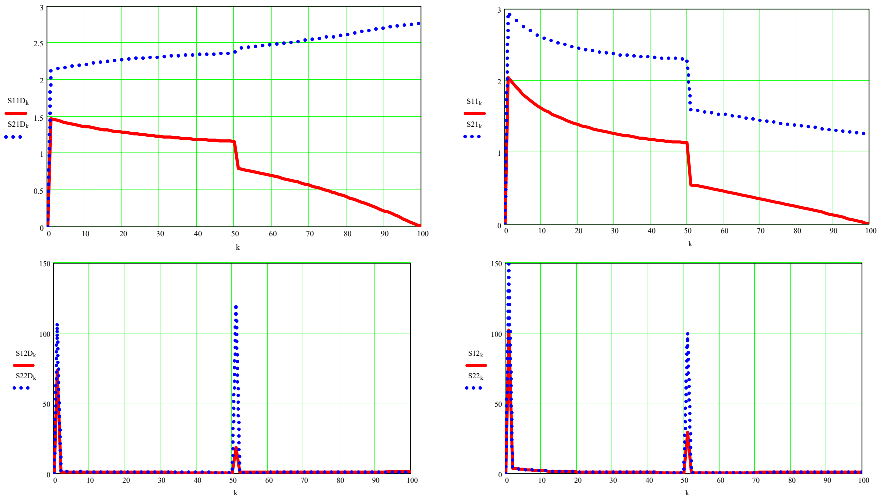

Figure A13. Here are represented the absolute velocities of point S1 (in red) and point S2 (in blue) dynamic on the left and kinematic on the right, on the top, and below are similarly represented the absolute accelerations corresponding to the same points S1 and S2 also dynamic values on the left and kinematic values on the right.

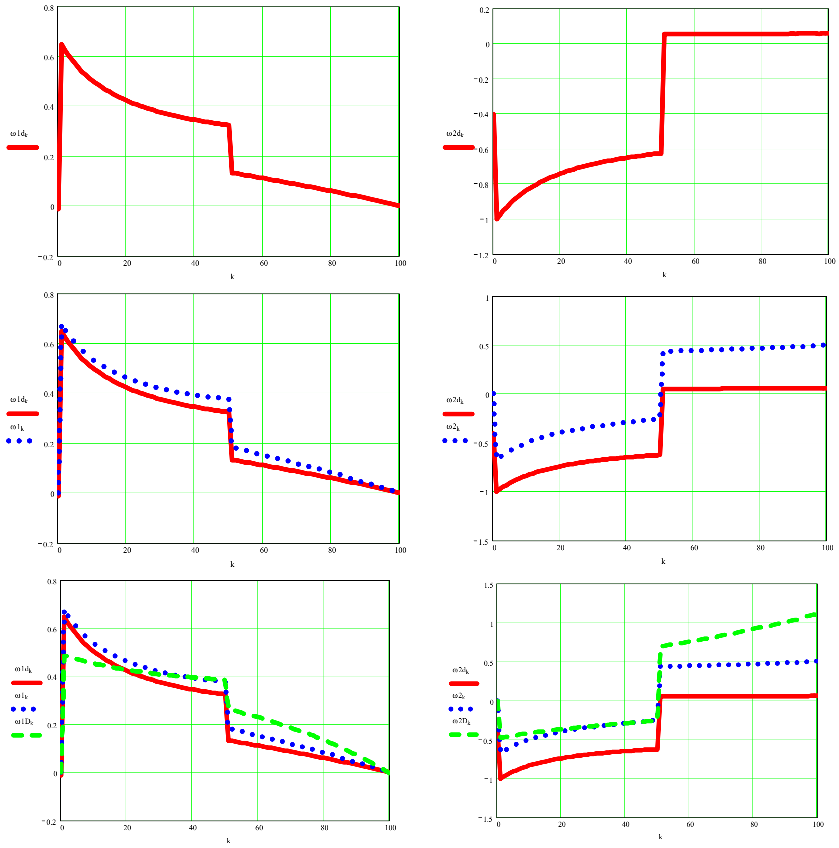

Figure A14. In the upper part are represented the dynamic angular speeds obtained with the new method presented in the paper, marked with d, on the left for the motor element 1 and on the right for the motor element 2; then below you can see the same dynamic speeds compared to the kinematic ones; in the end, bottom, all three will be compared, ie the dynamic angular velocity obtained by the absolutely new method (d), the kinematic angular velocity, and the dynamic angular velocity (D) determined by a classical method substantially improved by the author.

References

1. Merriam-Webster. *Definition of Dynamics (Entry 1 of 2)*; Merriam-Webster: Springfield, MA, USA, 2022. Available online: https://www.merriam-webster.com/dictionary/dynamics (accessed on 21 March 2022).
2. Giberti, H.; Abbattista, T.; Carnevale, M.; Giagu, L.; Cristini, F. A Methodology for Flexible Implementation of Collaborative Robots in Smart Manufacturing Systems. *Robotics* **2022**, *11*, 9. [CrossRef]
3. Maaroof, O.W.; Dede, M.I.C.; Aydin, L. A Robot Arm Design Optimization Method by Using a Kinematic Redundancy Resolution Technique. *Robotics* **2021**, *11*, 1. [CrossRef]
4. Chen, S.; Wen, J. Industrial Robot Trajectory Tracking Control Using Multi-Layer Neural Networks Trained by Iterative Learning Control. *Robotics* **2021**, *10*, 50. [CrossRef]
5. Hao, L.; Pagani, R.; Beschi, M.; Legnani, G. Dynamic and Friction Parameters of an Industrial Robot: Identification, Comparison and Repetitiveness Analysis. *Robotics* **2021**, *10*, 49. [CrossRef]
6. Fugal, J.; Bae, J.; Poonawala, H. On the Impact of Gravity Compensation on Reinforcement Learning in Goal-Reaching Tasks for Robotic Manipulators. *Robotics* **2021**, *10*, 46. [CrossRef]
7. Palomba, I.; Gualtieri, L.; Rojas, R.; Rauch, E.; Vidoni, R.; Ghedin, A. Mechatronic Re-Design of a Manual Assembly Workstation into a Collaborative One for Wire Harness Assemblies. *Robotics* **2021**, *10*, 43. [CrossRef]
8. Yamakawa, Y.; Katsuki, Y.; Watanabe, Y.; Ishikawa, M. Development of a High-Speed, Low-Latency Telemanipulated Robot Hand System. *Robotics* **2021**, *10*, 41. [CrossRef]
9. Pozzi, M.; Prattichizzo, D.; Malvezzi, M. Accessible Educational Resources for Teaching and Learning Robotics. *Robotics* **2021**, *10*, 38. [CrossRef]
10. Sun, J.; Han, X.; Li, T.; Li, S. Dynamic Parameter Identification of a Pointing Mechanism Considering the Joint Clearance. *Robotics* **2021**, *10*, 36. [CrossRef]
11. Stuhlenmiller, F.; Weyand, S.; Jungblut, J.; Schebek, L.; Clever, D.; Rinderknecht, S. Impact of Cycle Time and Payload of an Industrial Robot on Resource Efficiency. *Robotics* **2021**, *10*, 33. [CrossRef]
12. Gierlak, P. Adaptive Position/Force Control of a Robotic Manipulator in Contact with a Flexible and Uncertain Environment. *Robotics* **2021**, *10*, 32. [CrossRef]
13. Geng, J.; Arakelian, V.; Chablat, D.; Lemoine, P. Balancing of the Orthoglide Taking into Account Its Varying Payload. *Robotics* **2021**, *10*, 30. [CrossRef]
14. Colan, J.; Nakanishi, J.; Aoyama, T.; Hasegawa, Y. Optimization-Based Constrained Trajectory Generation for Robot-Assisted Stitching in Endonasal Surgery. *Robotics* **2021**, *10*, 27. [CrossRef]
15. Liu, R.; Nageotte, F.; Zanne, P.; de Mathelin, M.; Dresp-Langley, B. Deep Reinforcement Learning for the Control of Robotic Manipulation: A Focussed Mini-Review. *Robotics* **2021**, *10*, 22. [CrossRef]
16. Engelbrecht, D.; Steyn, N.; Djouani, K. Adaptive Virtual Impedance Control of a Mobile Multi-Robot System. *Robotics* **2021**, *10*, 19. [CrossRef]
17. Alizade, R.; Soltanov, S.; Hamidov, A. Structural Synthesis of Lower-Class Robot Manipulators with General Constraint One. *Robotics* **2021**, *10*, 14. [CrossRef]
18. Scalera, L.; Seriani, S.; Gallina, P.; Lentini, M.; Gasparetto, A. Human–Robot Interaction through Eye Tracking for Artistic Drawing. *Robotics* **2021**, *10*, 54. [CrossRef]
19. Petrescu, R.V.; Aversa, R.; Apicella, A.; Petrescu, F.I. Future Medicine Services Robotics. *Am. J. Eng. Appl. Sci.* **2016**, *9*, 1062–1087. [CrossRef]
20. Petrescu, F.I.T.; Petrescu, R.V.V. Forces at the Main Mechanism of a Railbound Forging Manipulator. *Indep. J. Manag. Prod.* **2015**, *6*, 904–921. [CrossRef]
21. Petrescu, F.I.; Petrescu, R.V.V. Kinematics at the Main Mechanism of a Railbound Forging Manipulator. *Indep. J. Manag. Prod.* **2015**, *6*, 711–729. [CrossRef]
22. Petrescu, F.I.; Petrescu, R.V. Dynamic Cinematic to a Structure 2R. *GEINTEC J.* **2016**, *6*, 3143–3154.
23. Petrescu, F.I.T.; Petrescu, R.V.V. Direct kinematics of a manipulator with three mobilities. *Indep. J. Manag. Prod.* **2021**, *12*, 1875–1900. [CrossRef]
24. Essomba, T. Design of a Five-Degrees of Freedom Statically Balanced Mechanism with Multi-Directional Functionality. *Robotics* **2021**, *10*, 11. [CrossRef]
25. Miguel-Tomé, S. The Heuristic of Directional Qualitative Semantic: A New Heuristic for Making Decisions about Spinning with Qualitative Reasoning. *Robotics* **2021**, *10*, 17. [CrossRef]
26. Petrescu, F.I.T.; Comanescu, A. Kinetostatics of a 2T9R Robot Mechanism. *Am. J. Eng. Appl. Sci.* **2022**, *15*, 59–80. [CrossRef]
27. Alpers, B. On Fast Jerk-, Acceleration- and Velocity-Restricted Motion Functions for Online Trajectory Generation. *Robotics* **2021**, *10*, 25. [CrossRef]
28. Caruso, M.; Gallina, P.; Seriani, S. On the Modelling of Tethered Mobile Robots as Redundant Manipulators. *Robotics* **2021**, *10*, 81. [CrossRef]
29. Ebel, L.; Maaß, J.; Zuther, P.; Sheikhi, S. Trajectory Extrapolation for Manual Robot Remote Welding. *Robotics* **2021**, *10*, 77. [CrossRef]
30. Thompson, L.; Badache, M.; Brusamolin, J.; Savadkoohi, M.; Guise, J.; Paiva, G.; Suh, P.; Guerrero, P.S.; Shetty, D. Multidirectional Overground Robotic Training Leads to Improvements in Balance in Older Adults. *Robotics* **2021**, *10*, 101. [CrossRef]

31. Vatsal, V.; Hoffman, G. The Wearable Robotic Forearm: Design and Predictive Control of a Collaborative Supernumerary Robot. *Robotics* **2021**, *10*, 91. [CrossRef]
32. Al Younes, Y.; Barczyk, M. Nonlinear Model Predictive Horizon for Optimal Trajectory Generation. *Robotics* **2021**, *10*, 90. [CrossRef]
33. Pacheco-Gutierrez, S.; Niu, H.; Caliskanelli, I.; Skilton, R. A Multiple Level-of-Detail 3D Data Transmission Approach for Low-Latency Remote Visualisation in Teleoperation Tasks. *Robotics* **2021**, *10*, 89. [CrossRef]
34. Stodola, M.; Rajchl, M.; Brablc, M.; Frolík, S.; Křivánek, V. Maxwell Points of Dynamical Control Systems Based on Vertical Rolling Disc—Numerical Solutions. *Robotics* **2021**, *10*, 88. [CrossRef]
35. Raviola, A.; Guida, R.; De Martin, A.; Pastorelli, S.; Mauro, S.; Sorli, M. Effects of Temperature and Mounting Configuration on the Dynamic Parameters Identification of Industrial Robots. *Robotics* **2021**, *10*, 83. [CrossRef]
36. Medina, O.; Hacohen, S. Overcoming Kinematic Singularities for Motion Control in a Caster Wheeled Omnidirectional Robot. *Robotics* **2021**, *10*, 133. [CrossRef]
37. Malik, A.; Henderson, T.; Prazenica, R. Multi-Objective Swarm Intelligence Trajectory Generation for a 7 Degree of Freedom Robotic Manipulator. *Robotics* **2021**, *10*, 127. [CrossRef]
38. Petrescu, F.I.T. *2012 Serial Mechatronic Systems, Parallel and Mixed*; Create Space Publisher: Seattle, WA, USA, 12 February 2014; p. 224. Available online: https://www.amazon.com/Sisteme-Mecatronice-Seriale-Paralele-Romanian/dp/1495923819 (accessed on 21 March 2022).
39. Ungureanu, L.M.; Petrescu, F.I.T. Dynamics of Mechanisms with Superior Couplings. *Appl. Sci.* **2021**, *11*, 8207. [CrossRef]
40. Featherstone, R. The Calculation of Robot Dynamics Using Articulated-Body Inertias. *Int. J. Robot. Res.* **1983**, *2*, 13–30.
41. Luh, J.Y.S.; Walker, M.W.; Paul, R.P.C. On-Line Computational Scheme for Mechanical Manipulators. *J. Dyn. Syst. Meas. Control* **1980**, *102*, 69–76.
42. Walker, M.W.; Orin, D.E. Efficient Dynamic Computer Simulation of Robotic Mechanisms. *J. Dyn. Syst. Meas. Control* **1982**, *104*, 205–211.
43. Featherstone, R. *Robot Dynamics Algorithms*; Kluwer Academic Publishers: Boston, MA, USA; Dordrecht, The Netherlands; Lancaster, UK, 1987.
44. Rodriguez, G.; Jain, A.; Kreutz-Delgado, K. A Spatial Operator Algebra for Manipulator Modelling and Control. *Int. J. Robot. Res.* **1991**, *10*, 371–381.
45. Pelecudi, C. *Theory of Spatial Mechanisms*; Publishing House of the Academy of the Socialist Republic of Romania: Bucharest, Romania, 1972; p. 510.
46. Pelecudi, C.; Maros, D. *Mechanisms*; Didactic and Pedagogical Publishing House: Bucharest, Romania, 1985; p. 395.
47. Featherstone, R.; Orin, D.E. In Proceedings of the 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation, San Francisco, CA, USA, 24–28 April 2000; pp. 826–834.
48. Khalil, W.; Dombre, E. *Modeling, Identification, and Control of Robots*; Taylor and Francis: New York, NY, USA, 2002.
49. Sporns, O. Complexity. *Scholarpedia* **2007**, *2*, 1623.
50. Meiss, J. Dynamical systems. *Scholarpedia* **2007**, *2*, 1629.
51. Featherstone, R. *Rigid Body Dynamics Algorithms*; Springer: Boston, MA, USA, 2007.
52. Khatib, O. A unified approach for motion and force control of robot manipulators: The operational space formulation. *IEEE J. Robot. Autom.* **1987**, *3*, 43–53.
53. Siciliano, B.; Sciavicco, L.; Villani, L.; Oriolo, G. *Robotics-Modelling, Planning, and Control. Advanced Textbooks in Control and Signal Processing*; Springer: London, UK, 2009. [CrossRef]
54. Zhang, X.; Pan, W.; Scattolini, R.; Yu, S.; Xu, X. Robust tube-based model predictive control with Koopman operators. *Automatica* **2022**, *137*, 110114. [CrossRef]
55. Meera, A.A.; Wisse, M. Dynamic Expectation Maximization Algorithm for Estimation of Linear Systems with Colored Noise. *Entropy* **2021**, *23*, 1306. [CrossRef]
56. Lv, M.; Li, Y.; Pan, W.; Baldi, S. Finite-Time Fuzzy Adaptive Constrained Tracking Control for Hypersonic Flight Vehicles with Singularity-Free Switching. *IEEE/ASME Trans. Mechatron.* **2021**. [CrossRef]
57. Aguado, E.; Milosevic, Z.; Hernández, C.; Sanz, R.; Garzon, M.; Bozhinoski, D.; Rossi, C. Functional Self-Awareness and Metacontrol for Underwater Robot Autonomy. *Sensors* **2021**, *21*, 1210. [CrossRef]
58. Yang, Y.; Zhou, H.; Song, Y.; Vink, P. Identify dominant dimensions of 3D hand shapes using statistical shape model and deep neural network. *Appl. Ergon.* **2021**, *96*, 103462. [CrossRef]
59. Han, M.; Tian, Y.; Zhang, L.; Wang, J.; Pan, W. Reinforcement learning control of constrained dynamic systems with uniformly ultimate boundedness stability guarantee. *Automatica* **2021**, *129*, 109689. [CrossRef]
60. Bloesch, M.; Sommer, H.; Laidlow, T.; Burri, M.; Nuetzi, G.; Fankhauser, P.; Bellicoso, D.; Gehring, C.; Leutenegger, S.; Hutter, M.; et al. A Primer on the Differential Calculus of 3D Orientations. *Tech. Rep.* **2016**, arXiv:1606.05285.
61. van der Spaa, L.F.; Wolfslag, W.J.; Wisse, M. Unparameterized Optimization of the Spring Characteristic of Parallel Elastic Actuators. *IEEE Robot. Autom. Lett.* **2019**, *4*, 854–861. [CrossRef]
62. Calli, B.; Caarls, W.; Wisse, M.; Jonker, P.P. Active Vision via Extremum Seeking for Robots in Unstructured Environments: Applications in Object Recognition and Manipulation. *IEEE Trans. Autom. Sci. Eng.* **2018**, *15*, 1810–1822. [CrossRef]