




Article

Phase-Space Modeling and Control of Robots in the Screw Theory Framework Using Geometric Algebra

Jesús Alfonso Medrano-Hermosillo [†], Ricardo Lozoya-Ponce ^{*,†}, Abraham Efraím Rodríguez-Mata [†]
and Rogelio Baray-Arana [†]

Instituto Tecnológico de Chihuahua (ITCH), Chihuahua 31310, Mexico

* Correspondence: ricardo.lp@chihuahua.tecnm.mx

† This author contributed equally to this work.

Abstract: The following paper talks about the dynamic modeling and control of robot manipulators using Hamilton's equations in the screw theory framework. The difference between the proposed work with diverse methods in the literature is the ease of obtaining the laws of control directly with screws and co-screws, which is considered modern robotics by diverse authors. In addition, geometric algebra (GA) is introduced as a simple and iterative tool to obtain screws and co-screws. On the other hand, such as the controllers, the Hamiltonian equations of motion (in the phase space) are developed using co-screws and screws, which is a novel approach to compute the dynamic equations for robots. Regarding the controllers, two laws of control are designed to ensure the error's convergence to zero. The controllers are computed using the traditional feedback linearization and the sliding mode control theory. The first one is easy to program and the second theory provides robustness for matched disturbances. On the other hand, to prove the stability of the closed loop system, different Lyapunov functions are computed with co-screws and screws to guarantee its convergence to zero. Finally, diverse simulations are illustrated to show a comparison of the designed controllers with the most famous approaches.

Keywords: screw theory; geometric algebra; Hamilton's equations; sliding mode control; Lyapunov theory

MSC: 70e60; 70b15



Citation: Medrano-Hermosillo, J.A.; Lozoya-Ponce, R.; Rodríguez-Mata, A.E.; Baray-Arana, R. Phase-Space Modeling and Control of Robots in the Screw Theory Framework Using Geometric Algebra. *Mathematics* **2023**, *11*, 572. <https://doi.org/10.3390/math11030572>

Academic Editors: Maria Luminița Scutaru and Catalin I. Pruncu

Received: 15 December 2022

Revised: 13 January 2023

Accepted: 18 January 2023

Published: 21 January 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In the Lagrangian approach, the two fundamental variables, written as position (θ) and velocity ($\dot{\theta}$), are mutually dependent. However, in the Hamiltonian formalism, the fundamental variables, computed as position (θ) and momentum (p), provide more abstract and profound formulations mechanics [1]. For example, Hamiltonian formalism is very important in the study of the energy changes that are possible in molecules and atoms. In addition, it is crucial if it is interested in quantizing a dynamical system or in quantum theory. Thus, this field should be taken into account [1,2].

The Lagrangian formalism (traditionally used in robots) is based on the kinetic and potential energies of the robot. This function is used to construct the body dynamics and then the control of the system [3]. Therefore, the resulting dynamic model and its controller will be represented by positions and velocities. However, in robotics, it is possible to compute controllers using Hamiltonian formalism, because the momenta, in theory, change very quickly (in rate $1/10$) [4]. In addition, in practice, it is easier to measure these forces with sensors. Hence, there are motivations to implement controllers using the Hamiltonian approach. Some relevant works about Hamiltonian controllers are mentioned below.

In [5], the authors present a proportional-derivative control with gravity compensation (PD+G) using Hamilton's equations for robot manipulators with multiple degrees of

freedom, demonstrating and proving the analysis using a simulation for a planar robot. In [6], the authors propose a hybrid controller for a SCARA robot. The work is based on a port-controlled Hamiltonian system to reduce the position tracking error. In [7], using Hamilton's equations, the work describes a formalism to control electromechanical systems. On the other hand, there are other advanced works formulating Hamiltonian mechanics using a geometric framework. In [8], the work proposes Hamiltonian mechanics in terms of Geometric Calculus, where the author mentioned that their Hamiltonian formalism would be highly important for application with robots. On the other hand, other works are using GA to reduce the computational cost, as mentioned in [9]. In [10], geometric algebra is utilized to compute Hamiltonian mechanics and the Poisson bracket, where this work could be used in robotics. The authors in [11] illustrate the importance of Hamiltonians in robotics. The technique computes the dynamic equations using Newton–Euler and describes the local Hamiltonians in each joint to implement its law of control. Based on the above, the Hamiltonian equations can be used to control electromechanical systems. However, the previous works have been developed using the traditional Euler–Lagrange or Newton–Euler equations, where in some robots, the computational cost is high due to the considerable multiplications between the diverse components. In addition, the development is tedious for some robots. Hence, the new proposed method expands the Hamiltonian control approach using screw theory, where this powerful mathematical tool has been used in recent years for the analysis of spatial mechanisms and some works have expressed it as “the forgotten tool in multibody dynamics” [12,13].

Screw theory is a mathematical tool for the analysis of spatial mechanics, the main element of this theory is the screw [14]. The screw is constructed by two three-dimensional vectors, where these vectors are angular velocity and linear velocity [15,16]. Therefore, it is possible to study rigid bodies using this technique. The screw theory has gained importance because it is an elegant mathematical tool and can reduce the number of multiplications between the Lie group $SE(3)$ (as is common in traditional techniques) [17]. Moreover, some authors describe the analysis of the kinematics and dynamics of rigid bodies using screw theory as modern robotics [18]. In robotics, several works using screw theory have gained prominence in recent years. For example, diverse authors propose the screw theory approach in the process of inverse and/or forward kinematics [19–22]. Other relevant techniques are described below. In [23], the paper describes an analysis of error sources of industrial robots, where they proposed a pose error model of industrial robots with screw theory. In [24], the authors propose a mathematical model using Kane's dynamic equations in the framework of screw theory. In [25], they study the rigid-body dynamics of serial robots subject to time-invariant holonomic constraints on their end-effectors. The paper proposes a technique to compute the dynamic model of robots in the framework of screw theory. On the other hand, other interesting approaches for the dynamic models are mentioned in [26,27]. Alternatively, in [28], the method suggests a motion control approach with a focus on robotic manipulators based on screw theory and dual quaternions, where they add a stability analysis to propose a law of control for a desired trajectory. However, the drawback is that, while previous works consider kinematics and dynamics in rigid bodies, controllers are not directly proposed using screw theory. Therefore, it is an interesting motivation to compute controllers using the screw theory and the Hamiltonian approach. In addition, GA is proposed to compute the screws of the controllers and dynamics of the system; this novel approach reduces the number of operations and is intuitive for new researchers in the robotics field (due to previous knowledge of vector calculus not being required).

Taking into account the previous literature, it is possible to see that the Hamilton's equations can be used as a methodology to compute controllers for robots, where our work extends these equations for robots in the screw theory framework and iteratively. In addition, GA is proposed for computing the diverse screws and co-screws, which is an advantage for new users interested in robotics. Regarding the controllers, the equations of the robot are designed with co-screws and screws using the approach in [17]. Later, to prove

convergence, functions with screws and co-screws are computed to satisfy the Lyapunov theory and thus obtain the law of control. The main advantages of our approach are:

- Due to the use of GA, vector calculus knowledge is not required [29,30].
- Using the screw theory, the Denavit–rtenberg (D–H) representation and the homogeneous matrices are avoided. It is well known that with the traditional method, it is possible to have multiple results (this is due to the D–H technique). The above could be a risk of confusion by different designers [20].
- As the proposed technique is iterative, it is possible to be implemented in serial robots with any degrees of freedom. In addition, the Euler–Lagrange and Newton–Euler methods are avoided. Therefore, robot dynamics and control can be computed easily.
- Due to the laws of control being computed directly using screw theory, they are intuitive and can be programmed easily.
- Using sliding mode control, the robustness of the system under matched perturbations is obtained [31].

The document is organized as follows: In Section 2, we illustrate the theoretical bases of geometric algebra. In Section 3, the mathematical development necessary to construct the controllers is presented. In Section 4, we show the laws of control and their stability. In Section 5, we present a numerical example with a simulation to prove the designed located controllers. Finally, in Section 6, we illustrate the conclusions.

2. Geometric Algebra

In mathematics, geometric algebra is a term applied to Clifford’s theory of algebras [29,32]. The GA of an n -dimensional space is denoted by $\mathbb{G}_{p,q,r}$, where p, q and r represent the orthonormal basis vectors that square to 1, -1 and 0, respectively. On the other hand, in addition to scalar multiplication and vector addition, GA is endowed with a noncommutative product, this product is the Clifford product (or geometric product). For example, the Clifford product for two vectors a, b are:

$$ab = a \cdot b + a \wedge b \tag{1}$$

In Equation (1), the right side illustrates two elements: the first one is the inner product or dot product (symmetric part); the second one is the wedge product or exterior product (antisymmetric part), where the wedge product is a distributive, associative, and anti-commutative operator. The elements computed by the exterior product of k independent vectors span the k -th exterior power. In this space, each element is called a k -vector. The diverse multi-vectors are entities computed by the sum of elements of the set of \mathbb{G}_n , written as:

$$A = \langle A \rangle_1 + \langle A \rangle_2 + \dots + \langle A \rangle_n \tag{2}$$

In addition to Equation (2), contemplate two homogeneous multi-vectors A and B of grade r and s , respectively. The Clifford product can be shown as:

$$AB = \langle AB \rangle_{r+s} + \langle AB \rangle_{r+s-2} + \dots + \langle AB \rangle_{|r-s|} \tag{3}$$

where $\langle AB \rangle_t$, indicate the t -grade part of the multi-vector AB . Suppose a n -dimensional space with diverse orthonormal basis vectors $\{e_i\}, i = 1, \dots, n$, such that $e_i \cdot e_j = \delta_{i,j}$, where $\delta_{i,j} = 1 \mid i = j$ and $\delta_{i,j} = 0 \mid i \neq j$. The basis vectors for the entire GA are:

$$\{1; e_i; e_i \wedge e_j; e_i \wedge e_j \wedge e_k; \dots; I = e_1 \wedge \dots \wedge e_n\} \tag{4}$$

where I is called the pseudoscalar. Below, we include some important definitions that are useful in geometric algebra, for details consult [8,29,32,33].

Definition 1. Let a multi-vector A of grade r . After, the reverse of A , written as \tilde{A} , is defined by:

$$\tilde{A} = \sum_{i=0}^r (-1)^{\frac{i(i-1)}{2}} \langle A \rangle_i \tag{5}$$

Definition 2. Let a multi-vector A of grade r . After, the Clifford conjugate of A , written as \bar{A} , is defined by:

$$\bar{A} = \sum_{i=0}^r (-1)^{\frac{i(i+1)}{2}} \langle A \rangle_i \tag{6}$$

Definition 3. Let $a \in \mathbb{G}_3$. The rotation of a , written as a' , is defined by the following versor product:

$$a' = R_\theta \tilde{a} R_\theta = e^{-\frac{\theta}{2} L} a e^{\frac{\theta}{2} L} \tag{7}$$

where R_θ is the rotor operator, θ is the rotation angle, and L is the Lie algebra generator. The 2-vector L is the operator of rotation used in quaternions; in a different way, just consider $e_2 e_3 \rightarrow i$, $e_3 e_1 \rightarrow j$ and $e_1 e_2 \rightarrow k$.

Definition 4. Let $I \in \mathbb{G}_n$. After, the inverse of I , written as I^{-1} , is defined by:

$$I^{-1} = \frac{\tilde{I}}{I\tilde{I}} \tag{8}$$

Definition 5. Let a multi-vector A of grade r . After, the dual of A , written as A^* , is defined by:

$$A^* = \sum_{i=0}^r \langle A \rangle_i I^{-1} \tag{9}$$

Definition 6. Consider two homogeneous multi-vectors A and B . After, the commutator product between A and B is defined by:

$$A \underline{\times} B = \frac{1}{2} (AB - BA) \tag{10}$$

Definition 7. Consider two homogeneous multi-vectors A and B . After, the anti-commutator product between A and B is defined by:

$$A \overline{\times} B = \frac{1}{2} (AB + BA) \tag{11}$$

3. Mathematical Development

3.1. Screws

Suppose the following Lie group:

$$SE(3) := \left\{ \begin{pmatrix} R & x \\ 0 & 1 \end{pmatrix} : R \in SO(3), x \in \mathbb{R}^3 \right\} \tag{12}$$

The Lie algebra of $SE(3)$ is:

$$se(3) := \left\{ \begin{pmatrix} w & vs. \\ 0 & 0 \end{pmatrix} : w \in so(3), vs. \in \mathbb{R}^3 \right\} \tag{13}$$

where $se(3)$, w and v represent the Lie algebra of the Lie group $SE(3)$, the angular velocity, and the linear velocity, respectively [18]. On the other hand, the Lie algebra elements are often illustrated as follows:

$$s = \begin{pmatrix} w \\ vs. \end{pmatrix} \tag{14}$$

where s is constructed by two three-dimensional vectors called screws.

Lie Algebra Elements

Consider a path through the identity (e) in a group (G):

$$\gamma : \mathbb{R} \rightarrow G \tag{15}$$

where $\gamma(0) = e$. The Lie algebra element is calculated by following a path $\gamma(t)$ starting at the identity element of $SE(3)$ and evaluating $\frac{d}{dt}|_{t=0}\gamma(t)$ [14,17]. Therefore, suppose the following path:

$$\mathbf{t} = x - R_\theta x \widetilde{R}_\theta \tag{16}$$

where \mathbf{t} is the translation vector. After, computing the Lie algebra element:

$$v = -\left(\frac{d}{dt}(R_\theta) x \widetilde{R}_\theta + R_\theta x \frac{d}{dt}(\widetilde{R}_\theta)\right) \tag{17}$$

Considering $\frac{d}{dt}(R_\theta) = -\frac{\dot{\theta}}{2}LR_\theta$ and $\frac{d}{dt}(\widetilde{R}_\theta) = \frac{\dot{\theta}}{2}L\widetilde{R}_\theta$:

$$v = -\left(\left(-\frac{\dot{\theta}}{2}LR_\theta\right) x \widetilde{R}_\theta + R_\theta x \left(\frac{\dot{\theta}}{2}L\widetilde{R}_\theta\right)\right) \tag{18}$$

Now, evaluating around the entity, where $R_\theta(0) = \widetilde{R}_\theta(0) = 1$:

$$v(0) = -\frac{1}{2}(xL - Lx)\dot{\theta} \tag{19}$$

Finally, using Definition 6:

$$v(0) = (L \times x)\dot{\theta} \tag{20}$$

Therefore, the linear velocity is the commutator between the Lie algebra generator of the rotor and the point x . On the other hand, the angular velocity is solved using the same methodology. The path through the identity is $\gamma(t) = R_\theta$, where $R_\theta(0) = 1$ and $R_\theta \widetilde{R}_\theta = 1$. Differentiating the last relation:

$$\frac{d}{dt}(R_\theta)\widetilde{R}_\theta + R_\theta \frac{d}{dt}(\widetilde{R}_\theta) = 0 \tag{21}$$

Now, evaluating around the entity:

$$L\dot{\theta} - L\dot{\theta} = 0 \tag{22}$$

Hence, the Lie algebra element consists of a 2 - vector L . In consequence, the angular velocity is expressed by the Lie algebra generator of the rotor. However, traditionally, the angular velocity is written as a unit vector [34–36]. Thus, this notation can be used by Definition 5. Therefore, the angular velocity is:

$$w(0) = L^*\dot{\theta} \tag{23}$$

Finally, with our approach, the screw described above can be written as:

$$s(0) = \begin{pmatrix} L^* \\ L \times x \end{pmatrix} \dot{\theta} \tag{24}$$

It is possible to see that each screw can be computed easily with the previous equation. In each DoF, it is only necessary to record the Lie algebra element and its Cartesian position.

3.2. Velocity Kinematics

In robotics, the screws express the velocities of the diverse joints, where the velocity for a robot with n degrees of freedom (DoF) can be illustrated by:

$$V_n(0) = \sum_{j=1}^n s_j(0)\dot{\theta}_j \tag{25}$$

where the previous velocity is only valid in the identity or in the home position. To find the velocity in the current position of the robot, it is necessary to compute the current screw (s). To write the current screw, it is indispensable to use the following notation:

$$s_j = e^{\theta_i ad(s_i)} s_j(0)\dot{\theta} \tag{26}$$

Here, we use the adjoint representation and the exponential mapping to compute the current screw [37]. Therefore, the velocity of a serial robot with n DoF, in any position, is:

$$V_n = \sum_{j=1}^n s_j\dot{\theta}_j \tag{27}$$

From Equation (27), you can see that the different screws are the columns of the Jacobian matrix in robotics.

3.3. Co-Screws

Properly speaking, co-screws are linear functional on the velocities, satisfying the following:

$$\mathcal{F} : se(3) \rightarrow \mathbb{R} \text{ where } \mathcal{F}(as_1 + bs_2) = a\mathcal{F}(s_1) + b\mathcal{F}(s_2) \tag{28}$$

Here, $a, b \in \mathbb{R}$, and $\mathcal{F}(s)$ are often called the evaluation map. The co-screws are constructed using two three-dimensional vectors, but, contrary to screws, these elements are computed by the dual Lie algebra $se^*(3)$. Thus, in robotics, the momentum co-screw (\mathcal{P}) can be written as follows:

$$\mathcal{P} = \begin{pmatrix} j \\ p \end{pmatrix} \tag{29}$$

where j and p represent the angular and linear momentum, respectively, [17]. Furthermore, the evaluation map is illustrated as:

$$\mathcal{P}(V) = \mathcal{P}^T V \tag{30}$$

Another choice to construct the momentum co-screw is using the inertia operator, where the inertia provides an isomorphism as:

$$N : se(3) \rightarrow se^*(3) \tag{31}$$

and it is computed by:

$$N = e^{-\theta_i ad^T(s_i)} N(0) e^{-\theta_i ad(s_i)} \text{ and } N(0) = \begin{pmatrix} \mathcal{I} & mC \\ mC^T & mI_3 \end{pmatrix} \tag{32}$$

Here \mathcal{I} represents the inertia tensor, m the mass of the link, C the adjoint representation of the center of mass and I_3 the identity matrix in \mathbb{R}^3 . Hence, the momentum co-screw can be written as [17]:

$$\mathcal{P} = NV \tag{33}$$

3.4. Lagrangian Formulation of Dynamics Using Screw Theory

The kinetic energy can be described, in the context of the screw theory, as a combination of the momentum co-screw and the screw:

$$E_k(\theta, \dot{\theta}) = \sum_{j=1}^n \frac{1}{2} (j_j \cdot w_j + p_j \cdot v_j) = \sum_{j=1}^n \frac{1}{2} \mathcal{P}_j(V_j) = \sum_{j=1}^n \frac{1}{2} N_j V_j(V_j) \tag{34}$$

Using the evaluation map, Equation (34) changes to:

$$E_k(\theta, \dot{\theta}) = \sum_{j=1}^n \frac{1}{2} V_j^T N_j V_j \tag{35}$$

On the other hand, the potential energy is written by:

$$E_p(\theta) = \sum_{j=1}^n \tilde{g}^T \tilde{c}_j \tag{36}$$

where:

$$\tilde{g} = \begin{pmatrix} -ge_2 \\ 0 \end{pmatrix} \quad \tilde{c}_j = \begin{pmatrix} m_j c_j \\ m_j \end{pmatrix} \tag{37}$$

Here, g is the gravitational force, m_j is the mass of each link and c_j is the position of the center of mass. Then, using the Euler–Lagrange equation, it is computed the dynamic equations of the robotic system (for details see the Appendix A). Therefore:

$$\tau_i = \frac{d}{dt} \left(\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \dot{\theta}_i} \right) - \frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \theta_i} \tag{38}$$

where:

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \dot{\theta}_i} \right) = \sum_{j=i}^n s_i^T N_j \dot{V}_j + V_j^T N_j [s_i, V_j] + V_j^T N_j [V_i, s_i] \tag{39}$$

$$\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \theta_i} = \sum_{j=i}^n \mathcal{G}_j^T s_i - V_j^T N_j [s_i, V_i] \tag{40}$$

Therefore, the robot dynamics are described by:

$$\tau_i = \sum_{j=i}^n \dot{V}_j^T N_j s_i + V_j^T N_j [s_i, V_j] - \mathcal{G}_j^T s_i \tag{41}$$

where \mathcal{G} is the co-screw of the gravitational forces and $[s_i, V_j]$ is the Lie bracket [17,25].

3.5. Hamilton’s Equations

Hamilton’s equations can be computed using the Legendre transformation [1]:

$$\mathcal{H}(\theta, p, t) = \sum_{i=1}^n p_i \dot{\theta}_i - \mathcal{L}(\theta, \dot{\theta}, t) \tag{42}$$

where the partial derivation with respect to p_i provides:

$$\frac{\partial \mathcal{H}(\theta, p, t)}{\partial p_i} = \dot{\theta}_i \tag{43}$$

The previous equation is one of Hamilton’s equations. To compute the second equation is necessary to use the partial derivation of Equation (42) with respect to θ_i :

$$\frac{\partial \mathcal{H}(\theta, p, t)}{\partial \theta_i} = - \frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \theta_i} \tag{44}$$

Finally, the third Hamilton’s equation is computed using Equation (38) with Equation (44) and $p_i = \frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \dot{\theta}_i}$:

$$\dot{p}_i = \tau_i - \frac{\partial \mathcal{H}(\theta, p, t)}{\partial \theta_i} \tag{45}$$

4. Hamilton Control Using Screw Theory

With our approach using screw theory, Hamilton’s equations can be written as (for details see Appendix A):

$$\frac{\partial \mathcal{H}(\theta, p, t)}{\partial p_i} = \dot{\theta}_i \tag{46}$$

$$\frac{\partial \mathcal{H}(\theta, p, t)}{\partial \theta_i} = \sum_{j=i}^n \mathcal{P}_j^T [s_i, N_i^{-1} \mathcal{P}_i] - \mathcal{G}_j^T s_i \tag{47}$$

$$\dot{p}_i = \tau_i + \sum_{j=i}^n \mathcal{G}_j^T s_i - \mathcal{P}_j^T [s_i, N_i^{-1} \mathcal{P}_i] \tag{48}$$

The previous equations could be used to compute the dynamic equations of robots in the phase space. After, to ensure that the robot reaches the desired position using the Hamiltonian approach, it is necessary to consult the following theorems.

Theorem 1. *A serial robot system reaches a desired position using the following control law:*

$$\tau_i = K_i \text{sign}(\mathcal{S}_i) + K_{\mathcal{S}_i} \mathcal{S}_i - \frac{\partial \mathcal{H}(\theta, p, t)}{\partial p_i} - \sum_{j=i}^n \mathcal{P}_j^T [N_i^{-1} \mathcal{P}_i, s_i] - \mathcal{G}_j^T s_i \tag{49}$$

where $\mathcal{S}_i = \tilde{p}_i + \tilde{\theta}_i$ is the sliding surface, $\tilde{p}_i = p_{d_i} - p_i$ is the error between the desired and measured momentum, $\tilde{\theta}_i = \theta_{d_i} - \theta_i$ is the error between the desired and the measured joint position, and $K_i, K_{\mathcal{S}_i} \in \mathbb{R}^+$.

Proof. In sliding mode control, the convergence of the surface needs to be satisfied [38]. Hence, the following Lyapunov function is proposed:

$$\mathcal{V}(\mathcal{S}_i) = \frac{1}{2} \mathcal{S}_i \overline{\times} \mathcal{S}_i \tag{50}$$

Now, use Definition 7 and differentiate the Lyapunov function in terms of time:

$$\dot{\mathcal{V}}(\mathcal{S}_i) = \mathcal{S}_i \overline{\times} \dot{\mathcal{S}}_i = \mathcal{S}_i \overline{\times} \frac{d}{dt} (\tilde{p}_i + \tilde{\theta}_i) \tag{51}$$

Using $\dot{\tilde{\theta}}_i = -\dot{\theta}_i$, Equations (46) and (48):

$$\dot{\mathcal{V}}(\mathcal{S}_i) = \mathcal{S}_i \overline{\times} \left(-\tau_i - \sum_{j=i}^n \mathcal{P}_j^T [N_i^{-1} \mathcal{P}_i, s_i] - \mathcal{G}_j^T s_i - \frac{\partial \mathcal{H}(\theta, p, t)}{\partial p_i} \right) \tag{52}$$

Applying the controller in Equation (49):

$$\dot{\mathcal{V}}(\mathcal{S}_i) = -K_i |\mathcal{S}_i| - K_{\mathcal{S}_i} \mathcal{S}_i \overline{\times} \mathcal{S}_i \tag{53}$$

Thus, as the derivative of the Lyapunov function is negative definite, the convergence of the sliding surface is satisfied. Then, $S_i = 0 \rightarrow \tilde{\theta}_i = 0$. Hence, the serial robot with the proposed controller will reach the desired position [38]. On the other hand, the closed-loop system can be computed applying Equation (49) into Equation (48):

$$\dot{\tilde{p}}_i = \frac{\partial \mathcal{H}(\theta, p, t)}{\partial p_i} - K_i \text{sign}(S_i) - K_{S_i} S_i \tag{54}$$

□

Theorem 2. *A serial robotic arm will track a desired smooth function by applying the following law of control:*

$$\tau_i = K_{p_i} \tilde{\theta}_i + K_{v_i} \tilde{p}_i + \dot{p}_{d_i} - \sum_{j=i}^n \mathcal{P}_j^T [N_i^{-1} \mathcal{P}_i, s_i] - \mathcal{G}_j^T s_i \tag{55}$$

where $S_i = \tilde{p}_i + \tilde{\theta}_i$ is the sliding surface, $\tilde{p}_i = p_{d_i} - p_i$ is the error between the desired and measured momentum, $\tilde{\theta}_i = \theta_{d_i} - \theta_i$ is the error between the desired and the measured joint position, and $K_{p_i}, K_{v_i} \in \mathbb{R}^+$.

Proof. The proof of this theorem is easily solved. If one converts the law of control in Equation (55) into Equation (48), it provides:

$$K_{p_i} \tilde{\theta}_i + K_{v_i} \tilde{p}_i + \dot{\tilde{p}}_i = 0 \tag{56}$$

The previous equation is linear and the closed-loop system is globally asymptotically stable, if and only if, $K_{p_i}, K_{v_i} \in \mathbb{R}^+$ [39]. This law of control is similar to the traditional PD-CTC (computed-torque control with a proportional-derivative action), but the proposed controller is in the phase space [35]. On the other hand, the closed-loop system can be represented as follows:

$$\dot{\tilde{p}}_i = -K_{p_i} \tilde{\theta}_i - K_{v_i} \tilde{p}_i \tag{57}$$

□

5. Examples

5.1. Single Degree-of-Freedom Robot

Consider the manipulator in Figure 1. After, using Equation (48), the dynamic equation of the robot in the phase space is:

$$\dot{p}_1 = \tau_1 + \mathcal{G}_1^T s_1 - \mathcal{P}_1^T [s_1, N_1^{-1} \mathcal{P}_1] = \tau_1 + \mathcal{G}_1^T s_1 \tag{58}$$

The screw can be computed as $s_1(0) = \begin{pmatrix} L^* \\ L \times x \end{pmatrix} = \begin{pmatrix} e_1 e_2^* \\ e_1 e_2 \times 0 \end{pmatrix} = \begin{pmatrix} e_3 \\ 0 \end{pmatrix}$ and the wrench of gravitational forces as $\mathcal{G}_1 = \begin{pmatrix} -mgc(0) \times e_2 \\ -mg e_2 \end{pmatrix}$. Therefore:

$$\dot{p}_1 = \tau_1 + \begin{pmatrix} -mgc(0) \times e_2 \\ -mg e_2 \end{pmatrix}^T \begin{pmatrix} e_3 \\ 0 \end{pmatrix} \tag{59}$$

$$\dot{p}_1 = \tau_1 - mg e_1 \cdot c(0) \tag{60}$$

The previous equation is in the initial position, to transform to the actual position is necessary to change the initial position of the center of mass to the actual. Hence, as the

center of mass is concentrated at the end of the link, the dynamic equations of the system are:

$$\dot{p}_1 = \tau_1 - mg e_1 \cdot l(\cos(\theta_1) e_1 + \sin(\theta_1) e_2) = \tau_1 - mgl\cos(\theta_1) \tag{61}$$

The above result is the dynamic equation of the robot in the phase space, demonstrating the ease of computing the motion equation. Remember that if we use another approach, it is necessary to develop the Euler–Lagrange equation and then use the Legendre transform, the traditional process is tedious instead of the proposed technique. Now, to illustrate the behavior of the previous system with Theorem 1, it is necessary to apply Equation (49) in Equation (61). Therefore, the system can be written as:

$$\dot{p}_1 = K_1 \text{sign}(\mathcal{S}_1) + K_{S1} \mathcal{S}_1 - \frac{\partial \mathcal{H}(\theta, p, t)}{\partial p_1} = K_1 \text{sign}(\mathcal{S}_1) + K_{S1} \mathcal{S}_1 - \frac{p}{ml^2} \tag{62}$$

where:

$$\mathcal{S}_1 = \tilde{\theta}_1 + \tilde{p}_1 \tag{63}$$

Based on the above, it is possible to conclude that the behavior of the system will be determined by the constant values. However, it is easy to check that the previous second-order differential equation is stable if and only if the constant values are positive and then the desired position will be reached. On the other hand, the behavior of Equation (61) with Theorem 2 can be written as:

$$\dot{p}_1 = K_{p1} \tilde{\theta}_1 + K_{v1} \tilde{p}_1 + \dot{p}_{d1} \tag{64}$$

Similar as Equation (62), the previous second-order differential equation is stable if and only if the constant values are positive.

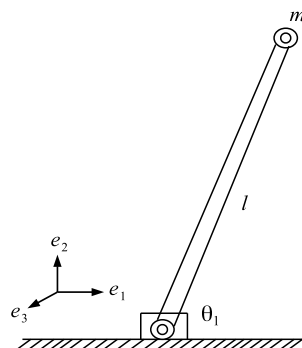


Figure 1. Single degree-of-freedom robot.

Comparison with Other Techniques

In the above section, the dynamic equations of the example were computed. It was seen the advantages of the proposed work instead of the traditional method. However, in this section, the diverse proposed laws of control are compared with other traditional techniques. To simulate the results of the example, it is necessary to implement the parameters in Table 1, where these elements are the proposed components for the robot. The laws of control to be compared are shown in Table 2, where it is possible to see the most famous controllers in the industry (PD and PID). The diverse gains for the controllers were calculated according to [40].

Table 1. Parameters of the robot.

| Parameter | Value | Unit |
|-----------|-------|-----------------|
| m | 0.25 | kg |
| l | 0.5 | m |
| g | 9.81 | $\frac{m}{s^2}$ |

Table 2. Feedback controllers for the single degree-of-freedom robot.

| Law of Control | Law of Control | Gains |
|----------------|--|--------------------------------|
| Theorem 1 | See Equation (49) | $K_1 = 1, K_{S1} = 10$ |
| Theorem 2 | See Equation (55) | $K_{p1} = 25, K_{v1} = 25$ |
| PD | $\tau = K_p e(t) + K_v \frac{de(t)}{dt}$ | $K_p = 25, K_v = 25$ |
| PID | $\tau = K_p e(t) + K_I \int e(t)dt + K_v \frac{de(t)}{dt}$ | $K_p = 25, K_I = 15, K_v = 25$ |

The simulations were conducted using the Euler integration method, with a step size of 0.001 s. The initial conditions were selected as $\theta_1(0) = p_1(0) = 0$ and the desired value as $\theta_{d1} = 45^\circ$. Thus, the behavior of the error, with diverse controllers, can be seen in Figure 2.

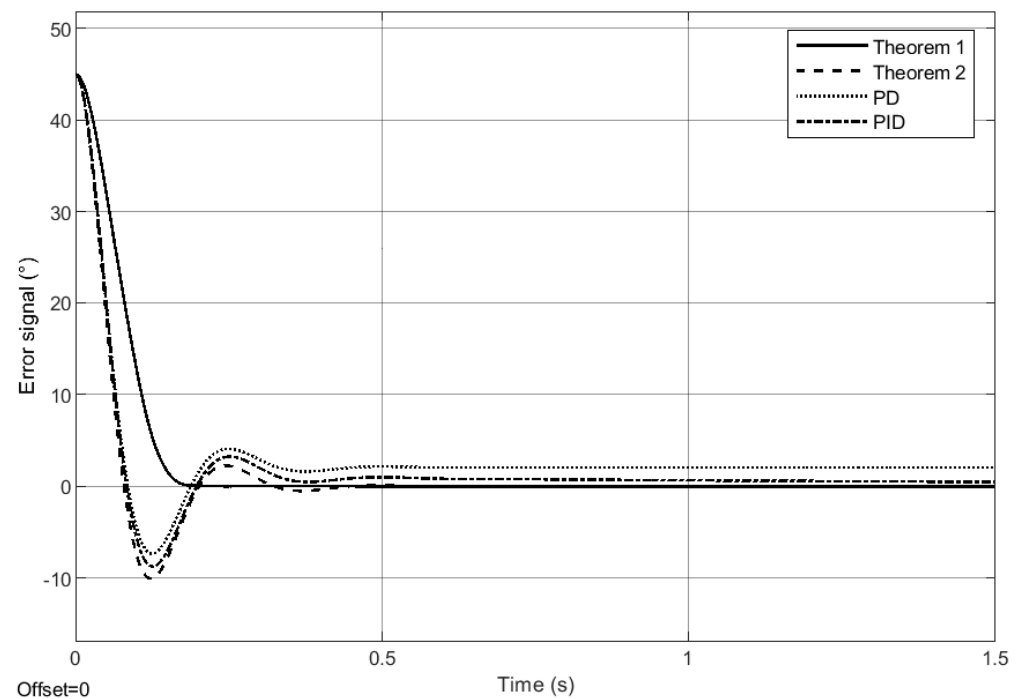


Figure 2. Error signals with diverse controllers in θ_1 for a single degree of freedom robot arm.

In Figure 2, the performances of the robot with different controllers are shown, where the two proposed laws of control reach convergence to zero. In addition, our approaches are faster than traditional techniques. Thus, the methodology proposed can be used to compute the dynamic equations of the robot in a simple manner and the controllers are efficient to be implemented physically.

5.2. Two Degrees-of-Freedom Robot

Suppose the two-link manipulator with pivot joints of Figure 3. The link lengths of the manipulator are l_1 and l_2 and the link masses are m_1 and m_2 , with these masses concentrated at the end of each link.

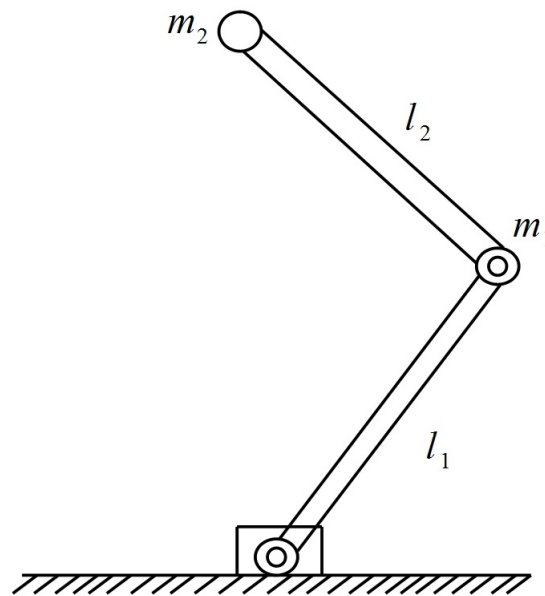


Figure 3. Two-link manipulator.

The Hamilton equations of the proposed robot are calculated using Equation (48). Therefore, these equations are written as follows:

$$\dot{p}_1 = \tau_1 + \mathcal{G}_1^T s_1 + \mathcal{G}_2^T s_1 - \mathcal{P}_1^T [s_1, N_1^{-1} \mathcal{P}_1] - \mathcal{P}_2^T [s_1, N_1^{-1} \mathcal{P}_1] \tag{65}$$

$$\dot{p}_2 = \tau_2 + \mathcal{G}_2^T s_2 - \mathcal{P}_2^T [s_2, N_2^{-1} \mathcal{P}_2] \tag{66}$$

Now, to illustrate the behavior of the previous system with the Theorem 1 is necessary to apply Equation (49). Therefore, the system can be written as:

$$\dot{p}_1 = K_1 \text{sign}(S_1) + K_{S1} S_1 - \frac{\partial \mathcal{H}(\theta, p, t)}{\partial p_1} \tag{67}$$

$$\dot{p}_2 = K_2 \text{sign}(S_2) + K_{S2} S_2 - \frac{\partial \mathcal{H}(\theta, p, t)}{\partial p_2} \tag{68}$$

where:

$$\begin{aligned} S_1 &= \tilde{\theta}_1 + \tilde{p}_1 \\ S_2 &= \tilde{\theta}_2 + \tilde{p}_2 \end{aligned} \tag{69}$$

On the other hand, the behavior of the robot with the Theorem 2 can be written as:

$$\dot{p}_1 = K_{p1} \tilde{\theta}_1 + K_{v1} \tilde{p}_1 + \dot{p}_{d1} \tag{70}$$

$$\dot{p}_2 = K_{p2} \tilde{\theta}_2 + K_{v2} \tilde{p}_2 + \dot{p}_{d2} \tag{71}$$

Comparison

In the same way as the previous example, to simulate the results of the example, it is necessary to implement the parameters in Table 3. However, in this example, the robot has two DoFs. Therefore, the system has two laws of control, where the constant values and equations are chosen as Table 4 (the gains were calculated using [40]). The simulations were performed using Euler’s integration method, with a step size of 0.01 s. The initial conditions were selected as $\theta_1(0) = \theta_2(0) = 0^\circ$, $p_1(0) = p_2(0) = 0 \text{ kg} \frac{\text{m}}{\text{s}}$ and the desired values as $\theta_{d1} = 175^\circ, \theta_{d2} = 45^\circ$. Hence, Figures 4 and 5 illustrate the results.

Table 3. Parameters of the robot.

| Parameter | Value | Unit |
|------------|-------|-----------------|
| m_1, m_2 | 0.25 | kg |
| l_1, l_2 | 2 | m |
| g | 9.81 | $\frac{m}{s^2}$ |

Table 4. Feedback controllers.

| Law of Control | Law of Control | Gains |
|----------------|--|--|
| Theorem 1 | See Equation (49) | $K_1 = K_2 = 1; K_{S1} = K_{S2} = 10$ |
| Theorem 2 | See Equation (55) | $K_{p1} = K_{p2} = 25; K_{v1} = K_{v2} = 25$ |
| PD | $\tau = K_p e(t) + K_v \frac{de(t)}{dt}$ | $K_p = K_v = \begin{bmatrix} 25 & 0 \\ 0 & 25 \end{bmatrix}$ |
| PID | $\tau = K_p e(t) + K_I \int e(t)dt + K_v \frac{de(t)}{dt}$ | $K_p = K_v = \begin{bmatrix} 25 & 0 \\ 0 & 25 \end{bmatrix}, K_I = \begin{bmatrix} 15 & 0 \\ 0 & 15 \end{bmatrix}$ |

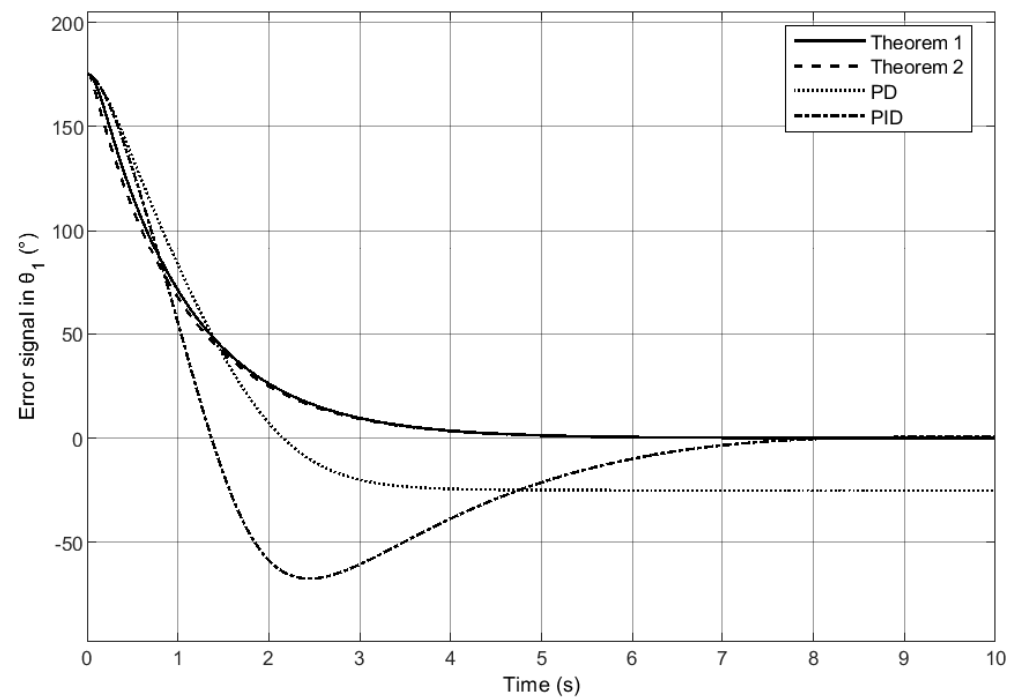


Figure 4. Error signals with diverse controllers in θ_1 for a two degrees-of-freedom robot.

In this example, the efficiency of the algorithm is proved instead of the traditional techniques. The dynamic equations are computed in an iterative form and are easy to develop, which is an advantage for new researchers in the robotic field. On the other hand, in accordance with the simulations, the controllers are faster than the typical laws of control in the industry and they are easy to program in hardware.

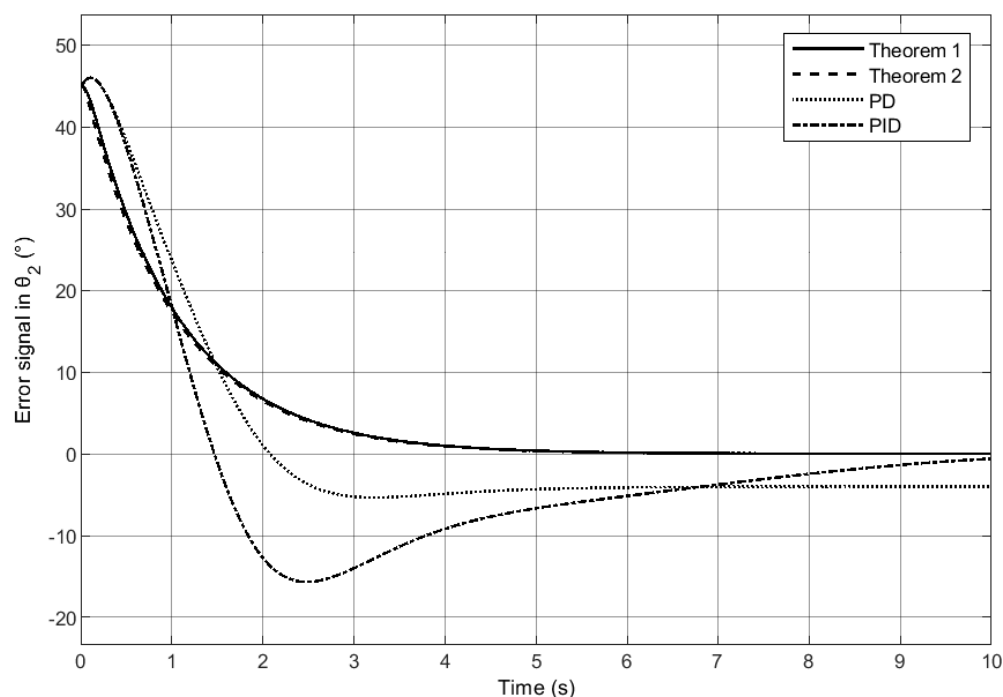


Figure 5. Error signals with diverse controllers in θ_2 for a two degrees-of-freedom robot.

6. Conclusions and Future Work

In the present article, the capacity of the proposed method for the modeling and control of robots in phase space is demonstrated. The dynamic equations, in phase space, of an articulated robot can be developed easily and iteratively (which cause it to be easy to program) using the suggested technique. In the examples, it is clearly seen that, with a few simple steps, the equations of motion of the system can be computed; otherwise, when using the traditional method, it is necessary to develop the Euler–Lagrange or Newton–Euler Equations (in addition, the kinematic model should be contemplated before) to later use the Legendre transform; this causes it to be a long and tedious process for new researchers in the field. Talking about the controllers, it is observed how easy it is to build controllers using the screw theory, such as the dynamic equations, where the control laws are obtained iteratively with a few steps. In the simulation, it was shown that the proposed controllers have a better performance than traditional techniques in phase space. Thus, it is illustrated that this technique can be used for the dynamic model and its control in the phase space.

On the other hand, this paper only contemplates the dynamic equations of the robot and its controller. However, the perturbations, unknown parameters, or other external elements are not added to the algorithm. Thus, the previous analysis and the dynamic equations of more complex robots will be taken into account for future work.

Author Contributions: Conceptualization, J.A.M.-H.; data curation, R.L.-P.; formal analysis, A.E.R.-M. and R.L.-P.; funding acquisition, R.B.-A.; investigation, J.A.M.-H.; methodology, J.A.M.-H.; project administration, J.A.M.-H. and R.L.-P.; resources, A.E.R.-M. and R.B.-A.; software, J.A.M.-H. and A.E.R.-M.; supervision, J.A.M.-H. and A.E.R.-M.; validation, R.L.-P. and R.B.-A.; visualization, A.E.R.-M. and R.B.-A.; writing—original draft, J.A.M.-H.; writing—review and editing, J.A.M.-H. and R.L.-P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

Appendix A

Appendix A.1. Robot Dynamics Using Screw Theory

The Lagrangian is written as:

$$\mathcal{L}(\theta, \dot{\theta}, t) = \sum_{j=1}^n \frac{1}{2} V_j^T N_j V_j - \tilde{g}^T \tilde{c}_j \tag{A1}$$

Using the Euler-Lagrange equation $\left(\tau_i = \frac{d}{dt} \left(\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \dot{\theta}_i} \right) - \frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \theta_i} \right)$. Hence:

$$\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \dot{\theta}_i} = \frac{\partial \left(\sum_{j=1}^n \frac{1}{2} V_j^T N_j V_j - \tilde{g}^T \tilde{c}_j \right)}{\partial \dot{\theta}_i} \tag{A2}$$

Here, $V_j = s_j \dot{\theta}_j$:

$$\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \dot{\theta}_i} = \frac{\partial \left(\sum_{j=1}^n \frac{1}{2} \left((s_j \dot{\theta}_j)^T N_j s_j \dot{\theta}_j \right) - \tilde{g}^T \tilde{c}_j \right)}{\partial \dot{\theta}_i} \tag{A3}$$

Therefore:

$$\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \dot{\theta}_i} = \sum_{j=i}^n s_i^T N_j V_j \tag{A4}$$

Now, differentiating with respect to time:

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \dot{\theta}_i} \right) = \sum_{j=i}^n \frac{d}{dt} \left(s_i^T N_j V_j \right) \tag{A5}$$

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \dot{\theta}_i} \right) = \sum_{j=i}^n \frac{d}{dt} \left(s_i^T \right) N_j V_j + s_i^T \frac{d}{dt} \left(N_j \right) V_j + s_i^T N_j \frac{d}{dt} \left(V_j \right) \tag{A6}$$

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \dot{\theta}_i} \right) = \sum_{j=i}^n V_j^T N_j ad(V_i) s_i - s_i^T ad^T(V_j) N_j V_j - s_i^T N_j ad(V_j) V_j + s_i^T N_j \dot{V}_j \tag{A7}$$

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \dot{\theta}_i} \right) = \sum_{j=i}^n V_j^T N_j [V_i, s_i] + s_i^T \{V_j, N_j V_j\} - s_i^T N_j [V_j, V_j] + s_i^T N_j \dot{V}_j \tag{A8}$$

where $\{V_j, N_j V_j\}$ represent the co-bracket. Considering $\{V_2, N_2 V_2\}^T s_1 = V_2^T N_2 [s_1, V_2]$:

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \dot{\theta}_i} \right) = \sum_{j=i}^n \dot{V}_j^T N_j s_i + V_j^T N_j [s_i, V_j] + V_j^T N_j [V_i, s_i] \tag{A9}$$

On the other hand:

$$\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \theta_i} = \frac{\partial \left(\sum_{j=1}^n \frac{1}{2} V_j^T N_j V_j - \tilde{g}^T \tilde{c}_j \right)}{\partial \theta_i} \tag{A10}$$

$$\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \theta_i} = \sum_{j=1}^n \frac{1}{2} \frac{\partial V_j^T}{\partial \theta_i} N_j V_j + V_j^T \frac{\partial N_j}{\partial \theta_i} V_j + V_j^T N_j \frac{\partial V_j}{\partial \theta_i} - \frac{\partial \tilde{g}^T \tilde{c}_j}{\partial \theta_i} \tag{A11}$$

$$\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \theta_i} = \sum_{j=i}^n V_j^T N_j ad(s_i) (V_j - V_i) + s_i^T \mathcal{G}_j - \frac{1}{2} \left(V_j^T ad^T(s_i) N_j V_j + V_j^T N_j ad(s_i) V_j \right) \tag{A12}$$

$$\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \theta_i} = \sum_{j=i}^n V_j^T N_j [s_i, V_j - V_i] + s_i^T \mathcal{G}_j - \frac{1}{2} \left(V_j^T N_j [s_i, V_j] - V_j^T \{s_i, N_j V_j\} \right) \tag{A13}$$

$$\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \theta_i} = \sum_{j=i}^n V_j^T N_j [s_i, V_j - V_i] + s_i^T \mathcal{G}_j + V_j^T N_j [V_j, s_i] \quad (\text{A14})$$

$$\frac{\partial \mathcal{L}(\theta, \dot{\theta}, t)}{\partial \theta_i} = \sum_{j=i}^n \mathcal{G}_j^T s_i - V_j^T N_j [s_i, V_i] \quad (\text{A15})$$

Finally, applying Equations (A9) and (A15) into the Euler–Lagrange equation:

$$\tau_i = \sum_{j=i}^n \dot{V}_j^T N_j s_i + V_j^T N_j [s_i, V_j] - \mathcal{G}_j^T s_i \quad (\text{A16})$$

Appendix A.2. Hamilton's Equations Using Screw Theory

Hamilton's equation in Equation (44) can be easily computed using Equation (A15). Thus, it can be illustrated by:

$$\frac{\partial \mathcal{H}(\theta, p, t)}{\partial \theta_i} = \sum_{j=i}^n \mathcal{P}_j^T [s_i, N_i^{-1} \mathcal{P}_i] - \mathcal{G}_j^T s_i \quad (\text{A17})$$

In addition to the foregoing, Hamilton's equation in Equation (45) can be computed using Equation (A17):

$$\dot{p}_i = \tau_i + \sum_{j=i}^n \mathcal{G}_j^T s_i - \mathcal{P}_j^T [s_i, N_i^{-1} \mathcal{P}_i] \quad (\text{A18})$$

References

- Goldstein, H. *Classical Mechanics*; Addison-Wesley: Boston, MA, USA, 1980.
- Taylor, J. *Classical Mechanics*; G—Reference, Information and Interdisciplinary Subjects Series; University Science Books: Melville, NY, USA, 2005.
- Peymani, E.; Fossen, T.I. A Lagrangian framework to incorporate positional and velocity constraints to achieve path-following control. In Proceedings of the 2011 50th IEEE Conference on Decision and Control and European Control Conference, Orlando, FL, USA, 12–15 December 2011; pp. 3940–3945. [\[CrossRef\]](#)
- Arimoto, S. *Control Theory of Non-Linear Mechanical Systems: A Passivity-Based and Circuit-Theoretic Approach*; Oxford Engineering Science Series; Clarendon Press: Oxford, UK, 1996.
- Záda, V.; Belda, K. Mathematical modeling of industrial robots based on Hamiltonian mechanics. In Proceedings of the 2016 17th International Carpathian Control Conference (ICCC), High Tatras, Slovakia, 29 May–1 June 2016; pp. 813–818. [\[CrossRef\]](#)
- Chi, J.; Yu, H.; Yu, J. Hybrid Tracking Control of 2-DOF SCARA Robot via Port-Controlled Hamiltonian and Backstepping. *IEEE Access* **2018**, *6*, 17354–17360. [\[CrossRef\]](#)
- Záda, V. Exponentially stable tracking control in terms of Hamiltonian mechanics. In Proceedings of the 2017 18th International Carpathian Control Conference (ICCC), Sinaia, Romania, 28–31 May 2017; pp. 483–487. [\[CrossRef\]](#)
- Hestenes, D. Hamiltonian Mechanics with Geometric Calculus. In *Proceedings of the Spinors, Twistors, Clifford Algebras and Quantum Deformations*; Oziewicz, Z., Jancewicz, B., Borowiec, A., Eds.; Springer: Dordrecht, The Netherlands, 1993; pp. 203–214.
- Bayro-Corrochano, E.; Zamora-Esquível, J. Differential and inverse kinematics of robot devices using conformal geometric algebra. *Robotica* **2007**, *25*, 43–61. [\[CrossRef\]](#)
- Dahab, E.A.E. A formulation of Hamiltonian mechanics using geometric algebra. *Adv. Appl. Clifford Algebr.* **2000**, *10*, 217–223. [\[CrossRef\]](#)
- Bayro-Corrochano, E.; Medrano-Hermosillo, J.; Osuna-González, G.; Uriostegui-Legorreta, U. Newton–Euler modeling and Hamiltonians for robot control in the geometric algebra. *Robotica* **2022**, *40*, 4031–4055. [\[CrossRef\]](#)
- Huang, Z.; Li, Q.; Ding, H. *Basics of Screw Theory*; Springer: Berlin, Germany, 2013; pp. 1–16. [\[CrossRef\]](#)
- Müller, A. Screw Theory—A forgotten Tool in Multibody Dynamics. *PAMM* **2017**, *17*, 809–810. [\[CrossRef\]](#)
- Ball, R. *A Treatise on the Theory of Screws*; Cambridge Mathematical Library; Cambridge University Press: Cambridge, UK, 1998.
- Toscano, G.S.; Simas, H.; Castelan, E.B.; Martins, D. A new kinetostatic model for humanoid robots using screw theory. *Robotica* **2018**, *36*, 570–587. [\[CrossRef\]](#)
- Hunt, K.H. Special configurations of robot-arms via screw theory. *Robotica* **1986**, *4*, 171–179. [\[CrossRef\]](#)
- Selig, J.M. *Geometric Fundamentals of Robotics (Monographs in Computer Science)*; Springer: Berlin/Heidelberg, Germany, 2004.
- Lynch, K.; Park, F. *Modern Robotics: Mechanics, Planning, and Control*; Cambridge University Press: Cambridge, UK, 2017.

19. Yi, B.J.; Kim, W.K. Screw-Based Kinematic Modeling and Geometric Analysis of Planar Mobile Robots. In Proceedings of the 2007 International Conference on Mechatronics and Automation, Harbin, China, 5–8 August 2007; pp. 1734–1739. [\[CrossRef\]](#)
20. Medrano-Hermosillo, J.A.; Lozoya-Ponce, R.; Ramírez-Quintana, J.; Baray-Arana, R. Forward Kinematics Analysis of 6-DoF Articulated Robot using Screw Theory and Geometric Algebra. In Proceedings of the 2022 XXIV Robotics Mexican Congress (COMRob), Mineral de la Reforma/State of Hidalgo, Mexico, 9–11 November 2022; pp. 1–6. [\[CrossRef\]](#)
21. Zou, Y.; Zhang, A.; Zhang, Q.; Zhang, B.; Wu, X.; Qin, T. Design and Experimental Research of 3-RRS Parallel Ankle Rehabilitation Robot. *Micromachines* **2022**, *13*, 950. [\[CrossRef\]](#) [\[PubMed\]](#)
22. Sun, T.; Lian, B.; Yang, S.; Song, Y. Kinematic Calibration of Serial and Parallel Robots Based on Finite and Instantaneous Screw Theory. *IEEE Trans. Robot.* **2020**, *36*, 816–834. [\[CrossRef\]](#)
23. Liang, Z.; Meng, S.; Changkun, D. Accuracy analysis of SCARA industrial robot based on screw theory. In Proceedings of the 2011 IEEE International Conference on Computer Science and Automation Engineering, Shanghai, China, 10–12 June 2011; Volume 3, pp. 40–46. [\[CrossRef\]](#)
24. Huang, Y.; Liao, Q.; Wei, S.; Guo, L. Research on Dynamics of a Bicycle Robot with Front-Wheel Drive by Using Kane Equations Based on Screw Theory. In Proceedings of the 2010 International Conference on Artificial Intelligence and Computational Intelligence, Sanya, China, 23–24 October 2010; Volume 1, pp. 546–551. [\[CrossRef\]](#)
25. Selig, J.M.; McAree, P.R. Constrained robot dynamics I: Serial robots with end-effector constraints. *J. Robot. Syst.* **1999**, *16*, 471–486. [\[CrossRef\]](#)
26. Cheng, J.; Bi, S.; Yuan, C.; Cai, Y.; Yao, Y.; Zhang, L. Dynamic Modeling Method of Multibody System of 6-DOF Robot Based on Screw Theory. *Machines* **2022**, *10*, 499. [\[CrossRef\]](#)
27. Qin, Q.; Gao, G. Screw Dynamic Modeling and Novel Composite Error-Based Second-order Sliding Mode Dynamic Control for a Bilaterally Symmetrical Hybrid Robot. *Robotica* **2021**, *39*, 1264–1280. [\[CrossRef\]](#)
28. Abaunza, H.; Chandra, R.; Özgür, E.; Ramón, J.A.C.; Mezouar, Y. Kinematic screws and dual quaternion based motion controllers. *Control Eng. Pract.* **2022**, *128*, 105325. [\[CrossRef\]](#)
29. Bayro-Corrochano, E. *Geometric Algebra Applications Vol. I: Computer Vision, Graphics and Neurocomputing*; Springer International Publishing: Berlin/Heidelberg, Germany, 2018.
30. Bayro-Corrochano, E. A Survey on Quaternion Algebra and Geometric Algebra Applications in Engineering and Computer Science 1995–2020. *IEEE Access* **2021**, *9*, 104326–104355. [\[CrossRef\]](#)
31. Ji, P.; Li, C.; Ma, F. Sliding Mode Control of Manipulator Based on Improved Reaching Law and Sliding Surface. *Mathematics* **2022**, *10*, 1935. [\[CrossRef\]](#)
32. Hestenes, D.; Sobczyk, G. *Clifford Algebra to Geometric Calculus: A Unified Language for Mathematics and Physics*; Fundamental Theories of Physics; Springer: Dordrecht, The Netherlands, 1987.
33. Perwass, C. *Geometric Algebra with Applications in Engineering*; Springer: Berlin, Germany, 2009; Volume 4. [\[CrossRef\]](#)
34. Lee, J. Velocity workspace analysis for multiple arm robot systems. *Robotica* **2001**, *19*, 581–591. [\[CrossRef\]](#)
35. Siciliano, B.; Sciavicco, L.; Villani, L.; Oriolo, G. *Robotics: Modelling, Planning and Control*; Advanced Textbooks in Control and Signal Processing; Springer: London, UK, 2010.
36. Craig, J. *Introduction to Robotics, Global Edition*; Pearson Education Limited: London, UK, 2021.
37. Selig, J.; Ding, X. A screw theory of static beams. In Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No.01CH37180), Maui, HI, USA, 29 October–3 November 2001; Volume 1, pp. 312–317. [\[CrossRef\]](#)
38. Utkin, V.I. *Sliding Modes in Control and Optimization*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 1992.
39. Khalil, H. *Nonlinear Systems*; Pearson Education; Prentice Hall: Kent, OH, USA, 2002.
40. Kelly, R. A tuning procedure for stable PID control of robot manipulators. *Robotica* **1995**, *13*, 141–148. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.