



Article Adaptive Control Design and Stability Analysis of Robotic Manipulators

Bin Wei

Department of Mechanical Engineering, York University, Toronto, ON M3J 1P3, Canada; binwei28@yorku.ca

Received: 9 October 2018; Accepted: 11 December 2018; Published: 14 December 2018



Abstract: In this paper, the author presents the adaptive control design and stability analysis of robotic manipulators based on two main approaches, i.e., Lyapunov stability theory and hyperstability theory. For the Lyapunov approach, the author presents the adaptive control of a 2-DOF (degrees of freedom) robotic manipulator. Furthermore, the adaptive control technique and Lyapunov theory are subsequently applied to the end-effector motion control and force control, as in most cases, one only considers the motion control (e.g., position control, trajectory tracking). To make the robot interact with humans or the environment, force control must be considered as well to achieve a safe working environment. For the hyperstability approach, a control system is developed through integrating a PID (proportional–integral–derivative) control system and a model reference adaptive control (MRAC) system, and also the convergent behavior and characteristics under the situation of the PID system, model reference adaptive control system, and PID+MRAC control system are compared.

Keywords: adaptive control; stability; Lyapunov; hyperstability; robotic manipulator

1. Introduction

Robotic mechanisms have been maturely employed in manufacturing industries [1–4]. In current robotic based manufacturing industries (e.g., in manufacturing assembly lines), robots are designed to work independently. In the situation where rapid changes in assembled products is required, traditional assembly robots are not capable to adapt to the rapid changes in assembled products. Robots working with humans is one of the effective solutions to the above situation. To make robots work with human operators, the most important issue is that robots must safely respond to contact forces while performing work. For example, a robot needs to stop or slow down when touched by a human operator, a robot must limit the amount of force it exerts while performing task, and a robot can be pushed out of the way by contact if necessary.

Furthermore, when a robot manipulator end-effector grasps an object to conduct work, it will change the dynamics of the robotic manipulator since the mass and initial properties of the grasped object may be unknown. Under this situation, traditional controls (e.g., PD (proportional–derivative) control) are not sufficient anymore. During the process of robotic mechanisms, the end-effector takes different weights of loads, usually the joint's output fluctuates along with time, and this phenomenon can deteriorate the end-effector's positioning accuracy. Rather, one approach to handle changing conditions is the adaptive control technique. In this paper, the adaptive control design and stability analysis for robotic manipulators based on two main approaches, i.e., Lyapunov stability theory and hyperstability theory, are presented. Regarding the adaptive control design and stability analysis for robotic manipulators based on Lyapunov stability theory, in most cases, one only considers the motion control. When a robot interacts with humans or the environment, force control also must be considered to achieve a safe working environment. For example, large force may damage the objects being manipulated; small force may not achieve the desired goal. The right amount of force is crucial for the human-robot interaction. Subsequently, the adaptive control technique is also applied to the

end-effector motion control and force control. The hybrid motion/force control concept can be traced back to [5]. In [5], Khabit systematically proposed the method of unified motion and force control of robotic manipulators [6]. The hybrid motion/force control is the combination of pure position control and pure force control. For example, we control force in the x direction and the control position in the y and z directions. In the hybrid motion/force control, one assumes that the environment is very stiff. In the literature, there is also an impedance control. Some recent impedance control studies for human-robot interaction can be found in [7–12]. The essential idea for impedance control is to change the position gains, i.e., impedance control can also be seen as the position control, but adjusting the position gains. This is the difference between the hybrid motion/force control and impedance control. Because in the hybrid motion/force control, one either has infinite stiffness, and regardless of the obstacles, if there are any in front of robots, this could cause damage (position control case) or have zero stiffness (force control case). So, in dealing with a soft environment, we want to set the stiffness value to be nonzero or infinite, therefore, the impedance control will become useful. The current industrial status is that in most robotic based industries, force control is not considered, only motion control is involved because of the large computation required, the parameter uncertainty in the dynamic model, and it is not intuitive to users, etc. Some industries use the "passive compliance" strategy. They insert some springs between the robot wrist and end-effector, or they use some soft materials in the end-effector. To make the robot interact with the environment, motion control and force control must be considered at the same time. In future manufacturing industries, force control also needs to be considered; in this way, robots and humans will cooperate with each other to complete a task.

Regarding adaptive control design and stability analysis for robotic manipulators based on hyperstability theory, in industries, often one employs the PID control system to every single joint of the serial robotic mechanisms to manipulate the robotic system. The ideal output performance is able to be obtained by altering the PID gains. Model reference adaptive control (MRAC) is a more advanced control technique, which was first developed by Landau [13], and it has been widely used after that [14–22]. As mentioned before, the necessity to employ the (model reference) adaptive control method to the robotic system is that the conventional control technique is not be able to make up for the load changes' situation. While if one employs the MRAC system, the above potential issue is able to be effectively rectified and therefore load changes' impacts are able to be addressed. For instance, the MRAC control system that was developed by Horowitz, and subsequently extended evolutions developed by other scholars [23], consists of an adaptation mechanism structure and a position feedback loop structure that is able to detect the error among the joint's ideal position and the joint's real position. This error is then served through the integral section of a PID-like control system, followed by the procedure that the position and velocity feedback values are being deducted from it. In this study, through integrating the PID controller and a MRAC control system, a hybrid control system is proposed based on the hyperstability approach.

The organization of this paper is as follows. Section 2 presents the adaptive control of robotic manipulators based on the Lyapunov approach, and the adaptive control technique and Lyapunov theory are subsequently applied to the end-effector motion control and force control. A control system is developed through integrating a PID control system and a model reference adaptive control (MRAC) system based on the hyperstability approach in Section 3. Section 4 gives the conclusion.

2. Lyapunov Approach

2.1. Background

There is an interconnection between the energy based Lagrange equation and the Lyapunov stability theory. Based on the adaptive control technique in [24] and Barbalat's lemma, one usually chooses the Lyapunov candidate function, *V*, as the quadratic form of some error, i.e.:

$$V = \frac{1}{2}S^2 \tag{1}$$

where *S* represent an error. Additionally, based on the Barbalat's lemma, if we choose a Lyapunov candidate function, *V*, that is lower bounded, $V \leq 0$ and V bounded, we will have the result of $V \to 0$. Subsequently, from $V \to 0$, one usually can achieve the condition that the actual value equals to the desired value, since V is usually the quadratic form of the error.

As an example, here, we use a mass-spring system (Figure 1) with no friction (it can also be seen as a 1-DOF robot) to illustrate the relationship and interconnection between the energy based Lagrange equation and the Lyapunov stability theory.



Figure 1. Mass-spring system.

The goal is to move the mass, *m*, from position x_0 to the desired position, x_d . The equation of motion is known as $m \stackrel{\bullet}{x} = f$.

The Lyapunov candidate function, *V*, is chosen as $V = \frac{1}{2}k_p(x - x_d)$, and $f = -\frac{\partial V}{\partial x} = -k_p(x - x_d)$, and therefore we have $m \overset{*>}{x} + k_p(x - x_d) = 0$. Figure 2 shows that if one considers the Lyapunov stability, it can be seen as a parabola. The minimum point in the bottom is the stable point, and every point on the parabola tends to approach the stable point.



Figure 2. Lyapunov candidate function and corresponding state.

From the Lagrange equation perspective, we have the Lagrange equation as:

$$\frac{d}{dt}\left(\frac{\partial K}{\partial x}\right) - \frac{\partial K}{\partial x} = f \tag{2}$$

where *K* represents the kinetic energy, with $f = -\frac{\partial V}{\partial x}$ and *V* represents the potential energy. After re-arranging Equation (2), we have the following:

$$\frac{d}{dt}\left(\frac{\partial K}{\partial x}\right) - \frac{\partial (K-V)}{\partial x} = 0 \tag{3}$$

The above Lagrange equation says that the kinetic energy and potential energy are on the left side of the equation, and on the right side of equation, there is no external force, which means the system is stable. To go one step further, even though the system is stable, it is clearly that the system will oscillate. To have an asymptotic stable system, we need to add a damping to the system, i.e.,

$$\frac{d}{dt}\left(\frac{\partial K}{\partial x}\right) - \frac{\partial (K-V)}{\partial x} = F_{damping} \tag{4}$$

If $F_{damping} \cdot x < 0$, we then will have an asymptotically stable system. For example, one can choose $F_{damping} = -k_v x$. Thus, we have the control as follows, $F = -k_p (x - x_d) - k_v x$, i.e., the PD control.

The reason that the PD controller works is that the it mimics the spring-damper system, and according to the global invariant set theorem, the system can globally tend to the stability point. Here, we will give detailed proof on why PD control works in the position control case. Before proceeding to the following, it should be known that the "I" term in the PID control can actually be seen as adaptive control.

It is known that the PD controller can be described as follows:

$$\tau = -K_P \tilde{q} - K_D \overset{\bullet}{q} \tag{5}$$

where $\tilde{q} = q - q_d$, q means the actual joint position, and q_d means the joint desired position. Considering the virtual physics, the PD controller can be seen as a virtual spring (P) and virtual damper (D). The reason why the PD controller works can be traced back to Lyapunov theory. The first step here is to have a Lyapunov candidate function as follows:

$$V = \text{kinetic energy} + \text{potential energy} = \frac{1}{2} \stackrel{\bullet}{q}^{T} H(q) \stackrel{\bullet}{q} + \frac{1}{2} \stackrel{\sim}{q}^{T} K_{P} \stackrel{\sim}{q}$$
(6)

where H(q) represents the inertia matrix. Therefore:

$$\overset{\bullet}{V} = \overset{\bullet}{q}^{T} (\tau - D\overset{\bullet}{q}) + \overset{\bullet}{q}^{T} K_{P} \tilde{q}$$
⁽⁷⁾

where Dq as a whole means viscous friction. Plugging the control law, $\tau = -K_P q - K_D q$, in the above equation, we have:

$$\overset{\bullet}{V} = -\overset{\bullet}{q}^{T}(K_{D} + D)\overset{\bullet}{q} \le 0$$
(8)

According to the global invariant set theorem, $\overset{\bullet}{V} = 0 \Rightarrow \overset{\bullet}{q} = 0 \Rightarrow H\overset{\bullet}{q} = -K_P \overset{\bullet}{q} \Rightarrow \overset{\bullet}{q} = -H^{-1}K_P \overset{\bullet}{q} \neq 0$ unless $\tilde{q} = 0$, in other words, unless q equals to q_d . This shows that starting anywhere, if we apply the PD controller, the system globally tends to $q = q_d$.

2.2. Adaptive Control of Robotic Manipulators

The control issue has been maturely developed in the last decades. PID control (most times PD control) is widely used in industries due to its simplicity. In controlling a robot end-effector position, industries use the PD controller in each joint of a robotic manipulator to control the robot end-effector goal position to accomplish a desired position control. The reason that the PD controller works in controlling a goal position of a robot end-effector in industries nowadays is illustrated above in detail, i.e., it mimics the spring-damper system, and according to the global invariant set theorem (an extended version of the Lyaponuv theorem), the system can globally tend to the stability point. However, if a robot needs to have a specified trajectory or fast motion control, the PD control approach is not good enough to handle the above situations since the PD controller does not tell you what is going on in the process of moving a robot. A more advanced control system is therefore required, for example, adaptive control. As an example, for a 2-DOF serial manipulator with revolute joints, as shown in Figure 3, we have the dynamic equation as follows:

$$H(q)\overset{\bullet\bullet}{q} + C(q,q)\overset{\bullet}{q} + D(q,q)\overset{\bullet}{q} + g(q) = \tau$$
(9)

where *q* is the joint angles, H(q) is the inertia matrix, C(q, q)q is the centripetal and Coriolis torques, D(q, q)q is the friction, and g(q) is the gravitational torques.

First, the Lyapunov candidate function is chosen as follows:

$$V = \frac{1}{2}S^T H(q)S \tag{10}$$

where $S = \overset{\bullet}{\widetilde{q}} + \lambda \widetilde{q}$ and $\widetilde{q} = q - q_d$, where λ is a positive constant number. Then one can calculate V as follows:

$$\overset{\bullet}{V} = S^{T}(\tau - H\overset{\bullet}{q_{r}} - C\overset{\bullet}{q_{r}} - D\overset{\bullet}{q} - g)$$
(11)

where q_r is defined as $q_r = q_d - \lambda \tilde{q}$. The above $-Hq_r - Cq_r - Dq - g$ can be written as $-y(q, q, q_r, q_r)a$, where $y(q, q, q_r, q_r)$ is a known function and a is an unknown constant vector, i.e., $-Hq_r - Cq_r - Dq - g$ can be written as a known function times an unknown constant:

$$\mathbf{\hat{V}} = S^T(\tau - y(q, \mathbf{\hat{q}}, \mathbf{\hat{q}}_r, \mathbf{\hat{q}}_r)a)$$
(12)

By choosing the control law:

$$\tau = y_a^{\wedge} - K_D S \tag{13}$$

where $\stackrel{\wedge}{a}$ means the estimate of *a*. We have:

$$\overset{\bullet}{V} = -S^T K_D S + S^T y_a^{\wedge} \tag{14}$$

It is noted that we have an extra $S^T y \hat{a}$. To eliminate this term, another term, $\frac{1}{2} \tilde{a}^T P^{-1} \tilde{a}$, is added to the above Lyapunov candidate function, then the Lyapunov candidate function can be rewritten as follows:

$$V = \frac{1}{2}S^{T}H(q)S + \frac{1}{2}\tilde{a}^{T}P^{-1}\tilde{a}$$
(15)

where *P* is a constant symmetrical positive definite matrix. Then:

$$\overset{\bullet}{V} = -S^T K_D S + S^T y \hat{a} + \frac{1}{2} \overset{\bullet}{a}^T P^{-1} \tilde{a}$$
 (16)

where $\tilde{a} = \hat{a} - a$. So, by making the last two terms equal to zero, i.e., by choosing the adaptation law as follows:

$$S^{T}y_{a}^{\wedge} + \frac{1}{2}a^{\wedge}P^{-1}a = 0$$

$$\Rightarrow a^{\wedge} = -PyS$$
(17)

Thus, we have:

$$\overset{\bullet}{V} = -S^T K_D S \tag{18}$$

Based on the Barbalet's lemma, when $\stackrel{\bullet}{V} \to 0 \Rightarrow S \to 0 \Rightarrow \stackrel{\bullet}{q} \to 0$ and $\stackrel{\bullet}{q} \to 0$.



Figure 3. Two-linkage mechanism.

2.3. The Role of Friction in Stability Analysis

As an extra note, sometimes the friction is useful in providing a stable condition in the system. The following section will illustrate how friction can be useful in the stability performance. It is known that $S = \dot{\tilde{q}} + \lambda \tilde{q} = \dot{q} - \dot{q_r}$, so in the dynamic equation, $D\tilde{q}$ can be written as $D\tilde{q} = D\tilde{q_r} + DS$, thus the dynamic equation can be rewritten as follows:

$$H(q)\overset{\bullet\bullet}{q} + C(q,q)\overset{\bullet}{q} + D(q,q)\overset{\bullet}{q}_r + g(q) = \tau - DS$$
⁽¹⁹⁾

Instead of throwing away the friction term, as illustrated above, one can use the friction term. In the previous approach, one has $Hq_r + Cq_r + Dq + g = ya$. In this case, one has $Hq_r + Cq_r + Dq_r + g = y_{new}a$. In the previous approach, $V = -S^T K_D S$, in this case, $V = -S^T (K_D + D)S$, and the control law and the adaptation law are therefore as follows:

$$\tau = y_{new} \hat{a} - K_D S \tag{20}$$

2.4. Limitations

There are, however, mainly two limitations of the above method: First, it is noticed that in $\tilde{a}(t) = \hat{a}(t) - a$, one assumes that the constant, *a*, does not change with time. This is only valid when *a* changes very slowly with respect to time, then we can assume *a* is constant, or when a robot grasps a load (because the moment when the robot grasps a load, we can assume that the time sets back to zero and *a* becomes constant again). This condition will not be valid in some situations where *a* is changing fast with respect to time (e.g., when a robot performs fast and constant load and unload tasks), if this is the case, then we need to remodel *a*. The second limitation is that the above approach is on the joint space control, as it is known that this requires pre-calculation of the inverse kinematic of the serial manipulators, which is very cumbersome.

The above stability analysis is based on the Lyapunov theory. Another approach in the stability analysis of adaptive control is based on the hyper-stability theory. An example can be found in [25].

2.5. Operational Space Control Case

The drawback of joint space control for a multi-DOF robotic manipulator is that we need to do the inverse kinematic calculation, which is very cumbersome work, as it is known that the inverse kinematic for a serial manipulator is complicated as compared to the forward kinematic calculation. Furthermore, since for the joint space control, first one needs to imagine the final goal position for the end-effector, and then we do the inverse kinematic calculation to determine what the corresponding joint angles need to be. Because of this, another drawback of joint space control is that in real time, when a robotic manipulator meets an unexpected obstacle, one needs to redo the whole motion planning in real time. Due to the above limitations, the operational space control is therefore put forward by Khatib [5].

By applying a force, *F*, at the end-effector, and through multiplying J^T , one can calculate the corresponding joint torque, τ , needed to achieve the task space control:

$$\tau = J^T F \tag{21}$$

In the operational space control, one has motion control and force control. Motion control is to control the motion of the end-effector, and force control is to control how much force the end-effector needs to have. In some situations (e.g., a manipulator cleans a window), motion control and force control need to be considered at the same time. In this way, the effector has the proper force at the effector so that the effector will not break the window. If a robot is interacting with the environment, we must deal with the force control and motion control. A unified motion and force control was

studied in [5]. Recent research in [26] was focused on the compliant motion and force control, where the robot end-effector was controlled by the contacting force, and this type of control does not need trajectory planning.

The operational space control has a similar control structure with the joint space control case. The difference is that only the joint variables are changed to the operational space variable. The end-effector equation of motion in the operational space is as follows [27]:

$$M_x(x) \overset{\bullet}{x} + V_x(x, \overset{\bullet}{x}) + G_x(x) = F$$
 (22)

where *x* is the end-effector position and orientation, $M_x(x)$ is the end-effector kinetic energy matrix, $V_x(x, \hat{x})$ is the end-effector centripetal and Coriolis forces, and $G_x(x)$ is the end-effector gravitational forces. By using the decoupling technique, one has the control structure as follows, as shown in Figure 4.



Figure 4. Decoupling control structure.

When the end-effector grasps an object to conduct work, it will change the dynamics of the robotic manipulator as the mass and initial properties of the object may be unknown. In this case, we will apply the adaptive control technique. By applying the adaptive control technique, first, the Lyapunov candidate function is chosen as follows:

$$V = \frac{1}{2}S^T M_x(x)S \tag{23}$$

where $S = \overset{\bullet}{\overset{\circ}{x}} + \lambda \overset{\circ}{x}$ and $\overset{\circ}{x} = x - x_d$. Then, one can calculate $\overset{\bullet}{V}$ as follows:

where $\dot{x}_r = \dot{x}_d - \lambda \tilde{x}$. The above $-M\dot{x}_r - V\dot{x}_r - G$ can be written as $-y(x, \dot{x}, \dot{x}_r, \dot{x}_r)a$, where $y(x, \dot{x}, \dot{x}_r, \dot{x}_r)$ is a known function and a is an unknown constant, i.e., $-M\dot{x}_r - V\dot{x}_r - G$ can be written as a known function times an unknown constant:

$$\mathbf{\hat{V}} = S^T (F - y(x, \mathbf{\hat{x}}, \mathbf{\hat{x}}_r, \mathbf{\hat{x}}_r)a)$$
(25)

By choosing the control law:

$$F = y\hat{a} - K_D S \tag{26}$$

we have:

$$\stackrel{\bullet}{V} = -S^T K_D S + S^T y_a^{\wedge} \tag{27}$$

Actuators 2018, 7, 89

It is noted that we have an extra $S^T y_a^{\wedge}$. To eliminate this term, another term, $\frac{1}{2} a^T P^{-1} a$, is added to the above Lyapunov candidate function, then the Lyapunov candidate function can be rewritten as follows:

$$V = \frac{1}{2}S^{T}M_{x}(x)S + \frac{1}{2}a^{T}P^{-1}a^{\widetilde{a}}$$
(28)

Then:

where $\tilde{a} = \hat{a} - a$. So by making the last two terms equal to zero, i.e., by choosing the adaptation law as follows:

$$S^{T}y_{a}^{\wedge} + \frac{1}{2} \overset{\bullet^{T}}{a} P^{-1} \overset{\bullet^{T}}{a} = 0$$
(30)

Thus, we have:

$$\overset{\bullet}{V} = -S^T K_D S \tag{31}$$

Based on the Barbalet's lemma, when $\stackrel{\bullet}{V} \to 0 \Rightarrow S \to 0 \Rightarrow \stackrel{\sim}{x} \to 0$ and $\stackrel{\sim}{x} \to 0$. The control structure is as follows, as shown in Figure 5.



Figure 5. Adaptive control structure.

2.6. Force Control Case

For the force control case, however, it is very different from the motion control case because the dynamic equation is very different. For the motion control case, the dynamic equation is $H(q)\stackrel{\bullet}{q} + C(q, q)\stackrel{\bullet}{q} + D(q, q)\stackrel{\bullet}{q} + g(q) = \tau$ (joint space control) or $M_x(x)\stackrel{\bullet}{x} + V_x(x, x) + G_x(x) = F$ (operational space control case). We can see that they have the same structure in the dynamic equation. Whereas for the force control case, the dynamic equation is $m\stackrel{\bullet}{x} + k_e x = f$. In the motion control case, when the robot end-effector grasps an object, the unknown parameters are in the $H(q)\stackrel{\bullet}{q} + C(q, q)\stackrel{\bullet}{q} + D(q, q)\stackrel{\bullet}{q} + g(q)$ or $M_x(x)\stackrel{\bullet}{x} + V_x(x, x) + G_x(x)$. In the force control case, the unknown parameters are in the $m\stackrel{\bullet}{x}$.

Regarding the force control, in most studies, the controlled force is considered as a constant and does not change with time. Here, the variable force control (force changing with time) is presented. The force equation is:

$$m \overset{\bullet \bullet}{x} + k_e x = m \overset{\bullet \bullet}{x} + f_e = f \tag{32}$$

where *x* is the end-effector/sensor displacement, *m* is the mass of the end-effector, and k_e is the gain of the virtual spring. By using the decoupling technique, one has the control structure as follows, as shown in Figure 6.



Figure 6. Decoupling control structure.

Similarly, where the end-effector grasps an object to conduct work, by applying the adaptive control technique, first, the Lyapunov candidate function is chosen as follows:

$$V = \frac{1}{2}S^T m S \tag{33}$$

where $S = \stackrel{\bullet}{f_e} + \lambda \stackrel{\sim}{f_e}$ and $\stackrel{\sim}{f_e} = f_e f_d$. Then, one can calculate $\stackrel{\bullet}{V}$ as follows:

$$\overset{\bullet}{V} = S^T (f - m x_r) \tag{34}$$

where $\dot{x_r} = f_d - \lambda f_e$. The above $-mx_r$ can be written as $-y(f_e, f_e, x_r, x_r)a$, where $y(f_e, f_e, x_r, x_r)$ is a known function and a is an unknown constant, i.e., $-mx_r$ can be written as a known function times an unknown constant:

$$\mathbf{\hat{V}} = S^T (f - y(f_e, f_e, \mathbf{\hat{x}}_r, \mathbf{\hat{x}}_r)a)$$
(35)

By choosing the control law:

$$f = y_a^{\wedge} - K_D S \tag{36}$$

we have:

It is noted that we have an extra $S^T y_a^{\wedge}$. To eliminate this term, another term, $\frac{1}{2} \tilde{a}^T P^{-1} \tilde{a}$, is added to the above Lyapunov candidate function, then the Lyapunov candidate function can be rewritten as follows:

$$V = \frac{1}{2}S^{T}mS + \frac{1}{2}\tilde{a}^{T}P^{-1}\tilde{a}$$
(38)

Then:

$$\overset{\bullet}{V} = -S^T K_D S + S^T y \hat{a} + \frac{1}{2} \overset{\bullet^1}{a} P^{-1} \overset{\bullet^1}{a}$$

$$(39)$$

where $\tilde{a} = \hat{a} - a$. So, by making the last two terms equal to zero, i.e., by choosing the adaptation law as follows:

$$S^T y_a^{\wedge} + \frac{1}{2} \overset{\bullet^1}{a} P^{-1} \overset{\bullet}{a} = 0 \tag{40}$$

Thus, we have:

$$\overset{\bullet}{V} = -S^T K_D S \tag{41}$$

Based on the Barbalet's lemma, when $V \to 0 \Rightarrow S \to 0 \Rightarrow \tilde{f_e} \to 0$ and $\tilde{f_e} \to 0$. The control structure is shown in Figure 7. The future work will focus on combining the adaptive motion control and force control.



Figure 7. Adaptive control structure.

3. Hyperstability Approach

3.1. PID+MRAC Control

Through incorporating the PID and model reference adaptive controls, a PID+MRAC structure is developed as shown in Figure 8. The synthesis of the PID+MRAC controller is based on the hyper-stability theory. The output from the manipulator compares to the output of the reference model, which will result in a difference. This difference is utilized by the adaptation section to tune the parameters, M and N, matrices of the dynamic model. The matrices, Fp, Fv, Cp, and Cv, are employed for the purpose of making the system stable [6]. In [23], the M and N matrices were treated as constant in the process of adaptation. For the 1-DOF link scenario, it is known that the inertia matrix, M, is not changing and also there is no nonlinear term (i.e., N matrix is zero) in its dynamic formulation, thus one is able to directly combine the PID and the model reference adaptive control system to develop a PID+MRAC controller, as shown in Figure 8. Nevertheless, for the multi degrees-of-freedom scenario, since the *M* and *N* matrices of their dynamic formulation are changing when robotic manipulators move and also due to the dynamic equations mismatch, one is not able to combine a PID and the model reference adaptive control directly as in indicated above. On the positive side, however, Sadegh [25] proposed an enhanced version of the model reference adaptive control system that was initially developed in [23], which can combine the PID and the enhanced version adaptive controller, and a combination of PID and the enhanced MRAC is therefore developed for multi degrees-of-freedom manipulators, as shown in Figure 9.

3.2. Modelling and Analysis of One-Linkage Scenario

First, a simple one-linkage mechanism (Figure 10) is employed as an example. For the aim of utilizing the PID control for the one-linkage mechanism scenario, the dynamic equation first needs to be modelled and obtained. Based on the Lagrange technique, one has:

$$\pi_1 = (m_1 l_1^2) \overset{\bullet\bullet}{\theta_1} + (m_1 l_1 \cos \theta_1) g = M \overset{\bullet\bullet}{\theta_1} + 0 + (m_1 l_1 \cos \theta_1) g = M \overset{\bullet\bullet}{\theta_1} + 0 + G g$$
(42)

where m_1 is the link mass, θ_1 is the angle between the link and x axis, and l_1 is the length of the link. By employing the PID, the controller output equals to the torque, i.e.:

$$K_p e + K_i \int e dt + K_d \overset{\bullet}{e} = \tau_1 \tag{43}$$

where error is $e = r_p - x_p$. One knows by the one-linkage mechanism's *M* and *N* matrices, the manipulator's output (i.e., the joint's acceleration) can be deduced:

$$K_{p}e + K_{i} \int edt + K_{d}e = \tau_{1} = (m_{1}l_{1}^{2})\theta_{1}^{\bullet \bullet} + (m_{1}l_{1}\cos\theta_{1})g$$
(44)

$$\Rightarrow \theta_1 = M^{-1}(K_p e + K_i \int e dt + K_d e)$$
(45)



Figure 8. Control system for 1-DOF manipulator.



Figure 9. Control system for more than 1-DOF linkage.

Thus, one obtains the joint 1's acceleration, by taking the integral regarding the time to deduce the joint 1's velocity and subsequently taking a second integral to deduce the joint 1's positon.

Regarding the MRAC, in a similar fashion with the PID control, the controller's output is able to be deduced as following:

$$\tau_1 = \hat{M} u + \hat{V} - F_p e - F_v \overset{\bullet}{e} \tag{46}$$

where $u = K_I \int (r_p - x_p) - K_p x_p - K_d x_v$, r_p is the desired input value, and x_p is the actual joint output, please see Figure 9 for reference. $\stackrel{\wedge}{M}$ and $\stackrel{\wedge}{V}$ are the parameters that have been adjusted.



Figure 10. One linkage mechanism.

The mechanism's dynamic formulation is:

$$\tau_1 = (m_1 l_1^2) \hat{\theta}_1^{\bullet} + (m_1 l_1 \cos \theta_1) g \tag{47}$$

So, the mechanism's output (i.e., the joint's acceleration) is deduced as following:

$$a = \overset{\bullet \bullet}{\theta_1} = M^{-1} (\hat{M}u + \hat{V} - F_p e - F_v \overset{\bullet}{e} - V)$$

$$\tag{48}$$

Furthermore:

$$\int_{0}^{T} y^{T}(t)w(t)dt = \int_{0}^{T} y^{T}(t)\widetilde{M}u(t)dt + \int_{0}^{T} y_{1}(t)x_{v}^{T}(N_{1} - N_{1})x_{v}dt$$
(49)

where $w(t) = -(M - \hat{M})u(t) - (V - \hat{V})$. The first term in Equation (49) is utilized to determine the adaptation algorithm for M. The second term is utilized to determine the adaptation algorithm for N. F_p and F_v are employed for the purpose of making the system stable according to the hyper-stability theory [6]. By proper selection of the positive definite gain matrices, Fp, Fv, Cp, and Cv, the transfer function, $G(s) = (C_v s + C_p)(Ms^2 + F_v s + F_p)^{-1}$, can be made strictly positive real. Based on the sufficient portion Popov's asymptotic hyperstability theorem [6], the system is asymptotic stable for all $T \ge 0$, $\int_0^T y(t)^T w(t) dt \ge -\gamma_0^2$.

After simulation, as illustrated in Figure 11, the color red in the figure denotes the joint's output when the MRAC control system is applied, and the color yellow in the figure denotes the joint's output when the hybrid control is applied, where Cp = 1, Cv = 20, Fp = 20, Fv = 20, adaptation gain is 5, and the proportional gain, integral gain, and derivative gain of the PID part are selected as 5, 1, and 3, respectively. Based on the result, it can be seen that for the PID control, it takes approximately 40 s to converge to zero. The proportional gain, integral gain, and derivative gain are selected as 5, 1, and 3, respectively. With respect to the MRAC system, it takes approximately 20 s to approach the expected spot, and compared to the PID's case, this is almost half of that period. With respect to the synthesized PID+MRAC system, it takes approximately 10 s to approach the expected spot, and compared to the MRAC's case, this further reduces to half of that time. In addition, one more difference among the MRAC and the synthesized PID+MRAC systems is that under the MRAC system case, it slowly approaches the expected spot while regarding the synthesized system, it first goes over the expected spot very quickly and then gradually comes back to the expected spot. After multiple tests and choosing different gains, a similar result can be obtained, as shown in Figure 12. Based on previous analysis, it can be seen that the convergent performance under the synthesized PID+MRAC system case beats the MRAC control's case, and the case under the MRAC system beats the PID system.



Figure 11. Joint output under three controls—scenario 1.



Figure 12. Joint output under three controls—scenario 2.

3.3. Modelling and Analysis of Two-DOF Link Scenario

Regarding the 2-DOF link scenario (Figure 13), according to the Lagrange technique, the dynamic equation is deduced as the following:

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = M \stackrel{\bullet \bullet}{\theta} + N = \begin{bmatrix} m_{11} & m_{12} \\ m_{12} & m_{22} \end{bmatrix} \begin{bmatrix} \bullet \bullet \\ \theta_1 \\ \bullet \bullet \\ \theta_2 \end{bmatrix} + \begin{bmatrix} n_{11} \\ n_{21} \end{bmatrix}$$
(50)

where:

$$m_{11} = (m_1 + m_2)l_1^2 + m_2l_2^2 + 2m_2l_1l_2\cos\theta_2, \ m_{12} = m_2l_2^2 + m_2l_1l_2\cos\theta_2, \ m_{22} = m_2l_2^2 + m_2l_1l_2\sin\theta_2, \ m_{22} = m_2l_2^2 + m_2l_1l_2\sin\theta_2, \ m_{22} = m_2l_2^2 + m_2l_1l_2\sin\theta_2, \ m_{23} = m_2l_2^2 + m_2l_1l_2\sin\theta_2, \ m_{24} = m_2l_2^2 + m_2l_1l_2\sin\theta_2, \ m_{25} = m_2l_2^2 + m_2l_1l_2\sin\theta_2, \ m_{26} = m_2l_2^2 + m_2l_1l_2\sin\theta_2, \ m_{26} = m_2l_2^2 + m_2l_1l_2\sin\theta_2, \ m_{26} = m_2l_2^2 + m_2l_1l_2\cos\theta_2, \ m_{26} = m_2l_2^2 + m_2l_1l_2\sin\theta_2, \ m_{26} = m_2l_2^2 + m_2$$

Through re-parametrization of the above formulation:

$$\begin{bmatrix} (m_{1}+m_{2})l_{1}^{2}+m_{2}l_{2}^{2}+2m_{2}l_{1}l_{2}\cos\theta_{2} & m_{2}l_{2}^{2}+m_{2}l_{1}l_{2}\cos\theta_{2} \\ m_{2}l_{2}^{2}+m_{2}l_{1}l_{2}\cos\theta_{2} & m_{2}l_{2}^{2} \end{bmatrix} \begin{bmatrix} u_{1} \\ u_{2} \end{bmatrix} + \begin{bmatrix} 2(-m_{2}l_{1}l_{2}\sin\theta_{2})\dot{\theta}_{1}^{2} + (-m_{2}l_{1}l_{2}\sin\theta_{2})\dot{\theta}_{2}^{2} \\ m_{2}l_{1}l_{2}\sin\theta_{2}\dot{\theta}_{1} \end{bmatrix} = W \cdot \begin{bmatrix} \Theta_{1} \\ \Theta_{2} \\ \Theta_{3} \end{bmatrix}$$
(51)

By choosing $\Theta_1 = (m_1 + m_2)l_1^2 + m_2l_2^2$, $\Theta_2 = m_2l_2^2$, $\Theta_3 = m_2l_1l_2$, one has:

$$\Rightarrow W = \begin{bmatrix} u_1 & u_2 & 2u_1\cos\theta_2 + u_2\cos\theta_2 - 2\theta_1\theta_2\sin\theta_2 - \theta_2\theta_2\sin\theta_2\\ 0 & u_1 + u_2 & u_1\cos\theta_2 + \theta_1\theta_1\sin\theta_2 \end{bmatrix}$$
(52)

Since:

$$\tau = W \begin{bmatrix} \Theta_1 \\ \Theta_2 \\ \Theta_3 \end{bmatrix} - F_v \cdot erv - F_p \cdot erp$$
(53)

Thus:

$$\Rightarrow \stackrel{\bullet \bullet}{\theta} = \left(W \begin{bmatrix} \Theta_1 \\ \Theta_2 \\ \Theta_3 \end{bmatrix} - (F_v \cdot erv) - (F_p \cdot erp) - N \right) / M$$
(54)

where *erv* and *erp* represent the joint output position error and velocity error. Once one obtains the joints' acceleration, taking the integral regarding the time to deduce the joints' velocity and subsequently taking a second integral to deduce the joints' positon.



Figure 13. Two-linkage mechanism.

The adaptation algorithm is derived as follows:

$$\int_{0}^{T} y^{T}(t) \widetilde{M}u(t) dt = \int_{0}^{T} \begin{bmatrix} y_{1} \\ y_{2} \end{bmatrix}^{T} \begin{bmatrix} \widetilde{m}_{11} & \widetilde{m}_{12} \\ \widetilde{m}_{12} & \widetilde{m}_{22} \end{bmatrix} \begin{bmatrix} u_{1} \\ u_{2} \end{bmatrix} dt
= \int_{0}^{T} [y_{1}, y_{2}] \begin{bmatrix} \widetilde{m}_{11} & \widetilde{m}_{12} \\ \widetilde{m}_{12} & \widetilde{m}_{22} \end{bmatrix} \begin{bmatrix} u_{1} \\ u_{2} \end{bmatrix} dt
= \int_{0}^{T} \widetilde{m}_{11} y_{1} u_{1} dt + \int_{0}^{T} \widetilde{m}_{12} (y_{1} u_{2} + y_{2} u_{1}) dt + \int_{0}^{T} \widetilde{m}_{22} y_{2} u_{2} dt$$
(55)

So, if one chooses $\frac{d}{dt} m_{11}^{\wedge}(t) = \frac{d}{dt} m_{11}^{\sim}(t) = k_{m11} y_1 u_1$, a stable system can therefore be achieved [6]. Applying the same analysis to the other two terms, the following can be obtained:

$$\frac{d}{dt}\hat{m}_{12}(t) = \frac{d}{dt}\tilde{m}_{12}(t) = k_{m12}(y_1u_2 + y_2u_1), \ \frac{d}{dt}\hat{m}_{22}(t) = \frac{d}{dt}\tilde{m}_{22}(t) = k_{m22}y_2u_2$$
(56)

The derivation for *M* is completed, and by resorting to the same technique, one is able to determine the adaptation algorithm for *N* as follows:

$$\frac{d}{dt}\hat{n}_{12}(t) = \frac{d}{dt}\hat{n}_{12}(t) = k_{n12}(2y_1x_{v1}x_{v2} - y_2x_{v1}^2)$$
(57)

The results show the synthesized PID+MRAC case converges quicker as compared to the model reference adaptive control case, and both the synthesized PID+MRAC and model reference adaptive

control cases converge quicker as compared to the PID case, as shown in Figure 14. Through employing the same procedure, it can be extended to the cases where the serial robotic mechanisms have multiple degrees of freedom.



Figure 14. Joints' 1 and 2 outputs under three controls.

4. Conclusions

The adaptive control design and stability analysis based on two main approaches, i.e., Lyapunov stability theory and hyperstability theory, were studied here. With respect to the Lyapunov approach, the adaptive control of a 2-DOF robotic manipulator was presented. The adaptive control technique was subsequently applied to the end-effector motion control and force control. With respect to the hyperstability approach, a control system was developed through integrating a PID and a MRAC, and also the convergent behavior and characteristics under the situation of the PID, MRAC, and PID+MRAC were compared. The outcome indicates the new enhanced PID+MRAC converges quicker than that of the model reference adaptive control. The experiments and comparisons with other hybrid controllers will be the topic of future work. The experiment will be conducted by using Simulink and dSpace. Motors and motor controllers will be purchased [28], and a multi-DOF serial robot will be set up and built as an illustration. The reason a new robot should be built instead of using a ready-on-market robot is that the control system for most of those robots are built-in controllers.

As an extra note, learning control was developed after the development of adaptive control of robotic manipulators, but it is still in its early development stage. Learning control is used to handle those complex models through learning those complex models, which sometimes have the characteristic of repetitive motions (e.g., most robotic machines used in factories nowadays have repetitive motions) rather than computing them. The main idea behind the leaning control approach is that developing a control approach such that those complex models and uncertainties (e.g., friction at joints) are not needed to model mathematically, but through learning them.

Funding: This research received no external funding.

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

References

- 1. Liang, Q.; Zhang, D.; Wu, W. Methods and Research for Multi-Component Cutting Force Sensing Devices and Approaches in Machining. *Sensors* **2016**, *16*, 1926. [CrossRef] [PubMed]
- Zhang, D.; Wei, B. Study on Payload Effects on the Joint Motion Accuracy of Serial Mechanical Mechanisms. Machines 2016, 4, 21. [CrossRef]
- 3. Li, Z.; Barenji, A.; Huang, G. Toward a Block-chain Cloud Manufacturing System as a Peer to Peer Distributed Network Platform. *Robot. Comput. Integr. Manuf.* **2018**, *54*, 133–144. [CrossRef]

- 4. Blanes, C.; Mellado, M.; Beltran, P. Novel Additive Manufacturing Pneumatic Actuators and Mechanisms for Food Handling Grippers. *Actuators* **2014**, *3*, 205–225. [CrossRef]
- 5. Khatib, O. A unified approach for motion and force control of robot manipulators: The operational space formulation. *IEEE J. Robot. Autom.* **1987**, *3*, 45–53. [CrossRef]
- 6. Landau, Y.D. Adaptive Control: The Model Reference Approach; Marcel Dekker: New York, NY, USA, 1979.
- 7. Song, A.; Pan, L.; Xu, G. Adaptive motion control of arm rehabilitation robot based on impedance identification. *Robotica* **2015**, *33*, 1795–1812. [CrossRef]
- 8. Koivumäki, J.; Mattila, J. Stability-guaranteed impedance control of hydraulic robotic manipulators. *IEEE/ASME Trans. Mechatron.* **2017**, *22*, 601–612. [CrossRef]
- 9. Xu, G.; Song, A. Adaptive impedance control for upper-limb rehabilitation robot using evolutionary dynamic recurrent fuzzy neural network. *J. Intell. Robot. Syst.* **2011**, *62*, 501–525. [CrossRef]
- Li, P.; Ge, S.S.; Wang, C. Impedance control for human-robot interaction with an adaptive fuzzy approach. In Proceedings of the 2017 29th Chinese Control and Decision Conference (CCDC), Chongqing, China, 28–30 May 2017; pp. 5889–5894.
- 11. Li, Z.; Liu, J.; Huang, Z.; Peng, Y.; Pu, H.; Ding, L. Adaptive Impedance Control of Human–Robot Cooperation Using Reinforcement Learning. *IEEE Trans. Ind. Electron.* **2017**, *64*, 8013–8022. [CrossRef]
- 12. Sharifi, M.; Behzadipour, S.; Vossoughi, G. Nonlinear model reference adaptive impedance control for human–robot interactions. *Control Eng. Pract.* **2014**, *32*, 9–27. [CrossRef]
- 13. Dubowsky, S.; Desforges, D. The application of model-referenced adaptive control to robotic manipulators. *J. Dyn. Syst. Meas. Control* **1979**, *101*, 193–200. [CrossRef]
- 14. Cao, C.; Hovakimyan, N. Design and Analysis of a Novel L1 Adaptive Control Architecture with Guaranteed Transient Performance. *IEEE Trans. Autom. Control* **2008**, *53*, 586–591. [CrossRef]
- 15. Jain, P.; Nigam, M.J. Design of a Model Reference Adaptive Controller Using Modified MIT Rule for a Second Order System. *Adv. Electron. Electr. Eng.* **2013**, *3*, 477–484.
- 16. Nguyen, N.; Krishnakumar, K.; Boskovic, J. An Optimal Control Modification to Model-Reference Adaptive Control for Fast Adaptation. In Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit, Honolulu, Hawaii, 18–21 August 2008; pp. 1–19.
- 17. Idan, M.; Johnson, M.D.; Calise, A.J. A Hierarchical Approach to Adaptive Control for Improved Flight Safety. *AIAA J. Guid. Control Dyn.* **2002**, *25*, 1012–1020. [CrossRef]
- Li, X.; Cheah, C.C. Adaptive regional feedback control of robotic manipulator with uncertain kinematics and depth information. In Proceedings of the American Control Conference, Montreal, QC, Canada, 27–29 June 2012; pp. 5472–5477.
- 19. Rossomando, F.G.; Soria, C.; Patiño, D.; Carelli, R. Model reference adaptive control for mobile robots in trajectory tracking using radial basis function neural networks. *Latin Am. Appl. Res.* **2011**, *41*, 177–182.
- 20. Sharifi, M.; Behzadipour, S.; Vossoughi, G.R. Model reference adaptive impedance control in Cartesian coordinates for physical human–robot interaction. *Adv. Robot.* **2014**, *28*, 1277–1290. [CrossRef]
- 21. Ortega, R.; Panteley, E. L1—Adaptive Control Always Converges to a Linear PI Control and Does Not Perform Better than the PI. In Proceedings of the 19th IFAC World Congress, Cape Town, South Africa, 24–29 August 2014; pp. 6926–6928.
- 22. Horowitz, R.; Tomizuka, M. An Adaptive Control Scheme for Mechanical Manipulators—Compensation of Nonlinearity and Decoupling Control. *J. Dyn. Syst. Meas. Control* **1986**, *108*, 1–9. [CrossRef]
- 23. Sadegh, N.; Horowitz, R. Stability Analysis of an Adaptive Controller for Robotic Manipulators. In Proceedings of the 1987 IEEE International Conference on Robotics and Automation, Raleigh, NC, USA, 31 March–3 April 1987; pp. 1223–1229.
- 24. Slotine, J.E.; Li, W. On the adaptive control of robotic manipulators. *Int. J. Robot. Res.* **1987**, *6*, 49–58. [CrossRef]
- 25. Sadegh, N.; Horowitz, R. Stability and Robustness Analysis of a Class of Adaptive Controllers for Robotic Manipulators. *Int. J. Robot. Res.* **1990**, *9*, 74–92. [CrossRef]
- 26. Sentis, L.; Park, J.; Khatib, O. Compliant control of multi-contact and center of mass behaviors in humanoid robots. *IEEE Trans. Robot.* **2010**, *26*, 483–501. [CrossRef]

- 27. Craig, J.J. *Introduction to Robotics: Mechanics and Control*, 3rd ed.; Pearson/Prentice Hall: New York, NY, USA, 2005.
- 28. Candelas, F.; García, G.; Puente, S. Experiences on using Arduino for laboratory experiments of automatic control and robotics. *IFAC-PapersOnLine* **2015**, *48*, 105–110. [CrossRef]



© 2018 by the author. 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 (http://creativecommons.org/licenses/by/4.0/).