ROBOT CONTROL AND PROGRAMMING

Haruhisa Kawasaki
Faculty of Engineering, Gifu University, JAPAN

Keywords: Adaptive control, Cartesian space, Compliance control, Dynamic control, Dynamics, Force control, Force feedback glove, Grasp planning, Haptic interface, Hybrid position/force control, Impedance control, Interaction control, Joint space, Kinematics, Lyapunov-like lemma, Lyapunov stability theorem, Manipulator, Motion control, Off-line programming, On-line programming, PD control, PID control, Path planning, Robot programming language, Servo control, Teaching by showing, Teaching playback, Virtual reality, World model

Contents

Summary

This chapter presents a perspective of the elements of control and programming of robot manipulators. The primary difference between industrial robots and automated machines with a fixed program is that the robots are adaptable to variation of objects in the work environment and to changes of tasks by re-programming. In order to make robots execute

various tasks such as welding, assembling, grinding, and so on, it is first necessary to enhance the robot control technologies. Moreover, programming technologies for instructing the motions of robots play an important role in making robots user friendly for humans.

In view of constraints in the environment, robot control technologies are categorized into motion control in free space and interaction control in constrained space. As motion controls, servo controls such as the joint space servo and the Cartesian space servo, dynamic controls such as the computed torque control and the Cartesian based inverse dynamics control, and adaptive control in joint space are typical methods. Interaction control schemes can be grouped into two types: indirect force control and direct force control. Impedance control is a typical indirect force control.

Hybrid position/force control and adaptive hybrid position/force control are direct force controls. The robot programming methods are categorized into on-line programming and off-line programming. As advanced off-line programming, a graphical task-level robot language system and robot teaching in a virtual reality environment are expected to be developed because of their highly user friendly interfaces.

1. Introduction

Robot control is an essential technology that enables a robot to move precisely and adaptively. The problem of robot control is formulated to include determination of the joint torques to be generated by the joint actuators, so as to guarantee the execution of the robot task while satisfying given transient and steady state requirements. In view of constraints within the environment, the robot controls are categorized into motion control in free space and interaction control in constrained space.

The motion control is the most essential of robot manipulator controls. One of the basic requirements for the successful completion of a robot task is the ability to handle an interaction between the end-effector of the robot and the environment. The contact force at the manipulator's end-effector should be controlled at the desired force, because a large force error causes damages of the manipulator and the manipulated object.

Sophistication of the interface between the human user and the industrial robot is becoming extremely important as robots are applied to more and more demanding industrial applications. The robot manipulators differ from fixed automation chiefly by their ability to for re-programming, which gives the robots the flexibility to be useful for various tasks. With regard to direct use of robots at the programming, robot programming technologies are categorized into on-line programming, which means a robot is located on the automated line and used at the programming, and off-line programming, which means a robot is located on the automated line but is not used at the programming.

The on-line programming is an early-established technology, opted for because of its simplicity. In the present-day off-line programming, a robot user is offered a robot language system with graphical user interface. In present-day robot teaching technologies, a graphical task-level robot language in small batch production is practical and easy for low-skill users. Teaching in a virtual reality environment, in which the human operator, rather than the robot, performs the task, the motion intention of the operator is analyzed, and robot instructions are generated from the analysis automatically; this is advanced off-line programming. Herein, these robot control and programming technologies are presented.

2. Robot Dynamics

In general, the equations of motion of *n* degrees of freedom (DOF) of the robot manipulator in joint space are represented by

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} - \mathbf{f}_r \quad (1)$$

where $\mathbf{q} \in \mathbb{R}^n$ is the joint variable vector, $\boldsymbol{\tau} \in \mathbb{R}^n$ is the joint torque vector, $\mathbf{f}_r \in \mathbb{R}^n$ is the friction vector at the joints, $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n\times n}$ is the manipulator inertia matrix, $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}$ is the quadratic velocity term, and $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^n$ is the gravity acceleration vector. This dynamic model has the following three properties:

Property 1: $\mathbf{M}(\mathbf{q})$ is symmetric and positive definite.

Property 2: For suitable definition of $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$, $\dot{\mathbf{M}}(\mathbf{q}) - 2\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$ is a skew-symmetric matrix.

Property 3: The equations of motion are linear with respect to suitable dynamic parameters; that is,

$$\mathbf{M}(\mathbf{q})\boldsymbol{\alpha} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\mathbf{v} + \mathbf{g}(\mathbf{q}) = \mathbf{Y}(\mathbf{q},\dot{\mathbf{q}},\mathbf{v},\boldsymbol{\alpha})\boldsymbol{\sigma} \quad (2)$$

where $\sigma \in \mathbb{R}^p$ is a dynamic parameter vector, $\mathbf{v}$ and $\boldsymbol{\alpha} \in \mathbb{R}^n$ are any velocity and acceleration vectors, and $\mathbf{Y}(\mathbf{q},\dot{\mathbf{q}},\mathbf{v},\boldsymbol{\alpha}) \in \mathbb{R}^{n\times p}$ is the regressor with respect to σ.

The property 2 is referred to as passivity. These properties are utilized in the stability analysis of robot controls. Moreover, the equations of motion in Cartesian space have the same three properties. (see *Robot Kinematics and Dynamics*).

Since the actuator that generates the joint torques also has dynamic characteristics, the dynamic characteristics of the actuator should be taken into consideration at a design of control input. The most typical actuator is the DC servomotor. Its circuit, as shown in Figure 1, is described by

$$E_{\mathrm{M}} = R_{\mathrm{M}} i_{\mathrm{M}} + L_{\mathrm{M}} \dot{i}_{\mathrm{M}} - K_{\mathrm{e}} \dot{\theta}_{\mathrm{M}}$$

and the output motor torque is given by

$$\tau_{\mathrm{M}} = K_{\mathrm{T}} i_{\mathrm{M}}$$

where E_{M} is the voltage source, i_{M} is a motor current, θ_{M} is a motor angle, R_{M} is a motor electric resistance, L_{M} is a motor inductance, K_{e} is a back electromotive force constant, and K_{T} is a motor torque constant.

It is generally desirable to control the output torque with a current amplifier motor driver, which senses the current through the motor armature and continuously adjusts the voltage source E_{M} so that a desired current i_{M} flows through the motor armature. This is accomplished by large current feedback in the motor driver. In general, the motor is connected through a gear reduction to a joint. A relation between the joint load torque τ and motor output torque τ_{M} is represented in terms of motor variables by

$$\tau_{\mathrm{M}} = J_{\mathrm{M}}\ddot{\theta}_{\mathrm{M}} + \frac{1}{\gamma}\tau ,$$

or in terms of joint variables by

$$\tau_{\mathrm{L}} = \gamma\tau_{\mathrm{M}} = \gamma^2 J_{\mathrm{M}}\ddot{\theta} + \tau$$

where J_{M} is a motor inertia, γ is a reduction ratio, θ is a joint angle and τ_{L} is a motor output torque at the link side. This yields the equation of the manipulator, rewritten as

$$(\mathbf{M}(\mathbf{q}) + \mathbf{\Gamma}^{\mathrm{T}}\mathbf{M}_{\mathrm{M}}\,\mathbf{\Gamma})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau}_{\mathrm{L}} - \mathbf{f}_r \qquad (3)$$

where $\mathbf{\Gamma}$ is a reduction ratio matrix, $\mathbf{M}_{\mathrm{M}}$ is a diagonal motor inertia matrix, and $\boldsymbol{\tau}_{\mathrm{L}}$ is a motor output torque vector at the link side. Since the motor inertia $\mathbf{\Gamma}^{\mathrm{T}}\mathbf{M}_{\mathrm{M}}\,\mathbf{\Gamma}$ is positive definite and constant, this dynamics also has the above-mentioned three properties. The term $\mathbf{M}(\mathbf{q}) + \mathbf{\Gamma}^{\mathrm{T}}\mathbf{M}_{\mathrm{M}}\,\mathbf{\Gamma}$ is sometimes called the effective inertia matrix. It is noted that in a highly geared joint, the motor inertia can be a significant portion of the combined effective inertia. This may allow us to make the assumption that the effective inertia matrix is a constant. Hereafter, for simplicity of description, it is assumed that the actuator is an ideal generator of joint torque.

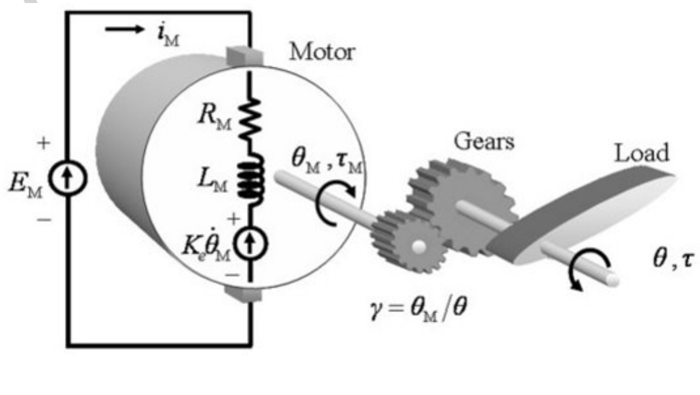

Figure 1: The circuit and mechanical model of a DC servomotor with gear reduction

3. Motion Control

Equations of motion of the manipulator are highly non-linear because the manipulator motion is affected by the gravitational acceleration, the centrifugal force, the Coriolis force, and the friction at the joints. The aim of manipulator motion control is to drive the end-effector to a desired point or to follow it to a desired trajectory in free space. Typical motion control techniques are presented; *i.e.*, servo control in joint space, servo control in task space, dynamic control, and adaptive control.

3.1. Servo Control

The servo control system is a control system that makes the position and/or velocity of a mechanical motion system follow desired values. In robot motion control, the desired values are given by the joint variables in joint space or by the position and orientation (pose) of the end-effector in Cartesian space. The former is called a joint servo control system, the latter a Cartesian servo control system.

3.1.1. Servo Control in Joint Space

As servo control techniques in joint space, PD control, PD control with gravity compensator, and PID control are presented.

PD control

For given desired constant joint variable $\mathbf{q}_{\mathrm{d}} = \left[q_{\mathrm{d}1}, \cdots, q_{\mathrm{d}n}\right]^T$, the most simplest control scheme is an independent joint control, the control law of which is given by

$$\boldsymbol{\tau}(t) = \mathbf{K}_{\mathrm{D}}\dot{\mathbf{e}}(t) + \mathbf{K}_{\mathrm{P}}\mathbf{e}(t) \qquad (4)$$

where $\mathbf{e}(t) = \mathbf{q}_{\mathrm{d}} - \mathbf{q}(t)$ is the position error vector, and $\mathbf{K}_{\mathrm{P}} = diag(K_{\mathrm{p}1}, \cdots, K_{\mathrm{p}n})$ and $\mathbf{K}_{\mathrm{D}} = diag(K_{\mathrm{D}1}, \cdots, K_{\mathrm{D}n}) \in \mathbb{R}^{n\times n}$ are the proportional and derivative feedback gain matrices, respectively. This is called PD control in joint space. When the joint velocities are adequately low, such that the effects of the centrifugal force and the Coriolis force can be ignored, that is, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} \cong \mathbf{0}$; the gravitational force can be ignored, that is, $\mathbf{g}(\mathbf{q}) \cong \mathbf{0}$; and the manipulator inertia matrix is assumed to be represented by $\mathbf{M}(\mathbf{q}) = diag(m_{11}, \cdots, m_{nn}) \equiv \tilde{\mathbf{M}}$, which means that the interaction between the joints can be ignored, then the equations of errors are represented by the second order linear constant coefficient differential equation

$$\tilde{\mathbf{M}}\ddot{\mathbf{e}}(t) + \mathbf{K}_{\mathrm{D}}\dot{\mathbf{e}}(t) + \mathbf{K}_{\mathrm{P}}\mathbf{e}(t) = \mathbf{0}.$$

If the feedback gains are settled by $K_{\mathrm{P}i} = m_{ii}\omega_i^2$ and $K_{\mathrm{D}i} = 2\xi_i\omega_i m_{ii}$, an error equation of the *i*-th joint variable is

$$\ddot{e}_i(t) + 2\xi_i\omega_i\dot{e}_i(t) + \omega_i^2 e_i(t) = 0 \qquad (5)$$

where ξ_i is a damping coefficient and ω_i is a natural angular frequency. This means e_i converges to zero as t approaches infinity. As shown in Figure 2, this control scheme consists of n independent single-input, single-output control systems. This is the design approach presently adopted by most industrial robot suppliers.

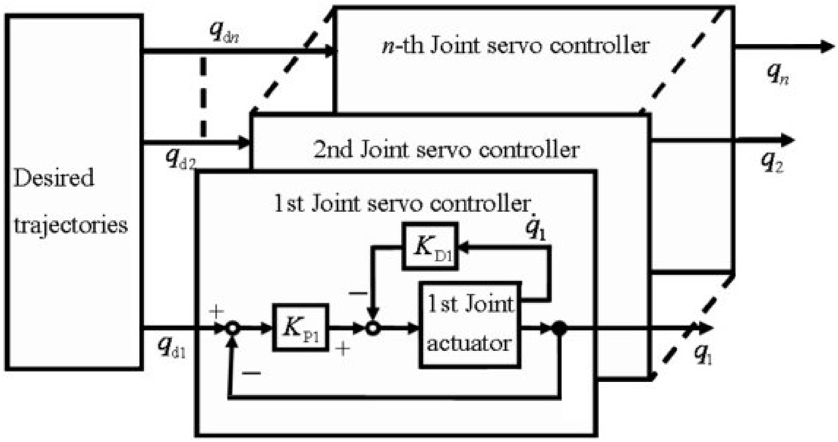

Figure 2: Joint servo controller.

PD control with gravity compensation

When the effect of gravity acceleration cannot be ignored, PD control with gravity compensation is adopted. Its control law is given by

$$\boldsymbol{\tau}(t) = \mathbf{K}_{\mathrm{D}}\dot{\mathbf{e}}(t) + \mathbf{K}_{\mathrm{P}}\mathbf{e}(t) + \hat{\mathbf{g}}(\mathbf{q}) \qquad (6)$$

where $\mathbf{K}_{\mathrm{D}} > 0$ and $\mathbf{K}_{\mathrm{P}} > 0$ are symmetric positive definite feedback gain matrices, and $\hat{\mathbf{g}}(\mathbf{q}) \in \mathbb{R}^n$ is a computed gravity term which is an estimate of $\mathbf{g}(\mathbf{q})$. Asymptotic stability of the closed system is proved based on the Lyapunov stability theorem on the assumption that $\hat{\mathbf{g}}(\mathbf{q}) = \mathbf{g}(\mathbf{q})$.

The closed system in which the PD control with gravity compensation is applied is represented by

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{K}_{\mathrm{D}}\dot{\mathbf{e}}(t) - \mathbf{K}_{\mathrm{P}}\mathbf{e}(t) = \mathbf{0}. \qquad (7)$$

Choose the following positive definite quadratic form as a Lyapunov function candidate:

$$V(t) = \frac{1}{2}\left\{\dot{\mathbf{q}}^{\mathrm{T}}\mathbf{M}(\mathbf{q})\dot{\mathbf{q}} + \mathbf{e}^{\mathrm{T}}\mathbf{K}_{\mathrm{P}}\mathbf{e}\right\}. \qquad (8)$$

An energy-based interpretation of (8) reveals a first term expressing the system kinematic energy and a second term expressing the potential energy stored in the system of equivalent stiffness $\mathbf{K}_{\mathrm{P}}$ provided by the feedback loop. Since $\mathbf{M}(\mathbf{q}) - 2\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$ is skew symmetric, a time derivative of (8) is found to be

$\dot{V}(t) = -\dot{\mathbf{q}}^{\mathrm{T}}\mathbf{K}_{\mathrm{D}}\dot{\mathbf{q}} \le 0$.

Hence, $V(t)$ is a Lyapunov function, which means that the system reaches an equilibrium posture satisfying $\dot{V}(t) = 0$. This means $\dot{\mathbf{q}}(t) = \mathbf{0}$. At the equilibrium satisfying (7), $\mathbf{q}(t) = \mathbf{q}_{\mathrm{d}}$ is the equilibrium posture. If $\mathbf{q}(t) \ne \mathbf{q}_{\mathrm{d}}$, then $\dot{\mathbf{q}}(t) \ne \mathbf{0}$ and $V(t) < 0$. Hence, $\mathbf{q}$ tends toward $\mathbf{q}_{\mathrm{d}}$ as t tends toward infinity. This shows that any manipulator equilibrium posture is globally asymptotically stable under PD control with gravity compensation. *(see Lyapunov Stability).*

PID control

If the computation of gravity term is not exact, and/or the friction at the joints can be ignored, steady state errors of joint variables will arise. In order to reduce the steady state errors, PID control is adopted. Its control law is

$$\boldsymbol{\tau}(t) = \mathbf{K}_{\mathrm{D}}\dot{\mathbf{e}}(t) + \mathbf{K}_{\mathrm{P}}\mathbf{e}(t) + \mathbf{K}_{\mathrm{I}}\int_0^t \mathbf{e}(t)dt \qquad (9)$$

where $\mathbf{K}_{\mathrm{I}} > \mathbf{0} \in \mathbb{R}^{n\times n}$ is an integral gain matrix. It is shown that the PID control scheme ensures asymptotic stability only when the initial states belong to a restricted region about the equilibrium state. (see *PID-control*).

3.1.2. Servo Control in Cartesian Space

Most robot tasks are designed in Cartesian space, and desired positions and orientations (posture) of the end-effector are given in Cartesian space. In such situations, servo controls in Cartesian space are desirable. When link parameters of the manipulator are not precise, desired joint variables $\mathbf{q}_{\mathrm{d}}$ that are computed using the desired posture of the end-effector $\mathbf{r}_{\mathrm{d}} \in \mathbb{R}^m$ through the inverse kinematic model do not lead to the precise posture of the end-effector. In order to converge the posture errors of the end-effector $\mathbf{e}_{\mathrm{r}} = \mathbf{r}_{\mathrm{d}} - \mathbf{r}$ on zero without computation of the inverse kinematic model, a PD servo control with gravity compensation in Cartesian space is effective. Its control law is

$$\boldsymbol{\tau}(t) = \mathbf{K}_{\mathrm{D}}\dot{\mathbf{q}}(t) + \mathbf{J}^{\mathrm{T}}(\mathbf{q})\mathbf{K}_{\mathrm{P}}\mathbf{e}_{\mathrm{r}}(t) + \hat{\mathbf{g}}(\mathbf{q}) \qquad (10)$$

where $\mathbf{J}(\mathbf{q}) = \partial\mathbf{r}(\mathbf{q})/\partial\mathbf{q}$ is the geometric Jacobian of the manipulator, and $\mathbf{K}_{\mathrm{P}} > \mathbf{0}$ and $\mathbf{K}_{\mathrm{D}} > \mathbf{0}$ are the proportional and derivative symmetric feedback gain matrices, respectively, in Cartesian space. By the duality of velocity kinematics and statics, the following physical interpretation is admitted: $\mathbf{J}^{\mathrm{T}}(\mathbf{q})\mathbf{K}_{\mathrm{P}}\mathbf{e}_{\mathrm{r}}(t)$ is a compensatory term in joint torque space that balances to force $\mathbf{K}_{\mathrm{P}}\mathbf{e}_{\mathrm{r}}(t)$, which is generated by the spring with stiffness $\mathbf{K}_{\mathrm{P}}$. If $\hat{\mathbf{g}}(\mathbf{q}) = \mathbf{g}(\mathbf{q})$ and a Lyapunov function candidate is chosen as

$$V(t) = \frac{1}{2}\left\{ \dot{\mathbf{q}}^{\mathrm{T}}\mathbf{M}(\mathbf{q})\dot{\mathbf{q}} + \mathbf{e}_{\mathrm{r}}{}^{\mathrm{T}}\mathbf{K}_{\mathrm{P}}\mathbf{e}_{\mathrm{r}} \right\},$$

then the stability argument is similar to that in joint space. On the assumption that the Jacobian is full rank, it is shown that $\mathbf{r}$ converges to $\mathbf{r}_d$ as t tends toward infinity. This controller needs measurements of $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\mathbf{r}$. If measurements of $\mathbf{r}$ are made directly in the Cartesian space through an external sensor such as the vision sensor, this controller does not need the computation of the inverse kinematic model. (see *Lyapunov Stability*).

3.2. Dynamic Control

In the previous section, the position control is discussed. This section deals with a trajectory control in which desired joint variables $\mathbf{q}_d(t)$, joint velocities $\dot{\mathbf{q}}_\mathbf{d}(t)$, and joint acceleration $\ddot{\mathbf{q}}_d(t)$ are given as the time function. The control scheme that takes the manipulator dynamics into consideration is called dynamic control. As the dynamic controls, the computed torque control and the Cartesian based inverse dynamics control is presented. In this case, it is assumed that the manipulator dynamics are known precisely. The basic idea is to realize not a local linearization but a global linearization of the manipulator dynamics by means of a nonlinear state feedback. (see *Feedback Linearization*)

Computed torque control

In order to perform a global linearization of the manipulator dynamics in joint space, computed torque control as shown in Figure 3 is adopted. Its control law is given by

$$\boldsymbol{\tau} = \hat{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}}^* + \hat{\mathbf{C}}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \hat{\mathbf{g}}(\mathbf{q}) \qquad (11)$$

where

$$\ddot{\mathbf{q}}^* = \ddot{\mathbf{q}}_d + \mathbf{K}_v\dot{\mathbf{e}}(t) + \mathbf{K}_p\mathbf{e}(t)$$

and $\hat{\mathbf{M}}(\mathbf{q}), \hat{\mathbf{C}}(\mathbf{q},\dot{\mathbf{q}})$, and $\hat{\mathbf{g}}(\mathbf{q})$ are computed values of $\mathbf{M}(\mathbf{q}),\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$, and $\mathbf{g}(\mathbf{q})$, respectively, and $\mathbf{K}_p$ and $\mathbf{K}_v$ are the respective position and velocity feedback gain matrices. The manipulator system is linearized by the second and third terms on the right hand side of (11), and the linearized system is controlled by the linear servo compensator given by the first term on the right hand side of (11). If the dynamic model is exact, $\hat{\mathbf{M}}(\mathbf{q}) = \mathbf{M}(\mathbf{q})$, $\hat{\mathbf{C}}(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$, and $\hat{\mathbf{g}}(\mathbf{q}) = \mathbf{g}(\mathbf{q})$, then the closed system is given by $\ddot{\mathbf{q}}^* = \ddot{\mathbf{q}}_d$. Hence, equations of the trajectory errors $\mathbf{e}(t) = \mathbf{q}_d(t) - \mathbf{q}(t)$ are described by a second order linear differential equation as follows:

$$\ddot{\mathbf{e}}(t) + \mathbf{K}_v\dot{\mathbf{e}}(t) + \mathbf{K}_p\mathbf{e}(t) = \mathbf{0}\,. \qquad (12)$$

If $\mathbf{K}_p$ and $\mathbf{K}_v$ are chosen as diagonal positive definite matrices, the closed system is decoupled and globally asymptotically stable. For example, by choosing *i*-th diagonal elements for $\mathbf{K}_p$ and $\mathbf{K}_v$ of $2\xi\omega$ and ω^2, respectively, the closed system with the damping coefficient ξ and the natural angular frequency ω is constructed at each joint. The nonlinear control law (11) needs the efficient computation algorithm of inverse dynamics and exact dynamic parameters. If the dynamic parameters are not known precisely, the mismatch between actual and modeled parameters will cause servo errors. The parameters of dynamics can be obtained by the dynamic parameter identification. (see *Robot Kinematics and Dynamics*).

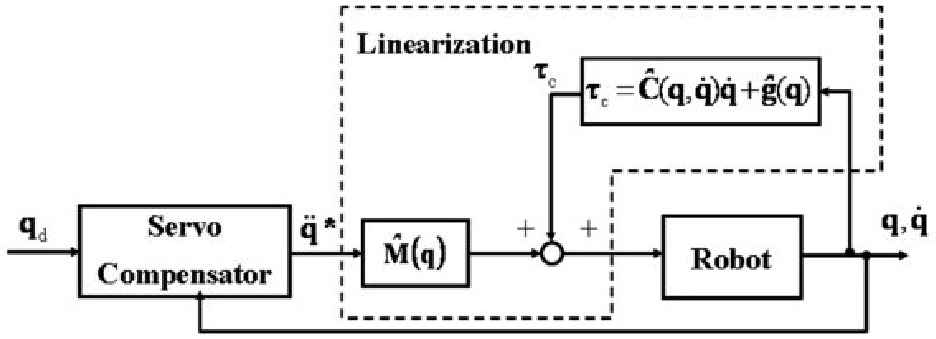

Figure 3: Computed torque method.

Cartesian based inverse dynamics control

When the desired posture of end-effector $\mathbf{r}_d(t)$, its velocity $\dot{\mathbf{r}}_d(t)$ and acceleration $\ddot{\mathbf{r}}_d(t)$ are given as a time function, dynamic control in Cartesian space is desirable. Cartesian-based inverse dynamics control consists of the feedback linearization in Cartesian space and position and velocity feedback compensation as shown in Figure 4. On the assumption that the manipulator geometric Jacobian $\mathbf{J}(\mathbf{q})$ is non-singular, its control law is given by

$$\boldsymbol{\tau} = \hat{\mathbf{M}}(\mathbf{q})\mathbf{J}^{-1}(\mathbf{q})\left(\ddot{\mathbf{r}}^* - \dot{\mathbf{J}}(\mathbf{q})\dot{\mathbf{q}}\right) + \hat{\mathbf{C}}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \hat{\mathbf{g}}(\mathbf{q}) \qquad (13)$$

where

$$\ddot{\mathbf{r}}^* = \ddot{\mathbf{r}}_d + \mathbf{K}_v\dot{\mathbf{e}}_r(t) + \mathbf{K}_p\mathbf{e}_r(t)$$

and $\mathbf{e}_r(t) = \mathbf{r}_d(t) - \mathbf{r}(t)$ is the posture error of the end-effector in Cartesian space. If the dynamic model is exact, $\hat{\mathbf{M}}(\mathbf{q}) = \mathbf{M}(\mathbf{q})$, $\hat{\mathbf{C}}(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$, and $\hat{\mathbf{g}}(\mathbf{q}) = \mathbf{g}(\mathbf{q})$, then the closed system satisfies $\ddot{\mathbf{r}} = \ddot{\mathbf{r}}^*$, which yields equations of the posture error in Cartesian space that are represented by

$$\ddot{\mathbf{e}}_r(t) + \mathbf{K}_v\dot{\mathbf{e}}_r(t) + \mathbf{K}_p\mathbf{e}_r(t) = \mathbf{0}.$$

If $\mathbf{K}_p$ and $\mathbf{K}_v$ are chosen as diagonal positive definite matrices, the closed system is decoupled and the posture error converges on zero.

Controlling a manipulator in the Cartesian space is in general more complex than controlling it in the joint space because of the presence of singularities and/or redundancy of the geometric Jacobian. Basically, it should avoid passing a singularity configuration of the manipulator. For the redundant manipulator, the Cartesian-based control scheme should incorporate a redundancy handling technique in the feedback loop.

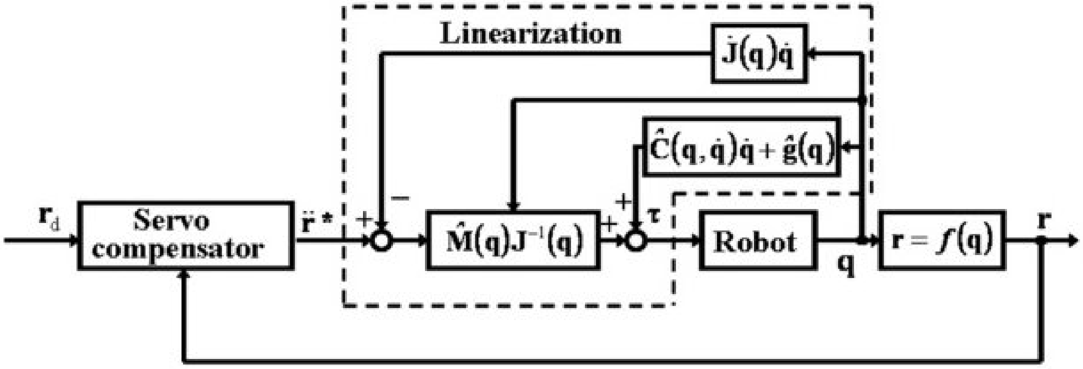

Figure 4: Cartesian based inverse dynamics control.

-
-
-

Bibliography

Arai T., Itoko T., and Yago H. (1997), *A Graphical Robot Language Developed in Japan*, Robotica 15, pp. 99-103. [This paper presents a graphical task-level robot language.]

Arimoto S. (1996), *Control Theory of Non-Linear Mechanical Systems*, Oxford University Press. [This book deals with stability of robot servo control.]

Asada H. and Izumi H. (1989), *Automatic program generation from teaching data for hybrid control of robots*, IEEE Transaction on Robotics and Automation, Vol.5, No.2, pp.163-173. [This paper presents a teaching of feedback control law.]

Burdea G. C. (1996), *Force and Touch Feedback for Virtual Reality*, John Wiley & Sons, Inc. [This book deals with haptic interfaces and their force feedback control.]

Craig J. J. (1989), *Introduction to Robotics: Mechanics and Control*, Addison-Wesley Publishing Company, Inc. [This is a widely used textbook that presents the essential of robotics.]

Ikeuch K. and Shuehiro T. (1994), *Assembly plan from observation*, IEEE Transaction on Robotics and Automation, Vol. 10, No.3, pp.368-385. [This paper presents a teaching by showing method termed the assembly plan from observation.]

Kawasaki H., Nakayama K., and Parker G. (2000), *Teaching for Multi-Fingered Robots Based on Motion Intention in Virtual Reality*, Proc. of IECON-2000. [This paper presents a teaching for mufti-fingered robots in a virtual reality environment.]

Li S. and Asada H. (1995), *Teaching Human Motion/Force Skills to Robots*, Journal of Robotics Society of Japan, Vol.13, No.5, pp.592-598. [This paper is a review of teaching based on human motion/force skills.]

Sciavico L. and Siciliano B. (1996), *Modelling and Control of Robot Manipulators*, Springer-Verlag. [This is a textbook that presents aspects of robot kinematics, dynamics, and control.]

Slotine J. J. E. and Weiping Li (1991), *Applied Nonlinear Control*, Prentice Hall. [This is a text book that presents the Lyapunov stability theory and adaptive control.]

Biographical Sketch

Haruhisa Kawasaki received the Master of Engineering Degree and the Doctor of Engineering degree both from Nagoya University in 1974 and 1986, respectively. He was a researcher at NTT's Laboratories from 1974 to 1990. He was Professor of Kanazawa Institute of Technology from 1990 to 1994. He is Professor of Faculty of Engineering of Gifu University since 1994. He was the chairman of executive committee of the International Conference on Virtual Systems and Multimedia in 1995 and 1996 (VSMM'95 and VSMM'96). He served as the Director of Virtual System Laboratory of Gifu University from 1997 to 1998. He was a guest professor at University of Surrey from July 1998 to January 1999. He is the editor of Journal of Robotics and Mechatronics since 1989. He is mainly engaged in the research fields of robot control, humanoid robot hand system, symbolic robot analysis system, robot teaching in a virtual realty environment and intelligent mechatronics. He is a member of the Institute of Electrical and Electronic Engineers (IEEE), the Japan Society of Mechanical Engineers (JSME), the Robotics Society of Japan (RSJ), the Society of Instrument and Control Engineers (SICE) and the Virtual Reality Society of Japan (VRSJ).