Abstract

A hybrid position/force control scheme was used to simulate the control of a two-degree-of-freedom robotic arm. The contact force between the end-effector and a surface was controlled while the end-effector moved parallel to the surface under position control. The end-effector was modeled as having passive compliance. Five simulations were performed using different levels of random noise in the force feedback loop. Stability was achieved and significant force control was maintained despite high levels of noise in the force feedback.

Introduction

Present-day industrial robots are capable of simple pick and place operations and tasks which do not require contact with the environment. With position control, a robot is capable of following a specified position trajectory within a certain tolerance. The amount of error is often acceptable for tasks such as paint spraying and welding. However, for a task requiring contact with a surface, the control of position alone is not adequate. Scraping paint from a pane of glass serves as an illustration. A small amount of position error may result in the scraper being slightly above the surface, thus failing to scrape off the paint, or if the error is in the other direction, the glass is likely to be broken. What is needed is the capability to control contact force perpendicular to the surface while controlling position in the directions parallel to the surface. The ability to mix the control of force and position is often referred to as "compliant motion control."

One of the problems in compliant motion control has been noise in the force feedback loop. This work shows by simulation how various levels of random noise in the force feedback affect the performance of a simple feedback manipulator. Stability appeared to be achievable only if the end-effector was modeled as having passive compliance. But having once achieved stability, the system showed considerable immunity to noise in the force feedback.

Manipulator Dynamics

The dynamic equations for any manipulator having \( n \) joints can be expressed in the following form \([1]\):

\[
\dot{\mathbf{q}} = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{V}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}) + \mathbf{F}(\mathbf{q}, \dot{\mathbf{q}})
\]

Where, \( \mathbf{q} \) is the \( \times 1 \) vector of joint displacements, \( \mathbf{M}(\mathbf{q}) \) is the \( \times \times \) inertia matrix of the manipulator, \( \mathbf{V}(\mathbf{q}, \dot{\mathbf{q}}) \) is an \( \times \) vector of coriolis terms, \( \mathbf{G}(\mathbf{q}) \) is an \( \times \) vector of gravity terms, and \( \mathbf{F}(\mathbf{q}, \dot{\mathbf{q}}) \) is an \( \times \) vector of friction effects. The \( \times \) vector of actuator torques and forces is denoted by \( \mathbf{T} \). For a revolute joint the corresponding element in \( \mathbf{T} \) is a torque. If any joint in the manipulator is prismatic the corresponding element of \( \mathbf{T} \) is a force. Each of the vectors and matrices in Eq. 1 are in terms of the joint positions and their time derivatives. Thus, Eq. 1 is said to be written with respect to joint space.

The incorporation of the manipulator dynamics into a control algorithm is facilitated by expressing the dynamic equations in terms of Cartesian variables which describe the position and orientation of the end effector. In terms of cartesian variables, Eq. 1 has the form \([2]\):

\[
\mathbf{F} = \mathbf{M}(\mathbf{\dot{q}}) + \mathbf{V}(\mathbf{\dot{q}}) + \mathbf{G}(\mathbf{\dot{q}}) + \mathbf{F}(\mathbf{\dot{q}})
\]
where F is a force-torque vector acting on the end-effector and \( \mathbf{X} \) is the Cartesian vector representing the position and orientation of the end-effector. \( \mathbf{M}_{\mathbf{X}}(\mathbf{q}) \) is the Cartesian mass matrix. The quantities \( \mathbf{V}_{\mathbf{X}}(\mathbf{\theta}, \dot{\mathbf{q}}) \), \( \mathbf{G}_{\mathbf{X}}(\mathbf{\theta}) \), and \( \mathbf{F}_{\mathbf{X}}(\mathbf{\theta}, \dot{\mathbf{q}}) \) are expressed in Cartesian space and are analogous to corresponding terms in Eq. 1. By Eq. 2, the dynamic equations of the manipulator are said to be "reflected" to the end-effector. It is shown in reference [3] that Eq. 2 can be obtained from Eq. 1 by the following relations:

\[
\mathbf{M}_{\mathbf{X}}(\mathbf{\theta}) = \mathbf{J}^T(\mathbf{\theta}) \mathbf{M}(\mathbf{\theta}) \mathbf{J}(\mathbf{\theta})^T
\]

\[
\mathbf{V}_{\mathbf{X}}(\mathbf{\theta}, \dot{\mathbf{q}}) = \mathbf{J}^T(\mathbf{\theta}) \left[ \mathbf{V}(\mathbf{\theta}, \dot{\mathbf{q}}) - \mathbf{M}(\mathbf{\theta}) \mathbf{J}(\mathbf{\theta}) \dot{\mathbf{q}} \right]
\]

\[
\mathbf{G}_{\mathbf{X}}(\mathbf{\theta}) = \mathbf{J}^T(\mathbf{\theta}) \mathbf{G}(\mathbf{\theta})
\]

\[
\mathbf{F}_{\mathbf{X}}(\mathbf{\theta}, \dot{\mathbf{q}}) = \mathbf{J}^T(\mathbf{\theta}) \mathbf{F}(\mathbf{\theta}, \dot{\mathbf{q}})
\]

Where \( \mathbf{J}(\mathbf{\theta}) \) is the manipulator Jacobian matrix, \( \mathbf{J}^T(\mathbf{\theta}) \) is the inverse of \( \mathbf{J} \), and \( \mathbf{J}^T(\mathbf{\theta}) \) is the transpose of \( \mathbf{J}^T(\mathbf{\theta}) \). The symbol for the Jacobian matrix indicates dependency upon the instantaneous manipulator configuration. If \( F \) is computed in Eq. 2 based upon a desired Cartesian motion of the end-effector, the required actuator torques can be obtained by,

\[
\tau = \mathbf{J}^T(\mathbf{\theta}) \mathbf{F}
\]

For practical control implementation equation 2 may be written in the form,

\[
\mathbf{F} = \mathbf{M}_{\mathbf{X}}(\mathbf{\theta}) \ddot{\mathbf{X}} + \mathbf{B}_{\mathbf{X}}(\mathbf{\theta})[\dot{\mathbf{q}}] + \mathbf{C}_{\mathbf{X}}(\mathbf{\theta})[\dot{\mathbf{q}}]^2 + \mathbf{G}_{\mathbf{X}}(\mathbf{\theta}) + \mathbf{F}_{\mathbf{X}}(\mathbf{\theta}, \dot{\mathbf{q}})
\]

Where \( \mathbf{B}(\mathbf{\theta}) \) is a matrix of dimension \( n \times n(n-1)/2 \) of Coriolis coefficients, \( [\dot{\mathbf{\theta}}, \dot{\mathbf{\theta}}] \) is an \( n(n-1)/2 \times 1 \) vector of joint velocity products given by,

\[
[\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3, \ldots, \dot{\theta}_n, \dot{\theta}_n, \dot{\theta}_n, \ldots, \dot{\theta}_1]^T
\]

and \( \mathbf{C}_{\mathbf{v}}(\mathbf{\theta}) \) is an \( n \times n \) matrix of centrifugal coefficients, and \( [\dot{\theta}^2] \) is an \( n \times 1 \) vector given by,

\[
[\dot{\theta}_1^2, \dot{\theta}_2^2, \ldots, \dot{\theta}_n^2, \dot{\theta}_n^2, \ldots, \dot{\theta}_1^2]^T
\]

The form of Eq. 8 is important in control applications because the parameters appear as functions of joint position only and may be updated at a variable rate that depends on how fast the manipulator is changing configuration.

**Position Control**

A conceivable approach to position control is shown in Fig. 1. It is based on the partitioned control law concept [1] consisting of a model-based portion shown within the dotted outline and a servo-based portion, which is the position control law. \( \mathbf{X}_d \) is the Cartesian vector representing the desired position and orientation of the end-effector. To the \( n \times 1 \) output vector \( \mathbf{F} \) of the position control law, the system within the dotted lines appears to be an identity matrix, or rather a diagonal mass matrix with unit masses everywhere on the diagonal. This can be seen by first writing an expression for \( \mathbf{F} \) directly from the block diagram in Fig. 1. The following is obtained:

\[
\mathbf{F} = \mathbf{M}_{\mathbf{X}}(\mathbf{\theta}) \mathbf{F}' + \mathbf{B}_{\mathbf{X}}(\mathbf{\theta})[\dot{\mathbf{q}}] + \mathbf{C}_{\mathbf{X}}(\mathbf{\theta})[\dot{\mathbf{q}}]^2 + \mathbf{G}_{\mathbf{X}}(\mathbf{\theta}) + \mathbf{F}_{\mathbf{X}}(\mathbf{\theta}, \dot{\mathbf{q}})
\]

Combining Eq. 11 with Eq. 8 gives,

\[
\mathbf{F}' = \ddot{\mathbf{X}}
\]

or,

\[
\mathbf{F}' = \ddot{\mathbf{X}}
\]

\[
\mathbf{E} = \mathbf{X}_d - \mathbf{X}
\]

A reasonable position control law is,

\[
\mathbf{F}' = \mathbf{X}_d + \dot{\mathbf{X}} + \mathbf{K}_p \mathbf{E} + \mathbf{K}_t \int \mathbf{E} \, dt
\]
Where $K_p$, $K_f$, and $K_i$ are diagonal gain matrices. In Eq. 15 if $K_i$ is small, the values for $K_p$ and $K_f$ can be chosen so as to achieve critical damping of errors.

**Hybrid Position/Force Control**

The schematic for a hybrid position/force controller after Rabert and Craig (4) is shown in Fig. 3. Related concepts are described in references (5) and (6). The position control portion is the same as in Fig. 2. The cartesian vector $F_d$ represents the desired forces to be exerted on the environment by the end-effector. It is clearly impossible to control both force and position in the same degree of freedom, thus $S$ and $S'$ in Fig. 3 are diagonal constraint matrices used to distinguish between a position controlled degree of freedom and a force controlled degree of freedom. The diagonal elements are either a 1 or 0. If a degree of freedom is position controlled the corresponding element in $S$ is 1, and 0 in $S'$, and vice versa if the degree of freedom is force controlled.

Let $K_e$ be a diagonal stiffness matrix representing the built-in passive compliance of the end-effector. Also, let $K_e$ be a diagonal stiffness matrix representing the environment with which interaction is required. The effective stiffness $K_{eq}$ of the contact is given by,

$$K_{eq} = (K_c + K_{eq}) K_e K_e$$

A force control law that is capable of implementation is given by,

$$F' = K_p E - K_f X_f + K_i f$$

$$\int E_f dt$$

Where, $F'$ is the vector of control actions and $E = F_d - F'$. $K_p$, $K_f$, and $K_i$ are diagonal gain matrices.

**Computer Simulation**

A computer program was written to study the dynamic behavior of a simple manipulator which moves while in contact with a surface. The control system architecture of Fig. 3 served as the basic flow chart. The position and force control laws used were Eqs. 15 and 17 respectively. The simulation was based on the two-degree-of-freedom model shown in Fig. 4a. Random noise was injected into the force feedback loop as shown in Fig. 3. The mass and length values for the links are shown in Fig. 4a, as well as the stiffness values for the surface and the end-effector compliance. Simple rectangular integration (Euler integration) was used. The integration time step was $0.001$ sec. Gravitational and frictional effects were neglected.

The value for $K_i$ in Eq. 15 was taken as zero. A value of 10 was chosen for $K_p$ in Eq. 17. Several attempts were made to stabilize the system before adding the passive compliance $K_c$ shown in Fig. 4a. All attempts failed perhaps because of the significant figure limitation of DASICA programming language on the IBM-XT personal computer. $K_c$ was chosen as $10000$ N/m, more or less arbitrarily. Stability was then achieved by trial and error selection of values for $K_p$ and $K_f$ in Eqs. 15 and 17, respectively. The value chosen for $K_p$ was $1000$ N/m and the value for $K_f$ was $50000$ N/m. $K_i$ in Eq. 15 was taken as $66.66$ N/sec/m (240) to achieve critical damping of errors in position. $K_c$ was similarly obtained, $K_c' = 447$, although with less justification since there is only a partial correspondence of terms between Eq. 15 and Eq. 17.

After stability was achieved there were 5 simulation runs, each using a different level of random noise in the force feedback. The peak-to-peak noise levels were 50, 100, 200, 300, and 400 N. In each simulation the end-effector started at a distance of 5 cm under position trajectory control as indicated in Fig. 4a. Once contact was made the end-effector moved along the surface with simple harmonic motion as shown in Fig. 4b. The frequency of the motion was 5 radians per second and the amplitude was 0.25 M. In each simulation the desired contact force to be maintained was 50 N.

Figure 5 shows the simulation results for the first case in which the peak-to-peak noise level was 50 N. Figures 6 through 9 show results of simulations using the other 4 noise levels. It is not until the noise level is somewhere between 300 and 400 N that control is completely lost. It can be seen in Fig. 9 that regions of the graph have zero value indicating that the end-effector is no longer being kept in contact with the surface.

**Conclusion**

A considerable number of trial and error attempts may be necessary to establish the various parameters in a hybrid position force control system. Passive
compliance of the end-effector may be essential for stability if the contacted surface is of high stiffness. The simulations suggest that high immunity to noise in the force feedback may be realizable, perhaps owing largely to the passive compliance of the end-effector.

Bibliography


Fig. 3. Hybrid Position/Force Control System

Fig. 4 Model of Robot Arm Used in Simulation. a) Initial Position of Arm. b) Motion After Contact.
Fig. 5 Simulation Results for 50 N Peak-to-Peak Noise.  
- a) Motion of End-Effector.  
- b) Contact Force.

Fig. 6 Contact Force from Simulation with 100 N Noise Level.

Fig. 7 Contact Force from Simulation with 200 N Noise Level.

Fig. 8 Contact Force from Simulation with 400 N Noise Level.