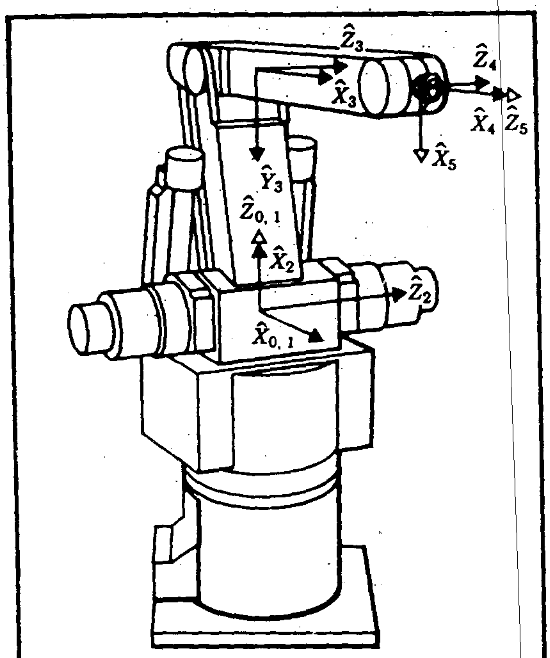
1. Control of Robotic Actuators

◎ Industrial Robot with Coordinate Frames



Assignment of link frames for the Yasukawa L-3.

- generally 6 joint 범용 로보트
- Positioning Device
- Kinematics

$$\underline{u} = f(\underline{\theta})$$

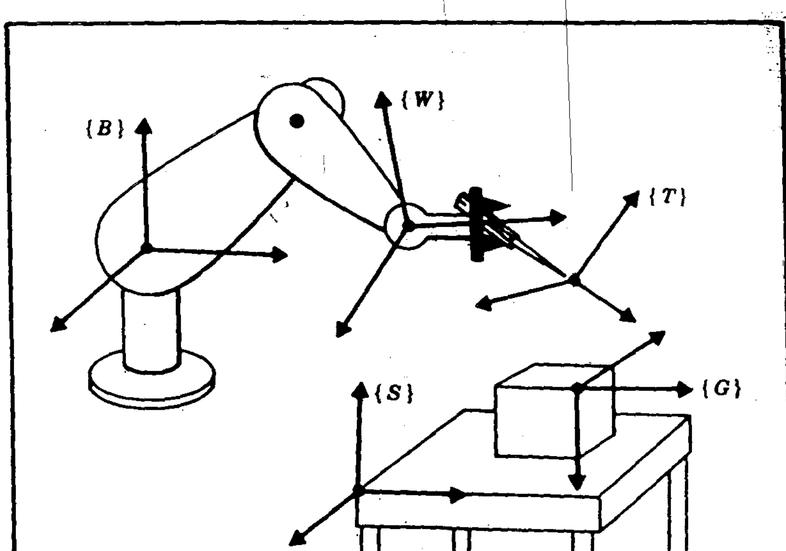
$$\dot{\underline{u}} = J\,\dot{\underline{\theta}}$$

$$\ddot{\underline{u}} = J\,\ddot{\underline{\theta}} + H(\theta, \dot{\theta})$$

- Dynamics

$$\underline{\tau} = H(\theta)\,\ddot{\underline{\theta}} + \underline{c}(\theta, \dot{\theta}) + \underline{g}$$

- Highly nonlinear mechanism



The standard frames.

- 행하여야 하는 작업은 {B}나 {T}에 의하여 지정된다

  → needs Inverse kinematic

$$\underline{\theta} = f^{-1}(\underline{u})$$

- 작업이 teaching에 의해서 주어지는 경우

  → teaching된 point는 $(\underline{\theta})$ memory에 기록

◎ Control 방법에 따른 로봇 분류

- Non-Servo Controlled Robot : Sequence 제어 방식 (bang bang type)
- Servo Controlled Robot
  - Point to Point Control (ex. Asea, Cincinnati Milacron) → pick and place
  - Continuous Path Control (ex. De Vilbis)
    - ↳ spray painting
      grinding
      arc welding
  - PTP 방식의 로봇으로 상당히 많은 점을 teaching해 CP control을 행하는 경우도 많다.
- 대다수의 로보트가 PTP 방식으로 제어됨

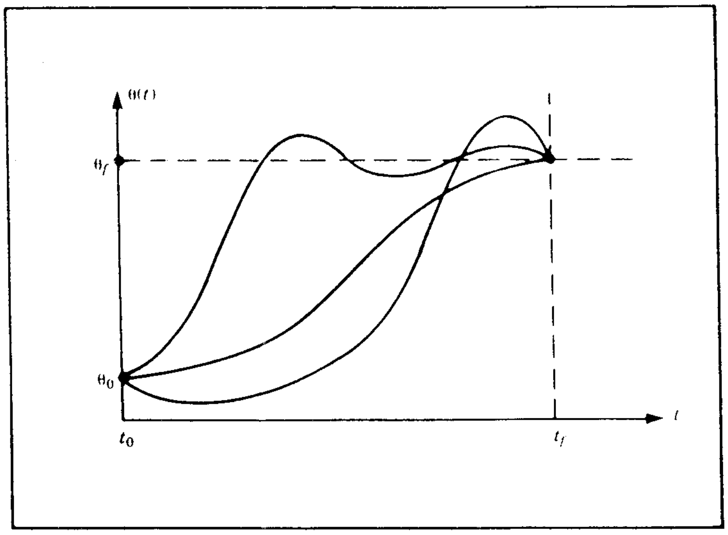
Question) teaching된 point와 point 사이는 어떠한 방식으로 이동하게 되는가?

→ Needs trajectory planner

◎ Trajectory Generation (Planning)

- Joint Space Schemes → joint trajectory
- Cartesian Space Schemes → Cartesian trajectory

- Joint Space Schemes 의 예 : Cubic Polynomial



Several possible path shapes for a single joint.

Given :

$$\theta(0) = \theta_0, \quad \theta(t_f) = \theta_f. \tag{7.1}$$

$$\dot{\theta}(0) = 0, \quad \dot{\theta}(t_f) = 0. \tag{7.2}$$

Cubic Polynomial :

$$\theta(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3. \tag{7.3}$$

이외에도

사다리꼴 planning

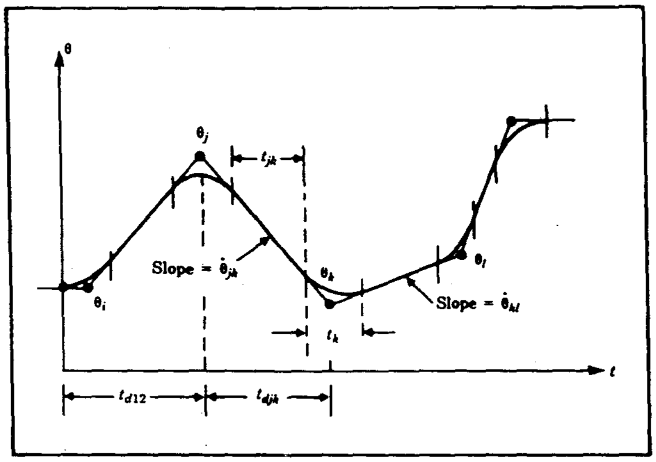
5차 order 〃

등이 있다.

Result :

$$a_0 = \theta_0,$$
$$a_1 = 0,$$
$$a_2 = \frac{3}{t_f^2}(\theta_f - \theta_0), \tag{7.6}$$
$$a_3 = -\frac{2}{t_f^3}(\theta_f - \theta_0).$$

○ 중간 경유점들이 있는 경우



Multisegment linear path with blends.

· 동일한 계산 방법

· Via point

· through point

• Optimal Planning (+) Optimal Control

앞의 경우에는 tf가 결정된 경우이지만 실제적으로 생산성을 향상시키기 위하여 Hardware가 허용하는 한 tf를 minimization이 필요하다

→ Off Line NLP problem

Ex)

$$ET = \tau_0 + \sum_{i=1}^{m} T_i + \tau_m$$

$0 \le \| v(t) \| \le K_v$,

$0 \le | \omega(t) | \le K_\omega$,

$0 \le | \dot{v}(t) | \le K_a$,

$0 \le \| \dot{\omega}(t) \| \le K_\alpha$,

for all t

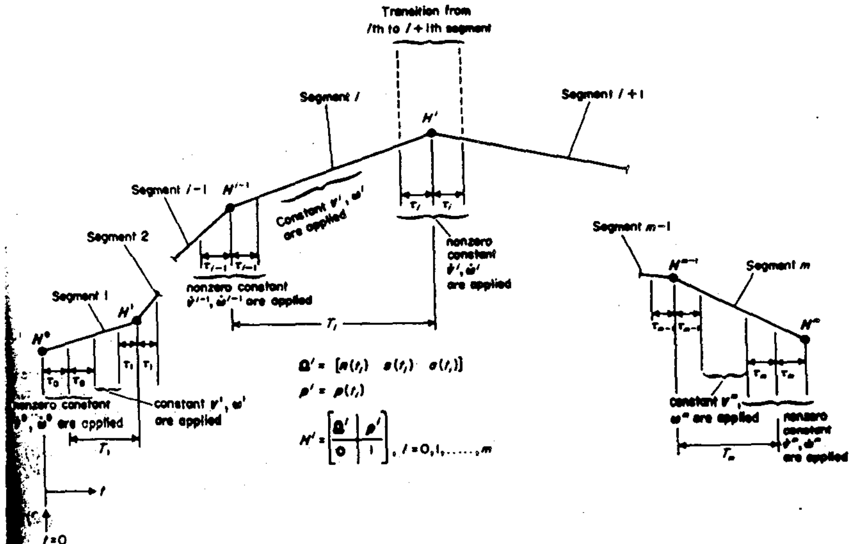


Figure 2
Straight-line segments of travelling path

• Optimal Planning의 예 (by using cubic polynomial)

An illustrative example having eight selected knots

Knot	Joint 1 (degrees)	Joint 2 (degrees)	Joint 3 (degrees)	Joint 4 (degrees)	Joint 5 (degrees)	Joint 6 (degrees)
1	10	15	45	5	10	6
2			(extra knot)			
3	60	25	180	20	30	40
4	75	30	200	60	−40	80
5	130	−45	120	110	−60	70
6	110	−55	15	20	10	−10
7	100	−70	−10	60	50	10
8	−10	−10	100	−100	−40	30
9			(extra knot)			
10	−50	10	50	−30	10	20

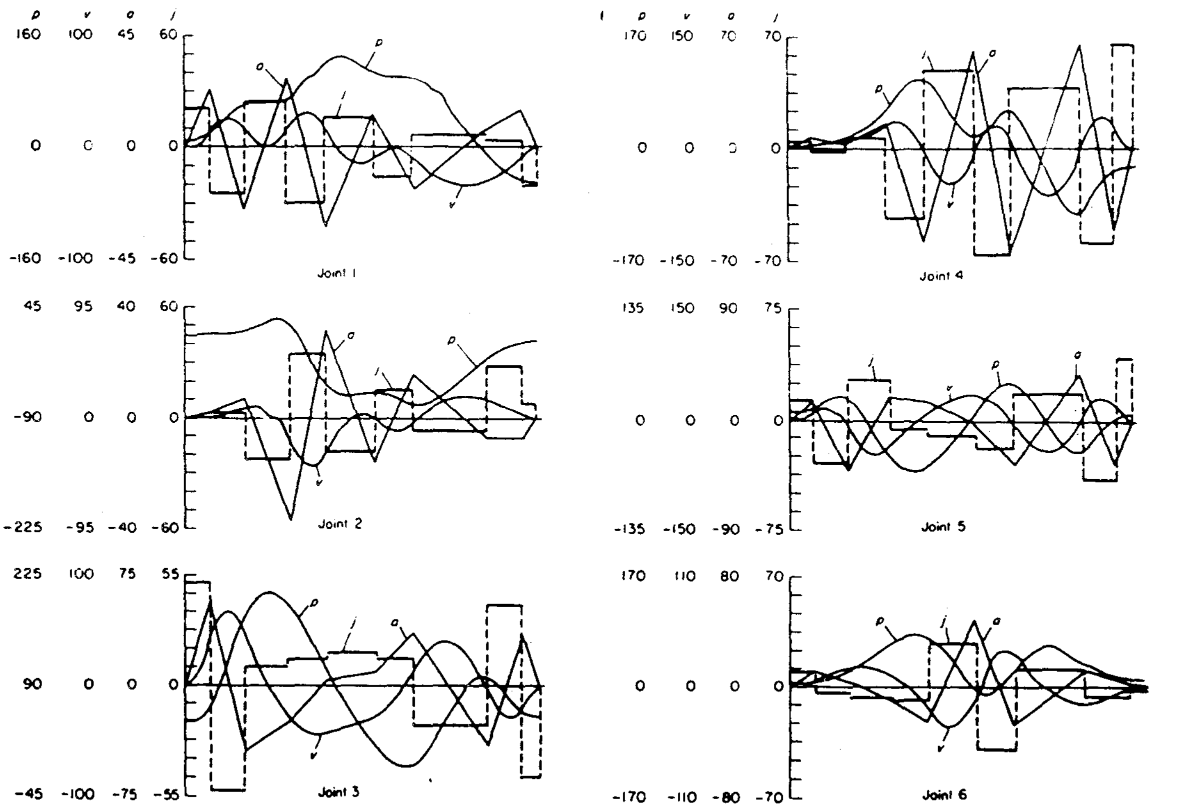
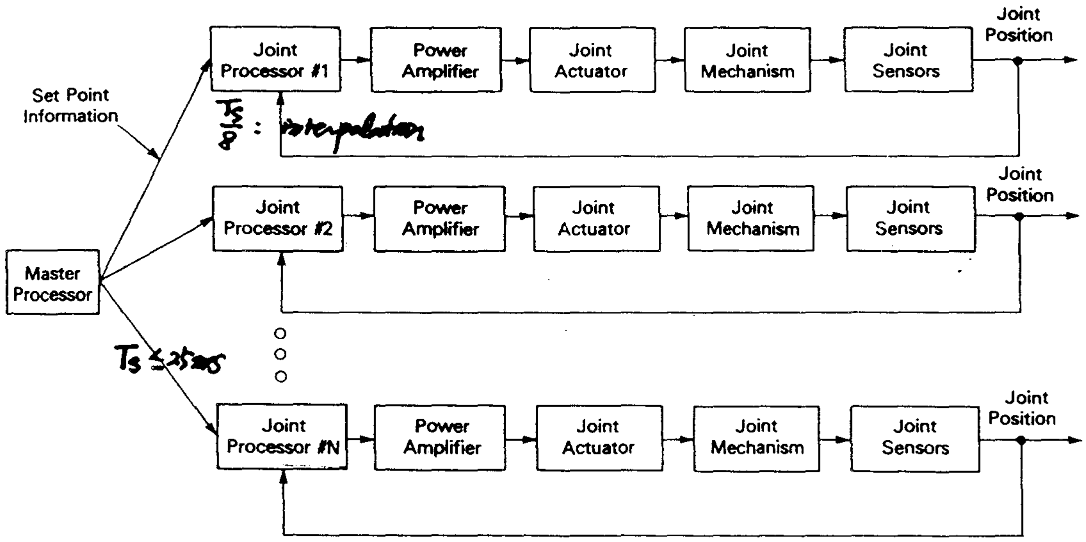


Figure 6
Optimum joint trajectories for the example. *p* is position, *v* velocity, *a* acceleration, *j* jerk

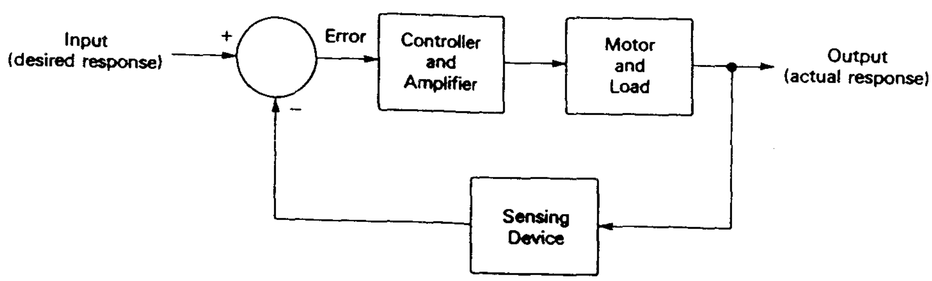
• 이와 같이 얻어진 joint profile은 시간이 경과됨에 따라 각 joint controller에 set position 값으로 보내진다.

⊙ Common Robot Control Architecture

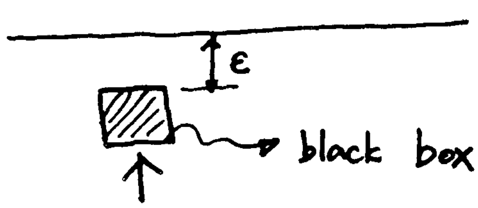


Common robot control architecture. It will be seen later that usually, the sensors are actually mounted at the actuator output.

◦ Independent Joint Control : PID based Control



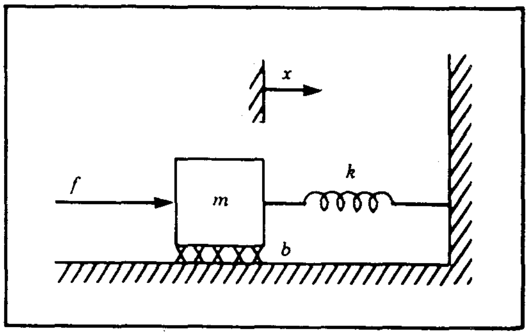
General closed-loop control system.



Linear and NonLinear Control

⊙ Linear Control (Review)

① Simple Feedback Control



A damped spring-mass system with an actuator.

$m\ddot{x} + b\dot{x} + kx = f$

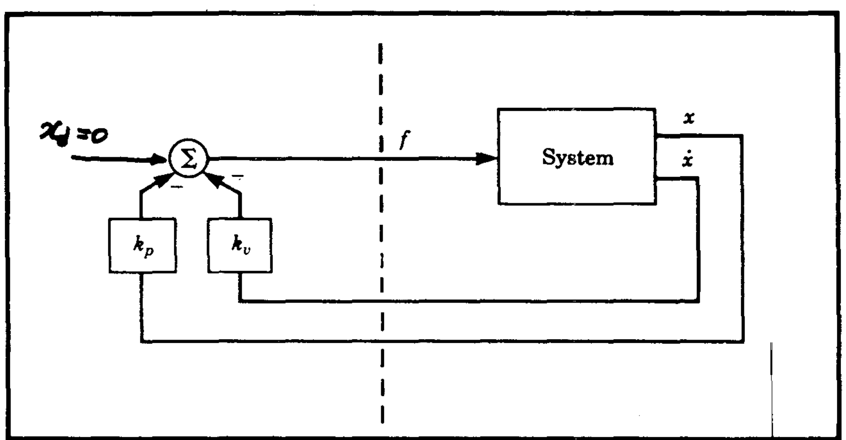


FIGURE 9.7 A closed loop control system. The control computer (to the left of the dashed line) reads sensor input and writes actuator output commands.

m, b, k unknown

$f = -k_p x - k_v \dot{x}$

$m\ddot{x} + b\dot{x} + kx = -k_p x - k_v \dot{x}$

$m\ddot{x} + (k_v + b)\dot{x} + (k_p + k)x = 0$

k_p, k_v 조정으로 performance 및 stability 해결 → critically damped

$\ddot{x} + 2\zeta\omega_n \dot{x} + \omega_n^2 x = 0$

$\zeta = 1$ 이면 critically damped

② Model-Based Feedback Control

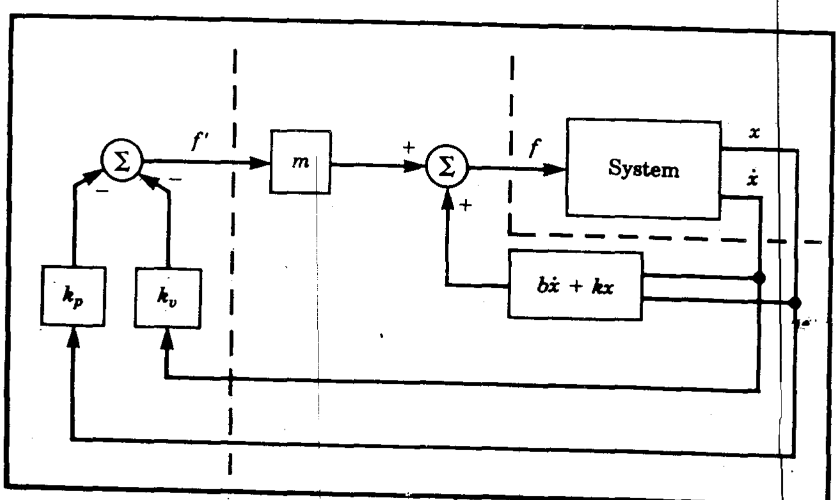


FIGURE 9.8 A closed loop control system employing the partitioned control method.

m, b, k should be known.

$f = \alpha f' + \beta$, $\alpha = m$, $\beta = b\dot{x} + kx$

$m\ddot{x} + b\dot{x} + kx = mf' + b\dot{x} + kx$

$\ddot{x} = f'$

set $f' = -k_v \dot{x} - k_p x$

$\ddot{x} + k_v \dot{x} + k_p x = 0$

easy to make it critically damped

⊙ Disturbance Rejection

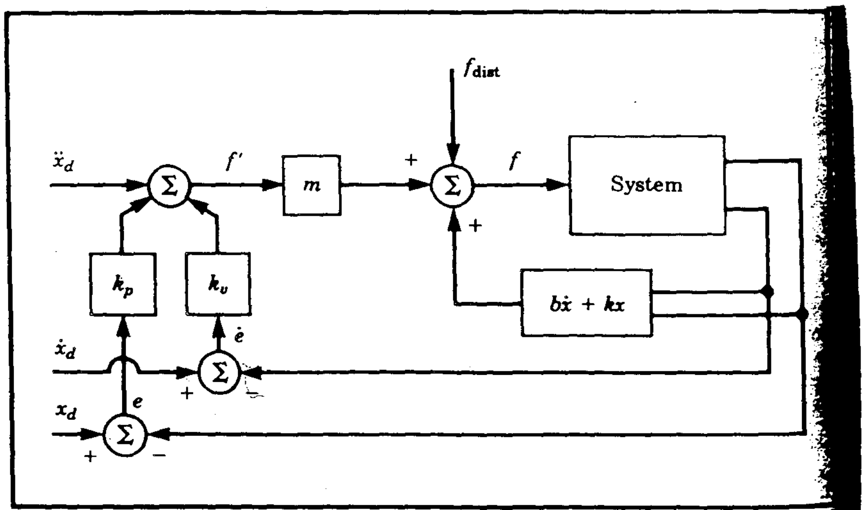


FIGURE 9.10 A trajectory-following control system with a disturbance acting.

$$\ddot{e} + k_v\dot{e} + k_pe = f_{dist}$$

From steady-state analysis;

$$k_pe = f_{dist}$$

$$e = f_{dist}/k_p \quad \rightarrow \text{ steady-state error 발생}$$

To eliminate steady-state error, add the integral term to control law

$$f' = \ddot{x}_d + k_v\dot{e} + k_pe + k_i\int e\,dt$$

$$\ddot{e} + k_v\dot{e} + k_pe + k_i\int e\,dt = f_{dist}$$

$$\dddot{e} + k_v\ddot{e} + k_p\dot{e} + k_ie = \dot{f}_{dist}$$

From s-s analysis, and if f_{dist} = Const,

$$k_ie = 0$$

$$\therefore e_{ss} = 0$$

- Trajectory - following Control

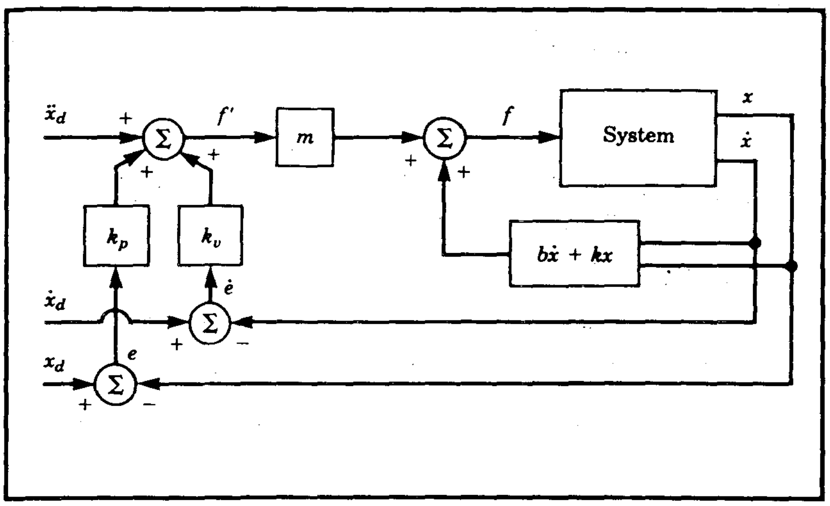


FIGURE 9.9 A trajectory following controller for the system in Fig. 9.6

x_d가 아니라
$x_d(t)$를 제어

$$m\ddot{x} + b\dot{x} + kx = f$$

$$f = \alpha f' + \beta \quad , \quad \alpha = m, \quad \beta = b\dot{x} + kx$$

$$f' = \ddot{x}_d + k_v(\dot{x}_d - \dot{x}) + k_p(x_d - x)$$

$$= \ddot{x}_d + k_v \dot{e} + k_p e$$

Total dynamic behavior

$$m\ddot{x} + b\dot{x} + kx = m(\ddot{x}_d + k_v\dot{e} + k_p e) + b\dot{x} + kx$$

$$\ddot{e} + k_v\dot{e} + k_p e = 0$$

k_v, k_p 조정으로 critically damped (ie. $k_v = 2\sqrt{k_p}$)

Nonlinear Control of Manipulators

- System nonlinear behavior

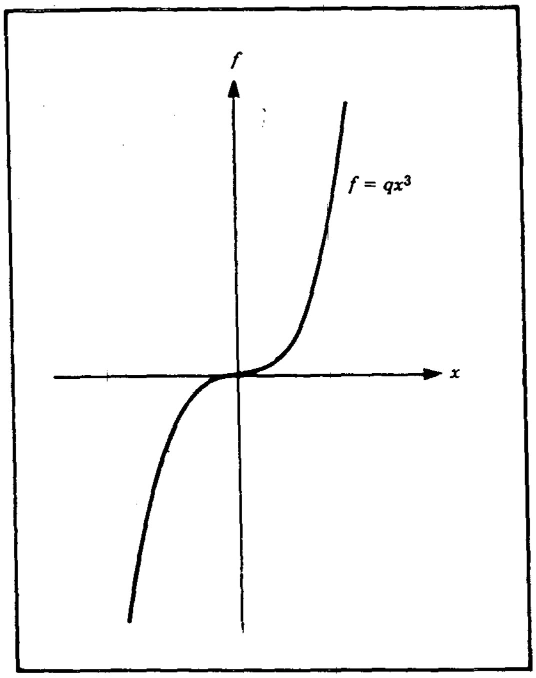


FIGURE 10.1 The force vs. distance characteristic of a nonlinear spring.

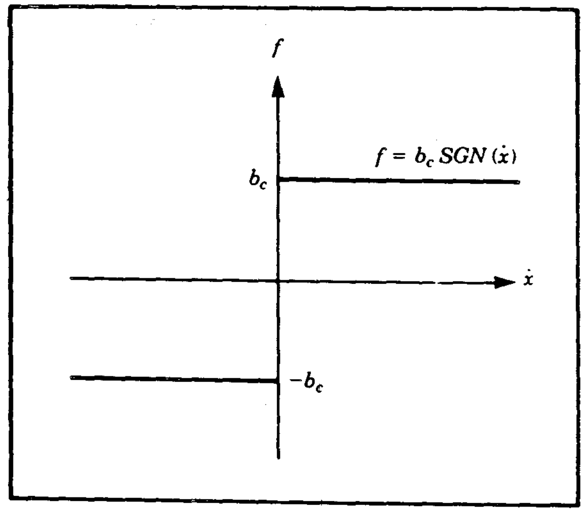


FIGURE 10.3 The force vs. velocity characteristic of Coulomb friction.

If we know the system's nonlinear characteristics,
easily build the following control system

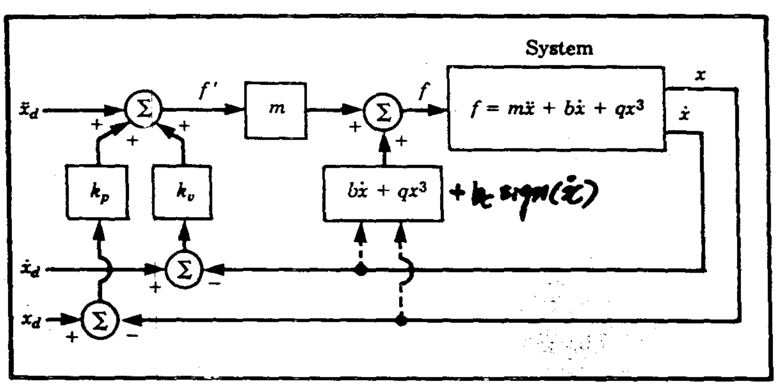


FIGURE 10.2 A nonlinear control system for a system

Robot Manipulators : Highly nonlinear system

MIMO based control is necessary

($\because$ Coupling effect)

⊙ Computed Torque Control Method

Manipulator dynamic eq. : $\tau = M(\theta)\ddot{\theta} + V(\theta,\dot{\theta}) + G(\theta) + F(\theta)$

partitioning control law :

$$\tau = \alpha\tau' + \beta$$

$$\alpha = M(\theta)$$

$$\beta = V(\theta,\dot{\theta}) + G(\theta) + F(\theta,\dot{\theta})$$

$$\tau' = \ddot{\theta}_d + K_v\dot{e} + K_p e$$

$$e = \theta_d - \theta$$

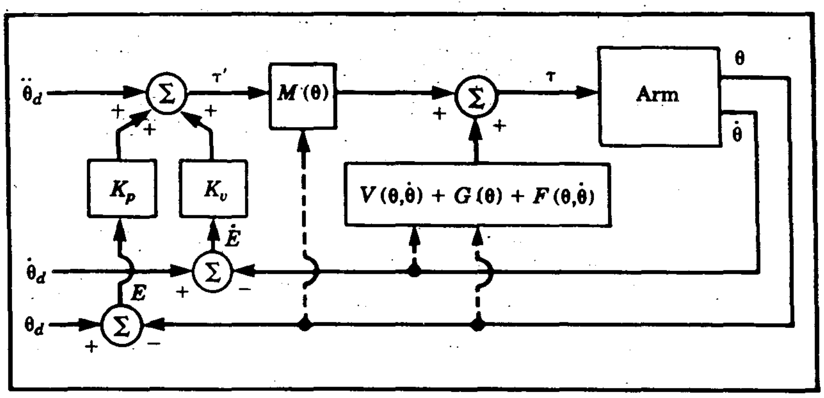


FIGURE 10.5 A model-based manipulator control system.

If we compute $M(\theta)$, $V(\theta,\dot{\theta})$, $G(\theta)$, $F(\theta,\dot{\theta})$ correctly,

error equation will be

$$\ddot{e} + k_v\dot{e} + k_p e = 0$$

⊙ Main problem of Computed Torque Control

- parameters를 구하는데 많은 시간이 걸린다.

  → real time control이 어렵다 (60Hz requirement)

- parameter의 값이 불확실하다.

  → mass, center of mass , inertia , friction 등의 정확한 값을 알기 어렵다.

Real time problem을 해결하기 위하여, Feedforward Controller

$$\tau_{ff} = \hat{M}(\theta_d)\ddot{\theta}_d + \hat{V}(\theta_d,\dot{\theta}_d) + g(\dot{\theta}_d) + F(\theta_d,\dot{\theta}_d)$$

$$\tau = \tau_{ff} + K_v(\dot{\theta}_d - \dot{\theta}) + K_p(\theta_d - \theta)$$

off-line computation

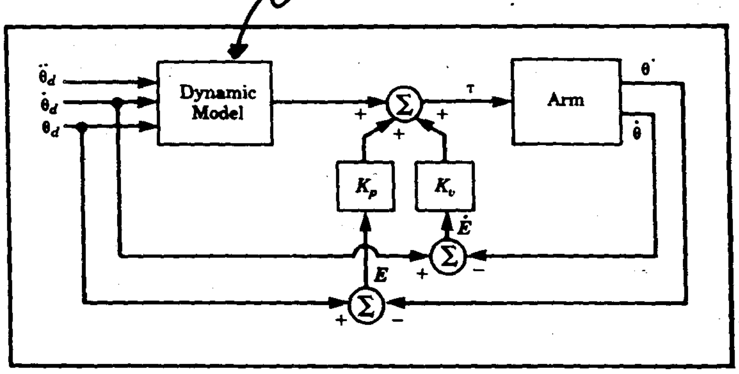


FIGURE 10.6 Control scheme with the model-based portion "outside" the servo loop.

• Model Parameter 값이 불확실 하므로 생기는 효과

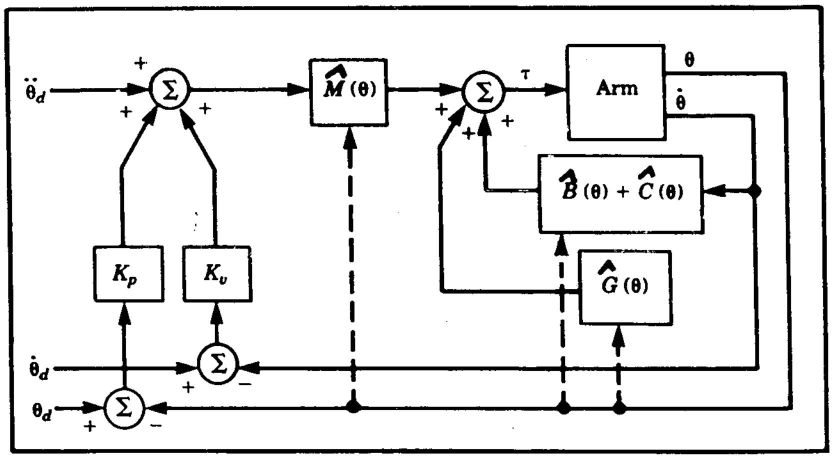


FIGURE 10.7 An implementation of the model-based manipulator control system.

⇓

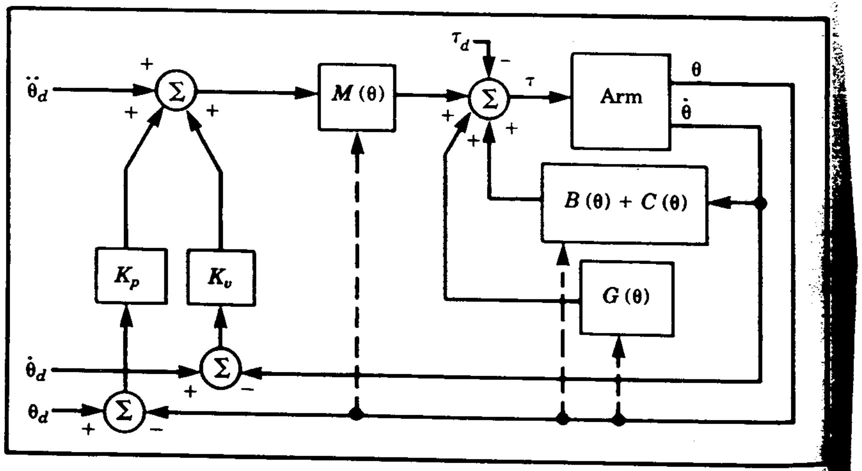


FIGURE 10.8 The model-based controller with an external disturbance acting.

Therefore, although the manipulator dynamics are given by

$$\tau = M(\Theta)\ddot{\Theta} + V(\Theta,\dot{\Theta}) + G(\Theta) + F(\Theta,\dot{\Theta}), \tag{10.23}$$

our control law computes

$$\begin{aligned} \tau &= \alpha\tau' + \beta, \\ \alpha &= \hat{M}(\Theta), \\ \beta &= \hat{V}(\Theta,\dot{\Theta}) + \hat{G}(\Theta) + \hat{F}(\Theta,\dot{\Theta}). \end{aligned} \tag{10.24}$$

Decoupling and linearizing will not therefore be perfectly accomplished when parameters are not known exactly. Writing the closed loop equation for the system, we have

$$\ddot{E} + K_v\dot{E} + K_pE = \hat{M}^{-1}\overbrace{\left[\left(M-\hat{M}\right)\ddot{\Theta} + \left(V-\hat{V}\right) + \left(G-\hat{G}\right) + \left(F-\hat{F}\right)\right]}^{\tau_d}, \tag{10.25}$$

Steady state error 뿐만 아니라 Stability 에도 영향을 줌.

◎ Model uncertainty를 해결하고자 하는 노력

① gain adaptive control

② parameter adaptive control

③ robust control – sliding mode, H∞ control

Controller Performance 비교 : PD Arm

1. PD controller with position reference only:

$$\tau = -\mathbf{K}_v\dot{\boldsymbol{\theta}} + \mathbf{K}_p(\boldsymbol{\theta}_d - \boldsymbol{\theta})$$

2. PD controller with position reference and feedforward of gravity torques:

$$\tau = \hat{\mathbf{g}}(\boldsymbol{\theta}_d) - \mathbf{K}_v\dot{\boldsymbol{\theta}} + \mathbf{K}_p(\boldsymbol{\theta}_d - \boldsymbol{\theta})$$

3. PD controller with position and velocity references:

$$\tau = \mathbf{K}_v(\dot{\boldsymbol{\theta}}_d - \dot{\boldsymbol{\theta}}) + \mathbf{K}_p(\boldsymbol{\theta}_d - \boldsymbol{\theta})$$

4. PD controller with position and velocity references plus feedforward of gravity torques:

$$\tau = \hat{\mathbf{g}}(\boldsymbol{\theta}_d) + \mathbf{K}_v(\dot{\boldsymbol{\theta}}_d - \dot{\boldsymbol{\theta}}) + \mathbf{K}_p(\boldsymbol{\theta}_d - \boldsymbol{\theta})$$

5. PD controller with position and velocity references plus feedforward of full dynamics:

$$\tau = \hat{\mathbf{H}}(\boldsymbol{\theta}_d)\ddot{\boldsymbol{\theta}}_d + \dot{\boldsymbol{\theta}}_d \; \hat{\mathbf{C}}(\boldsymbol{\theta}_d)\cdot\dot{\boldsymbol{\theta}}_d + \hat{\mathbf{g}}(\boldsymbol{\theta}_d) + \mathbf{K}_v(\dot{\boldsymbol{\theta}}_d - \dot{\boldsymbol{\theta}}) + \mathbf{K}_p(\boldsymbol{\theta}_d - \boldsymbol{\theta})$$

○ Gear 사용 Problem

- Friction and backlash
- Joint Flexibility
- Speed limitation
- dominance of rotor inertia

PD Arm 선정 이유

- more precise control
- back drivability
- more responsive to the envir.
- force control에 적합

○ Joint ~~Con~~ Torque Control 이 산업용 robot에 사용되지 않는 이유

- Motor current 로 가지고 정확한 torque를 구할 수 없다 (nonlinear dynamics due to gear friction and backlash)
- 산업용 로보트는 position Controller
  (Current or joint torque를 set value로 주기 어렵다)
- position value를 통한 force control은 정확한 control이 되지 않는다.

· Joint Trajectory Planning

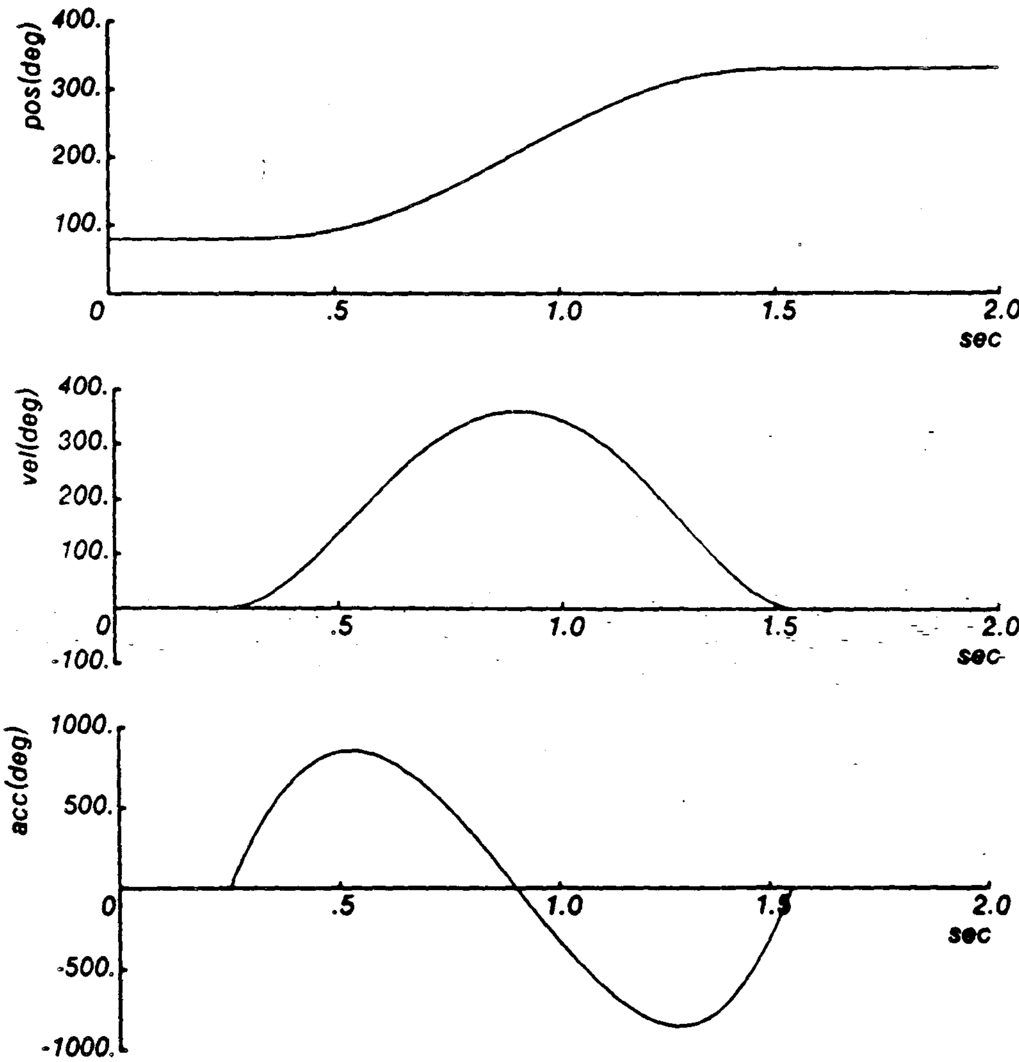


Figure 6.1: A fifth order polynomial trajectory.

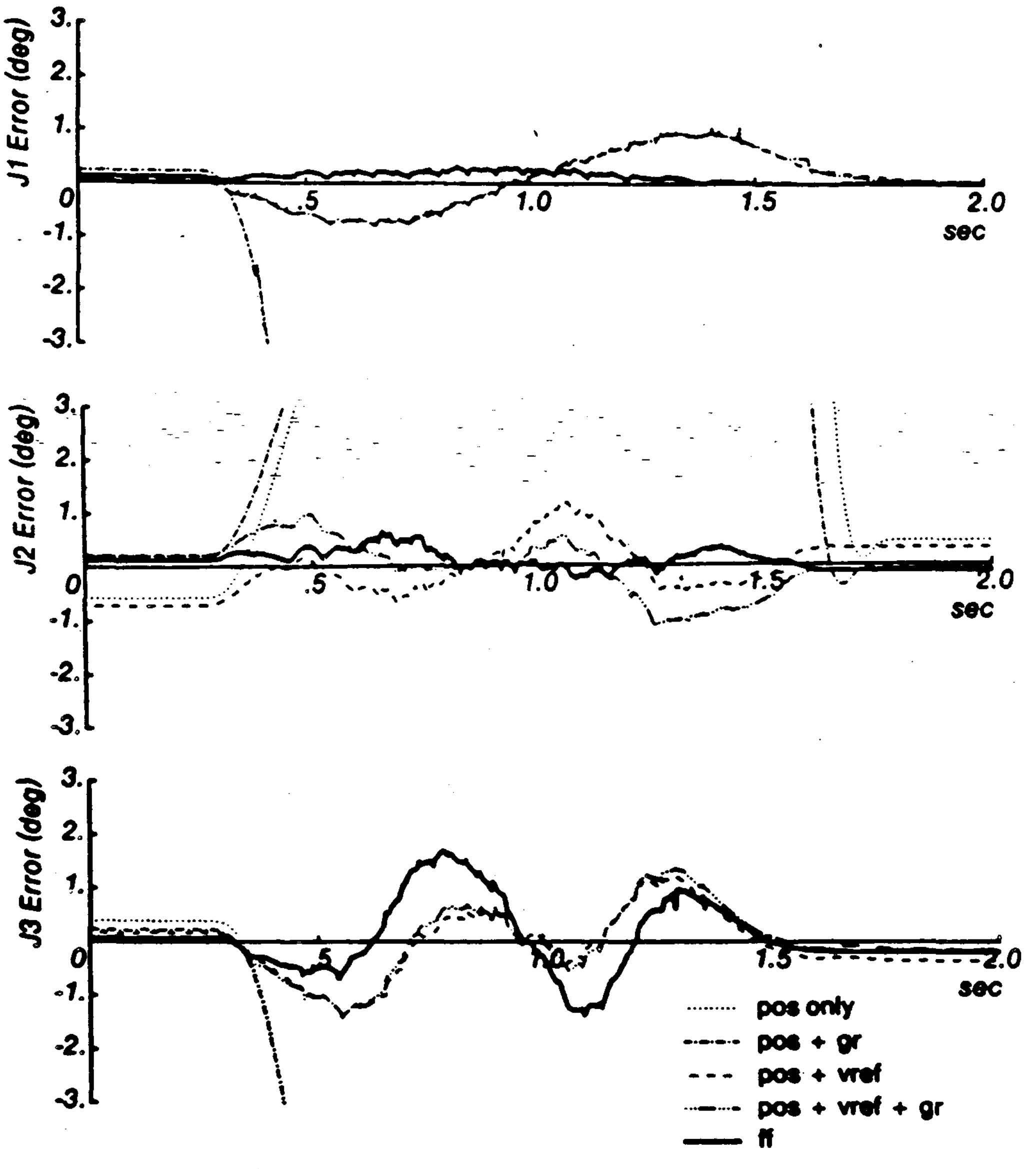


Figure 6.2: Trajectory errors of the 5 controllers for full 1.3s motion.

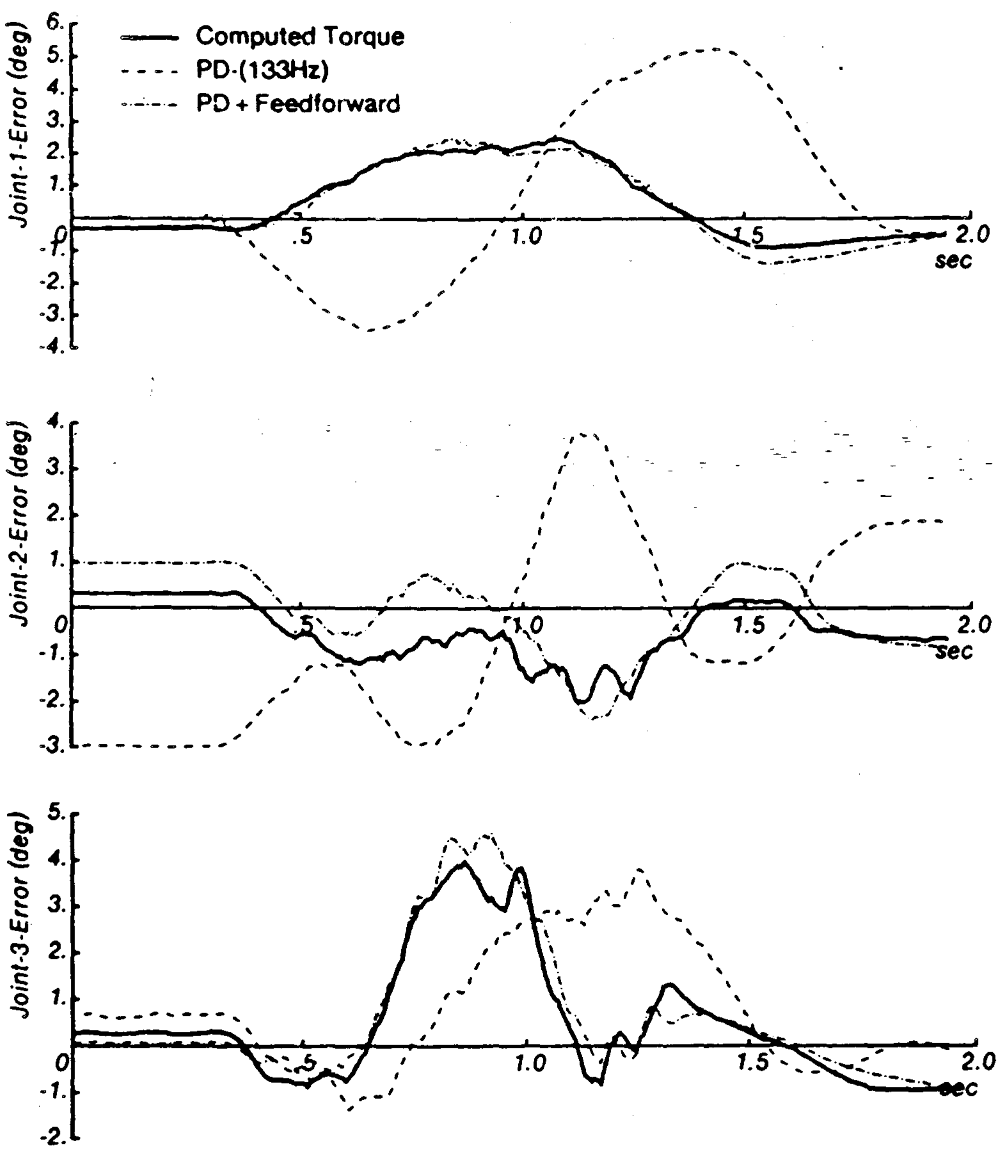


Figure 6.3: Trajectory errors of the three digital controllers for full 1.3s motion.

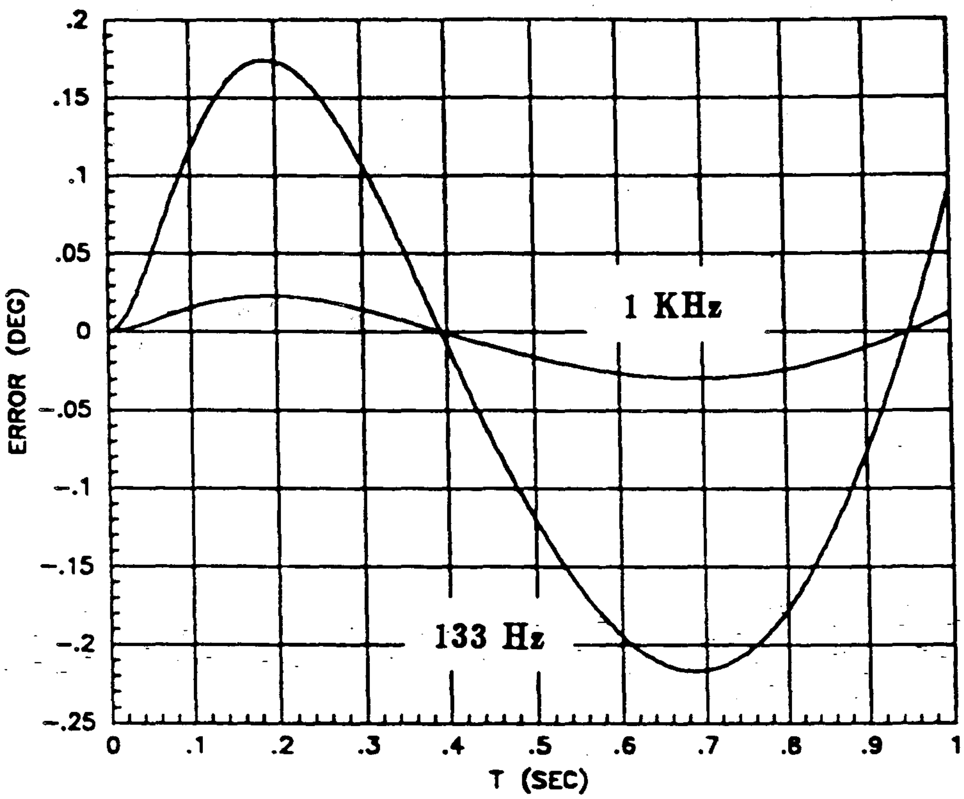


Figure 6.4: Trajectory errors for different sampling frequencies.

- Feedforward Adaptive Control

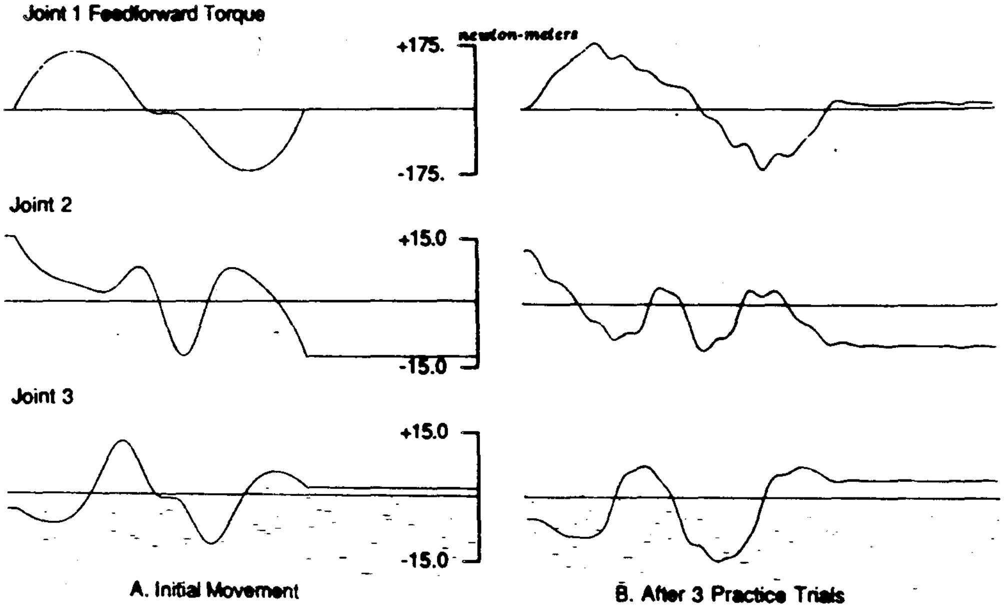


Figure 7.5: Feedforward Torques.

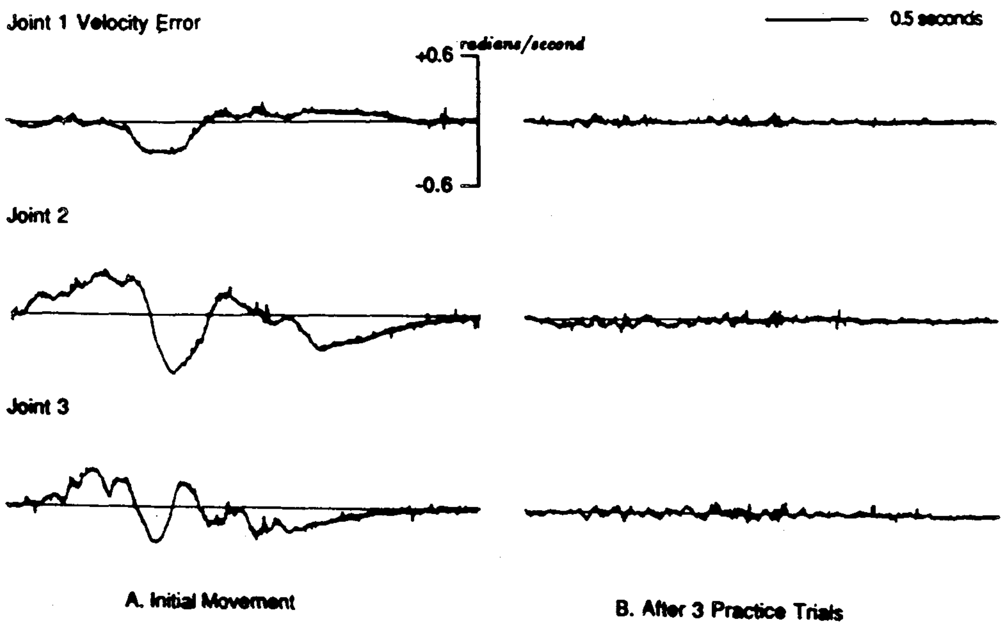


Figure 7.6: Velocity Errors.

⊙ Cartesian-based Control

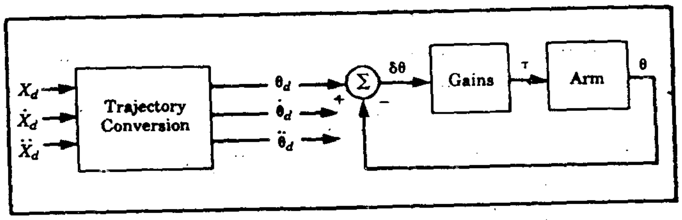


FIGURE 10.10 A joint-based control scheme with Cartesian path input.

Trajectory Conversion:

$$\Theta_d = INVKIN(\mathcal{X}_d),$$
$$\dot{\Theta}_d = J^{-1}(\Theta)\dot{\mathcal{X}}_d,$$
$$\ddot{\Theta}_d = \dot{J}^{-1}(\Theta)\dot{\mathcal{X}}_d + J^{-1}(\Theta)\ddot{\mathcal{X}}_d.$$

⊙ Concept

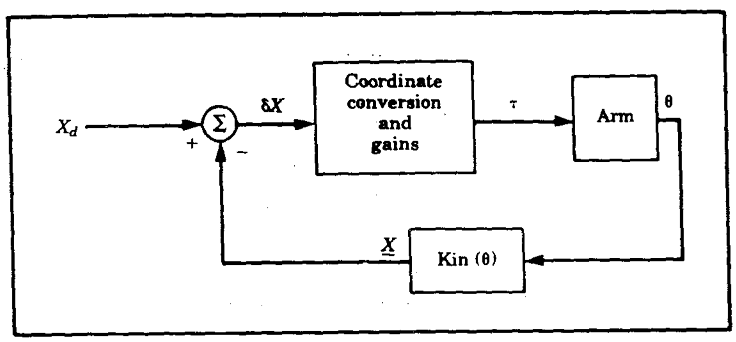


FIGURE 10.11 The concept of a Cartesian-based control scheme.

How to get τ ?

⊙ ~~position control~~ Inverse Jacobian Cartesian Control scheme

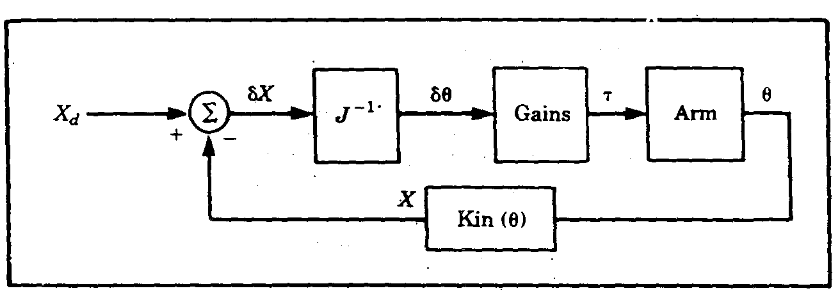


FIGURE 10.12 The inverse Jacobian Cartesian control scheme.

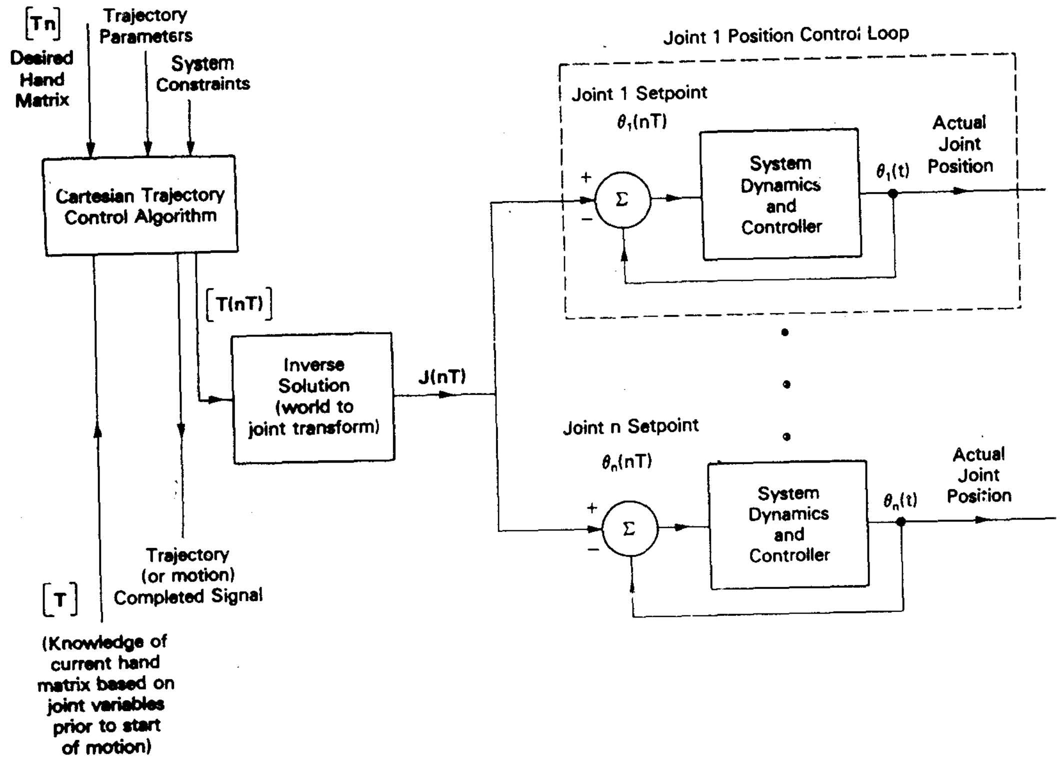
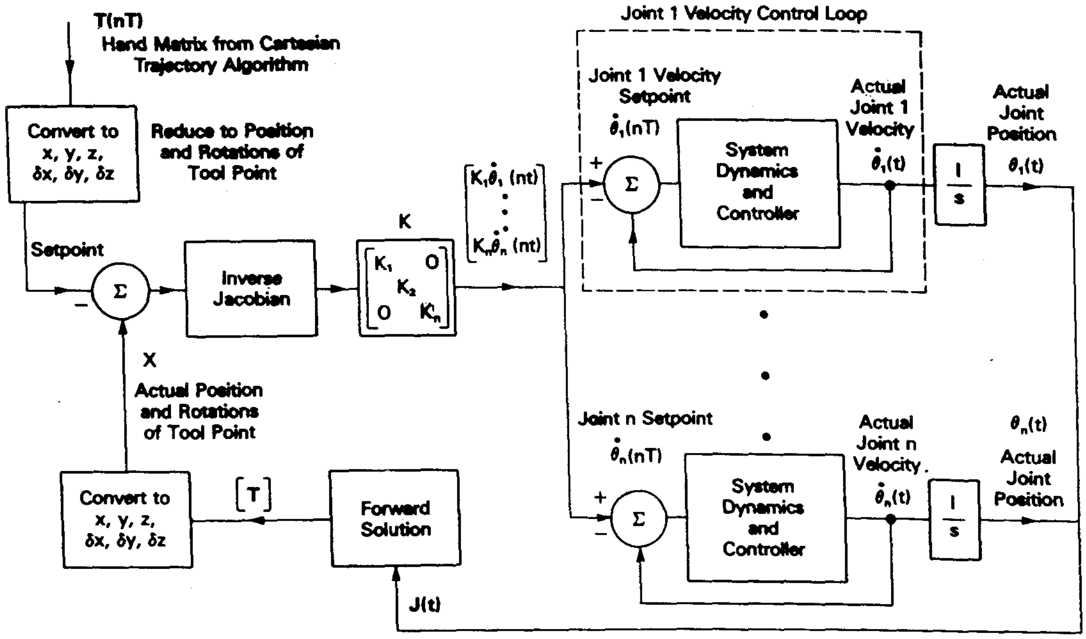


Figure 8.10.2. Resolved motion position control (RMPC).



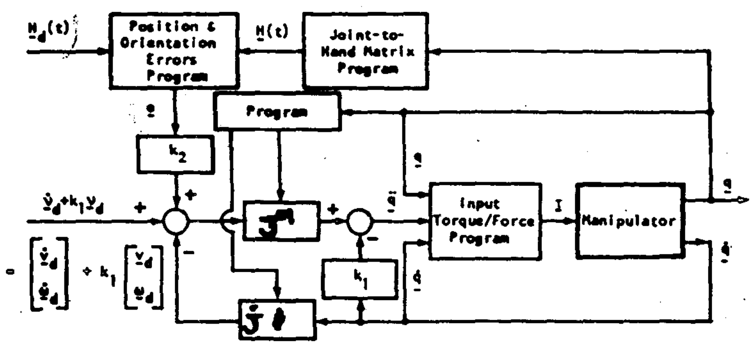
$\dot{\underline{X}} = J\dot{\underline{\theta}}$

<Closed-loop control for hand>

Figure 8.10.3. Resolved motion rate control (RMRC).

<Computed Torque Control 방법과 같은 개념>

- Model Based control



$M^{-1} = J^{-1}$

$\dot{M} = \dot{J}$

$\dot{\underline{X}} = J\dot{\theta}$

$\ddot{\underline{X}} = J\ddot{\theta} + \dot{J}\dot{\theta}$ -②

$\ddot{\theta} = -K_1\dot{\theta} + J^{-1}(\ddot{X}_d + K_1\dot{X}_d + K_2 e - \dot{J}\dot{\theta})$

Figure 5

Block diagram of the resolved-acceleration control algorithm.

$\ddot{\underline{X}} = \ddot{X}_d + K_1(\dot{X}_d - \dot{X}) + K_2 e$ -①

⊙ Transpose Jacobian Cartesian Control scheme

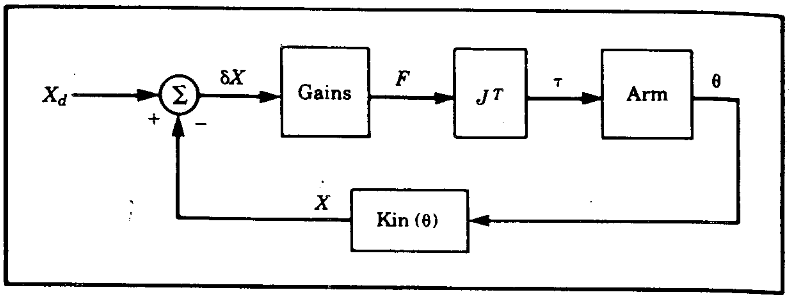


FIGURE 10.13 The transpose Jacobian Cartesian control scheme.

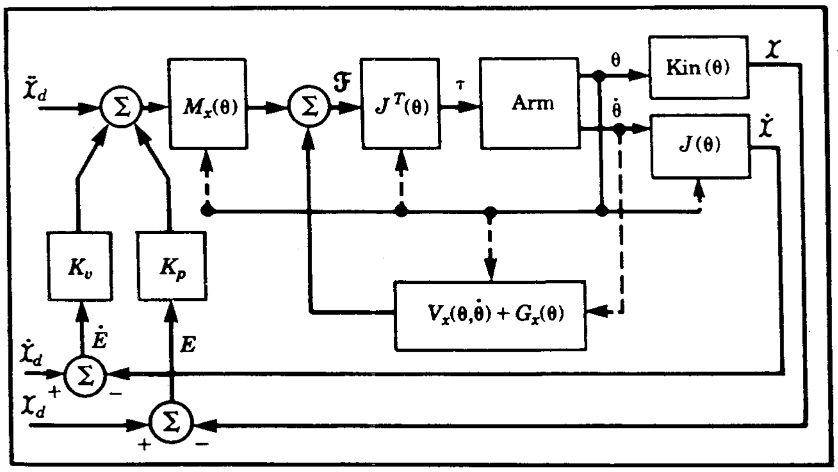


FIGURE 10.14 The Cartesian model based control scheme.

Operation Space Dynamic Modeling

$$\mathcal{F} = M_x(\theta)\ddot{x} + V_x(\theta,\dot{\theta}) + G_x(\theta)$$

$$\tau = J^T \mathcal{F}$$

$$M_x(\Theta) = J^{-T}(\Theta)\, M(\Theta)\, J^{-1}(\Theta),$$

$$V_x(\Theta,\dot{\Theta}) = J^{-T}(\Theta)\, \left(V(\Theta,\dot{\Theta}) - M(\Theta)\, J^{-1}(\Theta)\, \dot{J}(\Theta)\, \dot{\Theta}\right),$$

$$G_x(\Theta) = J^{-T}(\Theta)\, G(\Theta).$$

Compliance and Force Control

Industrial Robot : Positioning Device

(No contact to environment 가정)

Some Applications need the contact to environment

- washing a window
- scraping a paint off
- assemblying parts
- deburring, grinding

High precision robot은 이 작업을 할 수 있으나 too expensive

∴ Compliance & Force Control 고려

i) Washing a window with sponge

glass : high stiffness
sponge : high compliance

position control + high compliance

작업 가능

⇓

Compliance Control

Passive Compliance Control

Programmed Stiffness Control

ii) Scraping a paint off with a rigid scraping tool

glass : high stiffness

tool : high stiffness

tool or glass may be broken

작업이 어려움

⇓

Explicit Force Control

(f_d)

↓

Hybrid position/Force Control

A Framework for control in partially constrained tasks

- control 대상 : motion + force

- Contact이 발생하면, 그 contact을 통해 정의되는 Task는 일련의 constraints로 표현되는데, 이와같이 주어진 작업 구조의 기계적 또는 기하학적 특성으로부터 결정되는 Constraint를 'Natural' constraint라 한다

- Artificial Constraints : 제어 대상으로서, designer가 원하는대로 정해지는 desired motions and forces

- 동일 방향으로 force와 motion을 동시에 제어할 수 없음

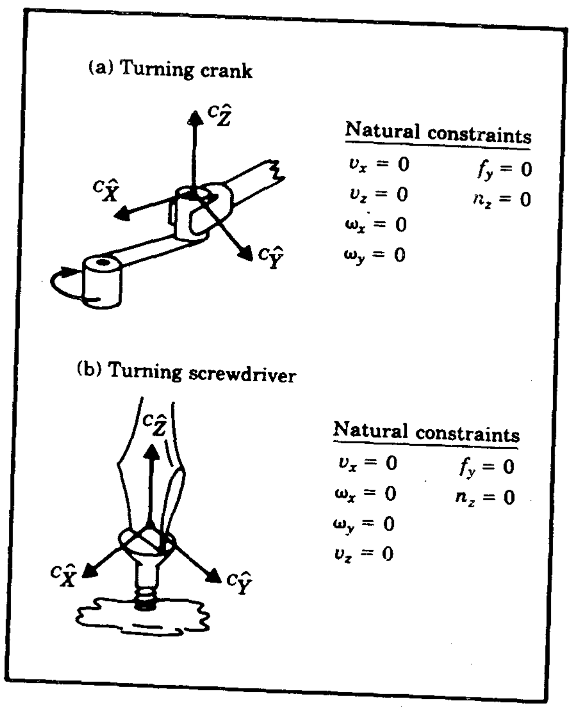


FIGURE 11.1 The natural constraints for two different tasks.

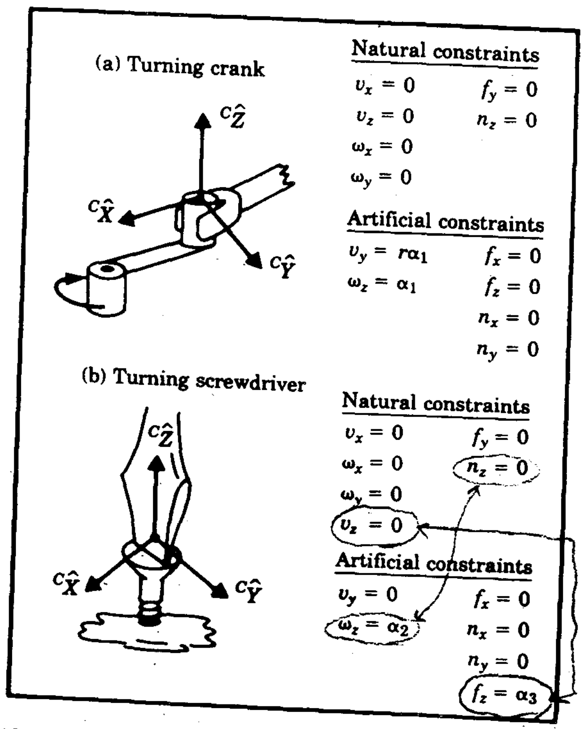


FIGURE 11.2 The natural and artificial constraints for two tasks.

- Assembly Strategy : Contact 상황 변화에 따른 natural constraints의 변화에 따라 new artificial constraints를 주는

Example of Assembly Strategy

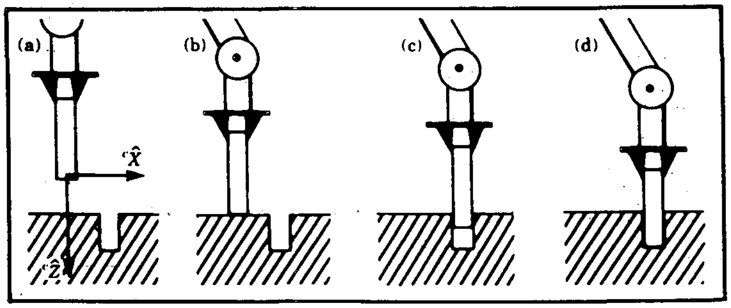


FIGURE 11.3 The sequence of four contacting situations for peg insertion.

Task	Natural constraints	Artifical Constraints	
(a)	$^{C}\mathcal{F} = 0.$	$^{C}\mathcal{V} = \begin{bmatrix} 0 \\ 0 \\ v_{approach} \\ 0 \\ 0 \\ 0 \end{bmatrix},$	Natural constraint인 $^{C}f_z$ 가 detect 되면
(b)	$^{C}v_z = 0,$ $^{C}\omega_x = 0,$ $^{C}\omega_y = 0,$ $^{C}f_x = 0,$ $^{C}f_y = 0,$ $^{C}n_z = 0.$	$^{C}v_x = v_{slide},$ $^{C}v_y = 0,$ $^{C}w_z = 0,$ $^{C}f_z = f_{contact},$ $^{C}n_x = 0,$ $^{C}n_y = 0.$	N.C.인 $^{C}v_z$ 가 detect 되면
(c)	$^{C}v_x = 0,$ $^{C}v_y = 0,$ $^{C}\omega_x = 0,$ $^{C}\omega_y = 0,$ $^{C}f_z = 0,$ $^{C}n_z = 0.$	$^{C}v_z = v_{insert},$ $^{C}w_z = 0,$ $^{C}f_x = 0,$ $^{C}f_y = 0,$ $^{C}n_x = 0,$ $^{C}n_y = 0,$	N.C인 $^{C}f_z$ 가 detect 되면
(d)	stop	stop	

⊙ Passive Compliance Control

RCC device

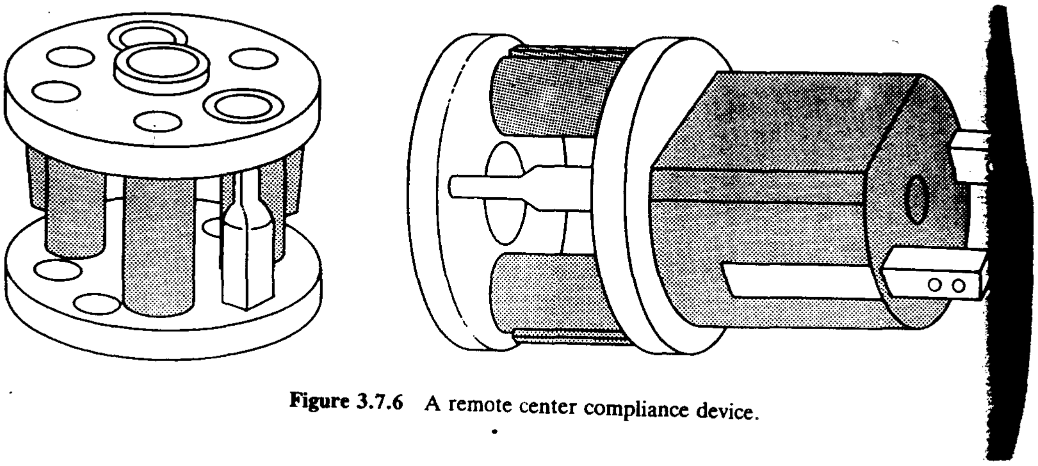


Figure 3.7.6 A remote center compliance device.

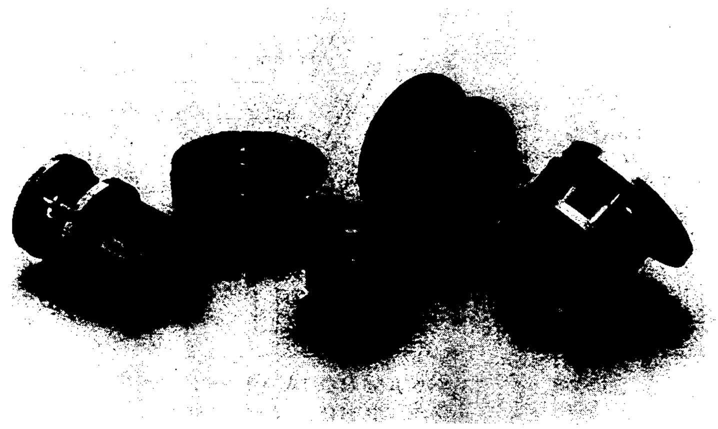


Figure 5.10.9. Family of remote center compliance (RCC) devices that combine high stiffness along the direction of the insertion force with low lateral and torsional stiffness (either perpendicular to or about the direction of the insertion force). (Courtesy of J. Rebman and the Lord Corp., Cary, NC.)

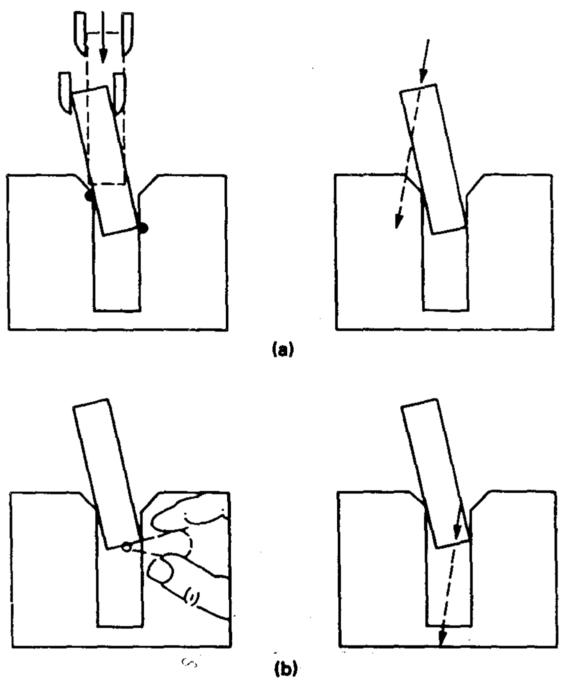


Figure 3.7.5: (a) peg being pushed into hole; (b) peg being pulled into hole. (Courtesy of J. Rebman and Lord Corporation, Cary, NC.)

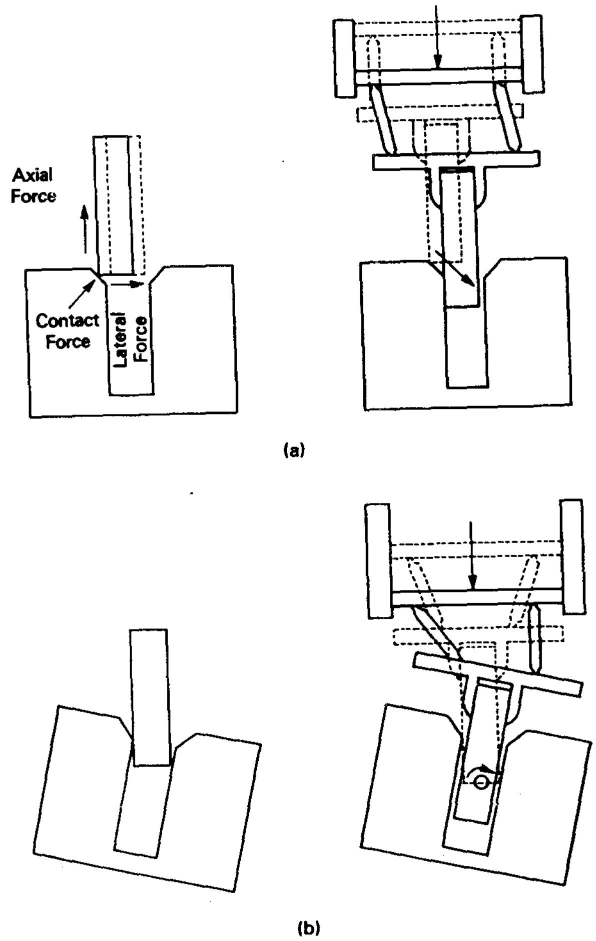


Figure 3.7.7 RCC for inserting a peg into chamfered hole: (a) insertion with laterial error; (b) insertion with rotational error. (Courtesy of J. Rebman and Lord Corporation, Cary, NC.)

◎ Active stiffness Control

Step 1) Set desired stiffness matrix K_p, in task space

2) Cartesian error $\delta x = x - x_d$

필요한 restoring force $F = -K_p \delta x$

3) 필요한 F를 만들기 위하여 τ 계산

$$\tau = J^T F$$

4) δx와 τ를 바로 연결시키면

$$\tau = -J^T K_p \delta x$$

5) Joint servo를 이용한다면

$\delta x = J \delta q$ 이므로 $\quad (q = \theta)$

$$\tau = -\underbrace{J^T K_p J}\, \delta q$$

$$= -K_q \delta q \rightarrow (*) \quad \text{or} \quad \tau = -K_q \delta q - K_v \dot{\delta q}$$

δq에 따르는 τ를 $(*)$와 같이 제어한다면, 원하는

K_p를 얻을 수 있다.

cf) stiffness control에 의한 position control,

Impedance Control,

◎ Dynamic Hybrid Position/Force Control

∘ Force Control Review

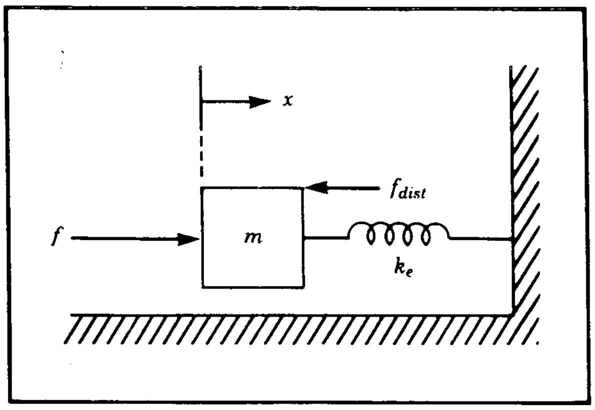


FIGURE 11.5 A spring-mass mystem.

~~[illegible]~~ Controlling Contact force : $f_e = k_e x$

System eq. : $f = m\ddot{x} + k_e x + f_{dist}$

Control input f를 f_e로 표현하면

$$f = m k_e^{-1} \ddot{f}_e + f_e + f_{dist}.$$

Using the partitioned controller concept, we use

$$\alpha = m k_e^{-1},$$
$$\beta = f_e + f_{dist}$$

to arrive at the control law,

$$f = m k_e^{-1} \left[\ddot{f}_d + k_{vf}\dot{e}_f + k_{pf} e_f \right] + f_e + f_{dist}, \tag{11.10}$$

where $e_f = f_d - f_e$ is the force error between the desired force, f_d, and the sensed force on the environment, f_e. If we could compute (11.10), we would have the closed loop system

$$\ddot{e}_f + k_{vf}\dot{e}_f + k_{pf} e_f = 0. \tag{11.11}$$

실제로 f_{dist}는 unknown, 그러므로 $\beta = f_e$

따라서, error equation은

$$\ddot{e}_f + k_{vf}\dot{e}_f + k_{pf}e_f = \frac{f_{dist}}{m k_e^{-1}}$$

Steady-state error $e_f = \dfrac{f_{dist}}{m k_e^{-1} k_{pf}}$ $\quad k_e$ big $\rightarrow$ e_f big

개선책으로 $\beta = f_d$ 하면, error eq.은

$$\ddot{e}_f + k_{vf}\dot{e}_f + k_{pf}e_f = \frac{f_e + f_{dist} - f_d}{m k_e^{-1}} = \frac{f_{dist} - e_f}{m k_e^{-1}}$$

S-S error $e_f = \dfrac{f_{dist}}{1 + m k_e^{-1} k_{pf}} \doteqdot f_{dist}$

~~force control은 range control 개념~~,

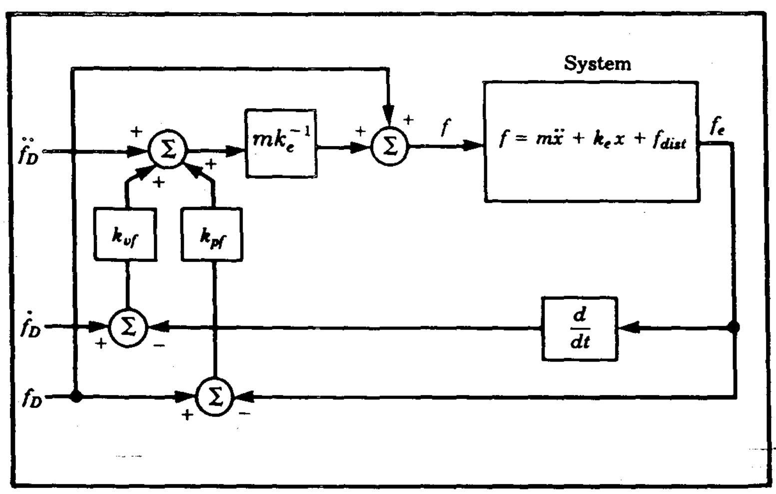


FIGURE 11.6 A force control system for the spring-mass system.

Practical Consideration

① f_d is not ~~time~~ varing $\rightarrow$ $f_d = \text{const}$

$$\dot{f}_d = \ddot{f}_d = 0$$

② $\dot{f}_e$ is not feasible (f_e is Sensing noisy signal)

$\dot{f}_e = k_e \dot{x}$ 으로 구함

Control law from $f = m k_e^{-1} (\ddot{f}_d + k_{vf} \dot{e}_f + k_{pf} e_f) + f_d$

$$f = m k_e^{-1} [k_{pf} e_f - k_{vf} \dot{f}_e] + f_d$$

$$= m k_e^{-1} [k_{pf} e_f - k_{vf} k_e \dot{x}] + f_d$$

$$= m [\frac{k_{pf}}{k_e} e_f - k_{vf} \dot{x}] + f_d$$

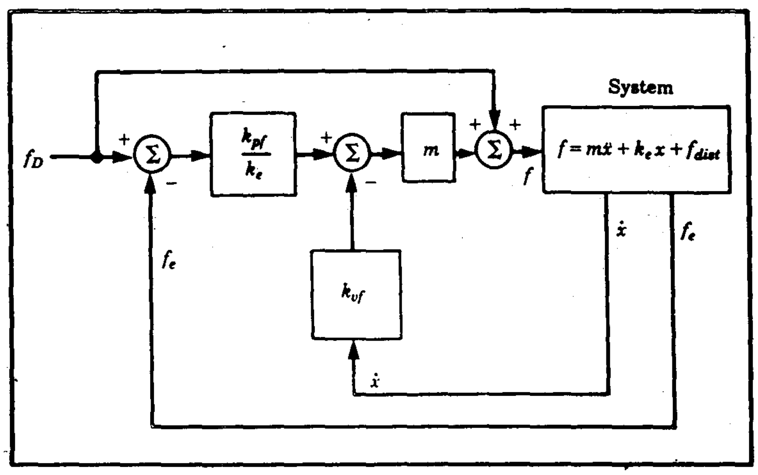


FIGURE 11.7 A practical force control system for the spring-mass.

· Hybrid Position / force Control

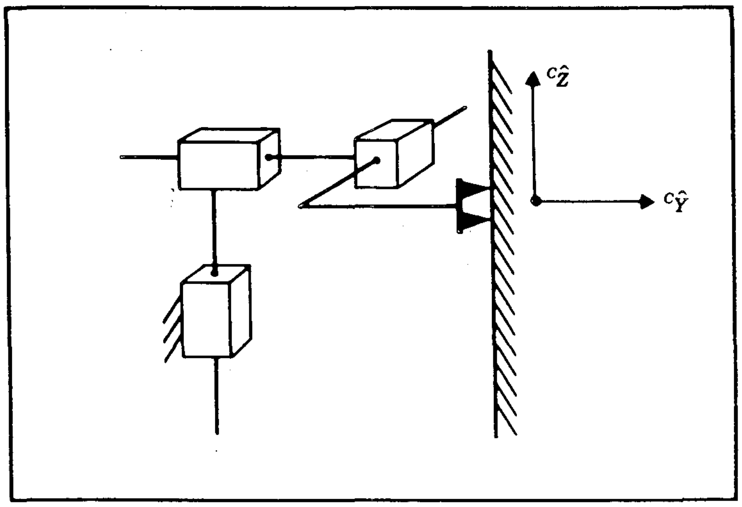


FIGURE 11.9 A Cartesian manipulator with three degrees of freedom contacting a surface.

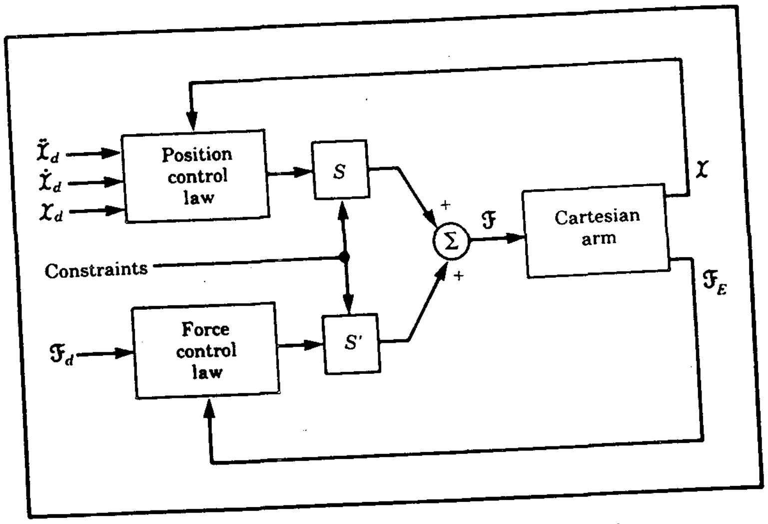


FIGURE 11.10 The hybrid controller for a 3-DOF Cartesian arm.

Ex) $\hat{x}, \hat{z}$ 방향 position Control
$\hat{y}$ 방향 force Control

$$S = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$S' = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

For general manipulator,

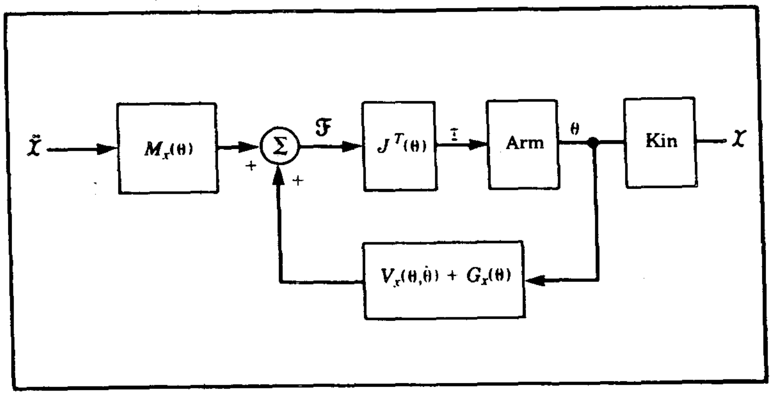


FIGURE 11.11 The Cartesian decoupling scheme introduced in Chapter 10.

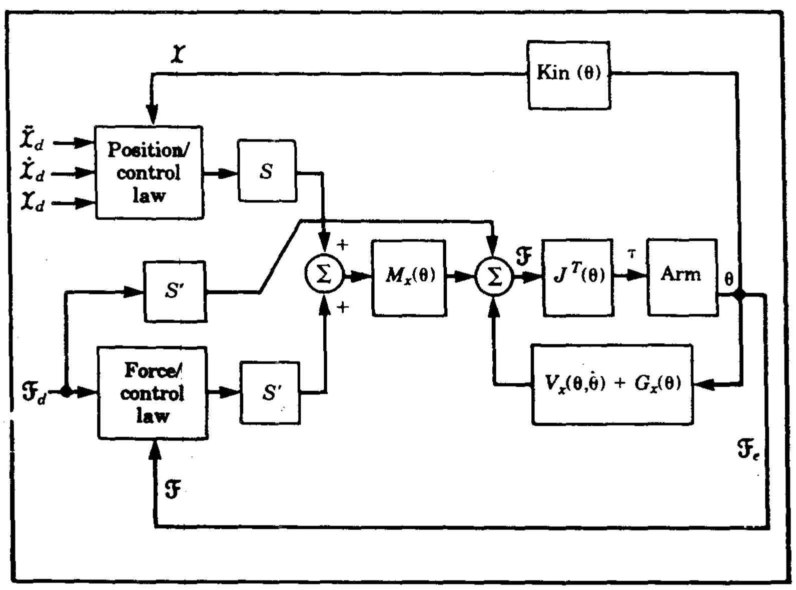
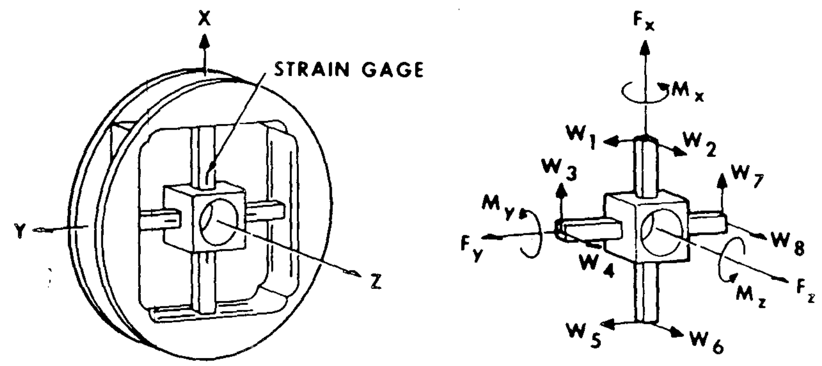


FIGURE 11.12 The hybrid position/force controller for a general manipulator. For simplicity, the velocity feedback loop has not been shown.

Force Measure



TRANSFORMATION MATRIX
UNDER IDEAL CONDITIONS

FORCES AND TORQUES REFERENCED TO X-Y-Z SENSOR COORDINATES

$$
\begin{bmatrix} F_x \\ F_y \\ F_z \\ M_x \\ M_y \\ M_z \end{bmatrix}
=
\begin{bmatrix}
0 & 0 & k_{13} & 0 & 0 & 0 & k_{17} & 0 \\
k_{21} & 0 & 0 & 0 & k_{25} & 0 & 0 & 0 \\
0 & k_{32} & 0 & k_{34} & 0 & k_{36} & 0 & k_{38} \\
0 & 0 & 0 & k_{44} & 0 & 0 & 0 & k_{48} \\
0 & k_{52} & 0 & 0 & 0 & k_{56} & 0 & 0 \\
k_{61} & 0 & k_{63} & 0 & k_{65} & 0 & k_{67} & 0
\end{bmatrix}
\begin{bmatrix} W_1 \\ W_2 \\ W_3 \\ W_4 \\ W_5 \\ W_6 \\ W_7 \\ W_8 \end{bmatrix}
$$

FORCES SENSED AT SPOKE ELEMENTS

Figure 5.10.7. JPL "Maltese cross" version of the sensor shown in Fig. 5.10.5. The matrix transformation relating the strain gage outputs to the actual *x*, *y*, and *z* components of the forces and torques is also given. (Courtesy of A.K. Bejczy, Jet Propulsion Laboratory, Pasadena, CA.)

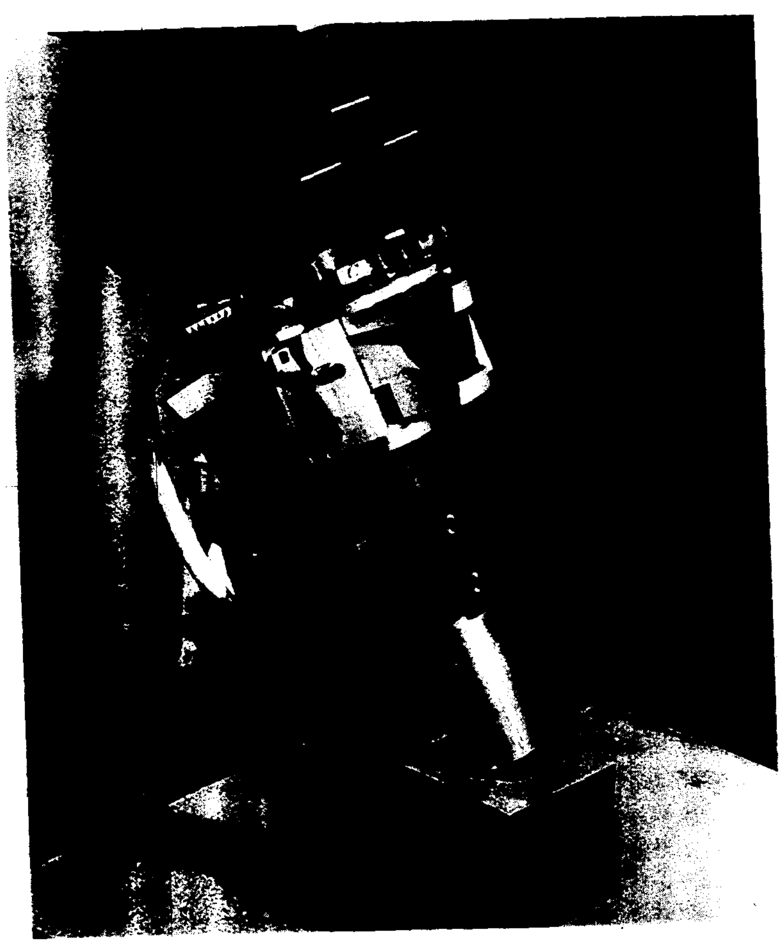


Figure 5.10.12. Chamferless insertion of a peg in a hole using an IRCC. (Courtesy of D. S. Seltzer, the Charles Stark Draper Laboratory, Inc., Cambridge MA.)

Figure 5.10.13. Edge following using an IRCC. Force information from the sensor is used to adjust the y position of the rotary cutting tool so that a constant force of 1 to 2 lb. is maintained against the plywood. (Courtesy of D.S. Seltzer, the Charles Stark Draper Laboratory, Cambridge, MA. From reference [39].)

(b)

Figure 5.10.14. A commercial instrumented remote center compliance: (a) artist's conception of an IRCC attached to a robot; (b) actual IRCC mounted on a robot shown grasping a motor armature. (Courtesy of P. Cholakis, Barry Wright Corp., Watertown, MA.)

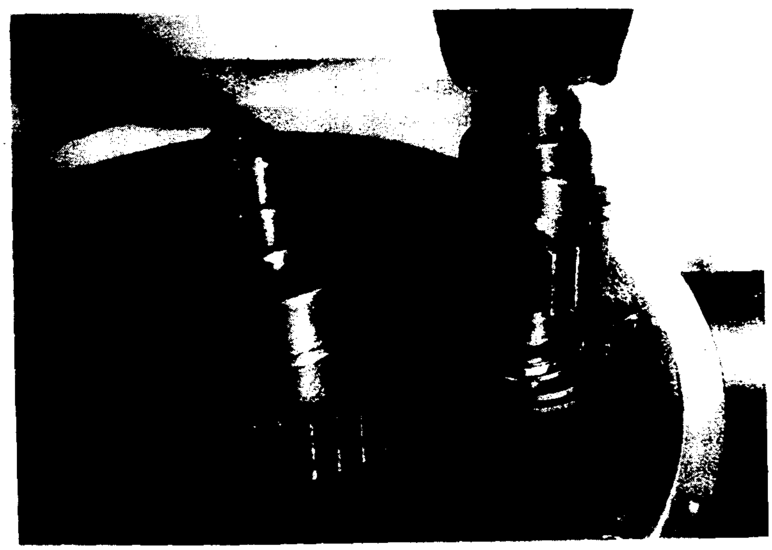


Figure 5.10.15. Successful insertion of a gear-shaft assembly into a bearing has been achieved by using a three-axis strain gage sensor mounted on the gripper's fingers to provide electronic compliance. (Courtesy of T.A. Brownell, General Electric Company, Corporate Research and Development, Schenectady, NY. From reference [40].)