ADVANCED CONTROL SYSTEMS

Motion Control - Operation Space -

Riccardo Muradore

UNIVERSITÀ di VERONA
Dipartimento di INFORMATICA

Outline

Operational Space

Previous control schemes assumes that the trajectories are provided in the joints space $(q_d(\cdot), \dot{q}_d(\cdot), \ddot{q}_d(\cdot))$.

However, desired trajectories are usually computed in the Operation space (i.e. Cartesian space) where the robot and the objects live $(x_d(\cdot), \dot{x}_d(\cdot), \ddot{x}_d(\cdot))$.

Two solutions are available:

$$x_d(t) = \begin{bmatrix} x(t) \\ y(t) \\ z(t) \\ \psi(t) \\ \varphi(t) \\ \theta(t) \end{bmatrix}$$

Previous control schemes assumes that the trajectories are provided in the joints space $(q_d(\cdot), \dot{q}_d(\cdot), \ddot{q}_d(\cdot))$.

However, desired trajectories are usually computed in the Operation space (i.e. Cartesian space) where the robot and the objects live $(x_d(\cdot), \dot{x}_d(\cdot), \ddot{x}_d(\cdot))$.

Two solutions are available:

1. map $x_d(\cdot)$, $\dot{x}_d(\cdot)$, $\ddot{x}_d(\cdot)$ into $q_d(\cdot)$, $\dot{q}_d(\cdot)$, $\ddot{q}_d(\cdot)$ using the inverse kinematics and then relay on joint space control laws.
   - 1a inversion of direct kinematics, inversion of first-order and second-order differential kinematics to transform the desired trajectories of end-effector position, velocity and acceleration into the corresponding quantities at the joint level;
   - 1b joint positions through kinematics inversion, and then compute velocities and accelerations via numerical differentiation (e.g. industrial robots).

$x_d(\cdot) \longrightarrow q_d(\cdot)$

Operational Space

Previous control schemes assumes that the trajectories are provided in the joints space $(q_d(\cdot), \dot{q}_d(\cdot), \ddot{q}_d(\cdot))$.

However, desired trajectories are usually computed in the Operation space (i.e. Cartesian space) where the robot and the objects live $(x_d(\cdot), \dot{x}_d(\cdot), \ddot{x}_d(\cdot))$.

Two solutions are available:

1. map $x_d(\cdot)$, $\dot{x}_d(\cdot)$, $\ddot{x}_d(\cdot)$ into $q_d(\cdot)$, $\dot{q}_d(\cdot)$, $\ddot{q}_d(\cdot)$ using the inverse kinematics and then relay on joint space control laws.
   - 1a inversion of direct kinematics, inversion of first-order and second-order differential kinematics to transform the desired trajectories of end-effector position, velocity and acceleration into the corresponding quantities at the joint level;
   - 1b joint positions through kinematics inversion, and then compute velocities and accelerations via numerical differentiation (e.g. industrial robots).
2. developed control scheme directly in the operational space.

   The measured joint space variables are transformed into the corresponding operational space variables through direct kinematics relations.

Operational Space

Pros

When considering the environment, it is necessary to control both the position and the interaction forces.

Cons

Operational Space algorithms suffer of higher computational space, because they use direct kinematics relations to transform joint space measurements into the operational space ones.

Operational Space PD Control with gravity compensation

Let x_d be the desired *constant* target position ($\dot{x}_d = 0$); we need to design a PD Control with gravity compensation on the Operational Space that steers the robot pose x to x_d. (Regulation problem)

$x(t) \rightarrow x_d$

The operational space *error at the end-effector* is

$$\tilde{x}(t) = x_d - x(t)$$

The goal of the Regulation problem is to get

$$\tilde{x}(t) \rightarrow 0 \quad \equiv \quad x(t) \rightarrow x_d$$

asymptotically.

As for the joint space scenario, we will start choosing a Lyapunov candidate function

$$V(\dot{q}, \tilde{x}) = \frac{1}{2}\dot{q}^T B(q)\dot{q} + \frac{1}{2}\tilde{x}^T K_P \tilde{x} \tag{1}$$

with K_P a symmetric positive definite matrix.

$V(\dot{q}, \tilde{x})$ is a positive definite function around the equilibrium point $\dot{q} = 0, \tilde{x} = 0$.

Operational Space PD Control with gravity compensation

Evaluating its time derivative

$$\dot{V}(\dot{q},\tilde{x}) = \dot{q}^T B(q)\ddot{q} + \frac{1}{2}\dot{q}^T \dot{B}(q)\dot{q} + \dot{\tilde{x}}^T K_P \tilde{x}, \qquad (2)$$

along the trajectories of

$$\boxed{B(q)\ddot{q} + C(q,\dot{q})\dot{q} + F\dot{q} + g(q) = \tau}$$

we have

$$\boxed{\dot{V}(\dot{q},\tilde{x}) = \underbrace{-\dot{q}^T F\dot{q}}_{\leq 0} + \underbrace{\dot{q}^T(\tau - g(q))}_{\tau} + \underbrace{\dot{\tilde{x}}^T K_P \tilde{x}}_{?}.}$$

where we exploited $\dot{q}^T N\dot{q} = 0$.

Using the analytical Jacobian $J_A(q)$ to relate Cartesian velocity and joint velocity $\dot{x} = J_A(q)\dot{q}$, we have

$$\boxed{\dot{\tilde{x}} = -J_A(q)\dot{q}}$$

since $\dot{\tilde{x}} = \dot{x}_d - \dot{x} = -\dot{x}$ because $\dot{x}_d = 0$.

$$
\begin{aligned}
\dot V(\dot q,\tilde x) &= -\dot q^T F\dot q + \dot q^T(\tau - g(q)) + \dot{\tilde x}^T K_P\tilde x \\
&= -\dot q^T F\dot q + \dot q^T(\tau - g(q)) - \dot q^T J_A^T(q)K_P\tilde x \\
&= -\dot q^T F\dot q + \dot q^T(\tau - g(q) - J_A^T(q)K_P\tilde x).
\end{aligned}
$$

P action

D action

By choosing the control law as

$$
\boxed{\tau = g(q) + J_A^T(q)K_P\tilde x - J_A^T(q)K_D J_A(q)\dot q} \tag{3}
$$

with K_D a symmetric positive definite matrix, we end up with

$$
\boxed{\dot V(\dot q,\tilde x) = -\dot q^T F\dot q - \dot q^T J_A^T(q)K_D J_A(q)\dot q.}
$$

≤ 0

$F > 0$ ≥ 0

The control law ~~u~~ τ performs a *nonlinear compensating action of joint space gravitational forces* and an *operational space linear PD control action*.

Since $\dot{V}(\dot{q},\tilde{x}) \preceq 0$, the Lyapunov function $V(\dot{q},\tilde{x})$ decreases as long as $\dot{q} \neq 0$. We have

$$\boxed{\dot{V} = 0 \quad \Longleftrightarrow \quad \dot{q} = 0}$$

To apply the Krasovskii-LaSalle theorem, we have to evaluate the invariant space characterized by $\dot{q} = 0$. Setting $\dot{q} = 0$ in

$$\underbrace{B(q)\ddot{q} + \cancel{C(q,\dot{q})\dot{q}} + \cancel{F\dot{q}} + g(q)}_{\text{robotic model}} = \underbrace{g(q) + J_A^T(q)K_P\tilde{x} - \cancel{J_A^T(q)K_D J_A(q)\dot{q}}}_{\text{control law}}$$

closed-loop system

we have

$$B(q)\ddot{q} + \cancel{g(q)} = \cancel{g(q)} + J_A^T(q)K_P\tilde{x}$$

$$B(q)\ddot{q} = J_A^T(q)K_P\tilde{x}$$

which implies that

B(q) is nonsingular

$$\ddot{q} = 0 \quad \Longleftrightarrow \quad \boxed{J_A^T(q)K_P\tilde{x} = 0}$$

Hp. If the analytical Jacobian has full rank, the previous equation is equivalent to

$J_A^T K_P$ is full rank

$$\boxed{\tilde{x} = 0,} \quad \text{i.e.} \quad x = x_d$$

IF NOT otherwise the state of the robot will converge to the set

direct kinematics

$$\{(q,\dot{q}) : \dot{q} = 0 \text{ and } \boxed{K_P(x_d - x) \in \ker(J_A^T(q))} \text{ where } x = \kappa(q)\}$$

Theorem (Regulation Problem)

Let x_d be the desired pose for the manipulator

$$B(q)\ddot{q} + C(q,\dot{q})\dot{q} + F\dot{q} + g(q) = \tau.$$

The PD controller with gravity compensation

$$\tau = g(q) + J_A^T(q)K_P(x_d - x) - J_A^T(q)K_D J_A(q)\dot{q}, \qquad K_P \succ 0, K_D \succ 0$$

guarantees that the state $(q,\dot{q})$ of the robot converges asymptotically to the set

$$\left\{(q,0) : K_P(x_d - x) \in \ker(J_A^T(q)) \text{ where } x = \kappa(q)\right\}.$$

If $J_A(q)$ has full row rank, then the robot state converges asymptotically to the set

$$\left\{(q,0) : x_d = \kappa(q)\right\}. \qquad \boxed{x = x_d}$$

Remark 1. Let n and m be the number of degrees of freedom of the manipulator and the dimension of the operational space, respectively.
For any initial condition $(q(0), \dot{q}(0))$, if there are no singularities along the path (i.e. values of q such that $\text{rank}(J^T(q)) < m \leq n$), then the robot asymptotically stabilizes to a configuration if $m = n$ or to a set of configurations is $m < n$; in both cases it results

$$\boxed{x = x_d, \qquad \dot{q} = 0}$$

Remark 2. It is possible to use instead of

$$\boxed{\tau = g(q) + J_A^T(q) K_P \tilde{x} - J_A^T(q) K_D J_A(q) \dot{q}}$$

the control law

$$\boxed{\tau = g(q) + J_A^T(q) K_P \tilde{x} - K_D \dot{q}}$$

$K_D \succ 0$

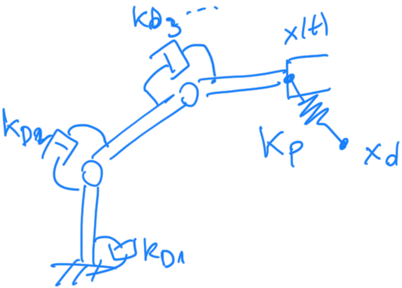
Check how the proof should be changed.

Operational Space PD Control with gravity compensation

Mechanical meaning

$$\tau = g(q) + J_A^T(q) \underbrace{K_P \tilde{x}}_{\text{spring at EE}} - \underbrace{K_D \dot{q}}_{\text{dampers at the joints}}$$

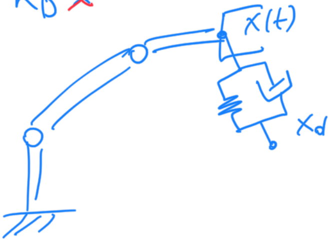
Derivative term



4 3 2 1

$$\begin{aligned} \tau &= g(q) + J_A^T(q) K_P \tilde{x} - J_A^T(q) K_D J_A(q) \dot{q} \\ &= g(q) + J_A^T(q) \underbrace{\left[K_P \tilde{x} + K_D \dot{\tilde{x}} \right]}_{\text{spring-damper at EE}} \end{aligned}$$

$$-J_A^T(q) K_D \underline{J_A(q)\dot{q}} =$$
$$= J_A^T K_D \cancel{\dot{\tilde{x}}}$$



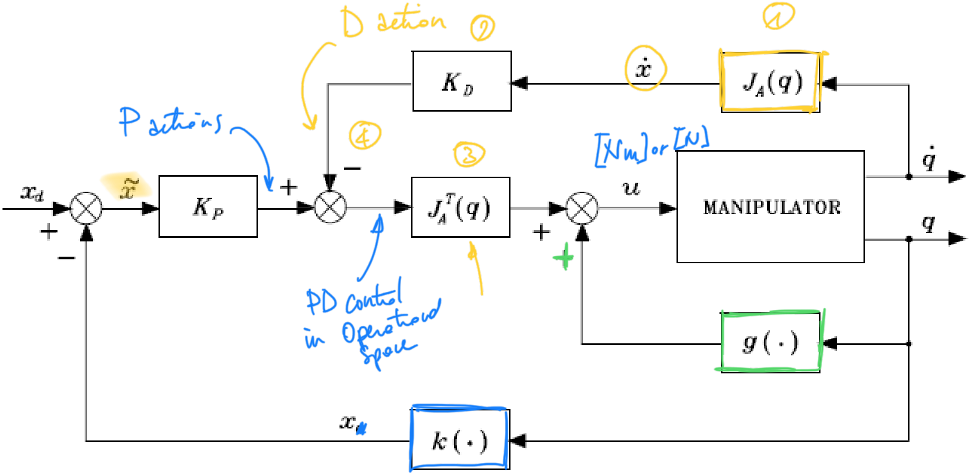


Figure: Operational Space PD control with gravity compensation block scheme.


Project

PROJECT – Assignment # 9

To do

- Design the Operational Space PD control law with gravity compensation

Michele Sandrini
Nicola

Operational Space Inverse Dynamics Control

The design of the *Operational Space Inverse Dynamics Control* follows the same line of reasoning of the *Joint Space Inverse Dynamics Control*.

we solve the tracking problem!

Given the equations of motion

$$\begin{aligned} B(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) &= \tau \\ B(q)\ddot{q} + n(q,\dot{q}) &= \tau \end{aligned}$$

we define τ in such a way to cancel out the nonlinearity and decouple the joint variables (*Inner control loop – Inverse dynamics control*)

$$\boxed{\tau = B(q)y + n(q,\dot{q}).}$$

y auxiliary signal

We end up again with

$$\boxed{\ddot{q} = y}$$

$\ddot{q}_i = y_i, \quad i = 1, \dots, n$

where y should be chosen to track the desired trajectory $\underline{x_d(t), \dot{x}_d(t), \ddot{x}_d(t)}$ (*Outer control loop – Stabilizing linear control law*)

Operational Space Inverse Dynamics Control

Assumption: non-redundant manipulators with $m = n$, i.e. J_A is a square nonsingular matrix. [$\Longleftrightarrow$ exact linearization in the operational space]

Using second-order differential geometry, we know that the acceleration of the end-effector is given by

$\dot{x} = J_A(q)\dot{q}$

$\frac{d}{dt}$

B

$$\boxed{\ddot{x} = J_A(q)\ddot{q} + \dot{J}_A(q,\dot{q})\dot{q}} \tag{4}$$

$\ddot{q} = y$

A

we can think of designing y as

A + B $\Rightarrow$

$$y = J_A^{-1}(q)\big(\underbrace{\ddot{x}_d + K_D\dot{\tilde{x}} + K_P\tilde{x}}_{\text{PD controller w.r.t. } \tilde{x}} - \dot{J}_A(q,\dot{q})\dot{q}\big) \tag{5}$$

where

$$\begin{aligned}\tilde{x} &= x_d - x\\ \dot{\tilde{x}} &= \dot{x}_d - \dot{x}\end{aligned}$$

$\gamma^0 \; \gamma^0$

and K_D, K_P are diagonal positive definite matrices.

UNIVERSITÀ di VERONA
Dipartimento di INFORMATICA
ALTAIR

Substituting (4) in (5) we have the n second-order linear differential equations for the error in the operational space

$$\begin{aligned} \ddot{q} &= y \\ \ddot{q} &= J_A^{-1}(q)\left(\ddot{x}_d + K_D\dot{\tilde{x}} + K_P\tilde{x} - \dot{J}_A(q,\dot{q})\dot{q}\right) \\ J_A(q)\ddot{q} &= \ddot{x}_d + K_D\dot{\tilde{x}} + K_P\tilde{x} \underline{- \dot{J}_A(q,\dot{q})\dot{q}} \\ J_A(q)\ddot{q} + \dot{J}_A(q,\dot{q})\dot{q} &= \ddot{x}_d + K_D\dot{\tilde{x}} + K_P\tilde{x} \\ \boxed{\ddot{x}} &= \ddot{x}_d + K_D\dot{\tilde{x}} + K_P\tilde{x} \end{aligned}$$

$\ddot{\tilde{x}} = \ddot{x}_d - \ddot{x}$

and finally

$$\ddot{\tilde{x}} + K_D\dot{\tilde{x}} + K_P\tilde{x} = 0 \qquad (6)$$

$\tilde{x}(t)$ depends on the i.c.

The matrices $K_D \succ 0, K_P \succ 0$ determine the convergence rate to zero. (independently of the configuration!)

$\tilde{x} \to 0 \quad \forall$ i.c.

K_D, K_P diagonal $\Rightarrow \ddot{\tilde{x}}_i + K_{D_i}\dot{\tilde{x}}_i + K_{P_i}\tilde{x}_i = 0$

$x_i \in \{x, y, z, \psi, \varphi, \theta\}$

The overall control law is

$$\tau = B(q)\left[J_A^{-1}(q)\big(\ddot{x}_d + K_D\dot{\tilde{x}} + K_P\tilde{x} - \dot{J}_A(q,\dot{q})\dot{q}\big)\right] + C(q,\dot{q})\dot{q} + g(q)$$

Exercise. It is possible to derive this law by starting from the dynamic model of the robot described in the operational space.

Operational Space Inverse Dynamics Control

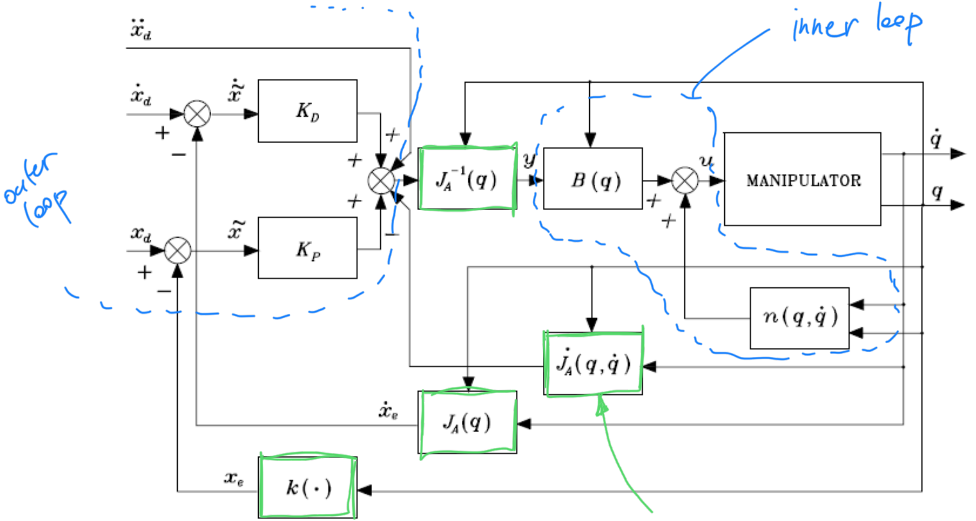


Figure: Operational Space Inverse Dynamics block scheme

Observations:

- singularities and/or redundancy influence the Jacobian, and the induced effects are somewhat difficult to quantify and handle with an operational space controller
- for redundant manipulators, an operational space control scheme should incorporate a redundancy handling technique inside the feedback loop
- for redundant manipulators ($m < n$) it is necessary to replace the matrix $J_A^{-1}(q)$ with the pseudo-inverse of the analytical Jacobian $J_A^{+}(q)$
- for redundant manipulators ($m < n$), there is an internal dynamics of dimension $n - m$ corresponding to the null-space torque.


Project

PROJECT – Assignment # 10

To do

- Design the Operational Space Inverse Dynamics Control law

Simone
Matteo