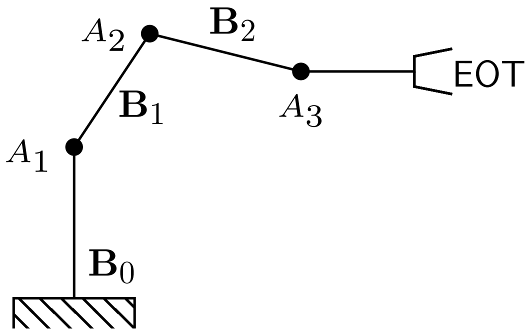
FEEDBACK CONTROL OF RIGID ROBOT MANIPULATORS
(open kinematic chains)

Patrick Danès
Univ. Toulouse III Paul Sabatier & LAAS-CNRS
(0)561.33.78.25. – (0)685.67.76.49.
patrick.danes@laas.fr
http://homepages.laas.fr/danes/UPS/2ASRI

FEATURES OF A MANIPULATOR ROBOT



★ Open rigid kinematic chain

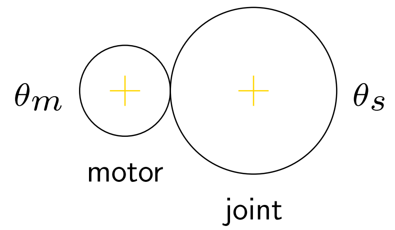
- bodies (links)
- articulation (joints): revolute (**R**) / prismatic (**P**)
- EOT = end of arm tooling (end effector)

★ Actuators / Actuating systems

- power supply
- power amplifier: control signal $\mapsto$ power to servomotor
- servomotor: ((pneumatic)), (hydraulic), electric
  - ▷ permanent magnet DC-motors
    - · stator (permanent magnet) which produces a radial magnetic flux
    - · rotating armature (winding around ferromagnetic core)
    - · commutation logic by means of brushes from external feed to armature /!\
  - ▷ brushless DC-motors
    - · rotor (permanent magnet)
    - · stationary armature (polyphase winding)
    - · electronic (static) commutator from external feed to armature winding phases (my means of motor shaft position)

★ Transmission: power from servomotor $\mapsto$ power to the joints

- (high speed,low torque) $\mapsto$ (low speed,high torque)
  - ▷ sometimes: rotating power $\mapsto$ translational motion
- typical gear ratio $r \in [\frac{1}{200}; \frac{1}{20}]$
  - ⊕ decouples the system and reduces nonlinearity effects
  - ⊖ friction, elasticity, backlash



$$\theta_s = r\theta_m$$

τ_l on joint axis

$\Downarrow$

$r\tau_l$ on actuator axis

★ Sensors

- proprioceptive
  - ▷ position (encoders. . .), velocity (tachometers. . .)
- exteroceptive
  - ▷ force, haptics, vision

★ Control architecture: successive mappings between 4 levels

- task (symbolic)
- actions (sequence of planned paths or way-points, handling collisions, joint limits, redundancy, singularities. . .)
- primitives (sequence of admissible reference motion trajectories and selection of the control algorithm)
- servoing (feedback control)

THE ROBOT CONTROL PROBLEM

★ [Siciliano] How to determine the time history of generalized forces to be developed by the joint actuators so as to guarantee the execution of the commanded task while satisfying given transient and steady-state requirements?

- specified motions in free space (motion control)

  vs

- specified motions and contact forces when EOT is constrained by the environment

★ When addressing motion control, the actuators evolve in the generalized space but the task is generally specified in the operational space

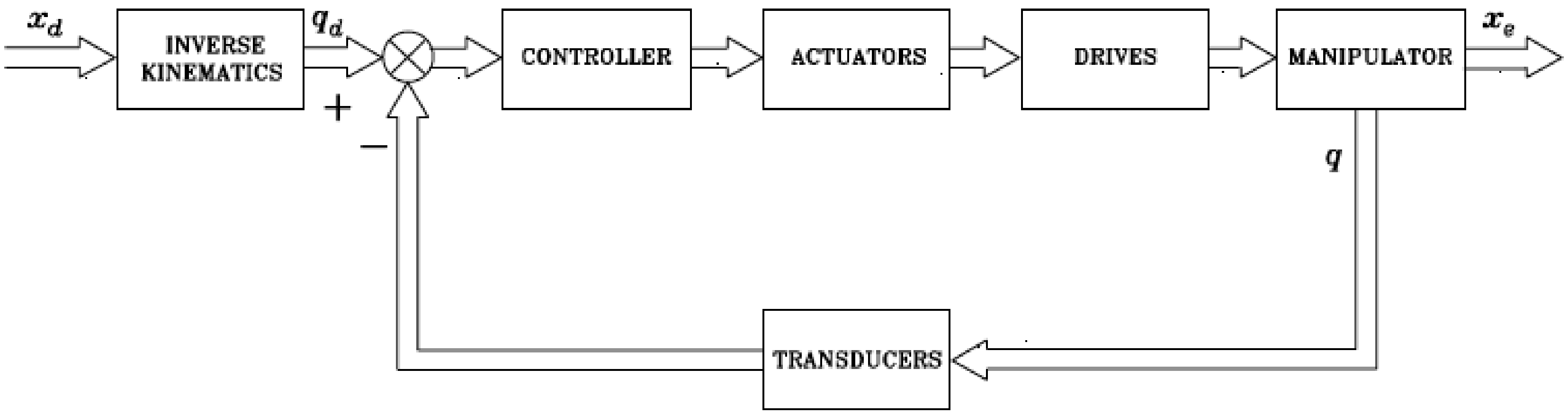
- point-to-point motion of the end effector (pick and place. . .)
  ↪ from the control viewpoint, this is regulation!
  - ▷ discrete set of (two or more) points
  - ▷ no control of the path of the end effector between them
- continous path & time law, a.k.a. trajectory (for welding, painting. . .)
  ↪ from the control viewpoint, this is tracking!

★ The problem is then how to elaborate the command signal which enables to control the robot manipulator (in closed-loop) on this reference. A wide variety of solutions exists, depending on

- the mechanical design of the robot
  - ▷ **P**,**R** joints
  - ▷ with/without reduction gear (cf. special case of direct-drive robots)
- the control input signal
  - ▷ velocity control
  - ▷ torque control
- joint *vs* operational/cartesian/task space control, see below

⋆ Joint space control (*vs* Operational/Cartesian/Task space control)

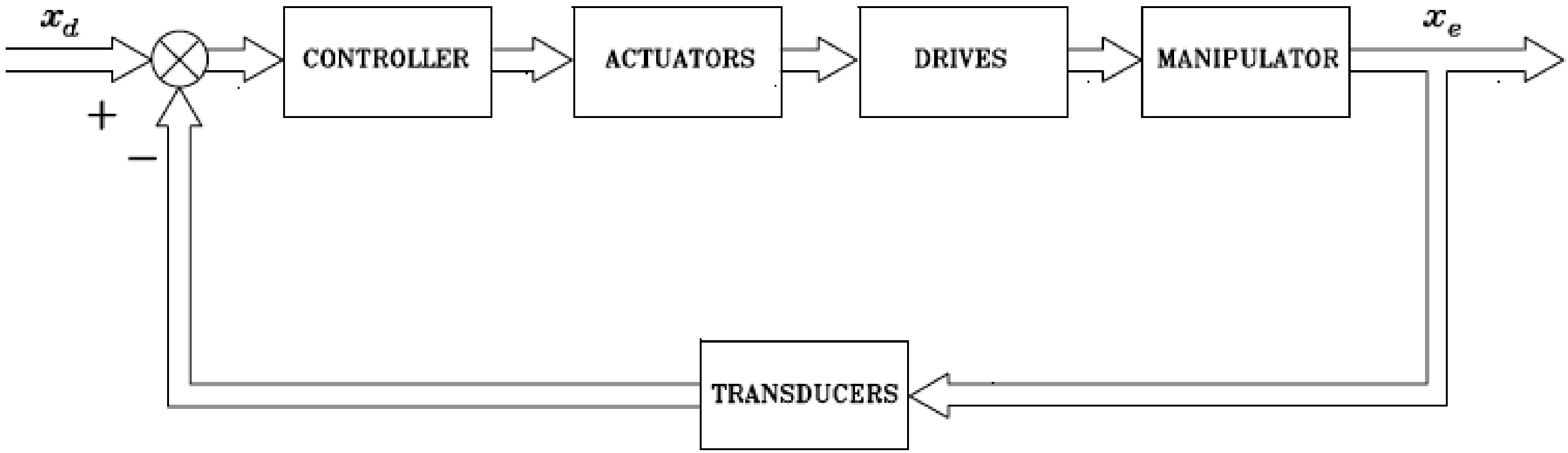
[Figures from Siciliano]



- $x^\star \mapsto q^\star$ by means of inverse kinematics
- /!\ x is controlled in open-loop w.r.t. $x^\star$! $\Rightarrow$ Potential robustness problems w.r.t.
  - ▷ uncertainty of the structure
  - ▷ imprecision on EOT pose relative to a manipulated object

★ Operational/Cartesian/Task space control (*vs* Joint space control)

[Figures from Siciliano]



- Conceptually: acts on operational space variables (in fact, no genuine sensor. . .)
- Inverse kinematics is embedded into the feedback control loop
- Expected better robustness though at the expense of greater complexity

★ Direct-drive robots

- joints driven by high-torque motors
- no friction, elasticity, nor backlash but nonlinearities and couplings between the joints imply distinct control strategies

SCARA and Staubli robots *vs* Adept direct-drive robot

★ These control strategies imply distinct implementations in terms of hardware and software. Always keep in mind that, as is usual in control,

Enhancement of the performances $\Downarrow$ Increase of the modeling "sharpness" and/or of the complexity of the required control scheme

★ A bird's view on robot feedback control general strategies

- joint space control
- operational (cartesian) space control
- control in contact with the environment
  - ▷ stiffness control, hybrid force/position control, hybrid impedance control
- (exteroceptive-) sensor based control
  - ▷ vision-based control, laser-based control...

CONNECTIONS WITH REFERENCE TRAJECTORY DESIGN (M. TAÏX)

★ Reference trajectories are setpoints of feedback controllers

★ Note: these are (at least) C^1 time signals (e.g., for a "trapezoidal velocity profile", the obtained S-shaped reference position trajectory is made of an arc of parabola with positive curvature, followed by a line segment, followed by another arc of parabola with negative curvature) as step-shaped setpoints would make no sense. However, keep in mind that

- when point-to-point motions are considered, the goal of feedback control is to regulate the controlled variable to a constant setpoint (in fact, the final value of the reference signal);
- when continous operational trajectories are considered, the goal of feedback control is to make the controlled variable track the reference signal.

OVERVIEW OF PROMINENT CONTROL APPROACHES

⋆ Decentralized control

- One (linear) feedback controller per axis, synthesized independently
  - ▷ simple structure (little communication between joints), low-cost hardware, scalability (same formulation on each joint)

↪ for point-to-point motion only

⋆ (Centralized) feedforward control

- One (linear decentralized) feedback controller per axis
- One (nonlinear centralized) feedforward controller
  - ▷ more involved, as offline computations of robot dynamics are required

↪ for "easy" trajectory tracking under some hypotheses

⋆ (Centralized) multivariable feedback control

- One (nonlinear centralized) feedback controller, maybe also including feedforward
  - ▷ much more involved, as online computations of robot dynamics are required

↪ for any trajectory tracking, any robot, etc.

BIBLIOGRAPHY

- ★ Robot Manipulator Control - Theory and Practice
  F.L. Lewis, D.M. Dawson, C.T. Abdallah
  Marcel Dekker Eds.

- ★ Robotics - Modelling, Planning and Control
  B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo
  Springer

- ★ Robot Dynamics and Control
  M. Spong, S. Hutchinson, M. Vidyasagar
  Wiley

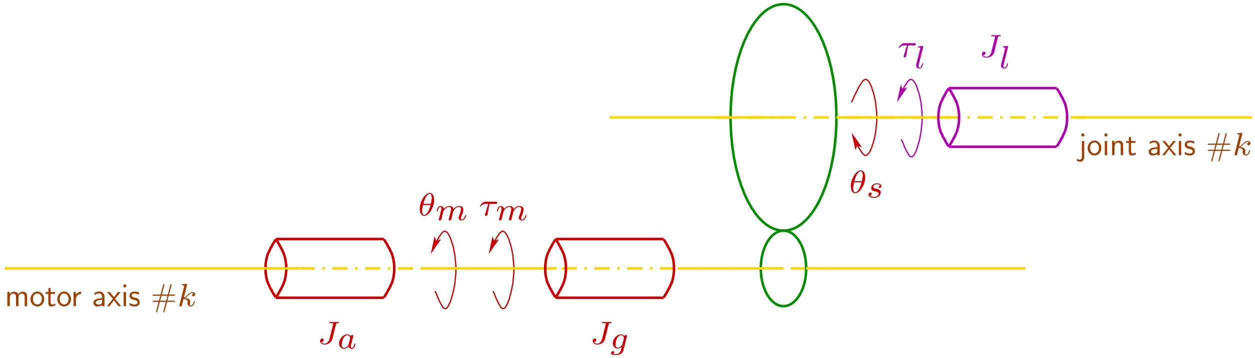
- ★ Handbook of Robotics
  B. Siciliano, O. Khatib Eds.
  Springer

CHAPTER I

TOOLS & MODELS

I.1 MODELING OF A PERMANENT MAGNET DC-MOTOR

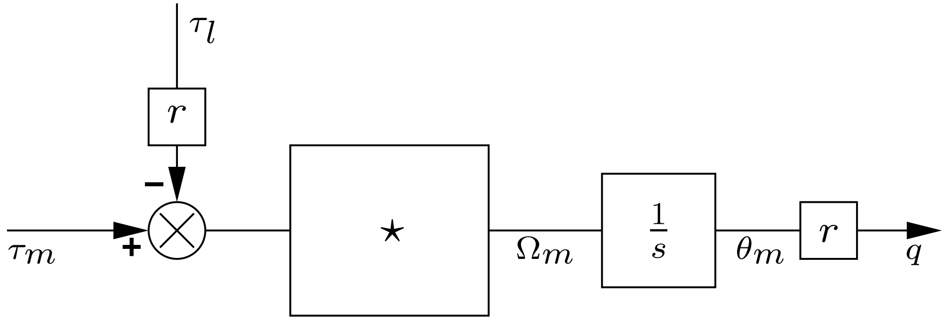
⋆ Schematic diagram (for each k^{th} joint – every variable should be subscripted by k)



- J_a: actuator inertia
- J_g: gear train inertia
- J_l: load inertia
- τ_m: generalized generated effort (force/torque)
- τ_l: generalized effort on the secondary (load) axis
  - nonlinear function of $\mathbf{q}$, $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}}$, etc.
  - equiv. effort brought back on primary axis $= r\tau_l$

★ Expressing causes and effects

- τ_m causes $\theta_m \leftrightarrow (\theta_s = q_k)$
  but τ_l opposes to τ_m via the effort $r\tau_l$ brought back on the primary axis



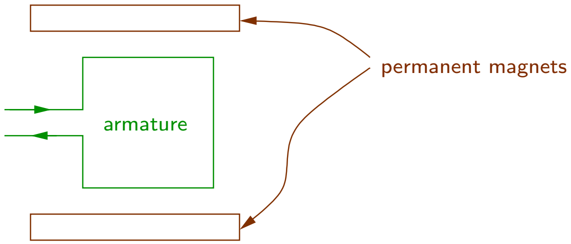
▷ fundamental law of dynamics: $J_m\ddot{\theta}_m = -B_m\dot{\theta}_m + \big(\tau_m - r\tau_l\big)$, with

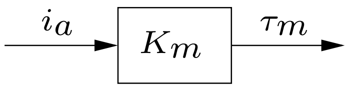
· $J_m \triangleq J_a + J_g$: motor (*i.e.* actuator+gear) inertia

· B_m: coefficient of motor (*i.e.* brushes+gear) friction

$$\Rightarrow \frac{\Omega_m(s)}{\tau_m(s) - r\tau_l(s)} = \frac{1}{B_m + J_m s}$$

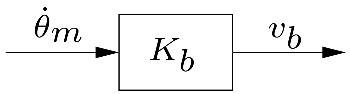
- what causes τ_m?...
  electromagnetic induction: the voltage v applied to the rotor (armature) of the motor, causes the current i_a, which in turn induces τ_m





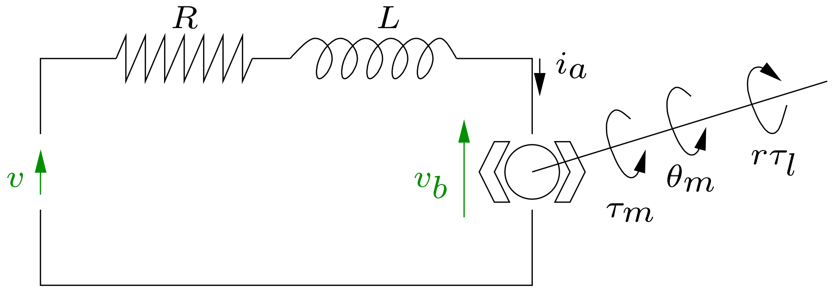
  ▷ $\tau_m = K_m i_a$, K_m torque constant

- but τ_m causes the motion θ_m, which itself gives rise to the back electromotive force (back emf) v_b which acts against v



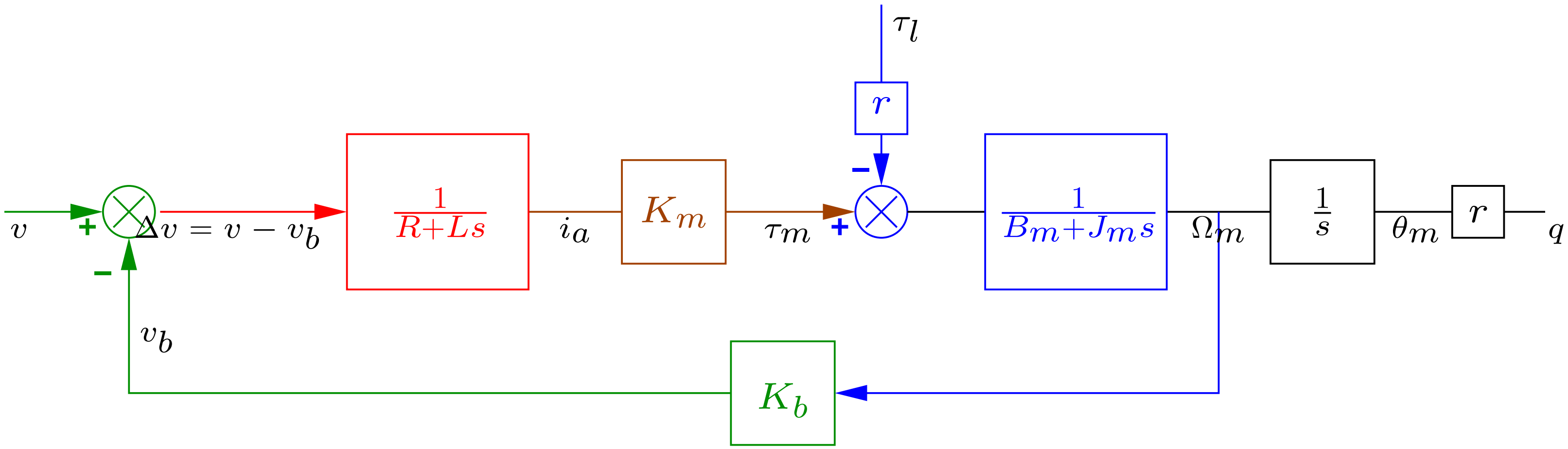
  ▷ $v_b = K_b\dot\theta_m$, K_b back emf constant

- last, $v - v_b$ causes i_a by (R, L being the armature's resistance and inductance)



  ▷ Kirchhoff's circuit laws: $L\dfrac{\mathrm{d}i_a}{\mathrm{d}t} + Ri_a = v - v_b \Rightarrow \dfrac{I_a(s)}{V(s) - V_b(s)} = \dfrac{1}{R + Ls}$

★ Finally, (for each k^{th} joint – every variable should be subscripted by k)



and $\Theta_m(s) = F_{\theta_m v}(s)V(s) + F_{\theta_m \tau_l}(s)\tau_l(s)$, with

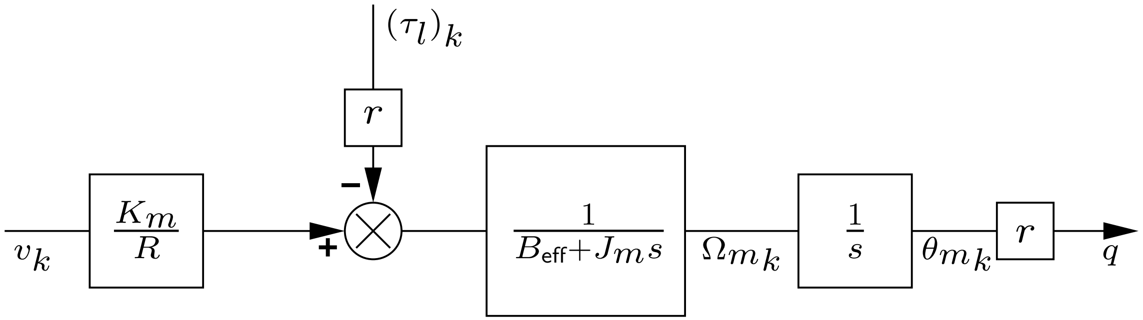
$$F_{\theta_m v}(s) = \frac{K_m}{s(K_b K_m + (R+Ls)(B_m + J_m s))}$$

$$F_{\theta_m \tau_l}(s) = \frac{-r(R+Ls)}{s(K_b K_m + (R+Ls)(B_m + J_m s))}$$

Note the "physical feedback" in the above model!

★ If $\frac{L}{R} << \frac{J_m}{B_m}$, then the above simplifies into
(for each k^{th} joint – every variable should be subscripted by k)

$$\Theta_m(s) = \frac{K_m}{R}\frac{1}{s(B_{\text{eff}} + J_m s)}V(s) - \frac{1}{s(B_{\text{eff}} + J_m s)}r\tau_l(s)$$



with $B_{\text{eff}} = B_m + \frac{K_b K_m}{R}$ effective friction coefficient

★ Notes

- Sometimes, the above model structure is kept for (each axis of) the whole robot (including its actuators), provided that for each k, J_m and $(\tau_l)_k$ are traded for the "effective inertia" J_{eff} and a disturbance d_k constant when robot is at rest
  More on this later...
- The contribution of B_m in B_{eff} is sometimes neglected...

I.2 INTRODUCTION TO DYNAMIC MODELS OF RIGID ROBOT MANIPULATORS

★ Direct and Inverse forms of the Dynamic Model

- Forward Dynamics: from the knowledge of $\tau(.)$, infer the time evolution of $\mathbf{q}(.)$
- Inverses Dynamics: from the knowledge of $\mathbf{q}(.), \dot{\mathbf{q}}(.), \ddot{\mathbf{q}}(.)$, infer τ

★ Expression

$$\underline{\mathbf{D}}(\mathbf{q})\ddot{\mathbf{q}} + \underline{\mathbf{C}}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \tau$$

with

▷ $\underline{\mathbf{D}}(\mathbf{q})$ the inertia matrix of the robot; $\underline{\mathbf{D}}(\mathbf{q}) = \underline{\mathbf{D}}^T(\mathbf{q}) > 0$

▷ $\underline{\mathbf{C}}(\mathbf{q},\dot{\mathbf{q}})$ the matrix of centrifugal and Coriolis efforts;
$\dot{\underline{\mathbf{D}}}(\mathbf{q}) - 2\underline{\mathbf{C}}(\mathbf{q},\dot{\mathbf{q}})$ skew-symmetric;

▷ $\mathbf{g}(\mathbf{q})$ the vector related to gravitation

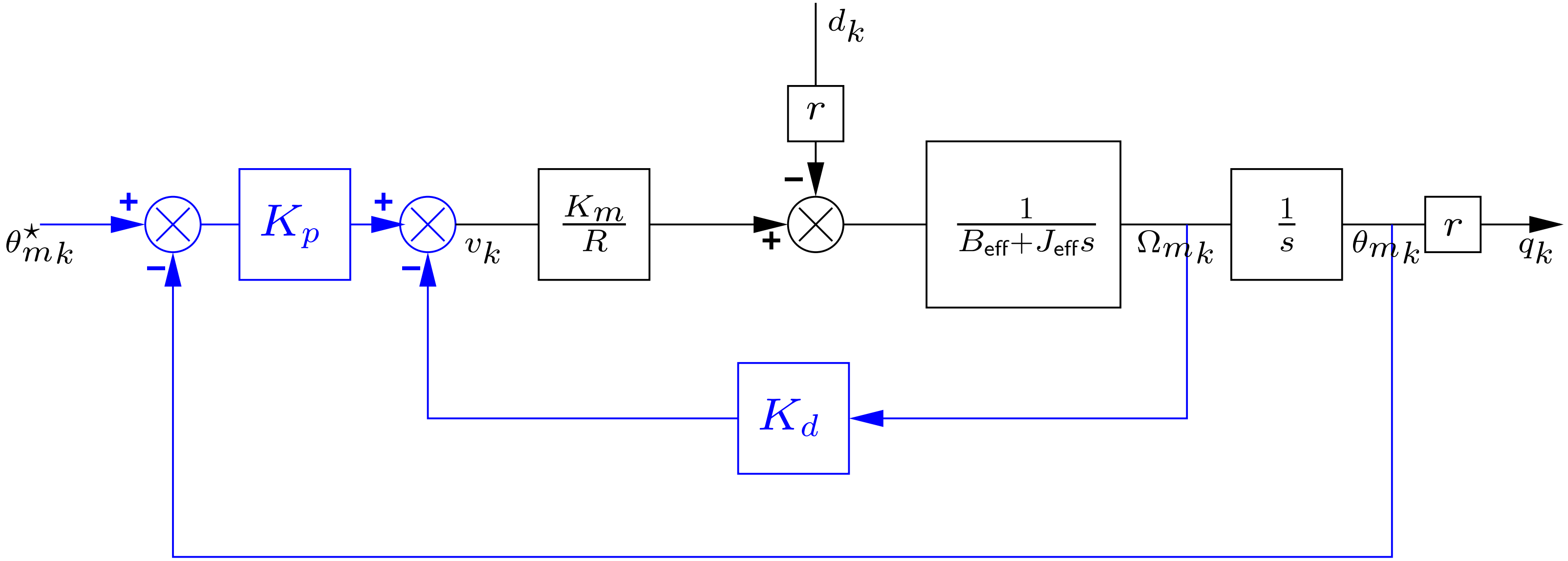
▷ τ the vector of generalized efforts applied on the robot' joints

- Note: possibly hard to determine experimentally, for a robot with non-polyhedral or non-ellipsoidal links. . .

I.3 PROPORTIONAL DERIVATIVE CONTROL

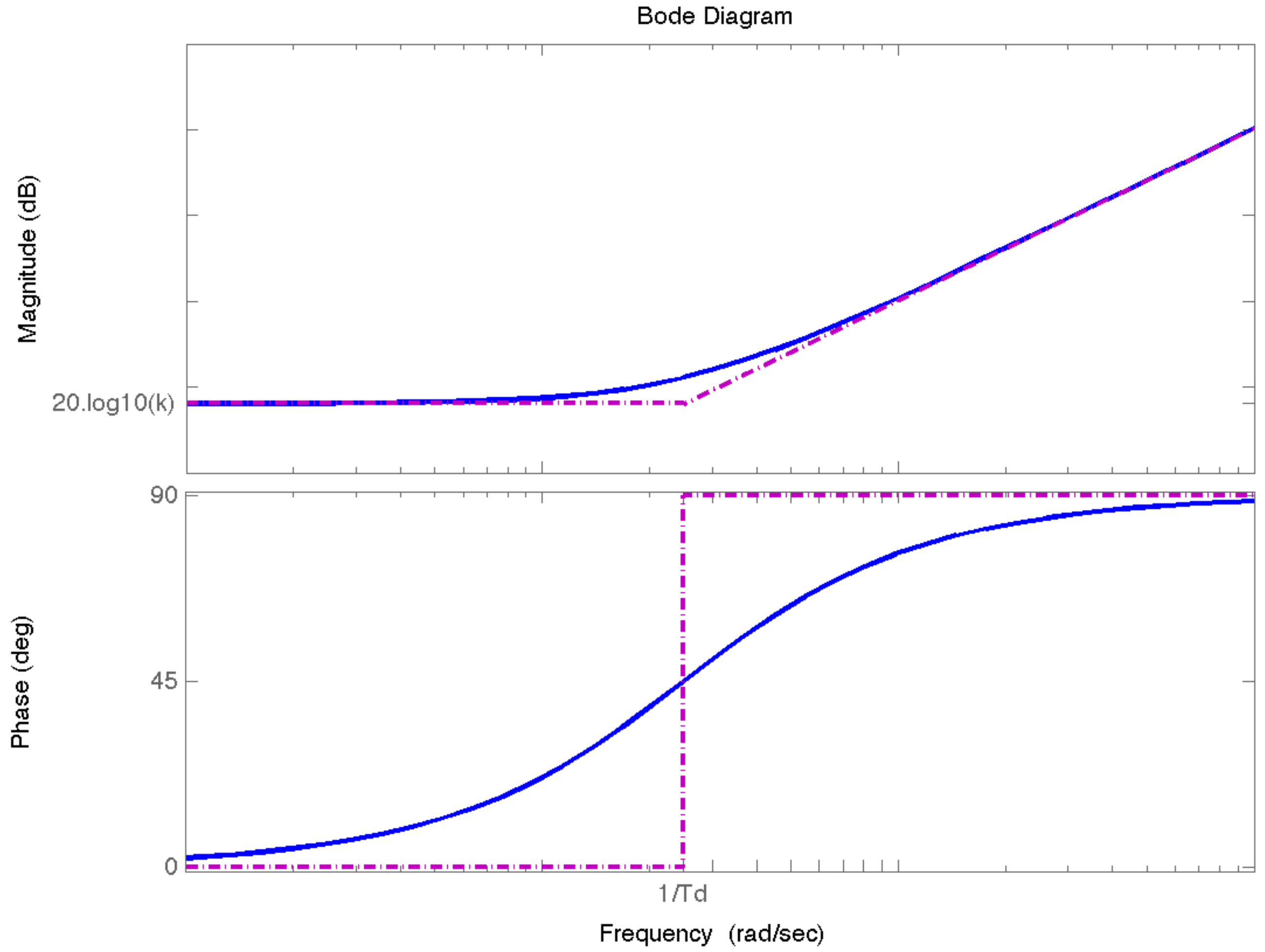
★ Control law for each k^{th} joint: $V(s) = K_p\big(\Theta^\star(s) - \Theta(s)\big) - K_d s\Theta(s)$

$$\Leftrightarrow v(t) = K_p\big(\theta^\star(t) - \theta(t)\big) - K_d\dot{\theta}(t)$$

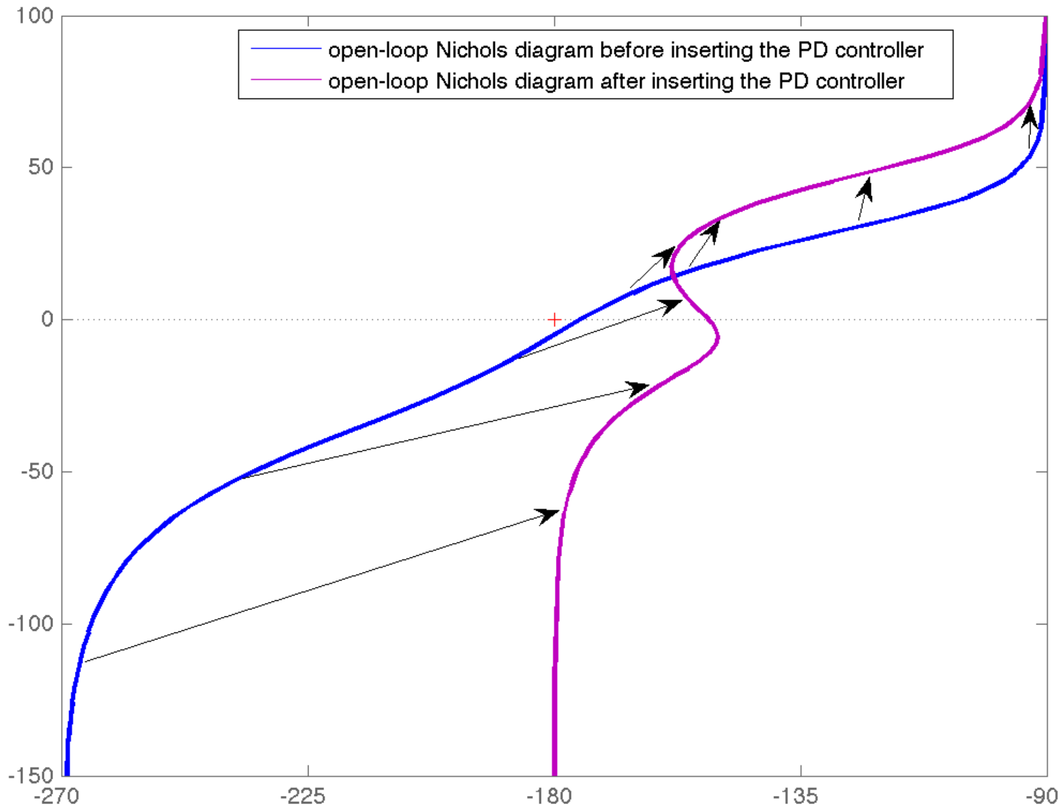


★ A short reminder on PD control and tachometer (velocity) feedback

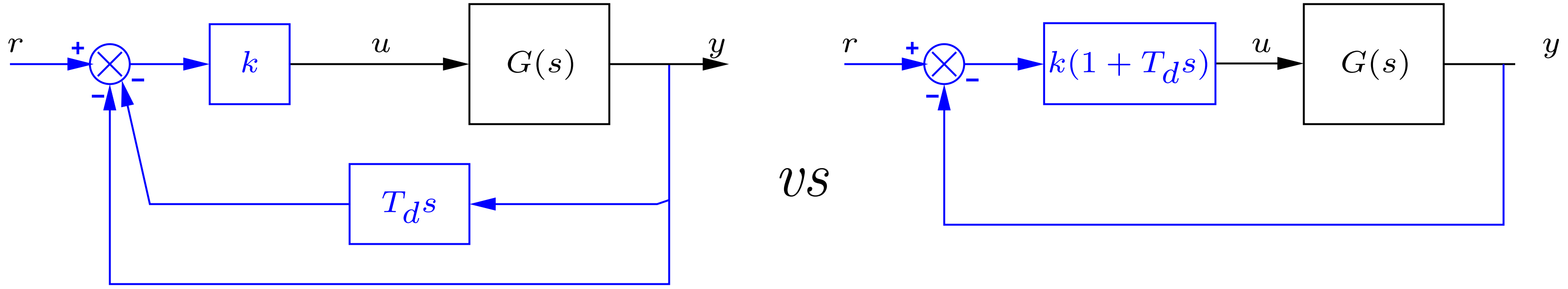
- Bode diagram of $D_{\text{PD}}(s) = k(1 + T_d s)$



- General guidelines to the selection of k, T_d
  - ▷ increase the stability margins of the feedback system, by adding a positive phase to the frequency response of the open-loop transfer function in the vicinity of the critical point



- Potential problems w.r.t. noise
  - ▷ PD $\mapsto$ phase-lead $D_{\text{PhLead}}(s) = \frac{D_{\text{PD}}(s)}{1+\tau s}$
- Why is tachometer (velocity) feedback prefered to PD control?



▷ a practical argument

- · whatever the control scheme, the control signal u entails the derivative of the output y, with y a "smooth" signal
- · in PD control, the derivative of the reference r (where r can be a step, for instance) is also involved in u, while this is not the case for tachometer feedback

▷ a theoretical explanation

- · the two closed-loop transfer functions $F_{\text{TACHO}}(s)$ and $F_{\text{PD}}(s)$ are such that

$$F_{\text{PD}}(s) = F_{\text{TACHO}}(s)(1 + T_d s)$$

…and the adverse effects in the transient of y originate from the zero at $-\frac{1}{T_d}$ in the transfer $F_{\text{PD}}(s)$

★ Application to the robotics problem

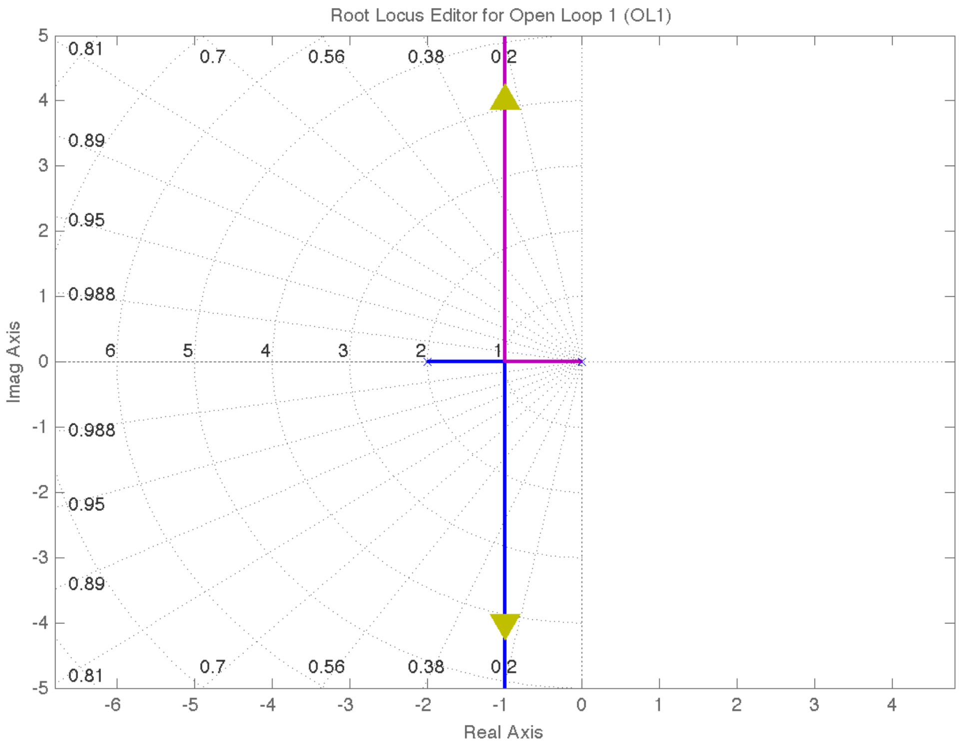
$$\Theta(s) = \frac{K_p \frac{K_m}{R}\,\Theta^\star(s)\ -\ r\,d(s)}{J_{\text{eff}} s^2 + (B_{\text{eff}} + K_d \frac{K_m}{R}) s + K_p \frac{K_m}{R}}$$

- by the Routh criterion, the feedback system is stable if and only if the coefficients of the denominator of the above expression are all positive
  - ▷ the well-known necessary condition of stability is also sufficient here
  - ▷ notice that this denominator is common to the transfers from any external input ($\theta^\star$ or d) to the controlled variable θ

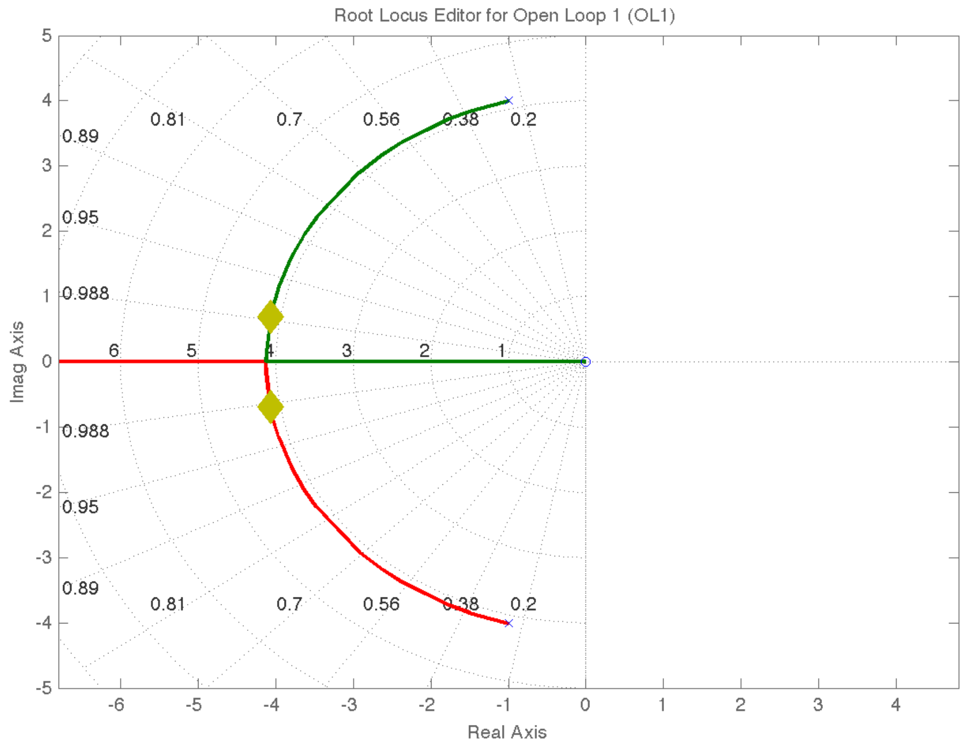
★ Tuning of the controller parameters K_p, K_d

- 2^{nd}-order characteristic polynomial – K_p, K_d are selected so that the feedback system is stable, with a unit damping ratio $(\zeta_{\text{closed-loop}} = 1)$
  - ▷ minimum settling time without overshoot

- qualitative effect of K_p, K_d on the closed-loop poles loci (Evans root loci)



effect of K_p for $K_d = 0$



effect of K_d for K_p fixed

★ Accuracy of the feedback system

- it can be easily shown that if d_k is a constant perturbation, then the position error—*i.e.* $\lim_{t\longrightarrow+\infty}\varepsilon(t)$ for a constant reference $\theta^\star(t)=\theta_0^\star$, with $\varepsilon(t)\triangleq\theta^\star(t)-\theta(t)$—reaches a constant steady state value ε_{pos} , which is all the smaller as K_p grows
  - ▷ prove that, for $d_k=d_0\Gamma(t)$,

$$\varepsilon_{\text{pos}}=\frac{rd_0}{K_p\frac{K_m}{R}}$$

  - ▷ in practice, though d_k is not a constant, the above indeed holds, because d_k amounts to gravitational effects in steady state
- similarly, the velocity error ε_{vel} for $\theta^\star(t)=\dot\theta_1 t\Gamma(t)$ and $d_k=d_0\Gamma(t)$ reads as

$$\varepsilon_{\text{vel}}=\frac{(K_d\frac{K_m}{R}+B_{\text{eff}})\dot\theta_1+rd_0}{K_p\frac{K_m}{R}}$$

  - ▷ left as an exercise ☺

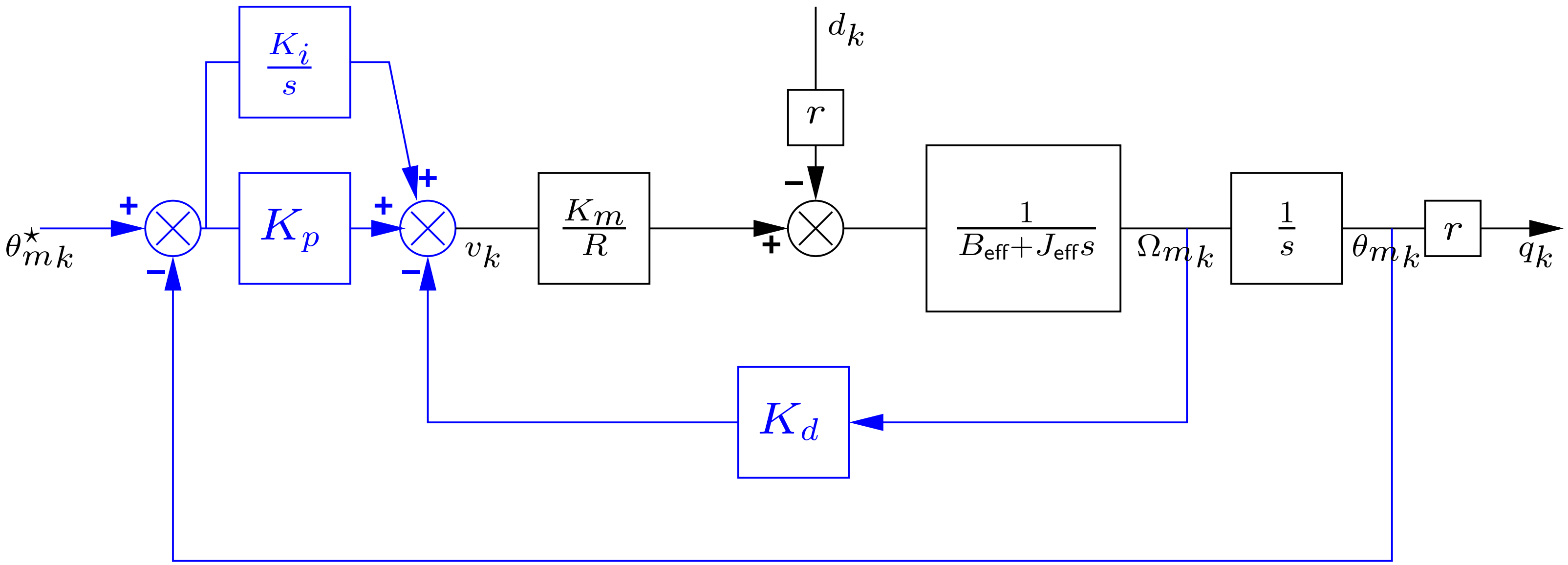
⋆ Limits

- actuators' saturations (mind the overshoot value!)
- this method can/must be extended in order to cope with flexibility in the drive train, if any

I.4 PROPORTIONAL INTEGRAL DERIVATIVE CONTROL

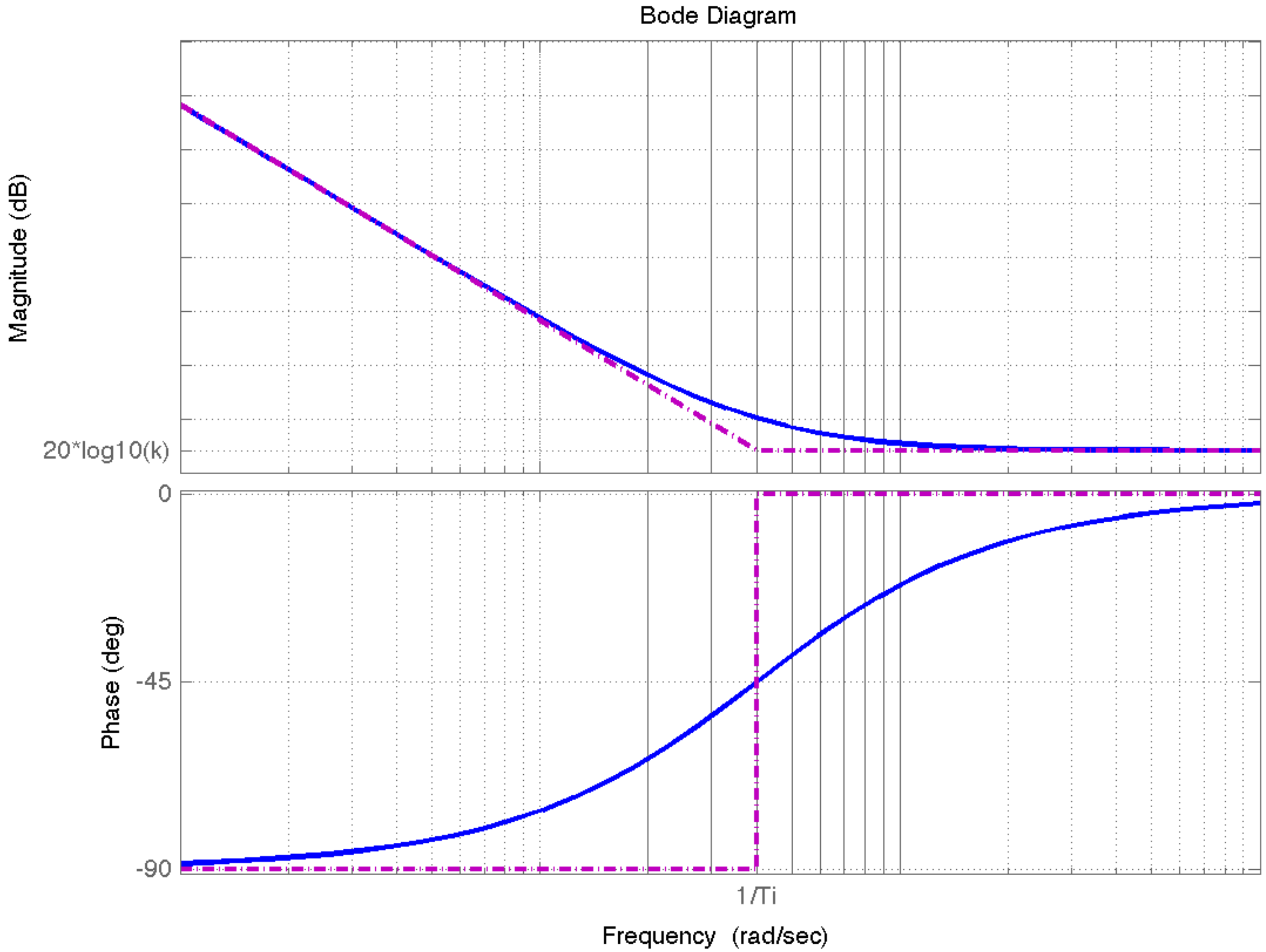
★ Control law for each k^{th} joint: $V(s) = K_p\big(\Theta^\star(s) - \Theta(s)\big) + \frac{K_i}{s}\big(\Theta^\star(s) - \Theta(s)\big) - K_d s\Theta(s)$

$$\Leftrightarrow v(t) = K_p\big(\theta^\star(t) - \theta(t)\big) + K_i \int_{-\infty}^{t} \big(\theta^\star(\tau) - \theta(\tau)\big)\mathrm{d}\tau - K_d\dot{\theta}(t)$$



★ A short reminder on Proportional Integral (PI) control

- Bode diagram of $D(s) = \frac{k}{T_i s}(1 + T_i s)$



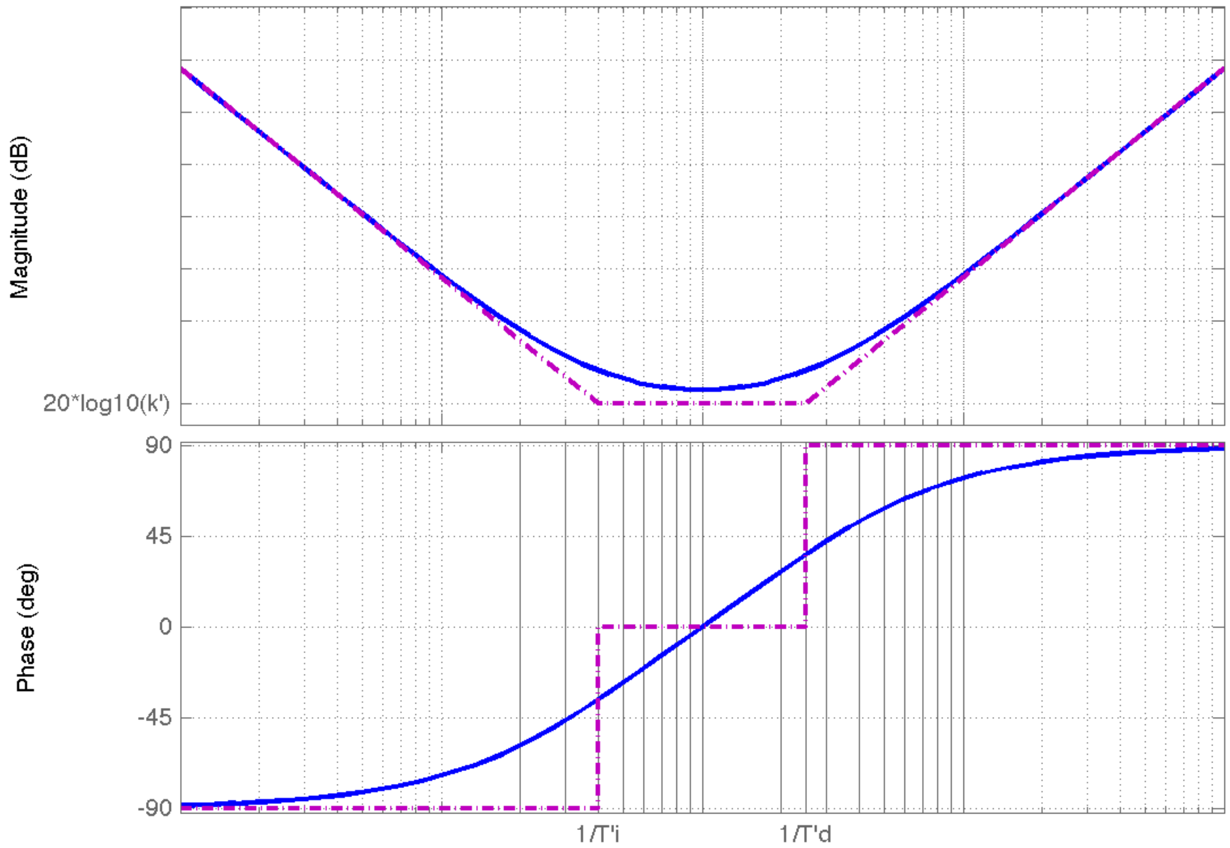
- Effect and tuning of PI control
  - ▷ suppresses the "highest-type" nonzero finite steady-state error
  - ▷ $\frac{1}{T_i}$ must be sufficiently low so that the open-loop frequency response is not modified in the vicinity of the critical point
  - ▷ a famous tuning method: the symmetric optimum rule
  - ▷ mind the actuators' saturations! note that there exists anti-windup systems to prevent the induced instability

★ Proportional Integral Derivative (PID) control

- Most industrial regulators, due to relative simplicity, expert knowledge, obtained robustness
  - ▷ including in robotics servo units
- Three basic PID configurations
  - ▷ standard: $D(s) = k\big(1 + \frac{1}{T_i s} + T_d s\big)$
  - ▷ parallel: $D(s) = K_p + \frac{K_i}{s} + K_d s$
  - ▷ series: $D(s) = \frac{k'}{T_i' s}(1 + T_i' s)(1 + T_d' s)$

$\hookrightarrow$ Remind they are unequivalent, as the series form assumes real zeros

- Bode diagram of a series form



▷ mind potential errors in manual plots of the Bode diagram of the standard and parallel PID configurations!

★ Tuning methods

- analytical, frequency-based, empirical
- benefits and drawbacks of PD & PI (cf. anti-windup systems...)
- see also *PID controllers: Theory, Design and Tuning* - K.J. Åström, T. Hägglund Instrument Society of America, 1995

★ Application to independent joint space control in robotics

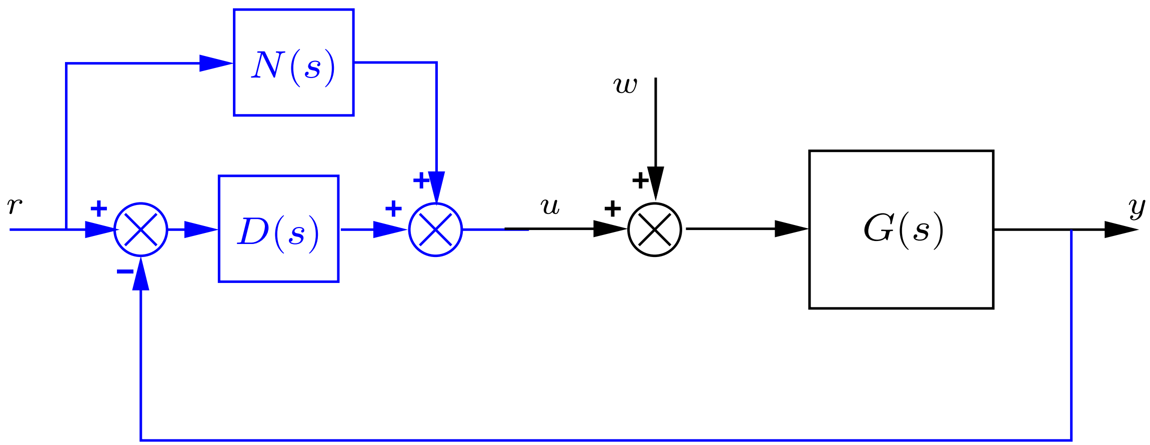
- as soon as the feedback system is stable, the steady-state position error is eliminated
- remind that the PD action is generally implemented as a velocity feedback
- once again: mind the actuators' saturations, and use anti-windup systems to prevent the induced instability

I.5 FEEDFORWARD CONTROL

★ Aims

- error-free tracking of time-varying reference inputs
- rejection of time-varying perturbations

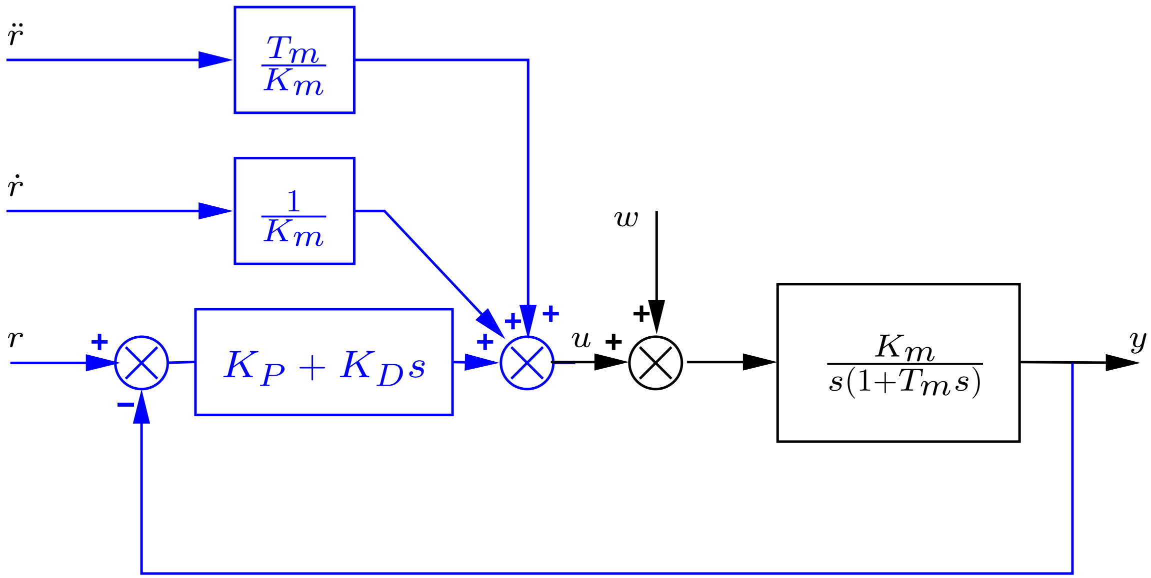
★ An introductory example



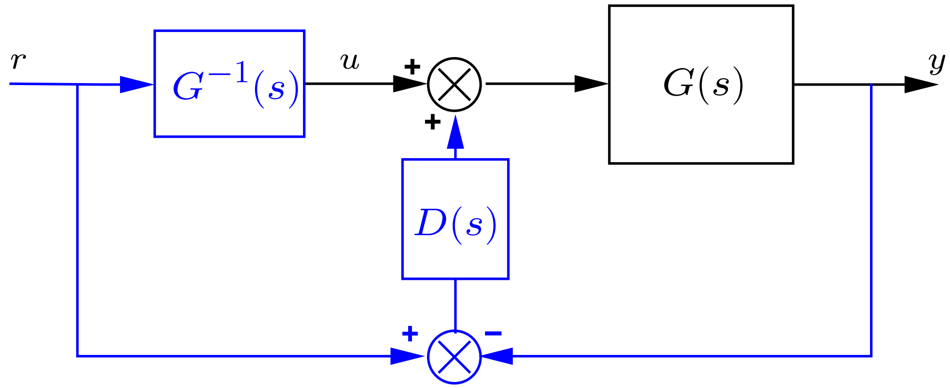
- $Y(s) = \dfrac{\Big(N(s)G(s)+D(s)G(s)\Big)R(s)+G(s)W(s)}{1+D(s)G(s)}$

★ When $w = 0$, if $N(s)G(s) = 1$, then $Y(s) = R(s)$, and the tracking is perfect !

- The controller adds in open-loop—via $N(s)$—what "was missing" in order to perform a perfect tracking
- Constraints
  - ▷ $N(s)$ must be stable $\Rightarrow$ $G(s)$ must be minimum-phase as $N(s) = G(s)^{-1}$
  - ▷ How can $N(s)$ be made causal?
    - · by deriving the reference input $r(t)$ as many times as necessary!
- Example: PD control with feedforward



★ Note: redrawing the feedback+feedforward control



★ Question: how to annihilate a disturbance thanks to feedforward control? thanks to feedback control?

I.6 VELOCITY-CONTROLLED VS TORQUE-CONTROLLED DRIVE

★ Consider again the block diagram of the DC-motor shown on page I-5, e.g., with L neglected therein. For the sake of completeness, this diagram can be completed by inserting

- a transfer function $F_{vv_0}(s) = \frac{V(s)}{V_0(s)} = \frac{K_0}{1+T_0 s}$ representing the input-output relationship of the power amplifier (high gain K_0, $T_0 \in [10^{-5}; 10^{-4}]$s)
- an inner current feedback, defined by $V_0(s) = C_i(s)(V_{\text{in}}(s) - k_i I_a(s)$, with v_{in} the outer reference signal

★ Consequently, for $\tau_l = 0$,

- if $k_i = 0$ and $C_i(0) = 1$, then $\omega_m \propto K_0 v_{\text{in}}$ in steady-state, and the drive system is said velocity-controlled
- if k_i is high, then $\tau_m \propto \frac{1}{k_i}(v_{\text{in}} - \frac{K_b}{K_0}\omega_m)$ so that $\tau_m \propto \frac{1}{k_i}(v_{\text{in}})$ in view of the high value of K_0, and the drive system is said torque-controlled

$\hookrightarrow$ left as an exercise ☺

★ Furthermore,

- for a velocity-controlled drive, a nonzero disturbance τ_l is rejected with a better (lower) gain and a better (shorter) time response than in the case when the drive is torque-controlled $\Rightarrow$ velocity-controlled drives will be used when high disturbance rejection is needed (e.g., decentralized independent joint control)

$\hookrightarrow$ left as an exercise ☺

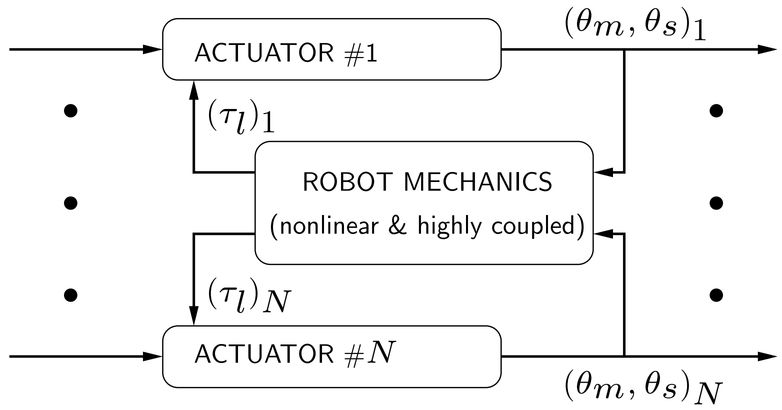
- torque-controlled drives will be better suited for centralized control schemes

CHAPTER II

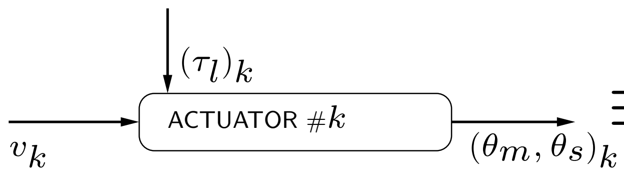
JOINT SPACE CONTROL

II.1 FULL MODELS FOR CONTROL

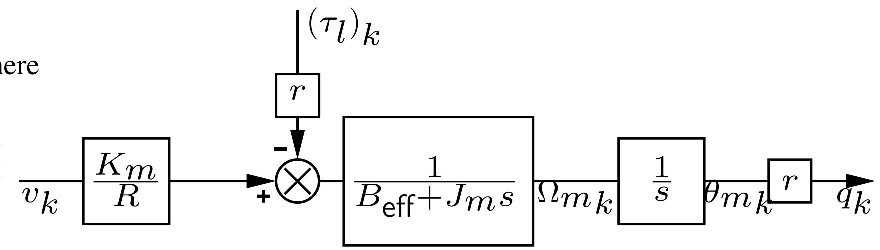
⋆ Model as per Lewis/Spong (other options do exist!)



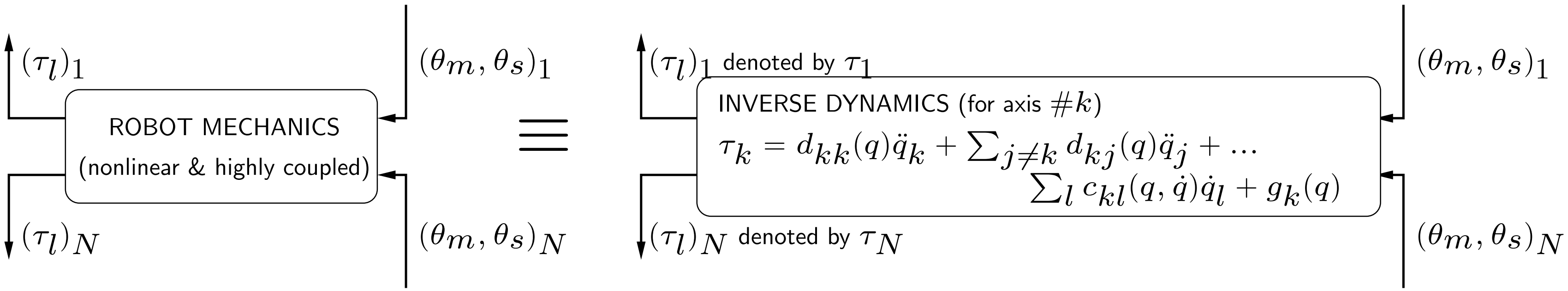
where



$\equiv$



and



★ Model as per Lewis/Spong (cont'd)

- So, what are the difficulties at this stage, and what can be done in order to synthesize a “simple” feedback control law?
  - ▷ as aforementioned, the generalized efforts $\left(r\tau_l\right)_k$ brought back on each k^{th} joint are coupled nonlinear functions of the robot configuration/velocities/accelerations $\mathbf{q}/\dot{\mathbf{q}}/\ddot{\mathbf{q}}$ (more on this later...)
  - ▷ so, $\left(r\tau_l\right)_k$ can be considered as mere perturbations under very specific hypotheses
- Note: the whole system can be modeled as

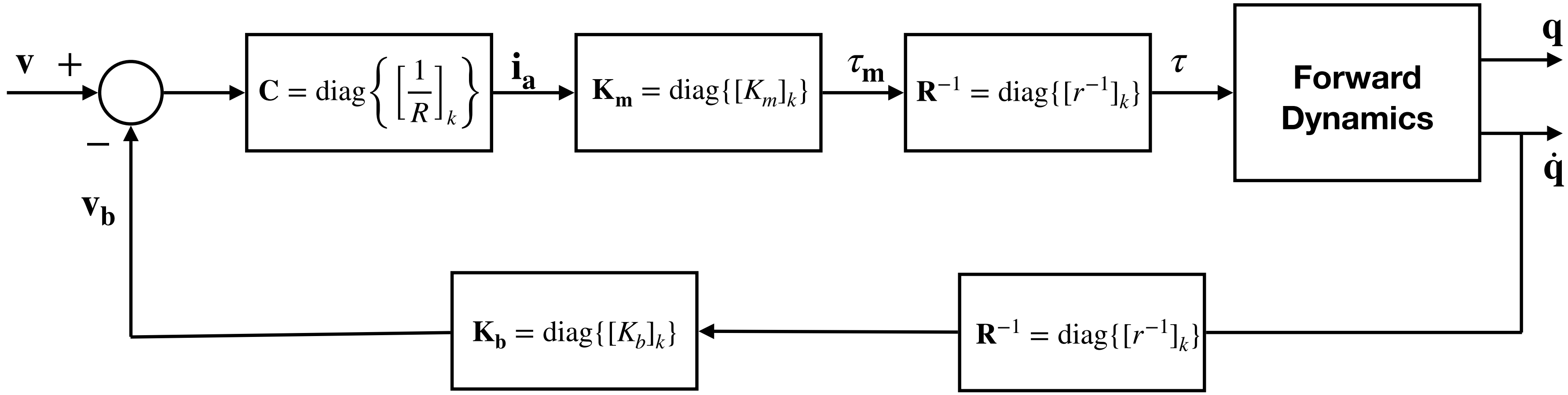
$$\underline{\mathbf{D}}_0(\mathbf{q})\ddot{\mathbf{q}} + \underline{\mathbf{C}}_0(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}_0(\mathbf{q}) = \mathbf{u}, \text{ with} \tag{1}$$

$$\underline{\mathbf{D}}_0(\mathbf{q}) = D(\mathbf{q}) + \operatorname{diag}\left(\left\{\frac{J_m}{r^2}\right\}_k\right), \tag{2}$$

$$\underline{\mathbf{C}}_0(\mathbf{q},\dot{\mathbf{q}}) = C(\mathbf{q},\dot{\mathbf{q}}) + \operatorname{diag}\left(\left\{\frac{B_{\text{eff}}}{r^2}\right\}_k\right), \tag{3}$$

$$\mathbf{u} = \operatorname{diag}\left(\left\{\frac{\frac{K_m}{R}}{r}\right\}_k\right)\mathbf{v}, \text{ where } \mathbf{v} = \begin{pmatrix} v_1 \\ \vdots \\ v_N \end{pmatrix}. \tag{4}$$

★ Model as per Siciliano *et al.*



$$\left.\begin{aligned}\tau &= \mathbf{R}^{-1}\mathbf{K_m}\mathbf{C}\Big(\mathbf{v} - \mathbf{K_b}\mathbf{R}^{-1}\dot{\mathbf{q}}\Big) \\ \tau &= \mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q})\end{aligned}\right\} \Rightarrow \mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \Big(\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{K_m}\mathbf{K_b}\mathbf{C}\mathbf{R}^{-2}\Big)\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \mathbf{K_m}\mathbf{C}\mathbf{R}^{-1}\mathbf{v}$$

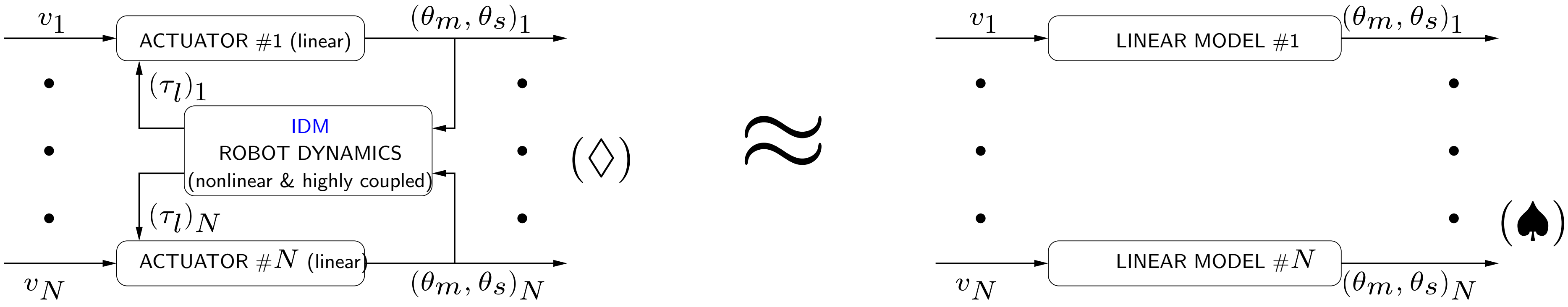
- Rather suited to centralized control, but the decentralized control approach presented in the sequel can be straightly extended to this model

II.2 DECENTRALIZED CONTROL

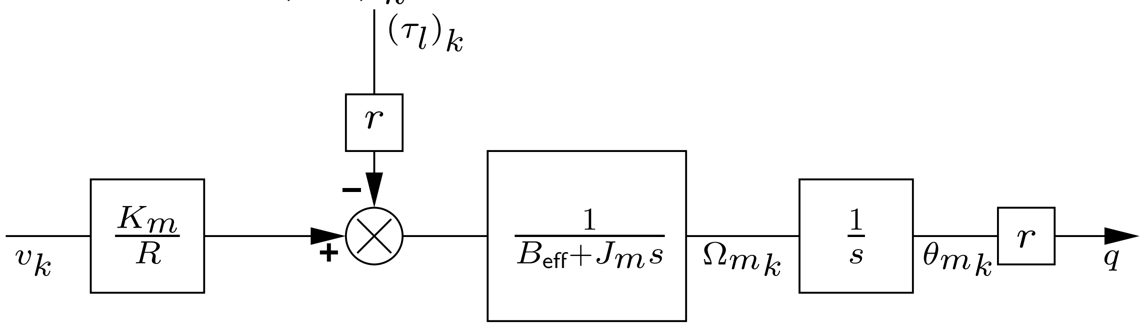
★ Aims

- Design a feedback controller on each axis independently from the other axes
- Work on simplified (linear) models
- Implement simplified (linear) control laws: PD, PID

★ How? Approximate the full nonlinear coupled model ($\Diamond$) of the N-joint robot and its actuators given below by N independent models $(\spadesuit)_k$, $k = 1, \ldots, N$

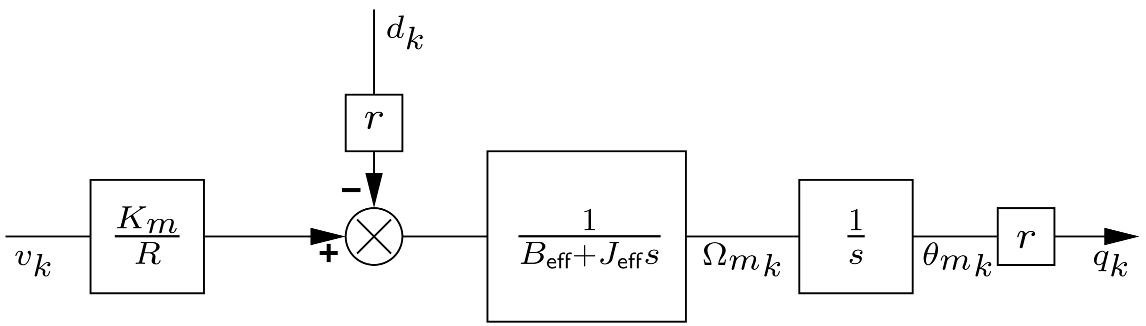


- The following schematic diagram was introduced for each k^{th} actuator, where $r\tau_k$ stands for the generalized effort $\left(r\tau_l\right)_k$ brought back by the robot dynamics



- We will trade off the full model $(\Diamond)$ by N simple parallel (noninteractive) linear diagrams$(\spadesuit)_k$, where each one writes as

(for each k^{th} joint – every variable should be subscripted by k)



$(\spadesuit)_k$

★ How? The data of the Inverse Dynamic Model of the robot enables us to express J_{eff} and d_k for each k^{th} axis, by combining the two following equations:

$$\tau_k = d_{kk}(\mathbf{q})\ddot{q}_k + \underbrace{\sum_{j\neq k} d_{kj}(\mathbf{q})\ddot{q}_j + \sum_{l} c_{kl}(\mathbf{q},\dot{\mathbf{q}})\dot{q}_l + g_k(\mathbf{q})}_{\text{henceforth denoted by } d_k(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})}$$

$$J_m\ddot{\theta}_{m_k} + B_{\text{eff}}\dot{\theta}_{m_k} = \frac{K_m}{R}v_k - r\tau_k$$

- After some computations, one gets:
  (for each k^{th} joint – every variable should be subscripted by k)

$$\big(J_m + r^2 d_{kk}(\mathbf{q})\big)\ddot{\theta}_{m_k} + B_{\text{eff}}\dot{\theta}_{m_k} = \frac{K_m}{R}v_k - r d_k(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})$$

  so that $\{(\spadesuit)_k\}$ can be obtained as soon as
  - ▷ the constant J_{eff} is set to the constant part, average value, or worst-case value of $J_m + r^2 d_{kk}(\mathbf{q})$;
  - ▷ the dependence of $d_k(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})$ on $\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}}$ is deliberately lost, so that $d_k(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})$ is replaced by a (exogeneous) disturbance signal $d_k(t)$ which is assumed constant when the robot does not move (due to gravitation)

★ What have we done?

- For each joint index k, $\left(\tau_l\right)_k$ has been written as the sum of two contributions, one of which can (in first approximation) be considered as a true exogeneous signal, *e.g.* $\left(\tau_l\right)_k$ can be splitted into
  - ▷ the inertial efforts due to the body which immediately follows the articulation $\#k$: $\left(J_m\right)_k \mapsto$ "effective inertia" $\left(J_{\text{eff}}\right)_k$
  - ▷ the inertial efforts due to the other links, plus the other efforts (centrifugal, gravitational, Coriolis, etc.): $\left(\tau_l\right)_k \mapsto \left(d\right)_k$
- Notice that many other ways to get a simplified open-loop model could be envisaged!

★ Controllers

- PD control
  - ▷ Can stabilize the feedback system
  - ▷ Nonzero position error due to constant d_k in steady-state
- PID control
  - ▷ Can stabilize the feedback system
  - ▷ Zero position error in spite of (asymptotically constant) d_k

★ Validity of the approximation

- point-to-point tasks
- slow motions
- robots equipped with important gear reduction, without flexibility in the joints

$\hookrightarrow$ but mind potential nonlinear effects! friction, backlash, etc.

II.3 CENTRALIZED FEEDFORWARD CONTROL

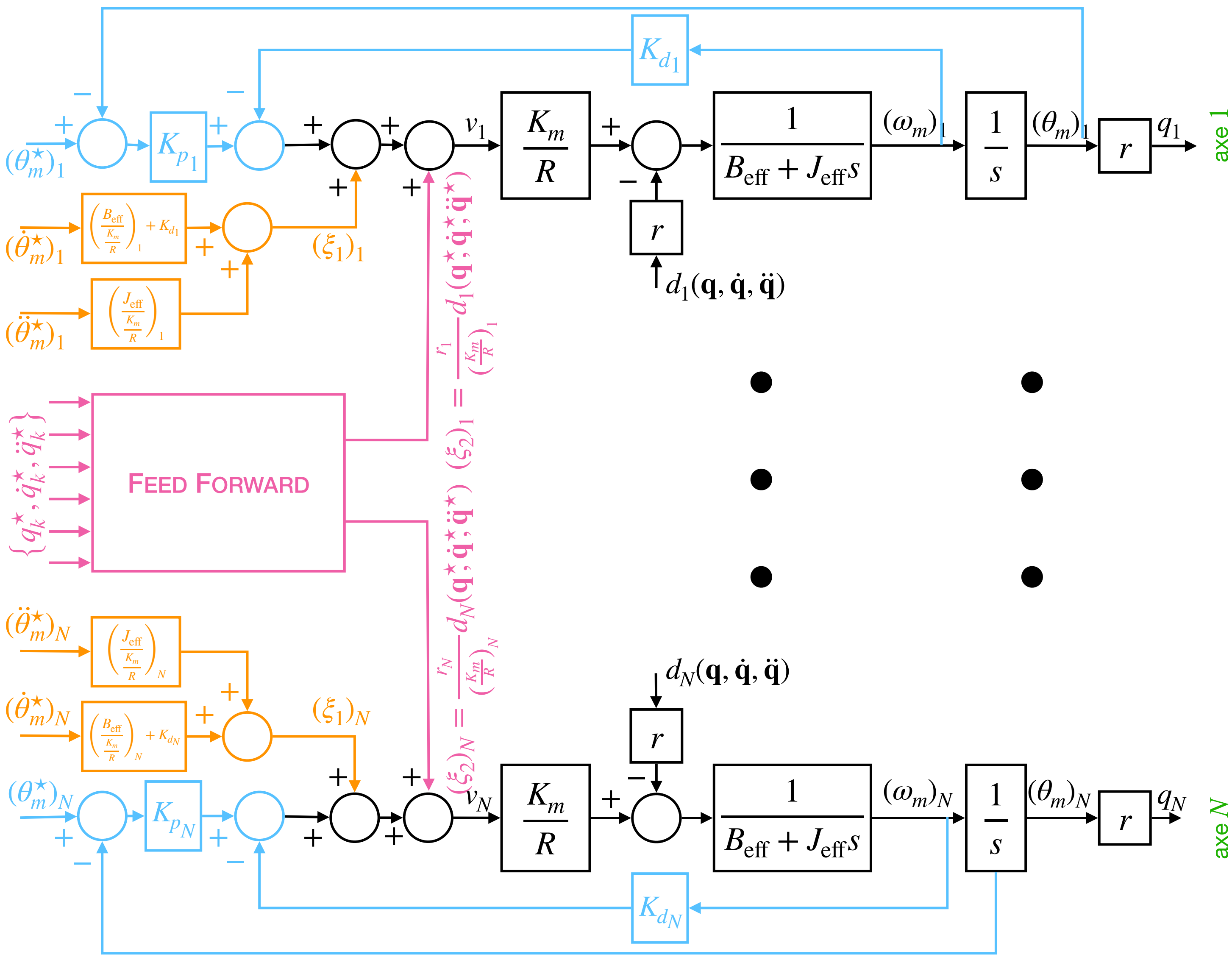
★ Aims

- Follow time-varying reference signals
- Take into account the very nature of $d_k(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ so as to improve closed-loop performances.

★ The Feedforward Computed Torque Method

- The basic idea is to superimpose on the output u_k from a classical feedback controller—*e.g.* PD, PID,...—dedicated to each k^{th} axis, a signal $\xi_k = (\xi_1)_k + (\xi_2)_k$ computed in open-loop where
  - ▷ $(\xi_1)_k$ would contribute to a perfect tracking if d_k were set to 0, following the guidelines in §I.5
  - ▷ $(\xi_2)_k$ nearly cancels the effect of $rd_k(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$

- Schematic diagram



- Let us focus on $(\xi_2)_k$ (because $(\xi_1)_k$ can be defined very easily). It can be set to

$$(\xi_2)_k = \frac{1}{\frac{K_m}{R}} r\, d_k(\mathbf{q}^\star, \dot{\mathbf{q}}^\star, \ddot{\mathbf{q}}^\star)$$
$$= \frac{1}{\frac{K_m}{R}} r \left\{ \sum_{j \neq k} d_{kj}(\mathbf{q}^\star)\ddot{q}_j^\star + \sum_l c_{kl}(\mathbf{q}^\star, \dot{\mathbf{q}}^\star)\dot{q}_l^\star + g_k(\mathbf{q}^\star) \right\}$$

  ▷ the reason of this definition of $(\xi_2)_k$ comes from the fact that $q_k = r\theta_{mk}$ is assumed to be close to $q_k^\star = r\theta_{mk}^\star$ during the task...

★ Pros, Cons, and Induced requirements

⊕ the stability of the whole feedback system stays unmodified

⊖ of course, $rd_k(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \neq \frac{K_m}{R}\xi_{2k}\ldots$, yet this leads to nice results

- Mind real-time constraints!
  ▷ an important research effort used to focus on the recursive computation of the dynamic model (Newton-Euler)
  ▷ $\mathbf{q}^\star, \dot{\mathbf{q}}^\star, \ddot{\mathbf{q}}^\star$ must be computed offline, and saved into a look-up-table

II.4 CENTRALIZED FEEDBACK CONTROL

★ Introduction

- Use of modern Automatic Control theories/techniques
  - ▷ powerful conclusions
  - ▷ sound theoretical bases
- No restrictive assumption, *e.g.* concerning the use of gear train

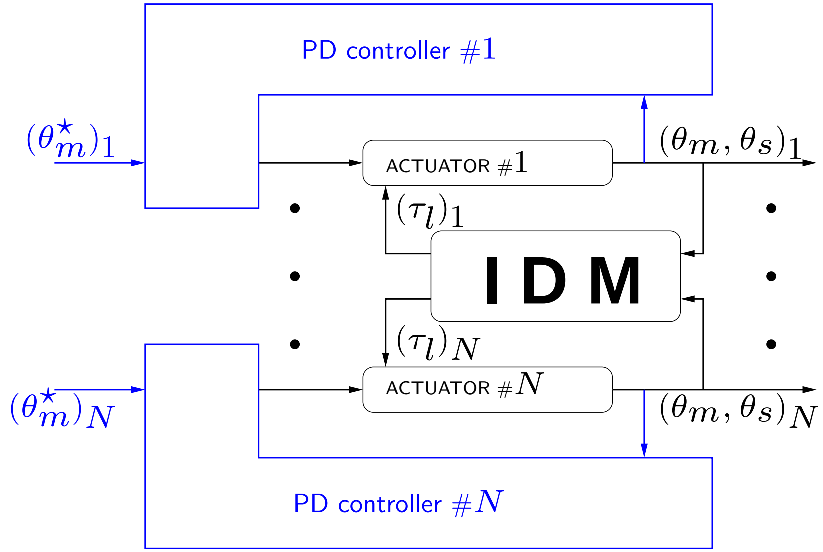
$\hookrightarrow$ possibility to consider mechanical structures enabling very high performances

II.4.1 PD control

★ A while ago, the analysis and synthesis of a PD control were studied, assuming that the robot could be handled through a very simplified model, in which some efforts brought back on the joint axes could be captured via "effective inertias" $(J_{\text{eff}})_k$. Nevertheless, the following question remains:

.../...

- What happens if a PD controller is connected to the genuine nonlinear model of the whole robot with its actuators?



★ It can be shown that

- the PD controller does stabilize the nonlinear feedback model, but $\varepsilon(t) \triangleq \theta^\star(t) - \theta(t)$ reaches a constant steady state value ε_{pos}, which is all the smaller as K_p grows
- ε_{pos} can be made lower by inserting a feedback term function of $\mathbf{q}$ which compensates for gravitational effects

II.4.2 Feedback Linearization (Feedback Computed Torque)

★ Introduction

- What about the connection of the system $\ddot{y}(t) = -\dot{y}(t) + a\sqrt{y(t)} + u(t)$ with the feedback controller $u(t) = -a\sqrt{y(t)} - y(t)$?

★ Assume that we start from the following model

$$\underline{\mathbf{D}}_0(\mathbf{q})\ddot{\mathbf{q}} + \underline{\mathbf{C}}_0(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}_0(\mathbf{q}) = \mathbf{u}$$

The feedback controller is based on two nested loops

- Inner loop: nonlinear controller which decouples and linearizes the robot dynamics
  - ▷ $\mathbf{u} = \underline{\mathbf{D}}_0(\mathbf{q})v + \underline{\mathbf{C}}_0(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}_0(\mathbf{q})$
  - ▷ the inner closed-loop system becomes a set of N parallel independent double integrators, described by $\ddot{q}_k = v_k$
- Outer loop: a set of non-interactive classical linear controllers, whose role is to stabilize the above double integrators
  - ▷ these controllers are theoretically fully independent of the robot dynamics
  - ▷ note that they can include feedforward as well!

★ Application and Consequent schematic diagram (sometimes a.k.a. "Computed Torque Method"...)

- The whole feedback system ($\Diamond$) can be modeled as

$$\underline{\mathbf{D}}_0(\mathbf{q})\ddot{\mathbf{q}} + \underline{\mathbf{C}}_0(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}_0(\mathbf{q}) = \mathbf{u}, \text{ with} \tag{5}$$

$$\underline{\mathbf{D}}_0(\mathbf{q}) = D(\mathbf{q}) + \mathrm{diag}\left(\left\{\frac{J_m}{r^2}\right\}_k\right), \tag{6}$$

$$\underline{\mathbf{C}}_0(\mathbf{q},\dot{\mathbf{q}}) = C(\mathbf{q},\dot{\mathbf{q}}) + \mathrm{diag}\left(\left\{\frac{B_{\mathrm{eff}}}{r^2}\right\}_k\right), \tag{7}$$

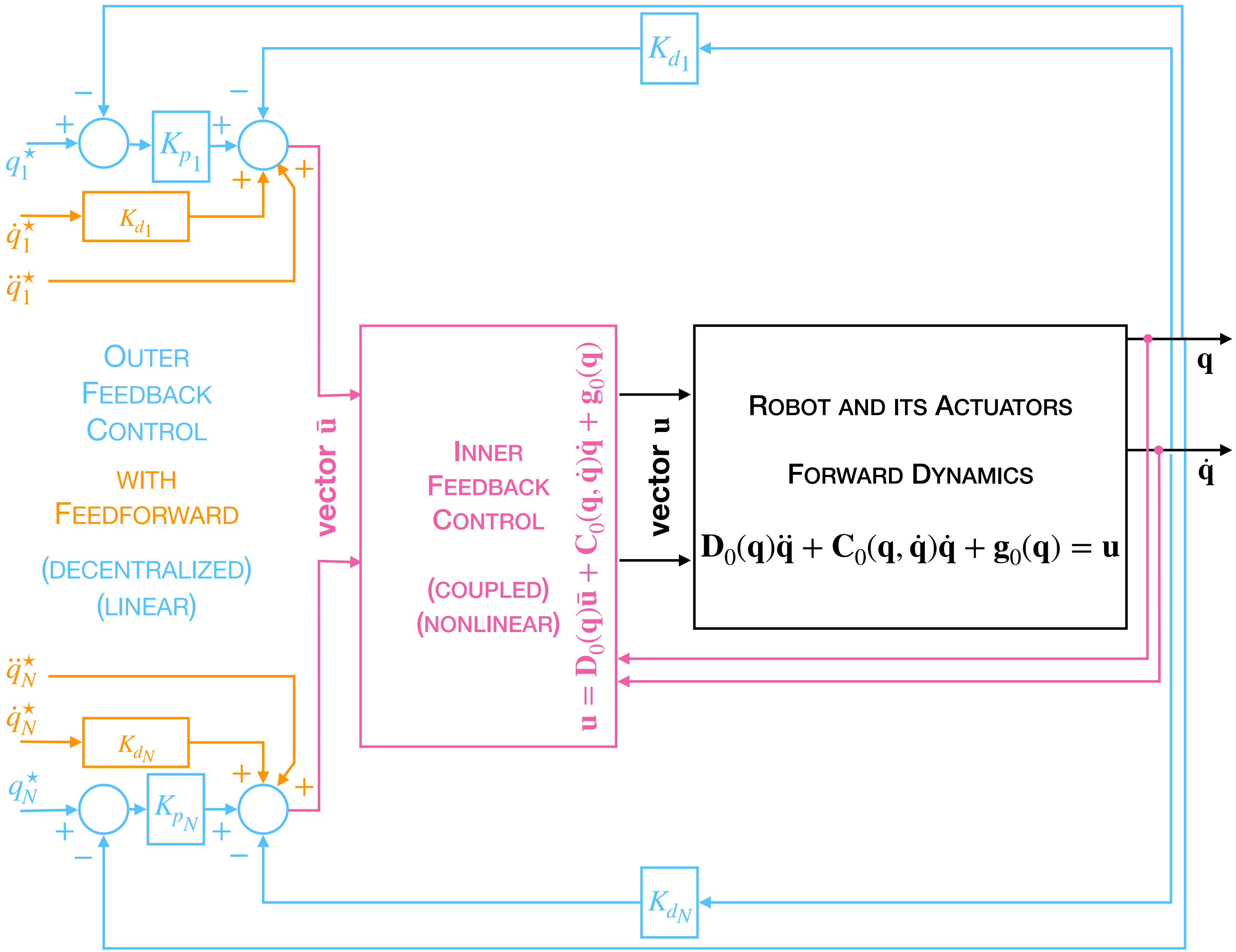
$$\mathbf{u} = \mathrm{diag}\left(\left\{\frac{\frac{K_m}{R}}{r}\right\}_k\right)\mathbf{v}, \text{ where } \mathbf{v} = \begin{pmatrix} v_1 \\ \vdots \\ v_N \end{pmatrix}. \tag{8}$$

- The controller writes as

$$\mathbf{u} = \underline{\mathbf{D}}_0(\mathbf{q})\bar{\mathbf{u}} + \underline{\mathbf{C}}_0(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}_0(\mathbf{q}) \tag{9}$$

$$\text{with } \bar{\mathbf{u}} = \ddot{\mathbf{q}}^\star + K_D(\dot{\mathbf{q}}^\star - \dot{\mathbf{q}}) + K_P(\mathbf{q}^\star - \mathbf{q}). \tag{10}$$

★ In short,


K_{d_1}
$q_1^\star$
K_{p_1}
$\dot{q}_1^\star$
K_{d_1}
$\ddot{q}_1^\star$
OUTER FEEDBACK CONTROL
WITH FEEDFORWARD
(DECENTRALIZED)
(LINEAR)
vector $\bar{\mathbf{u}}$
INNER FEEDBACK CONTROL
(COUPLED)
(NONLINEAR)
$\mathbf{u} = \mathbf{D}_0(\mathbf{q})\bar{\mathbf{u}} + \mathbf{C}_0(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}_0(\mathbf{q})$
vector $\mathbf{u}$
ROBOT AND ITS ACTUATORS
FORWARD DYNAMICS
$\mathbf{D}_0(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}_0(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}_0(\mathbf{q}) = \mathbf{u}$
$\mathbf{q}$
$\dot{\mathbf{q}}$
$\ddot{q}_N^\star$
$\dot{q}_N^\star$
K_{d_N}
$q_N^\star$
K_{p_N}
K_{d_N}

★ Pros and Cons

- $\oplus$ Very attractive strategy!
- $\oplus$ Decoupling and linearization
- $\ominus$ Requires an excellent model of the robot - Otherwise, adaptive or robust version
- $\ominus$ Requires a powerful hardware (because of many nonlinear computations)
- $\ominus$ The robot's DDM need to be simplified in order to synthesize the inner controller $\Rightarrow$ some important properties of the feedback system may be lost, *e.g.* stability, decoupling, etc.
  - ▷ a solution may consist in using linear robust control techniques to design the outer controller, the difference between the actual and ideal inner feedback system being embedded into an uncertainty which worst-case realization is considered