Joint-Space Control

Optimization-based Robot Control

Andrea Del Prete

University of Trento

Introduction

This document explains the control framework known as Task-Space Inverse Dynamics (TSID).

TSID is a popular control framework for legged robots.

Introduction

This document explains the control framework known as Task-Space Inverse Dynamics (TSID).

TSID is a popular control framework for legged robots.

It all started in 1987 with this paper by *Oussama Khatib*: "A unified approach for motion and force control of robot manipulators: The operational space formulation" [5]

Introduction

This document explains the control framework known as Task-Space Inverse Dynamics (TSID).

TSID is a popular control framework for legged robots.

It all started in 1987 with this paper by *Oussama Khatib*: "A unified approach for motion and force control of robot manipulators: The operational space formulation" [5]

Very active research topic between 2004 and 2015 [10, 6, 7, 9, 4, 3].

Introduction

This document explains the control framework known as Task-Space Inverse Dynamics (TSID).

TSID is a popular control framework for legged robots.

It all started in 1987 with this paper by *Oussama Khatib*: "A unified approach for motion and force control of robot manipulators: The operational space formulation" [5]

Very active research topic between 2004 and 2015 [10, 6, 7, 9, 4, 3].

Now not so active anymore (i.e. problem solved), but widely used.

Schedule

1. Theory of Joint Space Control ($\approx$ 1:15 hour)
2. Implementation ($\approx$ 1 hour)
3. Theory of Task Space Control ($\approx$ 1:15 hour)
4. Implementation ($\approx$ 1 hour)

Notation & Definitions

The state of the system is denoted $x \triangleq (q, v)$.
Configuration vector $q \in \mathbb{R}^{n_q}$ of (relative) joint angles.
Velocity vector $v = \dot{q} \in \mathbb{R}^{n_v}$ of (relative) joint velocities.

The control inputs are denoted $u \triangleq \tau$ (joint torques).

Notation & Definitions

The state of the system is denoted $x \triangleq (q, v)$.
Configuration vector $q \in \mathbb{R}^{n_q}$ of (relative) joint angles.
Velocity vector $v = \dot{q} \in \mathbb{R}^{n_v}$ of (relative) joint velocities.

The control inputs are denoted $u \triangleq \tau$ (joint torques).

The identity matrix is denoted I.
The zero matrix is denoted 0.
When needed, the size of the matrix is written as index, e.g., I_3.

Table of contents

Joint-Space Inverse Dynamics Control

Robot Manipulator

Given (nonlinear) manipulator dynamics:

$$M(q)\dot{v} + h(q, v) = \tau \tag{1}$$

Problem

Find $\tau(t)$ so that $q(t)$ follows reference $q^r(t)$.

Robot Manipulator

Given (nonlinear) manipulator dynamics:

$$M(q)\dot{v} + h(q, v) = \tau \tag{1}$$

Problem

Find $\tau(t)$ so that $q(t)$ follows reference $q^r(t)$.

Assumption

We know dynamics and can measure q and v.

Robot Manipulator

Given (nonlinear) manipulator dynamics:

$$M(q)\dot{v} + h(q, v) = \tau \tag{1}$$

Problem

Find $\tau(t)$ so that $q(t)$ follows reference $q^r(t)$.

Assumption

We know dynamics and can measure q and v.

Solution

Set $\tau = M(q)\dot{v}^d + h(q, v) \rightarrow$ closed-loop dynamics is $\dot{v} = \dot{v}^d$.

Robot Manipulator

Given (nonlinear) manipulator dynamics:

$$M(q)\dot{v} + h(q, v) = \tau \tag{1}$$

Problem

Find $\tau(t)$ so that $q(t)$ follows reference $q^r(t)$.

Assumption

We know dynamics and can measure q and v.

Solution

Set $\tau = M(q)\dot{v}^d + h(q, v) \rightarrow$ closed-loop dynamics is $\dot{v} = \dot{v}^d$.

Select $\dot{v}^d$ so that $q(t)$ follows $q^r(t)$:

Robot Manipulator

Given (nonlinear) manipulator dynamics:

$$M(q)\dot{v} + h(q, v) = \tau \tag{1}$$

Problem

Find $\tau(t)$ so that $q(t)$ follows reference $q^r(t)$.

Assumption

We know dynamics and can measure q and v.

Solution

Set $\tau = M(q)\dot{v}^d + h(q, v) \rightarrow$ closed-loop dynamics is $\dot{v} = \dot{v}^d$.

Select $\dot{v}^d$ so that $q(t)$ follows $q^r(t)$:

$$\dot{v}^d = \dot{v}^r$$

Robot Manipulator

Given (nonlinear) manipulator dynamics:

$$M(q)\dot{v} + h(q, v) = \tau \tag{1}$$

Problem

Find $\tau(t)$ so that $q(t)$ follows reference $q^r(t)$.

Assumption

We know dynamics and can measure q and v.

Solution

Set $\tau = M(q)\dot{v}^d + h(q, v) \rightarrow$ closed-loop dynamics is $\dot{v} = \dot{v}^d$.

Select $\dot{v}^d$ so that $q(t)$ follows $q^r(t)$:

$$\dot{v}^d = \dot{v}^r - K_d(v - v^r) - K_p(q - q^r) \tag{2}$$

where K_p, K_d are diagonal positive-definite gain matrices.

Convergence

Show that $q(t)$ converges to $q^r(t)$.

Convergence

Show that $q(t)$ converges to $q^r(t)$.

Closed-loop dynamics is

$$\dot{v} = \dot{v}^r - K_d \underbrace{(v - v^r)}_{\dot{e}} - K_p \underbrace{(q - q^r)}_{e}$$

Convergence

Show that $q(t)$ converges to $q^r(t)$.

Closed-loop dynamics is

$$\dot{v} = \dot{v}^r - K_d \underbrace{(v - v^r)}_{\dot{e}} - K_p \underbrace{(q - q^r)}_{e}$$

$$\ddot{e} = -K_d \dot{e} - K_p e$$

Convergence

Show that $q(t)$ converges to $q^r(t)$.

Closed-loop dynamics is

$$\dot{v} = \dot{v}^r - K_d \underbrace{(v - v^r)}_{\dot{e}} - K_p \underbrace{(q - q^r)}_{e}$$

$$\ddot{e} = -K_d \dot{e} - K_p e$$

$$\underbrace{\begin{bmatrix} \dot{e} \\ \ddot{e} \end{bmatrix}}_{\dot{x}} = \underbrace{\begin{bmatrix} 0 & I \\ -K_p & -K_d \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} e \\ \dot{e} \end{bmatrix}}_{x}$$

Convergence

Show that $q(t)$ converges to $q^r(t)$.

Closed-loop dynamics is

$$\dot{v} = \dot{v}^r - K_d \underbrace{(v - v^r)}_{\dot{e}} - K_p \underbrace{(q - q^r)}_{e}$$

$$\ddot{e} = -K_d \dot{e} - K_p e$$

$$\underbrace{\begin{bmatrix} \dot{e} \\ \ddot{e} \end{bmatrix}}_{\dot{x}} = \underbrace{\begin{bmatrix} 0 & I \\ -K_p & -K_d \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} e \\ \dot{e} \end{bmatrix}}_{x}$$

A is Hurwitz if K_p and K_d are diagonal and positive-definite $\rightarrow$ $\lim_{t\rightarrow\infty} x(t) = 0 \rightarrow \lim_{t\rightarrow\infty} q(t) = q^r(t)$

Many names for the same approach

This control law:

$$\tau = M(\dot{v}^r - K_d \dot{e} - K_p e) + h \tag{3}$$

is known as:

- Inverse-Dynamics (ID) Control: because based on inverse dynamics computation.
- Computed Torque: because it computes torques needed to get desired accelerations.
- Feedback Linearization (from control theory): because it uses state feedback to linearize closed-loop dynamics.

Many names for the same approach

This control law:

$$\tau = M(\dot{v}^r - K_d \dot{e} - K_p e) + h \tag{3}$$

is known as:

- Inverse-Dynamics (ID) Control: because based on inverse dynamics computation.
- Computed Torque: because it computes torques needed to get desired accelerations.
- Feedback Linearization (from control theory): because it uses state feedback to linearize closed-loop dynamics.

Another variant (with similar properties) exists:

$$\tau = M\dot{v}^r - K_d \dot{e} - K_p e + h \tag{4}$$

Other Control Laws for Manipulators

Simpler control laws often used for manipulators.

A common option is PD+gravity compensation:

$$\tau = \underbrace{-K_d\dot{e} - K_p e}_{PD} + \underbrace{g(q)}_{\text{gravity compensation}} \tag{5}$$

Other Control Laws for Manipulators

Simpler control laws often used for manipulators.

A common option is PD+gravity compensation:

$$\tau = \underbrace{-K_d \dot{e} - K_p e}_{PD} + \underbrace{g(q)}_{\text{gravity compensation}} \tag{5}$$

Another (even simpler) option is PID control:

$$\tau = -K_d \dot{e} - K_p e - \int_0^t K_i e(s) \mathrm{d}s \tag{6}$$

where integral replaces gravity compensation.

Other Control Laws for Manipulators

Simpler control laws often used for manipulators.

A common option is PD+gravity compensation:

$$\tau = \underbrace{-K_d \dot{e} - K_p e}_{PD} + \underbrace{g(q)}_{\text{gravity compensation}} \tag{5}$$

Another (even simpler) option is PID control:

$$\tau = -K_d \dot{e} - K_p e - \int_0^t K_i e(s) \mathrm{d}s \tag{6}$$

where integral replaces gravity compensation.

Both control laws are stable (so $q \to q^r$).

Other Control Laws for Manipulators

Simpler control laws often used for manipulators.

A common option is PD+gravity compensation:

$$\tau = \underbrace{-K_d\dot{e} - K_p e}_{PD} + \underbrace{g(q)}_{\text{gravity compensation}} \tag{5}$$

Another (even simpler) option is PID control:

$$\tau = -K_d\dot{e} - K_p e - \int_0^t K_i e(s)\mathrm{d}s \tag{6}$$

where integral replaces gravity compensation.

Both control laws are stable (so $q \to q^r$).

In theory "ID control" outperforms "PD+gravity", which outperforms "PID".

Other Control Laws for Manipulators

Simpler control laws often used for manipulators.

A common option is PD+gravity compensation:

$$\tau = \underbrace{-K_d \dot{e} - K_p e}_{PD} + \underbrace{g(q)}_{\text{gravity compensation}} \tag{5}$$

Another (even simpler) option is PID control:

$$\tau = -K_d \dot{e} - K_p e - \int_0^t K_i e(s) \mathrm{d}s \tag{6}$$

where integral replaces gravity compensation.

Both control laws are stable (so $q \to q^r$).

In theory "ID control" outperforms "PD+gravity", which outperforms "PID".

In practice the opposite could occur because of model errors.

Inverse Dynamics Control as Optimization Problem

Inverse Dynamics (ID) Control as Least-Squares Problem

Solution of optimization problem:

$$(\tau^*, \dot{v}^*) = \begin{aligned} \underset{\tau, \dot{v}}{\operatorname{argmin}} \quad & ||\dot{v} - \dot{v}^d||^2 \\ \text{subject to} \quad & M\dot{v} + h = \tau \end{aligned} \tag{7}$$

with $\dot{v}^d = \dot{v}^r - K_d \dot{e} - K_p e$

Inverse Dynamics (ID) Control as Least-Squares Problem

Solution of optimization problem:

$$
\begin{aligned}
(\tau^*, \dot{v}^*) = \quad & \underset{\tau, \dot{v}}{\text{argmin}} && ||\dot{v} - \dot{v}^d||^2 \\
& \text{subject to} && M\dot{v} + h = \tau
\end{aligned}
\tag{7}
$$

with $\dot{v}^d = \dot{v}^r - K_d \dot{e} - K_p e$, is exactly the ID control law:

$$
\tau^* = M\dot{v}^d + h, \tag{8}
$$

Inverse Dynamics (ID) Control as Least-Squares Problem

Solution of optimization problem:

$$\begin{aligned}(\tau^*, \dot{v}^*) = \quad & \underset{\tau, \dot{v}}{\text{argmin}} \quad && ||\dot{v} - \dot{v}^d||^2 \\ & \text{subject to} && M\dot{v} + h = \tau\end{aligned} \tag{7}$$

with $\dot{v}^d = \dot{v}^r - K_d\dot{e} - K_p e$, is exactly the ID control law:

$$\tau^* = M\dot{v}^d + h, \tag{8}$$

No advantage in solving (7) to compute (8), but (7) is starting point to solve more complex problems.

Inverse Dynamics (ID) Control as Least-Squares Problem

Solution of optimization problem:

$$\begin{aligned}(\tau^*, \dot{v}^*) = \quad & \underset{\tau, \dot{v}}{\text{argmin}} \quad && ||\dot{v} - \dot{v}^d||^2 \\ & \text{subject to} && M\dot{v} + h = \tau\end{aligned} \tag{7}$$

with $\dot{v}^d = \dot{v}^r - K_d \dot{e} - K_p e$, is exactly the ID control law:

$$\tau^* = M\dot{v}^d + h, \tag{8}$$

No advantage in solving (7) to compute (8), but (7) is starting point to solve more complex problems.

Problem (7) is Least-Squares Program/Problem (LSP).

Taxonomy of Convex Optimization Problems

Least-Squares Programs (LSP) have:

- linear equality/inequality constraints ($Ax \leq b$, or $Ax = b$)
- 2-norm of linear cost function ($||Ax - b||^2$)

Taxonomy of Convex Optimization Problems

Least-Squares Programs (LSP) have:

- linear equality/inequality constraints ($Ax \leq b$, or $Ax = b$)
- 2-norm of linear cost function ($||Ax - b||^2$)

LSPs are subclass of convex Quadratic Programs (QPs), which have:

- linear equality/inequality constraints ($Ax \leq b$, or $Ax = b$)
- convex quadratic cost function ($x^\top Hx + h^\top x$, with $H \geq 0$)

Taxonomy of Convex Optimization Problems

Least-Squares Programs (LSP) have:

- linear equality/inequality constraints ($Ax \leq b$, or $Ax = b$)
- 2-norm of linear cost function ($||Ax - b||^2$)

LSPs are subclass of convex Quadratic Programs (QPs), which have:

- linear equality/inequality constraints ($Ax \leq b$, or $Ax = b$)
- convex quadratic cost function ($x^\top Hx + h^\top x$, with $H \geq 0$)

LSPs and convex QPs can be solved extremely fast with off-the-shelf softwares

Taxonomy of Convex Optimization Problems

Least-Squares Programs (LSP) have:

- linear equality/inequality constraints ($Ax \leq b$, or $Ax = b$)
- 2-norm of linear cost function ($||Ax - b||^2$)

LSPs are subclass of convex Quadratic Programs (QPs), which have:

- linear equality/inequality constraints ($Ax \leq b$, or $Ax = b$)
- convex quadratic cost function ($x^\top Hx + h^\top x$, with $H \geq 0$)

LSPs and convex QPs can be solved extremely fast with off-the-shelf softwares
$\rightarrow$ We can solve LSP/QPs inside 1 kHz control loops!

Adding Torque Limits to ID Control

Take the ID control LSP:

$$\begin{aligned} \underset{\tau, \dot{v}}{\text{minimize}} \quad & ||\dot{v} - \dot{v}^d||^2 \\ \text{subject to} \quad & M\dot{v} + h = \tau \end{aligned} \tag{9}$$

Adding Torque Limits to ID Control

Take the ID control LSP:

$$\begin{aligned} \underset{\tau,\dot{v}}{\text{minimize}} \quad & ||\dot{v} - \dot{v}^d||^2 \\ \text{subject to} \quad & M\dot{v} + h = \tau \end{aligned} \tag{9}$$

LSPs allow for linear inequality constraints $\rightarrow$ we can add torque limits:

$$\begin{aligned} \underset{\tau,\dot{v}}{\text{minimize}} \quad & ||\dot{v} - \dot{v}^d||^2 \\ \text{subject to} \quad & M\dot{v} + h = \tau \\ & \tau^{min} \leq \tau \leq \tau^{max} \end{aligned} \tag{10}$$

Adding Torque Limits to ID Control

Take the ID control LSP:

$$\begin{aligned} \underset{\tau, \dot{v}}{\text{minimize}} \quad & ||\dot{v} - \dot{v}^d||^2 \\ \text{subject to} \quad & M\dot{v} + h = \tau \end{aligned} \tag{9}$$

LSPs allow for linear inequality constraints $\rightarrow$ we can add torque limits:

$$\begin{aligned} \underset{\tau, \dot{v}}{\text{minimize}} \quad & ||\dot{v} - \dot{v}^d||^2 \\ \text{subject to} \quad & M\dot{v} + h = \tau \\ & \tau^{min} \leq \tau \leq \tau^{max} \end{aligned} \tag{10}$$

Main advantage of optimization: inequality constraints.

Adding Current Limits for Electric Motors

In electric motors current i is proportional to torque τ:

$$i = k_\tau \tau \tag{11}$$

Adding Current Limits for Electric Motors

In electric motors current i is proportional to torque τ:

$$i = k_\tau \tau \tag{11}$$

Add current limits:

$$\begin{aligned} \underset{\tau, \dot{v}}{\text{minimize}} \quad & ||\dot{v} - \dot{v}^d||^2 \\ \text{subject to} \quad & M\dot{v} + h = \tau \\ & \tau^{min} \leq \tau \leq \tau^{max} \\ & i^{min} \leq k_\tau \tau \leq i^{max} \end{aligned} \tag{12}$$

Adding Joint Velocity Limits

Assuming constant accelerations $\dot{v}$ during time step Δt:

$$v(t + \Delta t) = v(t) + \Delta t \, \dot{v} \tag{13}$$

Adding Joint Velocity Limits

Assuming constant accelerations $\dot{v}$ during time step Δt:

$$v(t + \Delta t) = v(t) + \Delta t \, \dot{v} \tag{13}$$

Add joint velocity limits:

$$\begin{aligned} \underset{\tau, \dot{v}}{\text{minimize}} \quad & ||\dot{v} - \dot{v}^d||^2 \\ \text{subject to} \quad & M\dot{v} + h = \tau \\ & \tau^{min} \leq \tau \leq \tau^{max} \\ & i^{min} \leq k_\tau \tau \leq i^{max} \\ & v^{min} \leq v + \Delta t \dot{v} \leq v^{max} \end{aligned} \tag{14}$$

Adding Joint Position Limits

Could use same trick for position limits:

$$q(t + \Delta t) = q(t) + \Delta t \, v(t) + \frac{1}{2} \Delta t^2 \dot{v} \tag{15}$$

Adding Joint Position Limits

Could use same trick for position limits:

$$q(t + \Delta t) = q(t) + \Delta t \, v(t) + \frac{1}{2} \Delta t^2 \dot{v} \tag{15}$$

However, this can result in high accelerations, typically incompatible with torque/current limits $\rightarrow$ unfeasible LSP.

Adding Joint Position Limits

Could use same trick for position limits:

$$q(t + \Delta t) = q(t) + \Delta t\, v(t) + \frac{1}{2}\,\Delta t^2\,\dot{v} \tag{15}$$

However, this can result in high accelerations, typically incompatible with torque/current limits $\rightarrow$ unfeasible LSP.

Better approaches exist [1, 8, 2], but we don't discuss them here.

Summary

Inverse-Dynamics Control: $\tau = M(\dot{v}^r - K_d \dot{e} - K_p e) + h$

Summary

Inverse-Dynamics Control: $\tau = M(\dot{v}^r - K_d\dot{e} - K_p e) + h$

Other version: $\tau = M\dot{v}^r - K_d\dot{e} - K_p e + h$

Summary

$$\text{Inverse-Dynamics Control:} \qquad \tau = M(\dot{v}^r - K_d\dot{e} - K_p e) + h$$

$$\text{Other version:} \qquad \tau = M\dot{v}^r - K_d\dot{e} - K_p e + h$$

$$\text{PD + gravity compensation:} \qquad \tau = -K_d\dot{e} - K_p e + g(q)$$

Summary

Inverse-Dynamics Control: $\tau = M(\dot{v}^r - K_d\dot{e} - K_p e) + h$

Other version: $\tau = M\dot{v}^r - K_d\dot{e} - K_p e + h$

PD + gravity compensation: $\tau = -K_d\dot{e} - K_p e + g(q)$

PID: $\tau = -K_d\dot{e} - K_p e - \int_0^t K_i e(s)\mathrm{ds}$

Summary

Inverse-Dynamics Control: $\tau = M(\dot{v}^r - K_d \dot{e} - K_p e) + h$

Other version: $\tau = M\dot{v}^r - K_d \dot{e} - K_p e + h$

PD + gravity compensation: $\tau = -K_d \dot{e} - K_p e + g(q)$

PID: $\tau = -K_d \dot{e} - K_p e - \int_0^t K_i e(s) ds$

ID Control as LSP:

$$\begin{aligned} \underset{\tau, \dot{v}}{\text{minimize}} \quad & ||\dot{v} - \dot{v}^d||^2 \\ \text{subject to} \quad & M\dot{v} + h = \tau \end{aligned}$$

Summary

Inverse-Dynamics Control:	$\tau = M(\dot{v}^r - K_d\dot{e} - K_p e) + h$
Other version:	$\tau = M\dot{v}^r - K_d\dot{e} - K_p e + h$
PD + gravity compensation:	$\tau = -K_d\dot{e} - K_p e + g(q)$
PID:	$\tau = -K_d\dot{e} - K_p e - \int_0^t K_i e(s) \mathrm{d}s$

ID Control as LSP:

$$
\begin{aligned}
\underset{\tau,\dot{v}}{\text{minimize}} \quad & ||\dot{v} - \dot{v}^d||^2 \\
\text{subject to} \quad & M\dot{v} + h = \tau \\
& \tau^{min} \leq \tau \leq \tau^{max}
\end{aligned}
$$

Summary

Inverse-Dynamics Control: $\tau = M(\dot{v}^r - K_d\dot{e} - K_p e) + h$

Other version: $\tau = M\dot{v}^r - K_d\dot{e} - K_p e + h$

PD + gravity compensation: $\tau = -K_d\dot{e} - K_p e + g(q)$

PID: $\tau = -K_d\dot{e} - K_p e - \int_0^t K_i e(s)\mathrm{ds}$

ID Control as LSP:

$$\begin{aligned} \underset{\tau,\dot{v}}{\text{minimize}} \quad & ||\dot{v} - \dot{v}^d||^2 \\ \text{subject to} \quad & M\dot{v} + h = \tau \\ & \tau^{min} \leq \tau \leq \tau^{max} \\ & i^{min} \leq k_\tau \tau \leq i^{max} \end{aligned}$$

Summary

Inverse-Dynamics Control: $\tau = M(\dot{v}^r - K_d \dot{e} - K_p e) + h$

Other version: $\tau = M\dot{v}^r - K_d \dot{e} - K_p e + h$

PD + gravity compensation: $\tau = -K_d \dot{e} - K_p e + g(q)$

PID: $\tau = -K_d \dot{e} - K_p e - \int_0^t K_i e(s) \mathrm{d}s$

ID Control as LSP:

$$
\begin{aligned}
\underset{\tau, \dot{v}}{\text{minimize}} \quad & ||\dot{v} - \dot{v}^d||^2 \\
\text{subject to} \quad & M\dot{v} + h = \tau \\
& \tau^{min} \leq \tau \leq \tau^{max} \\
& i^{min} \leq k_\tau \tau \leq i^{max} \\
& v^{min} \leq v + \Delta t \dot{v} \leq v^{max}
\end{aligned}
$$

W. Decré, R. Smits, H. Bruyninckx, and J. De Schutter.
Extending iTaSC to support inequality constraints and non-instantaneous task specification.
In *IEEE International Conference on Robotics and Automation (ICRA)*, 2009.

A. Del Prete.
Joint Position and Velocity Bounds in Discrete-Time Acceleration / Torque Control of Robot Manipulators.
IEEE Robotics and Automation Letters, 3(1), 2018.

A. Del Prete, F. Nori, G. Metta, and L. Natale.
Prioritized Motion-Force Control of Constrained Fully-Actuated Robots: "Task Space Inverse Dynamics".
Robotics and Autonomous Systems, 63:150–157, 2015.

A. Escande, N. Mansard, and P.-B. Wieber.
Hierarchical Quadratic Programming: Fast Online Humanoid-Robot Motion Generation.
International Journal of Robotics Research, 33(7):1006–1028, 2014.

O. Khatib.
A unified approach for motion and force control of robot manipulators: The operational space formulation.
IEEE Journal on Robotics and Automation, 3(1):43–53, feb 1987.

M. Mistry, J. Buchli, and S. Schaal.
Inverse dynamics control of floating base systems using orthogonal decomposition.
2010 IEEE International Conference on Robotics and Automation, (3):3406–3412, may 2010.

L. Righetti, J. Buchli, M. Mistry, M. Kalakrishnan, and S. Schaal.
Optimal distribution of contact forces with inverse dynamics control.
The International Journal of Robotics Research, (January), jan 2013.

S. Rubrecht, V. Padois, P. Bidaud, M. Broissia, and M. Da Silva Simoes.
Motion safety and constraints compatibility for multibody robots.
Autonomous Robots, 32(3):333–349, 2012.

L. Saab, O. E. Ramos, N. Mansard, P. Soueres, and J.-y. Fourquet.
Dynamic Whole-Body Motion Generation under Rigid Contacts and other Unilateral Constraints.
IEEE Transactions on Robotics, 29(2):346–362, 2013.

L. Sentis and O. Khatib.
Synthesis of whole-body behaviors through hierarchical control of behavioral primitives.
International Journal of Humanoid Robotics, 2(4):505–518, 2005.