Robot Control
A Personal View on Some Limits and Perspectives

Vincent Padois – vincent.padois@inria.fr
Senior research scientist
Inria Bordeaux Sud-Ouest, Auctus

1st R4 Workshop,
ESTIA Bidart – 2021/11/8
Interactive robots do not exist for real

Real-world ...

Basic locomotion and manipulation skills

Advanced locomotion skills

Cognitive and physical interactions
Interactive robots do not exist for real

... vs Laboratory science and technology

Advanced control but no living bodies around

How many (truly) collaborative robots have you seen in the industry?

Why is it so?
The world is dynamic, complex and hard to predict (impact in 6s)
Outline of the presentation

1. Introduction

2. Limitations of existing control approaches

3. Real-life examples

4. Some potential solutions
   - Robot low-level control as an optimisation problem
   - Plan wise, perform wise

5. Conclusions
(Reactive) Optimal control

Ideally, solve reactively ...

\[
\min_{t_0, t_f, x(t), u(t)} \left( J_b(t_0, t_f, x(t_0), x(t_f)) + \int_{t_0}^{t_f} J_i(s, x(s), u(s)) \, ds \right)
\]

subject to:
- Dynamics: \( \dot{x}(t) = f(t, x(t), u(t)) \)
- Path constraints: \( h(t, x(t), u(t)) \leq 0 \)
- State constraints: \( x_l(t) \leq x(t) \leq x_u(t) \)
- Control bounds: \( u_l(t) \leq u(t) \leq u_u(t) \)
Ideally, solve reactively ...

\[
\min_{t_0, t_f, x(t), u(t)} \left( J_b(t_0, t_f, x(t_0), x(t_f)) + \int_{t_0}^{t_f} J_i(s, x(s), u(s)) \, ds \right)
\]

... but in practice

- infinite dimensional problem
- can generally not be solved, even once
- transformed in a finite dimensional problem: non linear program / constrained parameter optimization
- hard to solve, cannot be solved reactively
Looking closer

In dynamic environments, \( x(t) = \{ x_{rob}(t), x_{env}(t) \} \) requires **perception** for the state of the environment \( x_{env}(t) \) → no control over \( x_{env}(t) \) → reactive planning needed
Looking closer

In dynamic environments, \( x(t) = \{ x_{\text{rob}}(t), x_{\text{env}}(t) \} \)
\[ \rightarrow \] requires **perception** for the state of the environment \( x_{\text{env}}(t) \)
\[ \rightarrow \] no control over \( x_{\text{env}}(t) \) \( \rightarrow \) reactive planning needed

\[ \rightarrow \] compute an optimal control input trajectory \( \tau(t) \) **at each control instant** given
Looking closer

In dynamic environments, \( x(t) = \{x_{rob}(t), x_{env}(t)\} \)

\( \rightarrow \) requires **perception** for the state of the environment \( x_{env}(t) \)

\( \rightarrow \) no control over \( x_{env}(t) \) \( \rightarrow \) reactive planning needed

\( \rightarrow \) compute an optimal control input trajectory \( \tau(t) \) at each control instant given

\[\begin{align*}
\text{Control objectives: } & \{H_{1,f}, \ldots, H_{n_0,f}\} \\
\end{align*}\]
Looking closer

In dynamic environments, $x(t) = \{x_{rob}(t), x_{env}(t)\}$

$\Rightarrow$ requires **perception** for the state of the environment $x_{env}(t)$

$\Rightarrow$ no control over $x_{env}(t)$ $\Rightarrow$ reactive planning needed

$\Rightarrow$ compute an optimal control input trajectory $\tau(t)$ at each control instant given

- Control objectives : $\{H_1, f, \ldots, H_{n_0}, f\}$
- (Non-linear) Dynamics of the system :
  - $M(q)\dot{\nu} + b(q, \nu) = S^T(q)\tau \left( + \sum_{i}^{nc} J^T_{ci}(q)f_{ci} \right)$
  - $\nu_i = J(q)\dot{\nu}$ $\forall i \in [1, n_0]$ and $\nu_i := \dot{H}_i$
Looking closer

In dynamic environments, $x(t) = \{x_{rob}(t), x_{env}(t)\}$

$\mapsto$ requires **perception** for the state of the environment $x_{env}(t)$

$\mapsto$ no control over $x_{env}(t) \rightarrow$ reactive planning needed

$\mapsto$ compute an optimal control input trajectory $\tau(t)$ at each control instant given

- Control objectives: $\{H_{1,f}, \ldots, H_{n_o,f}\}$
- (Non-linear) Dynamics of the system:
  1. $M(q)\dot{\nu} + b(q, \nu) = S^T(q)\tau ( + \sum_{i}^{n_c} J_{ci}^T(q)f_{ci})$
  2. $\nu_i = J(q)\dot{\nu} \ \forall i \in [1, n_o]$ and $\nu_i := \dot{H}_i$
- Constraints:
  1. $\tau_l \leq \tau \leq \tau_u$
  2. $\dot{\tau}_l \leq \dot{\tau} \leq \dot{\tau}_u$
  3. $q_l \leq q \leq q_u$
  4. $\dot{\nu}_l \leq \dot{\nu} \leq \dot{\nu}_u$
  5. $h(x_{env}, q) \leq 0$
  6. ...

$\mapsto$ **very complex and computationally demanding control / optimization problem**
Outline of the presentation

1. Introduction

2. Limitations of existing control approaches

3. Real-life examples

4. Some potential solutions
   - Robot low-level control as an optimisation problem
   - Plan wise, perform wise

5. Conclusions
Optimal control vs real-life

Historically in the industry, the problem left to robots is simplified.
Optimal control vs real-life

Static environment → reactivity not required at the task planning level ...

... as constraints are met

- offline, through planning
- a posteriori through emergency stops or stereotypical safety zones definition
Optimal control vs real-life

Static environment → reactivity not required at the task planning level ...

... as constraints are met
- offline, through planning
- a posteriori through emergency stops or stereotypical safety zones definition

Yet finding a control trajectory is complex
- Decouple planning and control
  - Plan for \( q(t) \) or \( H(t) \)
  - Perform trajectory servoing and low level-control
Optimal control vs real-life

Static environment  →  reactivity not required at the task planning level ... 

... as constraints are met
- offline, through planning
- a posteriori through emergency stops or stereotypical safety zones definition

Yet finding a control trajectory is complex
- Decouple planning and control
  - Plan for $q(t)$ or $H(t)$
  - Perform trajectory servoing and low level-control

Still too complex!
- Simplification based on an underestimation of the true robot capacities
  - the industry is full of oversized and dangerous robots
- Highly expert manual tuning required
  - robots are not the promised versatile tools
Illustration with the Franka Emika Panda Robot

## Constants

Limits in the Cartesian space are as follows:

<table>
<thead>
<tr>
<th>Name</th>
<th>Translation</th>
<th>Rotation</th>
<th>Elbow</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\dot{p}_{\text{max}}$</td>
<td>1.7000 m/s</td>
<td>2.5000 rad/s</td>
<td>2.1750 rad/s</td>
</tr>
<tr>
<td>$p_{\text{max}}$</td>
<td>13.0000 m/s^2</td>
<td>25.0000 rad/s^2</td>
<td>10.0000 rad/s^2</td>
</tr>
<tr>
<td>$\ddot{p}_{\text{max}}$</td>
<td>6500.0000 m/s^3</td>
<td>12500.0000 rad/s^3</td>
<td>5000.0000 rad/s^3</td>
</tr>
</tbody>
</table>

Joint space limits are:

<table>
<thead>
<tr>
<th>Name</th>
<th>Joint 1</th>
<th>Joint 2</th>
<th>Joint 3</th>
<th>Joint 4</th>
<th>Joint 5</th>
<th>Joint 6</th>
<th>Joint 7</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_{\text{max}}$</td>
<td>2.8973</td>
<td>1.7628</td>
<td>2.8973</td>
<td>-0.0698</td>
<td>2.8973</td>
<td>3.7525</td>
<td>2.8973</td>
<td>rad</td>
</tr>
<tr>
<td>$q_{\text{min}}$</td>
<td>-2.8973</td>
<td>-1.7628</td>
<td>-2.8973</td>
<td>-3.0718</td>
<td>-2.8973</td>
<td>-0.0175</td>
<td>-2.8973</td>
<td>rad</td>
</tr>
<tr>
<td>$\dot{q}_{\text{max}}$</td>
<td>2.1750</td>
<td>2.1750</td>
<td>2.1750</td>
<td>2.1750</td>
<td>2.6100</td>
<td>2.6100</td>
<td>2.6100</td>
<td>rad/s</td>
</tr>
<tr>
<td>$\ddot{q}_{\text{max}}$</td>
<td>15</td>
<td>7.5</td>
<td>10</td>
<td>12.5</td>
<td>15</td>
<td>20</td>
<td>20</td>
<td>rad/s^2</td>
</tr>
<tr>
<td>$\dot{q}_{\text{max}}$</td>
<td>7500</td>
<td>3750</td>
<td>5000</td>
<td>6250</td>
<td>7500</td>
<td>10000</td>
<td>10000</td>
<td>rad/s^2</td>
</tr>
<tr>
<td>$\dddot{q}_{\text{max}}$</td>
<td>87</td>
<td>87</td>
<td>87</td>
<td>87</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>Nm</td>
</tr>
<tr>
<td>$\tau_{\text{max}}$</td>
<td>1000</td>
<td>1000</td>
<td>1000</td>
<td>1000</td>
<td>1000</td>
<td>1000</td>
<td>1000</td>
<td>Nm/s</td>
</tr>
</tbody>
</table>

---
→ Curse of "collaborative" robotics

- Safety in the collaboration requires small robots and controlled stops
- Small robots capabilities are small
- Underestimating the capabilities of small robots leads to "not much" capabilities
- Potentially safe robots are mostly useless
For systems making intermittent contacts with the environment (e.g. humanoids walking)...

... mostly two solutions

- Sequential simplified planning problem solving from contact sequence to center of mass trajectory under balance constraints and in purely static environment (plan once)

- Stereotypical walking gaits (planned once) on flat grounds and online planar trajectory adaptation

  + Trajectory servoing and multi-task whole-body control

[Ibanez 2017]
Optimal control vs complex robots (e.g. humanoids)

For systems making intermittent contacts with the environment (e.g. humanoids walking)...

Difficulties

- Planning performed with advanced models is costly → no reactivity
- Simplified models do not account for the true capabilities of the system → underestimation / overestimation → manual tuning
- Humanoids can't do much in real life

[Ibanez 2017]
Outline of the presentation

1 Introduction

2 Limitations of existing control approaches

3 Real-life examples

4 Some potential solutions
   ■ Robot low-level control as an optimisation problem
   ■ Plan wise, perform wise

5 Conclusions
Outline of the presentation

1. Introduction
2. Limitations of existing control approaches
3. Real-life examples
4. Some potential solutions
   - Robot low-level control as an optimisation problem
   - Plan wise, perform wise
5. Conclusions
Robot low-level control as an optimisation problem

In a dynamic environment, performance and safety requires to embed constraints in the low-level control problem: at each control instant, find the actuation torque $\tau^*$ optimizing under constraints some objective related task $\nu^* = J(q)\nu$.
Robot low-level control as an optimisation problem

In a dynamic environment, performance and safety requires to embed constraints in the low-level control problem: at each control instant, find the actuation torque $\tau^*$ optimizing under constraints some objective related task $\nu^* = J(q)\nu$

- Equation of motion and joint space to task space mappings: **equalities**
  - can be solved using Linear Algebra
    - $M(q)\dot{\nu} + b(q, \nu) = S^T(q)\tau + \sum_{i}^{nc} J_{ci}^T(q)f_{ci}$
    - $\nu_i = J(q)\dot{\nu} \quad \forall i \in [1, n_o]$ and $\nu_i := \dot{H}_i$

- Some limits on the system cannot or should never be crossed: **inequalities**
  - cannot be accounted for properly using Linear Algebra only
Robot low-level control as an optimisation problem

In a dynamic environment, performance and safety requires to embed constraints in the low-level control problem: at each control instant, find the actuation torque $\tau^*$ optimizing under constraints some objective related task $v^* = J(q)\nu$

- **Equation of motion and joint space to task space mappings:** equalities
  $\iff$ can be solved using Linear Algebra
  - $M(q)\dot{\nu} + b(q, \nu) = S^T(q)\tau + \sum_{i}^{n_c} J^T_{c_i}(q)f_{c_i}$
  - $v_i = J(q)\dot{\nu}$ $\forall i \in [1,n_o]$ and $v_i := \dot{H}_i$

- **Standard IVK and operational space control approaches**
  $\iff$ solution based on $J^+$ and null-space projections $\dot{\nu} = J^+(q)\nu + (I - J^+J)\dot{\nu}_0$

---

* see the work of [Liégeois 1977], [Khatib 1987], [Siciliano 1991], [Chiaverini 1997], [Mansard 2009], [Flacco 2012],...
Robot low-level control as an optimisation problem

In a dynamic environment, performance and safety requires to embed constraints in the low-level control problem: at each control instant, find the actuation torque $\tau^*$ optimizing under constraints some objective related task $v^* = J(q)v$

- Equation of motion and joint space to task space mappings: **equalities**
  - can be solved using Linear Algebra
    - $M(q)\dot{\nu} + b(q, \nu) = S^T(q)\tau + \sum_{i}^{n_c} J_{ci}^T(q)f_{ci}$
    - $v_i = J(q)\dot{\nu}$ \forall $i \in [1, n_o]$ and $v_i := H_i$

- **Standard IVK and operational space control approaches**
  - solution based on $J^+$ and null-space projections $\dot{\nu} = J^+(q)v + (I - J^+J)\dot{\nu}_0$

- Some limits on the system cannot or should never be crossed: **inequalities**
  - cannot be accounted for properly using Linear Algebra only
    $$D(q, \nu)X \leq h(q, \nu)$$

* see the work of  [Liégeois 1977], [Khatib 1987], [Siciliano 1991], [Chiaverini 1997], [Mansard 2009], [Flacco 2012], ...
Robot low-level control as an optimisation problem

In a dynamic environment, performance and safety requires to embed constraints in the low-level control problem: at each control instant, find the actuation torque $\tau^*$ optimizing under constraints some objective related task $\mathbf{v}^* = \mathbf{J}(q)\mathbf{v}$

- Equation of motion and joint space to task space mappings: equalities $\iff$ can be solved using Linear Algebra
  - $\mathbf{M}(q)\dot{\mathbf{v}} + \mathbf{b}(q, \mathbf{v}) = \mathbf{S}^T(q)\tau^* + \sum_{i=1}^{n_c} \mathbf{J}^T_{ci}(q)f_{ci}$
  - $\mathbf{v}_i = \mathbf{J}(q)\dot{\mathbf{v}} \forall i \in [1, n_o]$ and $\mathbf{v}_i := \dot{\mathbf{H}}_i$

- Standard IVK and operational space control approaches* $\iff$ solution based on $\mathbf{J}^+$ and null-space projections $\dot{\mathbf{v}} = \mathbf{J}^+(q)\mathbf{v} + (\mathbf{I} - \mathbf{J}^+\mathbf{J})\dot{\mathbf{v}}_0$

- Some limits on the system cannot or should never be crossed: inequalities $\iff$ cannot be accounted for properly using Linear Algebra only
  \[ \mathbf{D}(q, \mathbf{v})\mathbf{X} \leq \mathbf{h}(q, \mathbf{v}) \]

- These constraints are linear wrt control variables: convex solution space $\iff$ convex optimization (LQP) is a powerful tool to solve optimally the reactive control problem.

* see the work of [Liégeois 1977], [Khatib 1987], [Siciliano 1991], [Chiaverini 1997], [Mansard 2009], [Flacco 2012],...
3 reasons why Quadratic Programs are better than explicit Jacobian inversions
Robot low-level control as an optimisation problem

3 reasons why Quadratic Programs are better than explicit Jacobian inversions

1. Leave your robot alone
   - Methods based on $J^+$ forces constraints to be treated as tasks $\rightarrow$ active avoidance
   - QP allows to consider constraints as such $\rightarrow$ passive avoidance

[Rubrecht 2012a, Meguenani 2017b, Del Prete 2018a]
Robot low-level control as an optimisation problem

**3 reasons why Quadratic Programs are better than explicit Jacobian inversions**

1. **Leave your robot alone**
   - Methods based on $J^+$ forces constraints to be treated as tasks $\rightarrow$ active avoidance
   - QP allows to consider constraints as such $\rightarrow$ passive avoidance

2. **More constraints than DoFs : choose which one to consider at each time**
   - Methods based on $J^+$ use context specific heuristics to do so
   - QP comes with an optimal active constraints determination algorithm
Robot low-level control as an optimisation problem

3 reasons why Quadratic Programs are better than explicit Jacobian inversions

1. Leave your robot alone
   - Methods based on $J^+$ forces constraints to be treated as tasks $\rightarrow$ active avoidance
   - QP allows to consider constraints as such $\rightarrow$ passive avoidance

2. More constraints than DoFs: choose which one to consider at each time
   - Methods based on $J^+$ use context specific heuristics to do so
   - QP comes with an optimal active constraints determination algorithm

3. Infeasibility can’t be ignored
   - Methods based on $J^+$ can solve infeasible problems $\rightarrow$ constraints violation
   - QP can’t be solved if infeasible $\rightarrow$ deal with this problem first

[Rubrecht 2012a, Meguenani 2017b, Del Prete 2018a]
Constraints compliance as a control feature

For example:

\[ \tau_{k+1}^* = \arg \min_{\tau_{k+1}, \dot{q}_{k+1}} \left\| \text{obj} \left( \ddot{q}_{k+1}, \dot{x}_{k+1}^* \right) \right\|_{Q_t}^2 + \epsilon \left\| \begin{bmatrix} \tau_{k+1} \\ \ddot{q}_{k+1} \end{bmatrix} \right\|_{Q_r}^2 \]

such that

\[ M(q_k) \ddot{q}_{k+1} + b(q_k, \dot{q}_k) = S^T(q_k) \tau_{k+1} \]

\[ \tau_{\text{min}} \leq \tau_{k+1} \leq \tau_{\text{max}} \]

\[ q_{\text{min}} \leq q_{k+1} \leq q_{\text{max}} \]

\[ \dot{q}_{\text{min}} \leq \dot{q}_{k+1} \leq \dot{q}_{\text{max}} \]

\[ 0 \leq d_{k+1}^{rob, obj} \quad \forall j \in \{1, \ldots, n_{\text{obj}}\} \]

\[ \text{obj} \left( \ddot{q}_{k+1}, \dot{x}_{k+1}^* \right) = \dot{x}_{k+1}^{\text{des}} + PD(x_k, x_{k+1}^{\text{des}}) - J(q_k) \ddot{q}_{k+1} - \dot{J}(q_k) \dot{q}_k \]

\[ \ddot{q}_{k+1}^{*} \]

\[ \dot{x}_{k+1}^{*} \]
Constraints compliance as a control feature: the teleoperation case

- PhD thesis Sébastien Rubrecht, ANR TELEMACH, CIFRE Bouygues Construction [Rubrecht 2010, Rubrecht 2011, Rubrecht 2012a]
- **Context**: Teleoperation in tunnel boring machine cutter-heads
- Static environment, interactive task definition
Constraints compliance as a control feature

- PhD work of Lucas Joseph, CIFRE GE Healthcare [Joseph 2018c]
- Dynamic environment: perception in the loop and reactive constraints adaptation
Outline of the presentation

1. Introduction

2. Limitations of existing control approaches

3. Real-life examples

4. Some potential solutions
   - Robot low-level control as an optimisation problem
   - Plan wise, perform wise

5. Conclusions
Viability – Do not plan to do what you cannot do.

- Existence of a solution to the control problem over an $\infty$ time horizon?
  
  [Fraichard 2004], [Wieber 2008]
Viability – Do not plan to do what you cannot do.

Existence of a solution to the control problem over an $\infty$ time horizon?

[Fraichard 2004], [Wieber 2008]
Viability – Do not plan to do what you cannot do.

- Existence of a solution to the control problem over an $\infty$ time horizon? 
  
  [Fraichard 2004], [Wieber 2008]

- Modify the constraints expression to ensure compatibility [Rubrecht 2012b]

$$q'_{\min}(q_k, \nu_k, \dot{\nu}_{\min}, \dot{\nu}_{\max}) \leq q_{k+1} \leq q'_{\max}(q_k, \nu_k, \dot{\nu}_{\min}, \dot{\nu}_{\max})$$
Viability – Do not plan to do what you cannot do.

- **Existence of a solution to the control problem over an $\infty$ time horizon?**
  
  [Fraichard 2004], [Wieber 2008]

- **Modify the constraints expression to ensure compatibility** [Rubrecht 2012b]
  
  \[
  q'_\text{min}(q_k, \nu, \dot{\nu}_{\text{min}}, \dot{\nu}_{\text{max}}) \leq q_{k+1} \leq q'_\text{max}(q_k, \nu, \dot{\nu}_{\text{min}}, \dot{\nu}_{\text{max}})
  \]

- **Unfortunately** \( \dot{\nu}_{k+1} = M^{-1}(q_k)(S^T(q_k)\tau_{k+1} - b(q_k, \nu_k)) \) \( \rightarrow \dot{\nu}_{\text{max}, k+n} =? \)
Viability – Do not plan to do what you cannot do.

- Existence of a solution to the control problem over an $\infty$ time horizon? [Fraichard 2004], [Wieber 2008]

- Modify the constraints expression to ensure compatibility [Rubrecht 2012b]
  \[
  q'_\text{min}(q_k, \nu_k, \dot{\nu}_\text{min}, \dot{\nu}_\text{max}) \leq q_{k+1} \leq q'_\text{max}(q_k, \nu_k, \dot{\nu}_\text{min}, \dot{\nu}_\text{max})
  \]

- Unfortunately $\dot{\nu}_{k+1} = M^{-1}(q_k)(S^T(q_k)\tau_{k+1} - b(q_k, \nu_k)) \rightarrow \dot{\nu}_{\text{max}, k+n} = ?$

- Look for a minorant of the joint space acceleration capabilities [Meguenani 2017c], [Del Prete 2018b]
Viability – Do not plan to do what you cannot do.

- **Existence of a solution to the control problem over an ∞ time horizon?**
  
  \[ \text{[Fraichard 2004], [Wieber 2008]} \]

- **Modify the constraints expression to ensure compatibility** \([\text{Rubrecht 2012b]}\)
  \[
  q_{\min}(q_k, \nu_k, \dot{\nu}_{\min}, \dot{\nu}_{\max}) \leq q_{k+1} \leq q_{\max}(q_k, \nu_k, \dot{\nu}_{\min}, \dot{\nu}_{\max})
  \]

- **Unfortunately** \(\dot{\nu}_{k+1} = M^{-1}(q_k)\left(S^T(q_k)\tau_{k+1} - b(q_k, \nu_k)\right) \rightarrow \dot{\nu}_{\max, k+n} = ?\)

- **Look for a minorant of the joint space acceleration capabilities** \([\text{Meguenani 2017c}, \text{Del Prete 2018b}]\)

- **The problem gets even more complex when looking in the task space?**
  \[
  \ddot{x}_{k+1} = J(q_k)M^{-1}(q_k)\left(S^T(q_k)\tau_{k+1} - b(q_k, t\nu_k)\right) + J(q_k)\nu_k \rightarrow \ddot{x}_{\max, k+n} = ?
  \]
Global optimality does not exist

Try to be optimal given the current state of the world and its close future predicted evolution

Model Predictive Control
Model Predictive Control

- Global optimality does not exist
- Try to be optimal given the current state of the world and its close future predicted evolution
- Model Predictive Control

---

**Reactive Control**

**Predictive Control**

Instantaneous action

Previewed actions
The key ingredient to planning and model predictive control is ...

- ... a very good estimation of your motor capabilities in task space
- Complex : state dependant, polytopes
- MPC based motion replanning with state dependant robot capabilities
- PhD of Nicolas Torres (Cifre PSA) and Antun Skuric (Lichie Airbus)

[Skuric 2021] [Pickard 2021] [Skuric 2022]
Conclusions

- Global optimality does not exist a priori
  - Solving accurately for the full control trajectory does not make sense ...
  - ... and is hardly doable in a closed-loop way
- Closed-loop local planning at high level with low-level capabilities in mind
- Solve reactively at low-level with constraints

- Human modeling is a key prerequisite to human-robot interaction
  - Ergonomics [Maurice 2017], Motor variability [Savin 2020]
  - Physical capabilities [Benhabib 2020], [Skuric 2022]
  - Cognitive abilities (e.g. expertise) and biases
– Thank you for your attention –
Optimization-based control approaches to humanoid balancing.

L. Loher, O. Sigaud and V. Padois.

IEEE-RAS International Conference on Humanoid Robotics, Cancun, Mexico, November 2016.

References


A. Banet, P. Bidado and V. Padois.

Optimization-based control approaches to humanoid balancing.

L. Loher, O. Sigaud and V. Padois.

IEEE-RAS International Conference on Humanoid Robotics, Cancun, Mexico, November 2016.

B. Lober, O. Sigaud and V. Padois.


N. Mansard, O. Khalid and A. Kheddar.


Pauline Maurice, Vincent Padois, Yvan Measson and Philippe Bidaud.


O. Khalid.


O. Khalid.


A. Liégeois.


M. Liu, Y. Tan and V. Padois.


R. Lober, V. Padois and O. Sigaud.

Efficient Reinforcement Learning for Humanoid Whole-Body Control.

Proceedings of the IEEE-RAS International Conference on Humanoid Robots, Cancun, Mexico, November 2016.

S. Rubrecht, V. Padois, Philipp Bidaud, Michel de Broissia and Max Da Silva Simoes.

Motion safety and constraints compatibility for multibody robots.


J. Salini, V. Padois and P. Bidaud.

Synthesis of Complex Humanoid Whole-Body Behavior: a Focus on Sequencing and Tasks Transitions.


Movement Variability and Digital Human Models: Development of a Demonstrator Taking the Effects of Muscular Fatigue into Account.


Jonathan Savin.

Simulation of the variability of movement induced by the fatiguemusculaire par la contrainte ergonomique de postes de travail.


Evolution of movement variability patterns during a repetitive pointing task until exhaustion.


Antun Skurir, Vincent Padois and David Daney.


Antun Skurir, Vincent Padois, Nasser Rezoug and David Daney.


Divya Srinivasan and Swad Erik Mathiassen.

Motor variability in occupational health and performance.


P.-B. Wisser.

Viability and predictive control for safe locomotion.


Anis Megumu.

Safe control of robotic manipulators in dynamic contexts.


A. Megumi.


Anis Megumi.


Joshua K. Pickard, Milan Hladík, Vincent Padois and David Daney.

Efficient Sat-Based Approaches for the Computation of Robot Capabilities.

Submitted to IEEE TRO, 2021.

S. Rubrecht, V. Padois, Philippe Bidaud, Michel de Broissia and Max Da Silva Simoes.

Motion safety and constraints compatibility for multibody robots.