



Control and design of robots with tasks and constraints in mind

Vincent Padois

► To cite this version:

Vincent Padois. Control and design of robots with tasks and constraints in mind. Robotics [cs.RO].
Université Pierre et Marie Curie (Paris 6), 2016. tel-01398868

HAL Id: tel-01398868

<https://hal.science/tel-01398868>

Submitted on 17 Nov 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Habilitation à Diriger des Recherches

présentée devant

l'Université Pierre et Marie Curie (UPMC)

par

Vincent PADOIS

Laboratoire	Institut des Systèmes Intelligents et de Robotique - UMR UPMC/CNRS 7222
Ecole doctorale	Sciences Mécaniques, Acoustique, Électronique et Robotique de Paris

Control and design of robots with tasks and constraints in mind

Commande et conception de robots dans une optique orientée tâches et contraintes

Soutenue le 17 octobre 2016 devant le jury composé de:

M. Philippe FRAISSE	Professeur à l'Université Montpellier 2	Rapporteur
M. Philippe SOUÈRES	Directeur de Recherche au LAAS-CNRS	Rapporteur
M. Luigi VILLANI	Professeur à l'Università di Napoli Federico II	Rapporteur
M. Etienne DOMBRE	Directeur de Recherche au LIRMM	Examineur
M. Guillaume MOREL	Professeur à l'Université Pierre et Marie Curie	Examineur
M. Philippe BIDAUD	Professeur à l'Université Pierre et Marie Curie	Référent HDR

A mes parents qui ont fait de moi un humain fréquentable.
A Soizic qui a fait de moi un Homme sage.
A Léo qui a fait de moi une sage-femme.
A Karine qui patiemment m'attend.

“Tous les hommes ont un cancer qui les ronge, un excrément
quotidien, un mal récurrent : leur insatisfaction; le point de rencontre
entre leur être réel, squelettique, et l'infinie complexité de la vie.”

– Cesare PAVESE, *Le métier de vivre* (1952)

Remerciements

Le travail du chercheur est un travail souvent solitaire de part la réflexion personnelle incessante (et parfois intempestive) qu'il engendre. C'est aussi un travail très collectif qui repose pour beaucoup sur l'enthousiasme et l'investissement de nombreux collaborateurs. Je profite de ces quelques lignes pour évoquer quelques catégories de personnes ayant (eu) un impact positif non négligeable sur mon travail. Pour autant, je ne nomme personne par peur d'en oublier et par volonté de ne pas faire, malgré moi, un classement qui n'a pas lieu d'être. En outre, la majorité des personnes avec qui j'ai (eu) la chance de travailler n'a pas besoin de voir son nom explicitement écrit pour savoir ce qui nous lie.

Au premier rang de ces collaborateurs, je remercie les stagiaires, thésards et postdocs avec lesquels j'ai (eu) la chance de travailler et que j'ai (eu) plaisir à encadrer. Ils sont clairement les contributeurs majoritaires des travaux présentés dans ce manuscrit. J'espère que le plaisir est partagé, tant d'un point de vue scientifique qu'humain.

Mes collègues à l'ISIR (et précédemment au Stanford AI Lab, au LAAS et au LGP) constituent aussi un élément important du terreau fertile sur lequel je cultive mes idées. Je les remercie tous et j'espère jouer moi aussi ce rôle de terreau pour eux. Ma gratitude est aussi grande pour mes encadrants puis responsables d'équipe et directeurs de laboratoire successifs qui, au-delà du terreau, m'ont fourni l'essentiel : les outils, le lopin de terre et parfois même les graines à l'origine de ces idées.

Les projets auxquels j'ai (eu) la chance de participer sont aussi l'opportunité de rencontrer des chercheurs brillants avec lesquels les échanges d'idées sont souvent enrichissants et fructueux. Merci à eux.

Les membres du jury de cette HDR ont effectué un travail d'évaluation de mes travaux dont la profondeur est illustrée par la nature des échanges que nous avons pu avoir lors de ma soutenance. Je les remercie donc d'avoir accepté cet investissement ainsi que pour l'attitude bienveillante et ouverte que je leur ai toujours connue. Ils constituent à ce titre des exemples pour moi.

Finalement, même si mille mercis ne suffiraient pas pour dire ce que je lui dois, j'en ajoute un pour ne pas qu'on oublie celle qui me supporte sans jamais faillir.

Paris, le 25 octobre 2016.

Contents

Introduction	1
1 Constraints compatible, multi-tasks and reactive robot control	5
1.1 Robot control as an optimisation problem	7
1.1.1 Linear systems inversion and minimum norm solutions	8
1.1.2 Linear systems inversion under constraints	10
1.1.3 Dealing with constraints, the hard way	11
1.2 Dealing with multiple tasks and constraints	13
1.2.1 Contribution	13
1.2.2 Dealing with multiple tasks and constraints: going further	13
1.3 Constraints compatibility	14
1.3.1 Motion safety and constraints compatibility for multi-body robots	14
1.3.2 An energetic approach to safety	15
1.3.3 Constraints compatibility: going further	16
2 Optimisation-based control for balanced humanoid behaviours	19
2.1 A look at the dynamics equations	21
2.1.1 Lagrangian equations of motion	21
2.1.2 Newton-Euler equations of motion	22
2.1.3 Contact mechanics	23
2.2 Optimal control approach to balance	24
2.2.1 Balance?	24
2.2.2 General control problem	24
2.2.3 The balance problem	25
2.2.4 Challenges	26
2.2.5 Various approaches to the balance control problem	27
2.2.6 Balance indicators and criteria	28
2.3 Optimising balance at the planning level	29
2.3.1 Stochastic whole-body motion planning	30
2.3.2 Non-stochastic whole-body motion planning	31
2.4 Optimal regulation of balance tasks	31
2.4.1 Model Predictive Control	31
2.4.2 Instantaneous formulations	34
2.5 Optimisation-based whole-body control	34
2.5.1 Local optimality	34
2.5.2 Whole-body MPC	37
2.6 Conclusion	38

3	Performance based design and evaluation of robots	39
3.1	Context	40
3.2	Method	44
3.2.1	Ergonomic indicators for collaborative robotics	44
3.2.2	Simulation of co-manipulation activities	47
3.2.3	Sensitivity analysis of the ergonomic performance	49
3.2.4	Evolutionary design of a collaborative robot	52
3.3	Results	53
3.3.1	Acquisition of the initial situation	53
3.3.2	Sensitivity analysis	55
3.3.3	Evolutionary design of a robot morphology	57
3.4	Discussion and potential improvements	60
3.4.1	Limitations of the ergonomic indicators	60
3.4.2	Limitations of the optimisation	61
3.4.3	Limitations of the DHM	61
	Conclusions and Perspectives	63
	Bibliography	67
A	Linear system inversion	79
A.1	Weighted Euclidean norm solutions	79
A.2	Regularised, Weighted Euclidean norm solutions	80
A.3	Linear systems inversion under constraints	81
B	Selected publications	83
B.1	Generalised Hierarchical Control	84
B.2	Motion safety and constraints compatibility for multi-body robots	100
B.3	Control of robots sharing their workspace with humans: an energetic approach to safety . . .	118
B.4	Emergence of humanoid walking behaviours from mixed-integer model predictive control . . .	126
B.5	Variance modulated task prioritization in whole-body control	135

Introduction

A quick foreword

When talking about Robotics, there is always the temptation to resort to a regrettably misleading semantic shift in order to catch the attention of the readers. This shift is simple. It consists in talking about our collective imagination view of robots and their extraordinary, mysterious capabilities instead of mentioning what robots actually are. This shift is illustrated by Fig 1. It is symptomatic of many questionable practices and actually contributes to create an undesirable gap between reality and a phantasmatic version of it. This document is of course not focused on this shift. I just hope that what comes next stays on, what I consider to be, the right side of the line.

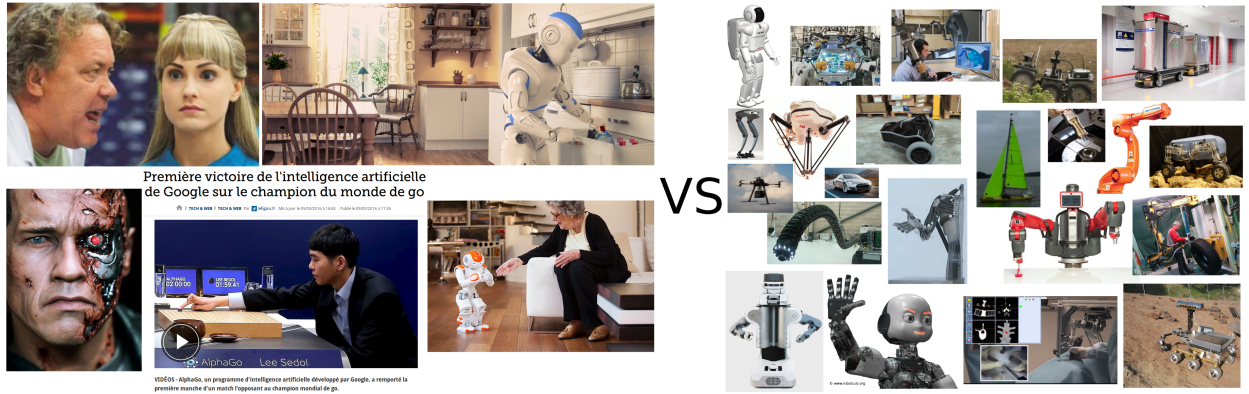


Figure 1: The regrettably misleading semantic shift between our collective imagination view of robots and what they actually are.

Description of the contribution

The work presented in this dissertation is mostly concerned with the problem of controlling robots. While a large part of the scientific literature in Robotics is dedicated to this problem, there still exists a gap between what has been proposed using advanced control techniques and the majority of existing applications. Indeed, in simulation or in lab conditions, *ad hoc* environment and situations can be generated in order to simplify the control problem and only address one of its sub-parts. This can be necessary in the preliminary stages of research, when trying to address a challenging problem. However, the risk is to provide solutions which are intrinsically incompatible with real life constraints. These constraints often constitute strong non-linearities which have to be accounted for to maintain the robot and its surrounding environment in proper working

conditions. Nevertheless, these limits have often been dealt with as exceptions or secondary objectives. If these approaches make sense in cases where all motions can be pre-planned, they do not apply to situations where motions have to be generated in reaction to the environment. In that sense, there is still a need for advanced control techniques that intrinsically account for real world constraints. These constraints both lie in the actuation and configuration spaces of the robot but also in the tasks spaces, *i.e.* in spaces where the tasks assigned to the robot can be expressed in a straightforward way by the programmer or by the user through interactive programming. Thus, robotics control paradigms have to provide the means to naturally express and optimally perform concurrent tasks while natively accounting for constraints in a computationally efficient way.

In an attempt to tackle this general problem, the first research direction presented in this dissertation describes some work performed in constraints compatible, multi-tasks robot control. The proposed contribution in that domain leads to the formulation of the robotics control problem as a constrained optimisation one. This formulation provides the benefit of computing locally optimal solutions intrinsically compatible with the constraints. This is particularly suitable for applications in constrained and dynamic environments.

In order to deal with robots or applications where the achievement of a complex activity requires the simultaneous realisation of several tasks, a novel hierarchical, multi-tasks control paradigm is also presented. The novelty of this paradigm lies in its capability to describe strict (null-space projection based) and soft (weights based) task hierarchies and switch from one to the other in a continuous way. Experiments on standard manipulators as well as humanoid robots, both in the quasi-static and dynamic cases, provide illustrations of the properties of this hierarchical controller.

Constraints compatibility also raises a fundamental question which is often overlooked in the literature: can we guarantee that a control solution always exist given the current state of the robot and the various constraints it is facing? This is far from obvious as it requires a prediction of the future evolution of the robot which is not always possible in open-ended robotics applications. Some contributions are presented to solve this problem. They are based on a thorough analysis of the constraints and a modification of their expressions to render them compatible one with another. Some theoretical and experimental results support these developments.

To extend this work to environments where physical interactions between the robot and its surroundings cannot be avoided, obstacle avoidance constraints expressed at the geometric level are replaced by energetic constraints. These energetic constraints are defined such that the kinetic energy to be dissipated by the robot in case of an expected or unexpected contact with the environment is bounded to avoid damage. Using vision sensors, the distance between moving objects in the environment and the robot can be estimated and fed to the controller at all time. This information is used to modulate the bound on the maximum energetic level, permitting to fully exploit the dynamics of the robot when obstacles are far and to gradually reduce it when they get closer. This energetic approach also encapsulates the cases where the robot is working in contact with the environment and allows to bound the storage of potential elastic energy in the robot/environment system. This paradigm is particularly suitable for Human-Robot workspace sharing and physical interaction applications.

While the first chapter advocates for the formulation of the reactive control problem as an optimisation one, the limits of reactive approaches are quite tangible when dealing with the constraints compatibility problem. Global optimality has to be tackled despite its complexity. This complexity reaches its climax with humanoid robots and the second contribution presented in this dissertation is centred on optimisation-based control for balanced humanoid behaviours. Balance is a complex notion and accounting for it in a general and efficient way at the control level is complex. Part of this complexity lies in the high dimensionality, non-linearity, under-actuation and hybrid nature of humanoid systems. These characteristics make it difficult to define computationally-friendly models of balance. This is all the more true that the impact of some given control action on balance cannot necessarily be anticipated for in a reactive way and requires its consequences to be previewed over a time horizon to conclude on its compatibility with the general balance

objective.

In a first section, the dynamics of free-floating systems are recalled and the non-linearities raised by contact mechanics are highlighted. Then, a general formulation of optimisation-based approaches to the balance control problem is proposed to expose the computational challenges it induces. Levers classically employed to tackle these challenges are then identified. The prevalent balance indicators and metrics, needed to describe control objectives and constraints, are therefore presented.

An overview of optimisation-based control approaches is finally proposed. These approaches are organised with respect to the level of abstraction at which they are employed to solve the balance control problem. They range from reactive controller for whole-body control to pure motion planning approaches but the most promising approaches are based on a Model Predictive Control formulation of the control problem, trying to achieve a compromise between optimality and reactivity.

Control is the central focus of the presented work. Nonetheless, designing the right robot for a given application is also a very interesting and important research topic if one envisions less traditional use of robots with respect to standard manufacturing applications. Thus, the final contribution presented in this document is related to the performance based design and evaluation of robots. While efficient control laws are necessary for robots to evolve in complex environments, their intrinsic morphological and inertial properties also have a great influence on their performance. Thus, optimising these properties is of interest and even primordial in highly constrained environments. In a “classical” design process, the retained solution is often relying on the strong expertise of the designer as well as on an iterative development process where several physical prototypes can be necessary. This may lead to suboptimal solutions as it can be difficult for the designer to foresee the performance of a given robot in novel robotics applications. It is also time consuming and costly.

Alternatively, the evaluation of the performances of a given robot architecture can be performed in a physically realist simulation environment. Using the generic control approach developed in this work, this allows to test in a systematic way a large variety of architectures in various contexts. By coupling this systematic evaluation to an optimisation process, here multi-objective evolutionary algorithms, families of solution can emerge and provide innovative initial solutions for a more classical design process.

In some applications, the performance is not solely evaluated in terms of achievement of the tasks. This is especially the case of collaborative robots where one dimension of the performance evaluation is related to ergonomics. Indeed, these robots often appear as a mean to relieve the worker from strong biomechanical demands and it is therefore important to account for it at the primary stage of design. Summarising the ergonomic performance of a robot is however complex as the evaluation of the biomechanical demands of a task on a human operator potentially relies on tenths of indicators. Hence, a methodology for analysing the sensitivity of the various indicators to the robot and task parameters is proposed. The goal of such an analysis is to reduce the number of ergonomic indicators which are considered in an evaluation, while sufficiently accounting for the global ergonomic level of the considered activity. Based on this analysis, morphological optimisation can then be performed. The example of the design of a manipulator for assisting drilling tasks is provided in details.

The contributions presented through these three chapters leverage new research questions which are discussed in conclusion.

Finally, this dissertation also contains two mandatory appendices which are required by the stylistic exercise of the *Habilitation à Diriger des Recherches*. The first one contains five selected publications. These publications are used and referred to in the different chapters as they can directly be mapped to some of the proposed contributions. The second mandatory appendix is an extended Curriculum Vitae which covers my work as an *enseignant-chercheur*.

Chapter 1

Constraints compatible, multi-tasks and reactive robot control

What is the control problem in Robotics? This is an opened question as objectives prescribed to robots can potentially be described in different spaces of interest – being them user-centred, 3-D world-centred, sensor-centred, joint-centred or actuator-centred – and defined at different time scales ranging from the robot life time to the millisecond.

While this question admits a large number of answers, one element is common to all these potential descriptions of the control problem: the constraints imposed by the laws of physics. These laws are classically summarised by the equations of motion of the robot. These equations relate the external and actuation forces acting on the robot to its acceleration. They can be written in the very general form

$$\mathbf{M}(\mathbf{q})\dot{\boldsymbol{\nu}} + \mathbf{n}(\mathbf{q}, \boldsymbol{\nu}) + \mathbf{g}(\mathbf{q}) + \boldsymbol{\varepsilon}(\mathbf{q}, \boldsymbol{\nu}, \dots) = \mathbf{S}^T(\mathbf{q})\boldsymbol{\tau} - \sum_i \mathbf{J}_{c_i}^T(\mathbf{q})\mathbf{w}_{c_i} \quad (1.1)$$

where $\mathbf{q}$ is the vector of generalised coordinates describing the configuration of the system and $\boldsymbol{\nu}, \dot{\boldsymbol{\nu}}$ are the n generalised velocities and accelerations respectively.¹ $\boldsymbol{\tau} \in \mathbb{R}^{n_a}$ is the actuation torque/force vector² with n_a the number of actuators and $\mathbf{S}(\mathbf{q})$ the transmission matrix accounting for the arrangements of the actuators with respect to the joints. $\mathbf{J}_{c_i}(\mathbf{q}) \in \mathbb{R}^{6 \times n}$ and $\mathbf{w}_{c_i} \in \mathbb{R}^6$ are the Jacobian matrix and the contact wrench associated to the i^{th} contact point, among n_c , of the robot with its environment respectively. $\mathbf{M}(\mathbf{q})$ is the generalised mass matrix. $\mathbf{n}(\mathbf{q}, \boldsymbol{\nu})$ and $\mathbf{g}(\mathbf{q})$ are the vector of joint torques induced by Coriolis-centrifugal and gravity effects respectively. $\boldsymbol{\varepsilon}(\mathbf{q}, \boldsymbol{\nu}, \dots)$ is the vector of joint torques associated to other physical effects such as dry and viscous friction.

Even though very general and complete, Eq. (1.1) is not sufficient to describe the overall dynamics and physical behaviour of a robot. Indeed, other intrinsic physical constraint have to be accounted for at the joint level³:

- joint position limits

$$\mathbf{q}_{min} \leq \mathbf{q} \leq \mathbf{q}_{max}, \quad (1.2)$$

¹For non-fixed based systems, some components of $\mathbf{q}$ describe the orientation (using a quaternion, the axis/angle representation or a rotation matrix) of the root body in the inertial frame of reference and their derivatives do not correspond to the angular velocity of the root body. This explains the introduction of the notation $\boldsymbol{\nu} \neq \frac{d\mathbf{q}}{dt}$.

²For the sake of clarity, and without loss of generality, the term “torque” is used in the remainder of this document.

³These limitations are conveniently written at the joint level but velocity and torque limits are mostly actuator related and should, as such, be specified at the actuator level when the transmission matrix $\mathbf{S}$ is configuration dependant. Also, the actuators dynamics is not accounted for in this description. This is equivalent to assuming that the actuators bandwidth is large enough to consider them as perfect torque source with saturations. While this is true for the most currently used actuators in Robotics, it does not hold for less conventional actuation means such as pneumatic ones.

- joint velocity limits

$$\boldsymbol{\nu}_{min} \leq \boldsymbol{\nu} \leq \boldsymbol{\nu}_{max}, \quad (1.3)$$

- joint torques limits

$$\boldsymbol{\tau}_{min} \leq \boldsymbol{\tau} \leq \boldsymbol{\tau}_{max}. \quad (1.4)$$

The impossibility for a body i of the robot to inter-penetrate any other surrounding body j has also to be considered and can be simply written

$$0 \leq d_{i,j}, \quad (1.5)$$

where $d_{i,j}$ is the shortest euclidean distance between bodies i and j . Considering all potential collision pairs, the associated distances can be gathered in a vector $\mathbf{d}$.

Strictly speaking, Eq. (1.1), (1.2), (1.3), (1.4) and (1.5) are sufficient to describe the possible motions of the system and their strict physical limitations. In practice, more conservative limitations are often considered in order to maintain the robot in a state⁴ such that control is eased. For example, rolling without slipping or contact existence can be enforced through constraints (assuming a good enough knowledge of the point kinematics and contact surface behaviour): the robot can slip or break contact, potentially without damage, but this is not necessarily convenient from a control point of view.

Overall all these constraints, being them physical or convenience constraints, can be written as (in)equalities of the very general form

$$\mathbf{c}(\boldsymbol{\sigma}) \leq \mathbf{h} \quad (1.6)$$

where $\mathbf{h}$ is a time-varying vector but considered constant over a given time step while $\mathbf{c}$ is a function of the extended-state

$$\boldsymbol{\sigma} = [\mathbf{q}^T, \boldsymbol{\nu}^T, \dot{\boldsymbol{\nu}}^T, \mathbf{w}_{c_1}^T, \dots, \mathbf{w}_{c_{n_c}}^T, \boldsymbol{\tau}^T, \mathbf{d}^T]^T. \quad (1.7)$$

The extend-state can be defined as the set of variables required to fully define the state of the system with respect to its environment and its physical limits. This notion is first introduced in [\[Rubrecht 2012\]](#)⁵ with a fairly similar meaning.

Eventually, the “simplest” control problem description in the joint space can be written: find a joint space trajectory such that there exists a finite horizon of actuation torque inputs $\boldsymbol{\tau}(t)$ allowing the robot to reach a given goal state $(\mathbf{q}_{goal}, \boldsymbol{\nu}_{goal})$ from an initial state $(\mathbf{q}_{init}, \boldsymbol{\nu}_{init})$, under the constraints defined by Eq. (1.6) and with minimum time or energy consumption. Stated this way, one could argue that this describes a trajectory generation problem rather than a control one. However, pre-computing and applying a full horizon of inputs to the actuators never make sense in practice as, even minor, uncertainties in the knowledge of the systems dynamics require the use of a feedback in order to reject disturbances induced by modelling errors. The actual control input can thus be seen as the sum of a term related to the achievement of the desired trajectory, which could potentially be pre-computed, and of a reactive term related to the feedback necessary for good tracking performances and robustness reasons. This simple example illustrates the fact that the control problem both relates to the generation of feasible trajectories and the reactive computation of a control input in order to robustly track the desired, feasible trajectory.

From the point of view of standard industrial applications, this control problem can be considered as solved. Indeed, these applications assume a perfectly known environment and repetitive tasks performed by very stiff robots. It is thus possible to compute offline trajectories that are both feasible, *i.e.* such that it can be guaranteed that the robot meets the physical constraints it has to comply with over an infinite horizon of time, and efficient, *i.e.* performing the prescribed actions with accuracy and in an optimal way

⁴Here the word “state” is to be understood in its most general sense.

⁵Underlined citations correspond to articles that I contributed to or to PhD thesis that I have advised.

time or energy wise. There is not a unique way to generate such trajectories and each robot maker has its own, usually not publicly disclosed, techniques to produce them. Generally speaking, “offline trajectory generation and motion planning” can be used as a general term to describe these techniques. Product Life Management (PLM) software tools also integrate such trajectory generators, relying on state-of-the-art planning [LaValle 2006] and trajectory generation algorithms [Kroger 2010].

Unfortunately, the following limiting factors do not allow to extend this approach to other robotic contexts:

- imperfectly known and dynamic environments forbids approaches where planning is performed offline, once and for all. Assuming that the necessary perception means are available to update the knowledge of the extend-state, online replanning is an option but is too computationally expensive to be applicable in real-time in complex environments.
- The control problem is often easier to express in the so-called task or operational space and several task spaces may co-exist to fully describe the operations to be performed by the robot. The forward model(s) relating the joint space to the task space(s) being generally non linear, this adds up to the overall computational complexity.
- A desired trajectory may not be available a priori but fed to the robot reactively based on sensor or user real-time inputs. Planning over a long horizon simply does not apply in these cases.
- Highly stiff robot behaviours may not always be suitable, especially when in contact with the environment. Deciding for the proper robot impedance depending on the context is also a difficulty which adds up to the overall computational complexity.

The control literature in Robotics has been focusing on these questions, trying to address them in the most general and suitable ways. Nonetheless, one of the historical pitfall at the control level is the use of analytical approaches for inverse velocity kinematics like problems. The first section of this chapter is an attempt to explain why the corresponding optimisation problem should explicitly be formulated instead. The second section introduces a general formalism to describe and solve hierarchical, multi-tasks control problems. This approach contrasts with strict and soft hierarchical approaches and, while imperfect, sheds new light on tasks hierarchies related questions. Finally, the constraints compatibility problem is introduced. This problem boils down to the capability to ensure that a control solution always exist. Some contributions in that domain are presented as an attempt to solve this, often underestimated, problem.

1.1 Robot control as an optimisation problem

The operations to be performed by a robot are conveniently described in the so-called task space. For a serial, fixed-base manipulator, the spaces associated to end-effector motions and forces are “natural” task spaces of choice. In the “early” Robotics control literature, the assumption was made that a feasible task-space trajectory was provided by “*either a human supervisor-operator or some sort of “artificial intelligence” or both operating cooperatively*” [Liégeois 1977] or that the end-effector of the manipulator was operated remotely [Whitney 1969], no trajectory being available in advance. In the latter case, the control problem is purely reactive and consists in computing the control input given some desired pose/twist/acceleration/wrench expressed in the task space.

Reasoning for example and without loss of generality at the velocity level, this problem explicitly consists in reactively computing⁶, *i.e.* at each control instant, the joint space velocity given some desired task space twist $\mathbf{t}^*$

$$\text{find } \boldsymbol{\nu} \text{ such that } \mathbf{t}^* = \mathbf{J}(\mathbf{q})\boldsymbol{\nu}. \quad (1.8)$$

⁶ $\mathbf{J}$ is supposed to be known given a measure of the configuration $\mathbf{q}$.

Constraints described by Equation (1.6) may actually prevent to actually achieve the desired twist $\mathbf{t}^*$. Thus, one more generally seeks for solutions that minimise some norm of the task-space error vector $(\mathbf{J}(\mathbf{q})\boldsymbol{\nu} - \mathbf{t}^*)$ while also minimising some norm of the solution $\boldsymbol{\nu}_{sol}$.

1.1.1 Linear systems inversion and minimum norm solutions

The inverse velocity kinematics problem is a particular case of inversion of linear systems of the type $\mathbf{Ax} = \mathbf{y}$ where $\mathbf{A} \in \mathbb{R}^{n_y \times n_x}$, with n_x and n_y the respective dimensions of vectors $\mathbf{x}$ and $\mathbf{y}$. Finding a solution to this problem boils down to the general optimisation problem of finding a minimum norm solution $\mathbf{x}_{sol}$ such that some norm of the error vector $(\mathbf{Ax}_{sol} - \mathbf{y})$ is also minimal. This is far from trivial. Indeed, in the most general case $\mathbf{A}$ is

- not a square matrix;
- neither full-row nor full-column rank ($\text{rank}(\mathbf{A}) = n_r$);
- ill-conditioned.

Weighted Euclidean norm solutions The solution to the inversion problem should be general enough to deal with both rank and conditioning issues. Using weighted Euclidean norms⁷ for both the error and the optimisation variable, A. Ben-Israel and T. Greville [Ben Israel 2003] (chapter. 3, section 5) write this as a cascade of two optimisation problems

$$\mathbf{x}_{sol} = \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{x}\|_{\mathbf{W}_x}^2 \quad (1.9a)$$

$$\text{s.t. } \mathbf{Cx} = \arg \min_{\mathbf{b}} \|\mathbf{Fb} - \mathbf{y}\|_{\mathbf{W}_y}^2 \quad (1.9b)$$

where:

- $\mathbf{W}_x \in \mathbb{R}^{n_x \times n_x}$ and $\mathbf{W}_y \in \mathbb{R}^{n_y \times n_y}$ are symmetric, positive definite matrices;
- $\mathbf{A} = \mathbf{FC}$ is a full rank decomposition of $\mathbf{A}$;
- $\mathbf{F} \in \mathbb{R}^{n_y \times n_r}$ and $\mathbf{C} \in \mathbb{R}^{n_r \times n_x}$ are respectively full-column rank and full-row rank;
- $\mathbf{b} = \mathbf{Cx}$.

It can be shown (see Appendix A) that the solution to this problem is given by

$$\mathbf{x}_{sol} = \mathbf{A}_{\mathbf{W}_x, \mathbf{W}_y}^+ \mathbf{y} \quad (1.10)$$

where

$$\mathbf{A}_{\mathbf{W}_x, \mathbf{W}_y}^+ = \mathbf{W}_x^{-1} \mathbf{C}^T \left(\mathbf{C} \mathbf{W}_x^{-1} \mathbf{C}^T \right)^{-1} \left(\mathbf{F}^T \mathbf{W}_y \mathbf{F} \right)^{-1} \mathbf{F}^T \mathbf{W}_y \quad (1.11)$$

is called the weighted pseudo-inverse of $\mathbf{A}$. It provides the minimum $\mathbf{W}_x$ -weighted euclidean norm (or least-squares) solution to the problem $\mathbf{y} = \mathbf{Ax}$ and is such that the $\mathbf{W}_y$ -weighted euclidean norm of the error $\|\mathbf{y} - \mathbf{Ax}_{sol}\|_{\mathbf{W}_y}$ is minimal.

⁷While the Euclidean norm offers in Robotics a good compromise between the variety of solutions it can lead to through the use of weighting matrices and the computational complexity it requires, alternative solutions exist. The recent work of V. M. Gonçalves *et al.* [Gonçalves 2016] proposes, for example, an efficient Linear Programming solution leading to L_0 -norm like solutions, i.e. solutions maximising the number of zero components in the solution vector.

Regularised, Weighted Euclidean norm solutions The previous solution does not handle the case where $\mathbf{A}$ is ill-conditioned which can be highly critical numerically speaking. To improve this result, one needs to introduce a particular decomposition of $\mathbf{A}$. This decomposition can be obtained based on the singular value decomposition (SVD)

$$\mathbf{A} = \mathbf{U} \mathbf{D} \mathbf{V} \quad (1.12)$$

where $\mathbf{U} \in \mathbb{R}^{n_y \times n_y}$ and $\mathbf{V} \in \mathbb{R}^{n_x \times n_x}$ are orthogonal and $\mathbf{D} \in \mathbb{R}^{n_y \times n_x}$ has the following structure

$$\mathbf{D} = \begin{bmatrix} \begin{bmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_r \end{bmatrix} & \mathbf{0}_{n_r \times (n_x - n_r)} \\ \mathbf{0}_{(n_y - n_r) \times n_r} & \mathbf{0}_{(n_y - n_r) \times (n_x - n_r)} \end{bmatrix}$$

$\underbrace{\hspace{10em}}_{\Sigma}$

with $\sigma_1, \dots, \sigma_r$ the singular values of $\mathbf{A}$. Based on (1.12), $\mathbf{A}$ can be written

$$\mathbf{A} = \mathbf{U} \underbrace{\begin{bmatrix} \mathbf{I}_{n_r} \\ \mathbf{0}_{(n_y - n_r) \times n_r} \end{bmatrix}}_{\mathbf{U}'} \underbrace{\Sigma \begin{bmatrix} \mathbf{I}_{n_r} & \mathbf{0}_{n_r \times (n_x - n_r)} \end{bmatrix}}_{\mathbf{V}'} \mathbf{V}. \quad (1.13)$$

Similarly to $\mathbf{F}$ and $\mathbf{C}$, $\mathbf{U}'$ and $\mathbf{V}'$ are respectively full-row and full column rank. However, the interesting feature of decomposition (1.13) lies in the explicit access to Σ . Indeed, the condition number of $\mathbf{A}$ only depends on Σ . Based on this decomposition, the overall optimisation problem can be defined as a cascade of three optimisation problems:

1. The first one aims at finding the optimal combination of the columns of $\mathbf{U}'$ such that the $\mathbf{W}_y$ -weighted Euclidean norm of the vector $(\mathbf{U}'\mathbf{c} - \mathbf{y})$ is minimal, with $\mathbf{c} = \Sigma\mathbf{V}'\mathbf{x}$.
2. Given $\mathbf{c}_{sol}$, the second sub-problem aims at finding the optimal combination of the columns of Σ such that the Euclidean norm of the vector $(\Sigma\mathbf{b} - \mathbf{c}_{sol})$ is minimal, with $\mathbf{b} = \mathbf{V}'\mathbf{x}$. Σ being full rank, this second problem admits a unique solution. However, to handle the potential conditioning problem, the inversion should be regularised. This can be achieved through the so-called damped least-squares or Thikhonov regularisation technique.
3. Given $\mathbf{b}_{sol}$, the last sub-problem aims finding the minimal $\mathbf{W}_x$ -weighted Euclidean norm solution $\mathbf{x}_{sol}$ among the set of all $\mathbf{x}$ such that $\mathbf{V}'\mathbf{x} = \mathbf{b}_{sol}$.

The three steps problem can be written

$$\mathbf{x}_{sol} = \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{x}\|_{\mathbf{W}_x}^2 \quad (1.14a)$$

$$\text{s.t. } \mathbf{V}'\mathbf{x} = \arg \min_{\mathbf{b}} \frac{1}{2} \left(\left\| \underbrace{\Sigma\mathbf{b} - \arg \min_{\mathbf{c}} \frac{1}{2} \|\mathbf{U}'\mathbf{c} - \mathbf{y}\|_{\mathbf{W}_y}^2}_{\mathbf{c}_{sol}} \right\|^2 + \|\mathbf{b}\|_{\Lambda}^2 \right) \quad (1.14b)$$

$\underbrace{\hspace{10em}}_{\mathbf{b}_{sol}}$

where $\Lambda \in \mathbb{R}^{n_r \times n_r}$ is a symmetric, positive definite regularisation matrix. It can be shown (see Appendix A) that the solution to this problem is given by

$$\mathbf{x}_{sol} = \mathbf{A}_{\mathbf{W}_x, \mathbf{W}_y, \Lambda}^+ \mathbf{y} \quad (1.15)$$

where

$$\mathbf{A}_{\mathbf{W}_x, \mathbf{W}_y, \Lambda}^+ = \mathbf{W}_x^{-1} \mathbf{V}'^T \left(\mathbf{V}' \mathbf{W}_x^{-1} \mathbf{V}'^T \right)^{-1} \left(\Sigma^T \Sigma + \Lambda \right)^{-1} \Sigma^T \left(\mathbf{U}'^T \mathbf{W}_y \mathbf{U}' \right)^{-1} \mathbf{U}'^T \mathbf{W}_y \quad (1.16)$$

is called the regularised, weighted pseudo-inverse of $\mathbf{A}$. An implementation of this type of solution for inverse velocity kinematics [Padois 2008] is present since 2008 in the Kinematics and Dynamics Library (KDL) of the OROCOS software [Bruyninckx 2003].

The solution to the inverse velocity kinematics problem (1.8) can thus be computed as

$$\boldsymbol{\nu}_{sol} = \mathbf{J}_{\mathbf{W}_t, \mathbf{W}_j, \Lambda}^+(\mathbf{q}) \mathbf{t}^*. \quad (1.17)$$

This solution exhibits interesting features in terms of norm minimisation and robustness to singularities but does not account for the inequality constraints described by Eq. (1.6). In practice, these constraints would have to be imposed *a posteriori* through various saturations. This is obviously suboptimal and cannot always be achieved as access to the low level control layers of robots is not always possible. Hence there is a need to account for constraints at the earliest stage of the computation of the control solution and solution (1.17) should not be used in practice. Instead, a proper optimisation problem should be formulated.

1.1.2 Linear systems inversion under constraints

Assuming:

- a convex optimisation problem with a quadratic cost function and linear constraints of the form

$$\mathbf{C}_{equ} \mathbf{x} = \mathbf{c}_{equ}, \quad (1.18a)$$

$$\mathbf{C}_{ine} \mathbf{x} \leq \mathbf{c}_{ine}; \quad (1.18b)$$

- a feasible problem⁸, *i.e* one that accepts at least one solution;

the optimal solution is such that $\mathbf{x}_{sol}$ necessarily lies at the intersection of the hyperplanes defined by the equality constraints (1.18a). However, it may lie on a subset only of the hyperplanes defined by the inequality constraints (1.18b). The corresponding constraints are said to be active and are such that for any given active inequality constraint i

$$\mathbf{c}_{ine,i} \mathbf{x}_{sol} = \mathbf{c}_{ine,i}. \quad (1.19)$$

Given the active set of constraints $\mathbf{C}_a \mathbf{x} = \mathbf{c}_a$ (including both equality and active inequality ones) and writing the basic optimisation problem

$$\mathbf{x}_{sol} = \arg \min_{\mathbf{x}} \frac{1}{2} \left(\|\mathbf{A}\mathbf{x} - \mathbf{y}\|^2 \right) \quad (1.20a)$$

$$\text{s.t. } \mathbf{C}_a \mathbf{x} = \mathbf{c}_a \quad (1.20b)$$

a minimal norm closed-form solution can be computed (see Appendix A) and is given by

$$\mathbf{x} = \mathbf{C}_a^+ \mathbf{c}_a + \mathbf{P}_{\mathbf{N}(\mathbf{C}_a)} (\mathbf{A} \mathbf{P}_{\mathbf{N}(\mathbf{C}_a)})^+ (\mathbf{y} - \mathbf{C}_a \mathbf{C}_a^+ \mathbf{c}_a) \quad (1.21)$$

where $\mathbf{P}_{\mathbf{N}(\mathbf{C}_a)}$ is an orthogonal projector into the kernel of $\mathbf{C}_a$. A regularised, weighted version of this solution can be formulated but is not derived here.

In fine, there is an analytical solution to the constrained optimisation problem. However, determining the active set is not trivial and actually requires an iterative solving process. In fact, although active-set algorithms are well established in the convex optimisation literature [Nocedal 1999] (chapter 16), only recent computational capabilities allow to envision the use of constrained optimisation algorithms at several

⁸Feasibility is ensured if the solution space described by Equation (1.18a) and (1.18b) is non empty. Conditions for feasibility (or constraints compatibility) are further discussed in section 1.3.

kHz frequency⁹. As a consequence, the explicit use of state-of-the-art constrained optimisation algorithms to deal with inverse velocity kinematics and structurally similar problems in Robotics is only recent and unfortunately far from generalised yet. The following section provides a brief overview of the alternative strategies historically developed in Robotics.

1.1.3 Dealing with constraints, the hard way

Constraints as tasks of low priority As a first attempt to explicitly account for constraints at the control level, A. Liégeois proposes in [Liégeois 1977] to make use of the robot redundancy in order to minimise some constraints related potential function in the null-space of the task-related Jacobian. Indeed, when the number of degrees of freedom m_t required to achieve the task is smaller than n_{dof} , the number of independently actuated degrees of freedom of the robot, the system is said to be redundant with respect to the task and the general solution to Eq. (1.8) writes

$$\boldsymbol{\nu}_{sol} = \mathbf{J}^+(\mathbf{q})\mathbf{t}^* + \underbrace{(\mathbf{I}_{n_{dof}} - \mathbf{J}^+(\mathbf{q})\mathbf{J}(\mathbf{q}))}_{\mathbf{P}_J} \mathbf{z}, \quad (1.22)$$

where $\mathbf{P}_J$ is a projector onto the null-space of $\mathbf{J}$ and $\mathbf{z}$ an arbitrary generalised velocity¹⁰. Based on this solution, the inverse velocity kinematics solution for an arbitrary number n_t of hierarchical tasks is exposed in [Siciliano 1991] and is written for any task $i \in 1, \dots, n_t$ associated to the Jacobian $\mathbf{J}_i$ and of desired value $\mathbf{t}_i^*$

$$\boldsymbol{\nu}_i = \boldsymbol{\nu}_{i-1} + (\mathbf{J}_i \mathbf{P}_{i-1})^+ (\mathbf{t}_i^* - \mathbf{J}_i \boldsymbol{\nu}_{i-1}) \quad (1.23)$$

with

$$\boldsymbol{\nu}_1 = \mathbf{J}_1^+(\mathbf{q})\mathbf{t}_1^*, \quad \mathbf{P}_i = (\mathbf{I}_{n_{dof}} - \mathbf{J}_{(i)}^+ \mathbf{J}_{(i)}), \quad \mathbf{J}_{(i)} = \begin{bmatrix} \mathbf{J}_1 \\ \mathbf{J}_2 \\ \vdots \\ \mathbf{J}_i \end{bmatrix}.$$

This solution is prone to algorithmic singularities and a less efficient but robust solution, including Thikhonov regularisation, is proposed in [Chiaverini 1997].

Based on this type of control schemes providing a way to hierarchise and solve the inverse velocity kinematics problem related to several objectives, numerous control architectures have been derived where compliance to constraints is ensured by defining avoidance tasks which are projected, in more or less sophisticated ways, into the null-space of the main task to achieve (*e.g.*, [Liégeois 1977], [Maciejewski 1985], [Seraji 1989], [Mansard 2009a], [Mansard 2009b], [Flacco 2012], see [De Luca 2016] for a recent informal review). Alternatively, the approach in [Baillieul 1985] consists in extending the main task Jacobian using supplementary kinematic constraints in order to “square” the problem to be solved. While this extension does not generate interference with the main task and can trivially be used to solve singularity problems, it does not directly allow the consideration of any secondary tasks.

If constraints-related tasks are projected into the nullspace of manipulation tasks, strict hierarchies are useless¹¹ and soft hierarchies have actually been used, notably in the seminal work of O. Khatib in [Khatib 1986] where tasks and constraints related tasks (in that case repulsive force fields) are considered with the same level of importance. The work in [Shen 2007] considers the use of different weights for each of these tasks, constraints related tasks being then potentially solved with higher (soft) priority. Unfortunately, this alternative approach does not provide any guarantee that the constraints are actually respected.

⁹Several implementations of this method exist but one can notably cite an efficient and widely used implementation developed with control application in mind: *qpOASES* [Ferreau 2014].

¹⁰The dependence to $\mathbf{q}$ is dropped for the sake of clarity

¹¹Unless the rank of all the concatenated Jacobians is less than or equal to the number of degrees of freedom of the robot.

Constraints as tasks of high priority A. Maciejewski and C. Klein [Maciejewski 1985] are the first to propose a control framework where the manipulation task Jacobian is explicitly projected into the null-space of the Jacobians of constraints related tasks. However, the number of constraints (here obstacles) being potentially large (not to say infinite), this null-space is potentially empty. For the robot to actually get its main task performed, what can be called heuristic constraints activation techniques are thus required. Based on the distance to constraints, these heuristics softly regulate the importance of all tasks ... which boils down to the same problems encountered with soft hierarchies as proposed in [Shen 2007].

To avoid situations where a compromise between constraints avoidance and task achievement has to be found (leading to control failure in the general case), one actually needs to introduce the concept of *passive* avoidance, as opposed to *active* avoidance. Adding a task to account for a constraint is indeed often performed in an active way, *i.e.* the added task is conceived as a motion away from the limit imposed by the constraint. Passive avoidance instead is based on the concept of preventing motions towards this limit. Conceptually, this leads to a similar requirement in terms of degrees of freedom mobilisation. However, there is no compromise to find between different motions of potentially antagonistic natures. Writing a given active avoidance task

$$\mathbf{t}_c^* = \mathbf{J}_c(\mathbf{q})\boldsymbol{\nu} \quad (1.24)$$

where $\mathbf{J}_c(\mathbf{q})$ is the constraint related task Jacobian and $\mathbf{t}_c^*$ the computed avoidance twist, the passive formulation “naturally” writes

$$\mathbf{0} = \mathbf{J}_c(\mathbf{q})\boldsymbol{\nu}. \quad (1.25)$$

Once again, the number of constraints being potentially large, projecting effector tasks in the null-spaces of constraint Jacobians can quickly lead to a complete freeze of the robot. This is annoying but safe, at least conceptually. To leave some degrees of freedom to the main task, the question is thus: when is a given passive avoidance task actually required? Considering joint position limits constraints only, P. Baerlocher and R. Boulic [Baerlocher 2004] answer this question by proposing a so-called “joint clamping algorithm”. At each control time step, it, overall, consists in determining which joints should be clamped (or in other words, not used any longer for computing the next control input) in order not to lead to a violation of the joint limits. This hand-crafted algorithm unfortunately leads to suboptimal solutions in some cases (see chapter 3 in the thesis of S. Rubrecht for an example [Rubrecht 2011a]). More importantly, it does not extend to any type of constraints. As anticipated in Section 1.1.2, a proper constraint activation mechanism is actually required to ensure both a strict respect of the constraints and an optimal control decision.

Some contributions on constrained optimisation based control approaches are presented in the next section.

1.2 Dealing with multiple tasks and constraints

Summary of the contribution

The contribution presented in this section is presented under the form of a published journal paper [Liu 2016b] corresponding to the postdoctoral work of M. Liu. The most illustrative video associated to this work can be seen [here^a](#). This work builds up on the PhD thesis work of J. Salini on “Dynamic control for the task/posture coordination of humanoids: toward synthesis of complex activities” [Salini 2012], illustrated in video [here^b](#).

Summary Multi-objective control systems for complex robots usually have to handle multiple prioritised tasks. Most existing hierarchical control techniques handle either strict task priorities by using null-space projectors or a sequence of quadratic programs; or non strict task priorities by using a weighting strategy. This paper proposes a novel approach to handle both strict and non-strict priorities of an arbitrary number of tasks. It can achieve multiple priority rearrangements simultaneously. A generalised projector, which makes it possible to completely project a task into the null-space of a set of tasks, while partially projecting it into the null-space of some other tasks, is developed. This projector can be used to perform priority transitions and task insertion or deletion. The control input is computed by solving one quadratic programming problem, where generalised projectors are adopted to maintain a task hierarchy, and equality or inequality constraints can be implemented. The effectiveness of this approach is demonstrated on a simulated robotic manipulator in a dynamic environment.

^ahttp://pages.isir.upmc.fr/~padois/website/fichiers/videos/liu_AutRob2015.webm

^bhttp://pages.isir.upmc.fr/~padois/website/fichiers/videos/salini_thesis.mp4

1.2.1 Contribution

The paper can be found in Appendix B.1.

1.2.2 Dealing with multiple tasks and constraints: going further

Posterior to this work, experimental results have been obtained on the KUKA LWR robot [Liu 2015] as well as on the iCub robot [Liu 2016a] using a quasi-static version of the proposed controller. While this more computationally efficient approach offers some perspectives, the overall question of hierarchy remains an opened one and trying to provide a universal and optimal answer to it is like opening the Pandora box. Indeed, why would one need to define hierarchies among tasks if these tasks were intrinsically well defined, *i.e.* using the right features, and compatible one with another as well as with constraints? This is of course a very complex problem, especially in open-ended environment, and hierarchies are actually an imperfect, conservative answer to the global difficulty of choosing the right space of expressions to describe tasks while keeping tractable computation times and some form of optimality.

While sub-optimal performances are fully acceptable, failure is not. In fact, before even trying to work on optimal multi-tasks control, one has to provide guarantees regarding the existence of constraints compatible solutions. This is the topic of the final section of this chapter.

1.3 Constraints compatibility

If most recent robot control approaches have made the switch from analytical to numerical control solution computation, one problem remains open: is there always a solution to the optimisation problem? In other words, are constraints always compatible? The answer is clearly “No” and the constraints compatibility problem can be illustrated by a very simple example. An automatic car is cruising towards a wall and its current velocity is $10m/s$. Its distance to the wall is $1m$, its maximum velocity capability is $50m/s$ and its maximum deceleration capability is $-2m/s^2$. At the considered control instant, none of the constraints of the car are violated but it is obvious that the car will inevitably collide with the wall in a very near future, no matter the next control action (expressed for example in terms of acceleration). The constraints on the position and velocity of the car are incompatible and this is due to the fact that their naive expression does not account for the bounds on the acceleration capabilities of the car. Rendering the constraints compatible in the “car and wall” example can for instance be done by expressing a virtual acceleration constraint as a function of the current position and velocity of the car as well as of its maximum deceleration capability.

This problem is actually much more general and can be generalised to all constraints (1.6). It somewhat relates to the notion of viability often used when dealing with the balance problem for humanoid robots: *A viable state can be defined as one from which it can be guaranteed, over an infinite time horizon, that proper control will avoid the robot to lose balance.* Constraints compatibility can be seen as an extension of this notion to the overall robot control problem: a viable (extended-)state can be defined as one from which it can be guaranteed, over an infinite time horizon, that a control solution satisfying all constraints will always exist. The naive constraints expression thus have to be modified in order to ensure that the control solution computed reactively and applied at the next control instant will maintain the system in a viable extended-state.

This modification of the constraints expression is of course not trivial as it requires a forecast of its effect over an infinite time horizon. Three distinctions can be made. 1) When the constraint expression is not a function of the extended-state itself, this can potentially be done. This is for example the case of joint space related constraints (position, velocity, torque, ... limits) which are generally constant. 2) However, when dealing with constraints relating to the external environment, their expression is generally a function of the state $(\mathbf{q}, \boldsymbol{\nu})$ of the robot. For example, a constraint of non-collision between a static object and the closest body of a surrounding robot can be written as a constraint on the robot velocity

$$\mathbf{J}_c(\mathbf{q}_k)\boldsymbol{\nu}_k T_e + d_k \geq 0 \quad (1.26)$$

where $\mathbf{J}_c$ is the Jacobian matrix associated to the closest body of the robot and projected along the unit vector associated to the shortest distance between the object and the closest robot body, $\boldsymbol{\nu}_k$ is the generalised velocity of the robot (here the control input to be computed), T_e is the control sample time and d_k is the current closest distance between the robot and the object. The evolution of $\mathbf{q}$ is a function of the control input which has itself to be computed, as mentioned preliminarily, reactively. The dependence of the constraints expression to the state generally being non-linear, reasoning over an infinite time horizon in that case is highly computationally complex. 3) It finally becomes impossible when the environment is dynamic and its future evolution is not known.

The remainder of this section introduces some research works to deal with the three cases, with some limitations which are also discussed.

1.3.1 Motion safety and constraints compatibility for multi-body robots

Summary of the contribution

The first contribution presented in this section is presented under the form of a published journal paper [Rubrecht 2012] corresponding to the PhD thesis work of S. Rubrecht [Rubrecht 2011a]. The most illustrative video associated to this work can be seen [here](http://pages.isir.upmc.fr/~padois/website/fichiers/videos/rubrecht_AutRob2.mp4)^a.

Summary In this work, a formal approach is proposed to ensure safe behaviours of multi-body robots in a reactive control framework. The permanent satisfaction of constraints being insufficient to ensure safety, this approach focuses on the constraints expression; the compatibility between these constraints is studied. For joint space related constraints, compatibility conditions are established. For the case of environment related constraints, where compatibility cannot be ensured in a simple way, a safe alternative behaviour is proposed. This alternative behaviour is based on the compilation of a feasible escape trajectory at each control time step. A complete case study involving obstacles, joint position, velocity and acceleration limits illustrates the approach. A particular method is developed to take full advantage of the usual avoidance techniques while maintaining safety. Experiments involving a 6-DOF manipulator operating in a cluttered environment confirms the reliability of the approach and validates the expected performances.

^ahttp://pages.isir.upmc.fr/~padois/website/fichiers/videos/rubrecht_AutRob2.mp4

The paper can be found in Appendix B.2.

1.3.2 An energetic approach to safety

As demonstrated in [Rubrecht 2012], the constraints compatibility problem can be solved including both joint space and operational space constraints in static environment. This is no longer the case in dynamic environments where the behaviour of the environment cannot be predicted over an infinite time horizon. This working context is becoming prevalent in Robotics, especially in collaborative applications where contacts between the robot and its environment (among which human operators) are actually desirable. This does not mean that safety is no longer an issue. It is just a more complex one as it requires to switch from historical collision avoidance and associated geometrical approaches to ones where contact wrenches become an important variable to monitor. However, it is not very satisfactory to reason in terms of contact wrenches only as they are information-less when the robot moves in free space. One could of course use different safety indicators depending on the operating mode of the robot but this would require a switch which can be anticipated as a source of discontinuity in the control law. Moreover, in collaborative applications, the notion of control mode is not necessarily well-defined: in what control mode is a robot following a given trajectory while a human operator is pushing it to momentarily get access to some zone of the workspace?

One thus need a more general safety indicator. The degree of danger induced by the presence of a robot has actually two main causes: the impact force created when collisions occur and the contact wrenches existing after the establishment of physical contact. Therefore, to ensure safety, an indicator whose value can be related to both theses causes and that can be expressed as a function of the robot control input has to be derived. The original contribution of the proposed approach is to rely on *energy* as the central quantity used to characterise safety.

Contrarily to velocity and force, energy is meaningful both in free-space and when in contact with the environment. More precisely, impact forces are a function of the kinetic energy to be dissipated at the impact time. Part of this kinetic energy is due to the robot motion and can thus be monitored. Contact wrenches can, on their end, be related to the potential energy virtually accumulated in the controller in order to achieve the control objective, being it a motion of force task. This energetic approach to safety

is developed in the framework of the ongoing PhD thesis of A. Meguenani. Some preliminary results are discussed hereafter.

Summary of the contribution

The contribution presented in the remainder of this section is presented under the form of a published conference paper [Meguenani 2015]. The most illustrative video associated to this work can be seen [here^a](http://pages.isir.upmc.fr/~padois/website/fichiers/videos/meguenani_ongoing.mp4).

Summary In this paper, a physically meaningful energy-related safety indicator for robots sharing their workspace with humans is proposed. Based on this indicator, a safety criterion accounting for the breaking capabilities of the robot is included as a quadratic constraint in the control algorithm. This constraint is modulated by the distance between the human operator and the end-effector of the robot. The control algorithm is formulated as an optimisation problem and computes the actuation torque of a robotic manipulator given some task to be performed and physical constraints to respect. The overall framework is validated in a physics simulation software on a Kuka LWR4 and different behaviours of the robot towards a considered obstacle in its environment are evaluated and discussed.

^ahttp://pages.isir.upmc.fr/~padois/website/fichiers/videos/meguenani_ongoing.mp4

The paper can be found in Appendix B.3.

1.3.3 Constraints compatibility: going further

The energetic approach seems a sound and general framework to safety with respect to traditional geometric approaches. The ongoing work on this topic is focused on:

- phases where the robot is physically interacting with its environment. During these phases, the potential energy virtually accumulated in the controller is considered and monitored. However, the control mode should have an influence on how this energy is monitored. If the current control mode considers physical interaction as a disturbance with respect to a trajectory following task, then the considered energy is very likely to grow and represent a high danger level. If the control mode accounts for this interaction such as in a gravity compensation or force amplification mode (parallel comanipulation), the considered energy level should not grow as much. It is thus important to define a typology of physical interaction cases and their associated feasible transitions. The control mode of the robot and the intention of the environment (obviously a human being here) are clearly discriminant factors.
- Reformulating the constraints on kinetic energy as a constraint on potential energy. Indeed, the motion of the robot and its corresponding kinetic energy is the outcome of the virtual potential energy stored in the controller and released through both compensatory effects (dynamics decoupling) and a resulting motion. Theoretical developments and the corresponding experimental results on this topic are presented in [Meguenani 2017]. This should lead to a generalised framework for safety.
- Extending the joint space compatibility study initiated in [Rubrecht 2012] in order to extend it to acceleration control including jerk constraints.

While essential in order to fully apprehend the constraints compatibility problem, this last development is actually doomed to produce limited results. Indeed, constraints compatibility over an infinite time horizon can only be guaranteed for state-independent constraints. Unfortunately, while actuation torques capabilities can be seen as constant for a given range of actuators velocities, the corresponding joint space

acceleration capabilities are not constant as clearly shown by Equation (1.1). This is due to non-linear dynamics couplings induced both by the structure of robots and by potentially complex transmission mechanisms from actuators to joints. This is all the more true that tasks requirements are more and more very likely to be dynamically updated. Global optimality of the control decision is thus meaningless and this advocates for control approaches computed with optimality over short time horizon in mind rather than for global planning or purely reactive ones.

The PhD thesis of Y. Tan [Tan 2016] is an attempt to introduce an horizon-based reasoning on constraints. In his work, a predictive control primitive based on Model Predictive Control (MPC) is developed to handle the presence of discontinuities in the constraints a robot must satisfy (for example when breaking contact with the environment). The controller takes advantage of predictive formulations to anticipate the evolution of the constraints by means of the control scenario and/or sensor information, and thus generate new continuous constraints to replace the original discontinuous constraints in a QP reactive controller. As a result, the rate of change in joint torques is minimised compared with the original discontinuous constraints. This predictive control primitive does not directly modify the desired task objectives, but the constraints to ensure that the worst case of changes in joint torques is well-managed. The effectiveness of the proposed control framework is validated by a set of experiments in simulation on the Kuka LWR robot and the iCub humanoid robot. The results show that the proposed approaches significantly decrease the rate of change in joint torques when task priorities switch or discontinuous constraints occur. Nonetheless, the overall control approach remains purely reactive and, as is mentioned in the next chapter, cannot generally guarantee the existence of a viable control decision in any context.

Chapter 2

Optimisation-based control for balanced humanoid behaviours

Note on this chapter

The contribution presented in this chapter is adapted from an article which has been recently submitted as a chapter entitled “Optimal Control Approach to Humanoid Balancing” for the book project “[Humanoid Robotics: a Reference](#)”. This article has been written together with A. Ibanez and partly builds up on his PhD thesis work [[Ibanez 2015](#)].

In chapter 1, some contributions in the domain of multi-tasks, constraint compliant and reactive robot control are presented. While this level of control appears as a necessity in order to compute the control input at each control instant while enforcing low-level constraints, it also has strong limitations. Firstly, it assumes that constraints compatibility can be ensured over a infinite time horizon. The work presented in section 1.3 clearly nullify the validity of this assumption. Secondly, it assumes that the multiple tasks to be achieved concurrently are compatible one with another or that a proper hierarchy of incompatible tasks can be found that will lead to a globally satisfactory behaviour. This is once again a strong assumption which cannot be considered as valid by default. Finally, the absence of anticipation over the future evolution of the desired tasks necessarily lead to suboptimal control decisions.

These limitations may be completely bearable in some classical robotic contexts where pure offline planning methods can be successfully applied. However, in many contexts the tasks dynamically evolve as a function of the environment dynamics and, in these cases, pure reactive approaches are doomed to fail. This is typically the case in humanoid robotics where balance motion have to be produced and for which purely reactive control approaches are not sufficient.

In this chapter, a review of optimisation-based control approaches to balanced humanoid behaviours is proposed. Balance is indeed a necessary condition in all use cases of humanoid robots as illustrated in figure 2.1. “Ensuring balance” is consequently a core feature of humanoid controllers and most of them rely on an explicit model of balance or on balance criteria extracted from these models.

However, balance is a complex notion and accounting for it in a general and efficient way at the control level is complex. Part of this complexity lies in the high dimensionality, non-linearity, under-actuation and hybrid nature of humanoid systems. These characteristics make it difficult to define computationally-

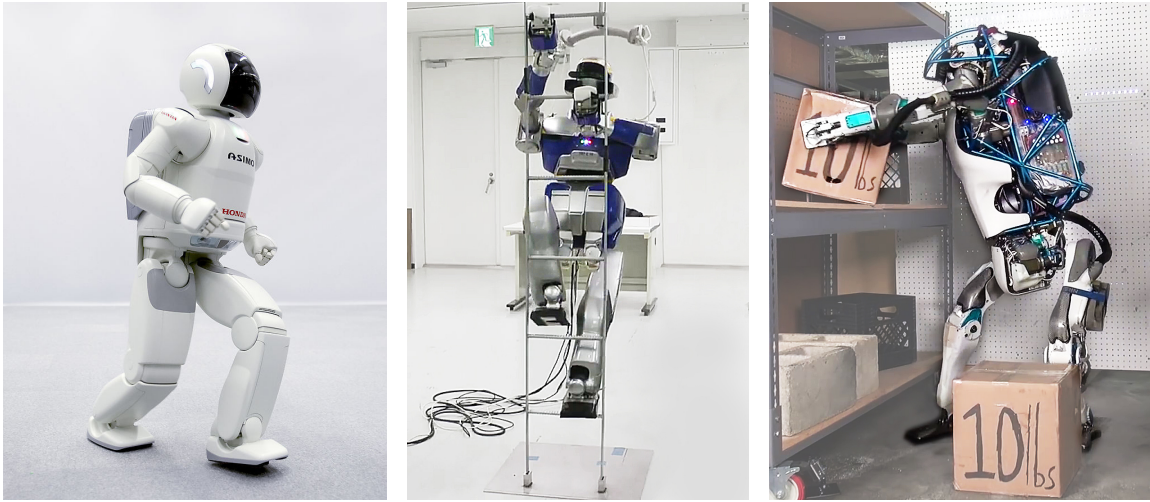


Figure 2.1: Current state of the art of balancing humanoids. From left to right: Asimo, HRP-2 and Atlas robots. *Courtesy of Honda Robotics, CNRS-AIST JRL, and Boston Dynamics.*

friendly models of balance. This is all the more true that the impact of some given control action on balance cannot necessarily be anticipated for in a reactive way and requires its consequences to be previewed over a time horizon to conclude on its compatibility with the general balance objective. As an example, unilateral contacts between the feet and ground provide the ability to modify the contacts configuration and generate feet motions with respect to the world, in other words locomotion. In these situations, breaking or making contact is a discrete action whose consequences are a function of the next actions of the corresponding foot: if the centre of mass of the overall system is launched forward but the flying foot never goes back to the ground the robot will inevitably fall whereas a proper next step may lead to dynamic equilibrium and balanced locomotion.

In scenarios where locomotion or corrective steps can be safely excluded, balance can be accounted for at the whole-body reactive control level (see figure 2.2), *i.e.* without the need for preview. In all other cases, predictions are required and, in the humanoid control literature, balance is most of the time first accounted for at higher abstraction and control levels (see figure 2.2). At the planning level, balance-compliant reference trajectories are computed in an optimal fashion over the complete time horizon. This is of course computationally costly and cannot be performed in real time, in closed-loop. At the task regulation level, the computational burden is solved by only considering some reduced part of the system's dynamics. This allows to refine the reference trajectories in a quasi real time way, accounting for the current state of the robot.

Balance models induce strong non-linearities such as the ones due to contact conditions. Moreover, unilateral contacts can be broken and discrete events (contact making/breaking) have to be accounted for at the control level. As a consequence, the control problems to be solved do not necessarily possess closed-form solutions and their solution space may not be convex and fully connected. These features advocate for a formulation of the control problems as constrained optimisation ones. While at the planning and task regulation levels, the problems are often naturally posed as optimisation ones, there has also been a shift toward optimisation based methods at the whole-body reactive control level: historical contributions [Khatib 2008] and their successors [Righetti 2011, Del Prete 2015] in this domain did not account for inequality constraints in an optimal way and the last decade has seen an important growth of alternative, optimisation-based, formulation of the control problem.

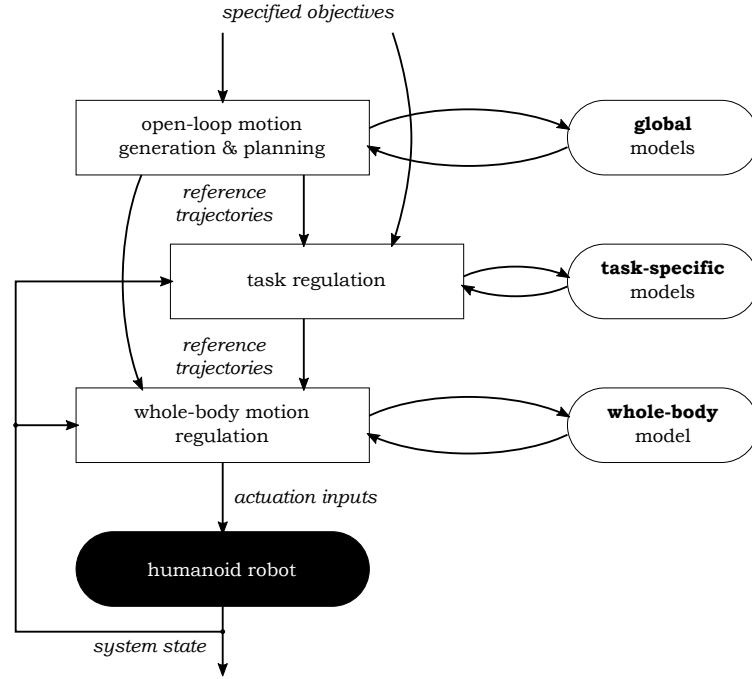


Figure 2.2: Generic overview of control architectures

While several review works provide a very complete, locomotion-centred, view of the balance problem [Kajita 2008, Wieber 2015], this chapter looks at the balance problem through the prism of optimisation. It is organised as follows: first, the dynamics of free-floating systems are introduced and the non-linearities raised by contact mechanics are highlighted. Second, a general formulation of optimisation-based approaches to the balance control problem is proposed to expose the computational challenges it induces. Levers classically employed to tackle these challenges are then identified. The prevalent balance indicators and metrics, needed to describe control objectives and constraints, are therefore presented. An overview of optimisation-based control approaches is finally proposed. These approaches are organised with respect to the level of abstraction at which they are employed to solve the balance control problem.

2.1 A look at the dynamics equations

Efficient balance control requires the exploitation of the whole-body dynamics of the system. Within this perspective, complexity rapidly arises from the high-dimensionality and non-linearity of these dynamics. Indeed, motions of legged robots are fundamentally supported by the contact wrenches from their environment, wrenches generally being subject to the complex mechanical laws of unilateral contacts.

2.1.1 Lagrangian equations of motion

Legged robots are generally modelled from the control point of view as systems composed of rigid bodies, arranged in a tree structure with a base body as their root, called *floating-base*. The displacement of the robot in space is captured with respect to the position and orientation of a reference frame $\mathcal{R}_b$ attached to this body, with respect to a given reference inertial frame $\mathcal{R}_0$, called *world frame*. Being free-floating systems, the base is henceforth treated as linked with a 6-DoF virtual non-actuated joint to the world, defining the pose $\mathbf{q}_b \in SE(3)$ of $\mathcal{R}_b$ with respect to $\mathcal{R}_0$, with $SE(3)$ the special Euclidean group, as illustrated

in figure 2.3. The associated twist ν_b is in $\mathbb{R}^6$.

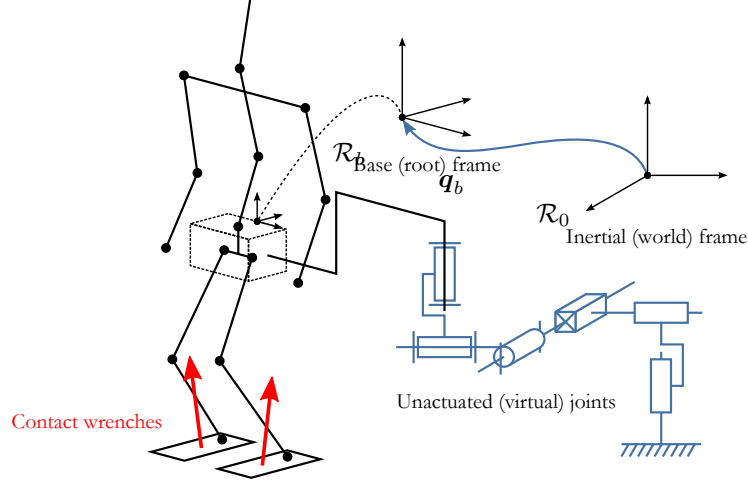


Figure 2.3: Kinematic representation of floating-base systems. The root body of the tree structure of the mechanism is free-floating in a reference inertial frame $\mathcal{R}_0$.

Having a closer loop to the equations of motion (1.1), which can be derived from the Lagrange formalism [Murray 1994], for such systems, they take the form

$$\underbrace{\begin{bmatrix} M_b & M_{bj} \\ M_{bj}^T & M_j \end{bmatrix}}_{M(q)} \underbrace{\begin{bmatrix} \dot{\nu}_b \\ \ddot{q}_j \end{bmatrix}}_{\dot{\nu}} + \underbrace{\begin{bmatrix} n_b \\ n_j \end{bmatrix}}_{n(q,\nu)} + \underbrace{\begin{bmatrix} g_b \\ g_j \end{bmatrix}}_{g(q)} = \underbrace{\begin{bmatrix} \mathbf{0}_6 \\ T^T(q) \tau \end{bmatrix}}_{S^T(q)} + \gamma_c, \quad (2.1)$$

where indices $\bullet_b$, $\bullet_j$ and $\bullet_{bj}$ denote definitions with respect to the base, the joints and both, respectively. $T \in \mathbb{R}^{(n-6) \times n_a}$ is a matrix representing the actuation characteristics. Vector γ_c captures wrenches applied to the system from the environment in the generalised coordinates space.

Equations (2.1) clearly exhibits that, the floating-base being non-actuated *i.e.* $\dim(\dot{\nu}) > \dim(\tau)$, legged robots are *underactuated*. It therefore appears that external wrenches γ_c play a prevalent role in the dynamics of the floating-base, that is the 6 first lines of equation (2.1).

2.1.2 Newton-Euler equations of motion

The 6 first lines of equation (2.1) are differential equations related to the floating-base describing the dynamics of the system as a whole. Written at the centre of mass of the system, they yield the Newton-Euler equations of motion

$$\begin{bmatrix} \dot{l}_x \\ M\ddot{x} \end{bmatrix} = w_x^c + w_x^g, \quad (2.2)$$

where w^c and w^g are respectively the net wrenches¹ issued from contact and gravity effects, M is the total mass of the system, and l_x is the angular momentum of the system at its centre of mass x . The Newton equation is the one related to linear momentum whereas the Euler one is related to angular momentum. These two equations plainly display the relation between contact wrenches and the dynamics of the centre of mass, as, from the structure of (2.1), actuation has no influence at this level. In the Newton equation, the

¹in 6D vector notation

motion of the centre of mass directly relates to the amplitude of external wrenches while the Euler equation captures their distribution.

It can be noted that differential equations (2.1) and (2.2) are generally constrained. Indeed, inputs $\boldsymbol{\tau}$, joint configurations $\mathbf{q}_j$ and their temporal derivatives are generally bounded by technological limits. Laws of contact mechanics might additionally impose constraints on the contact wrenches $\boldsymbol{\gamma}_c$ and, subsequently, on the system configuration and velocity and acceleration $(\mathbf{q}, \boldsymbol{\nu}, \dot{\boldsymbol{\nu}})$.

2.1.3 Contact mechanics

External wrenches $\boldsymbol{\gamma}_c$ in equation (2.1) are in the general case the result of contact forces between the system and its environment. A classical contact description is to consider non-adhesive contacts solely: contact forces are *unilateral*, *i.e.* they solely oppose penetration between the bodies in contact. With $\mathbf{n}$ the normal to the contact surface and $\mathbf{f}_c$ a contact force on a body, the unilaterality condition writes

$$f_c^n \geq 0, \quad \text{with } f_c^n \triangleq \mathbf{f}_c \cdot \mathbf{n}. \quad (2.3)$$

This characteristic (2.3) is not the only source of complexity: standard contact models indeed define contact laws as conditional equations, describing distinct possible contact cases. These contact cases involve both normal and tangential components of contact forces. In rigid body mechanics, which is the most widespread framework for multibody legged robots, normal contact mechanics are described as follows²

$$\begin{cases} f_c^n & \geq 0 & \text{if contact,} \\ f_c^n & = 0 & \text{otherwise.} \end{cases} \quad (2.4)$$

Tangential contact mechanics are more directly related to contact dynamics. Indeed, friction plays a role when a relative motion of the bodies in contact is involved. A standard friction model is the Coulomb dry friction model which states, with $\dot{\mathbf{x}}$ the relative velocity of the bodies

$$\begin{cases} \mathbf{f}_c^t = \mu_c f_c^n \frac{\dot{\mathbf{x}}^t}{\|\dot{\mathbf{x}}^t\|} & \Leftarrow \|\dot{\mathbf{x}}^t\| > 0 \\ \|\mathbf{f}_c^t\| \leq \mu_c f_c^n & \Rightarrow \|\dot{\mathbf{x}}^t\| = 0 \end{cases} \quad \text{with } \begin{cases} \mathbf{f}_c^t \triangleq \mathbf{f}_c - f_c^n \mathbf{n}, \\ \dot{\mathbf{x}}^t \triangleq \dot{\mathbf{x}} - (\dot{\mathbf{x}} \cdot \mathbf{n}) \mathbf{n}. \end{cases} \quad (2.5)$$

The dimensionless scalar μ_c is a parameter of the model, called coefficient of friction. The Coulomb model describes two regimes as illustrated in figure 2.4: in the *kinetic friction*, or sliding, regime the tangential contact force is given ; however, in the *static friction*, or sticking, regime this component is set-valued.

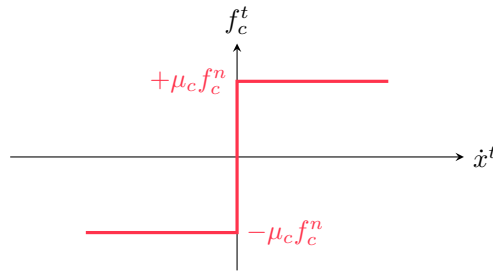


Figure 2.4: Non-smooth and nonlinear contact dynamics described by Coulomb's friction law.

Rigid-body models also induce non-smooth dynamics: discontinuities in the relative velocity of body entering contact, *impacts*, can indeed occur.

² It can be noted that these contact mechanics are fully compatible with the unilateral contact condition (2.3).

In a nutshell, the dynamics of multibody legged robots present several sources of complexity to tackle at the control level. First, their dynamics state is described by variables of high dimensionality. Second, despite this high degree of freedom these systems are under-actuated, posing inverse dynamics as an ill-posed problem and involving contact forces as a determining variable. Last, the dynamics governing these forces are non-smooth and subject to constraints.

2.2 Optimal control approach to balance

2.2.1 Balance?

At the whole-body motion regulation level, a humanoid robot is said to be balancing if, given its joint space and contact state (joint space and contact points positions and velocities), the external dynamic wrench induced by gravitational, inertial effects and other applied external wrenches applied to it can be compensated for by proper contact forces and joint torques. Achieving balance requires to respect the equation of motion while being able to maintain the contact state, *i.e.* not tipping over and not sliding. Given some tasks to perform, the global balance question boils down to some higher level problem where the evolution of the control input and of the state itself have to be determined to perform tasks at best while being able to reach a balanced terminal state. This general definition of balance includes jumping and running as potential types of balanced motions. However, in practice, ensuring balance at each time instant provides a better guarantee that a final balanced state can be achieved and jumping and running are generally not considered as a class of motion that can be achieved looking solely at the problem through the balance prism.

While this definition of balance can directly be translated into an optimisation problem aiming at finding a solution (either at the global or at the regulation level), it does not say much about the existence of solutions. Indeed, as mentioned in the section “Stability analysis, not falling down” in [Wieber 2015], understanding the long term behaviour requires to rely on *stability* and *robustness* notions. While many tools exist to define such notions for humanoids, the concept of *viability* is central. A viable state can be defined as one from which it can be guaranteed, over an infinite time horizon, that proper control will avoid the robot to lose balance. Ensuring the existence of a non empty and connected set of viable states and solving the control problem to maintain the future states of the system in this set is a very difficult, potentially intractable, problem. As a matter of fact, the balance literature offer much less ambitious visions of balance inducing limited behaviours but which guarantee balance in a tractable way.

This section describes potential ways of defining the optimisation problem related to balance and recalls some of the balance indicators/criteria defined in the literature and which can be seen as special cases of a more general one.

2.2.2 General control problem

The control problem can be, in the most general way, expressed as the problem of finding a series of control inputs which will drive the system from an initial state towards objectives.

Reaching objectives The objective of the activity the system is involved into is denoted $\mathcal{F}^d$. Stating that the system has reached this objective can be written as

$$\mathcal{F}(\mathcal{S}) = \mathcal{F}^d, \quad (2.6)$$

where $\mathcal{S}$ denotes the state of the system, and $\mathcal{F}$ relates the state of the system to its outputs.

In the general case however, there is no guarantee that this objective $\mathcal{F}^d$ is *feasible* nor that $\mathcal{F}$ is invertible. An intuitive and standard way to address this problem is to relax the constraint (2.6), and reformulate it

as the least-squares problem with the introduction of the slack variable $\mathcal{W}$

$$\begin{aligned} & \min_{\mathcal{S}, \mathcal{W}} \|\mathcal{W}\|^2 \\ \text{s.t. } & \mathcal{F}(\mathcal{S}) = \mathcal{F}^d + \mathcal{W}, \end{aligned}$$

which can be simplified to

$$\min_{\mathcal{S}} \|\mathcal{F}(\mathcal{S}) - \mathcal{F}^d\|^2, \quad (2.7)$$

where $\|\bullet\|$ is commonly the ℓ^2 -norm. Such an optimisation-based formulation is therefore mainly driven by the need for finding solutions bringing the system *as close as possible* to its objectives, while explicitly taking into account the notion of infeasibility. Infeasibility can indeed be considered in the optimisation problem with the constraints defining the set $\mathcal{K}^{\mathcal{S}}$ of admissible states over which the control problem is solved. That is, the objective-reaching problem is actually

$$\min_{\mathcal{S} \in \mathcal{K}^{\mathcal{S}}} \|\mathcal{F}(\mathcal{S}) - \mathcal{F}^d\|^2.$$

These constraints are nevertheless ignored in the rest of this section for the sake of simplicity.

Problems (2.6) and (2.7) describe the problem of finding a state $\mathcal{S}$ of the system which achieves the desired objective $\mathcal{F}^d$. The control problem therefore consists in finding a trajectory of control inputs that brings the system to this desired state.

Finding a way towards objectives In discrete time, let us describe the system as the dynamical system defined as

$$\begin{cases} \mathcal{S}_{k+1} = \mathcal{S}_{k+1}(\mathcal{S}_k, u_{k+1}) \\ \mathcal{F}_{k+1} = \mathcal{F}(\mathcal{S}_{k+1}) \end{cases}$$

In a dynamics framework, the system state at time t_i hence depends on an anterior initial state $\mathcal{S}_0$ at time t_0 and on an history of control inputs $\mathcal{U}_{0,i} \triangleq [u_1, \dots, u_i]^T$. This state at t_i is denoted

$$\mathcal{S}_{i|0} \triangleq \mathcal{S}_i(\mathcal{S}_0, \mathcal{U}_{0,i}).$$

The control problem is therefore written, with an objective $\mathcal{F}^d$ to be reached at time t_f

$$\min_{\mathcal{U}_{0,f}} \|\mathcal{F}(\mathcal{S}_{f|0}) - \mathcal{F}^d\|^2. \quad (2.9)$$

However, from the control point of view the dynamics of the environment and the system might be partially known. A *feedback* of the current system state $\mathcal{S}_k$ at each control time t_k is therefore introduced in order to account for potential resulting disturbances, as illustrated in figure 2.5. That is, the problem (2.9) is rewritten

$$\forall t_k, \quad \min_{\mathcal{U}_{k,f}} \|\mathcal{F}(\mathcal{S}_{f|k}) - \mathcal{F}^d\|^2, \quad (2.10)$$

to be solved at each control time t_k , accounting for the current system state $\mathcal{S}_k$. Note that problem (2.10) can naturally be written in the optimal control canonical form, in continuous time.

2.2.3 The balance problem

In the case of legged robots, final objectives $\mathcal{F}^d$ are generally considered as the outputs of *balanced* system states. Indeed, from a safety point of view balanced states are preferable as they are *controllable* [Dalibard 2013],

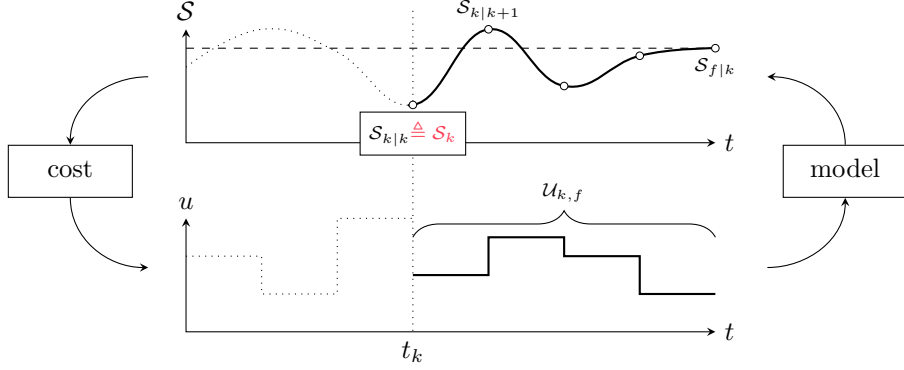


Figure 2.5: Model-based optimal control problem over a time horizon with state feedback.

i.e. the system is able to handle disturbances as a path exists allowing to recover the desired state. With $\mathcal{K}_k^{\mathcal{B}}$ the set of balanced states at instant t_k , the problem (2.10) writes

$$\begin{aligned} \forall t_k, \quad & \min_{\mathcal{U}_{k,f}} \|\mathcal{F}(\mathcal{S}_{f|k}) - \mathcal{F}^d\|^2, \\ \text{s.t.} \quad & \mathcal{S}_{f|k} \in \mathcal{K}_{f|k}^{\mathcal{B}}. \end{aligned} \quad (2.11)$$

The final constraint of problem (2.11) is nevertheless challenging as it involves the whole control history $\mathcal{U}_{k,f}$. A way to take on this challenge comes from the viability theory [Wieber 2002]: the control history $\mathcal{U}_{k,f}$ must induce system states which allows the system to maintain a balance state. While this problem is still intractable in the general case, a sufficient condition can yet be inferred: if all intermediate states $\{\mathcal{S}_{k+1|k}, \dots, \mathcal{S}_{f-1|k}\}$ are balanced, then a final state $\mathcal{S}_{f|k} \in \mathcal{K}_{f|k}^{\mathcal{B}}$ is reachable. Problem (2.11) is therefore rewritten

$$\begin{aligned} \forall t_k, \quad & \min_{\mathcal{U}_{k,f}} \|\mathcal{F}(\mathcal{S}_{f|k}) - \mathcal{F}^d\|^2, \\ \text{s.t.} \quad & \mathcal{S}_{i|k} \in \mathcal{K}_{i|k}^{\mathcal{B}} \quad \forall i \in [k+1, f]. \end{aligned} \quad (2.12)$$

In the robot control framework, various sources of disturbances are present. For robustness, stability and feasibility considerations, the control problem (2.12) is hence usually reformulated in order to keep the system away from stability boundaries. With $\mathcal{B}$ a metrics capturing the distance of the system to the stability boundaries, this yields the new formulation

$$\begin{aligned} \forall t_k, \quad & \begin{cases} \min_{\mathcal{U}_{k,f}} \|\mathcal{F}(\mathcal{S}_{f|k}) - \mathcal{F}^d\|^2, \\ \max_{\mathcal{U}_{k,i-1}} \mathcal{B}(\mathcal{S}_{i|k}) \quad \forall i \in [k+1, f], \end{cases} \\ \text{s.t.} \quad & \mathcal{S}_{i|k} \in \mathcal{K}_{i|k}^{\mathcal{B}} \quad \forall i \in [k+1, f]. \end{aligned} \quad (2.13)$$

2.2.4 Challenges

The control problem (2.13) raises several challenges to take on. First, the dimension of the control input history $\mathcal{U}_{k,f}$ can rapidly grow for legged robots and long-term objectives. Indeed legged robots present a high degree of freedom and the actuation vector τ is therefore of high dimension.

Second, the model presents strong non-linearities predominantly arising from the non-linear evolution of the dynamics model (2.1) with respect to the state $\mathcal{S}$ and from the non-smooth contact dynamics as discussed in paragraph 2.1.3.

Last, the optimisation problem (2.13) is not convex in the general case. Indeed, balance constraints do not necessarily describe a convex set and no characteristic of the cost functions suggests the existence of a single,

global optimum.

Additionally, the stabilisation of the system dynamics and the regulation of disturbances require the control problem to be solved at a frequency consistent with the overall dynamics. In the case of legged robots, this frequency is commonly around $100Hz$ to $1kHz$ which leads to the need for solutions every 1 to $10ms$. This computational requirement, with respect to the current state of computer science and optimisation techniques, prompts to consider different approaches to the control problem (2.13).

2.2.5 Various approaches to the balance control problem

From the preceding remarks, three main levers can be identified to reduce the computational complexity of the control problem (2.13): solving frequency, time horizon and model complexity.

Solving frequency The most straightforward approach to this challenge is to directly relax the computational requirement by considering a lower rate of control.

An extreme case is offline planning, where problem (2.13) is solved once and for all from an initial state $\mathcal{S}_0$ at an initial instant t_0 ; that is

$$\begin{aligned} & \begin{cases} \min_{\mathcal{U}_{0,f}} \|\mathcal{F}(\mathcal{S}_{f|0}) - \mathcal{F}^d\|^2, \\ \max_{\mathcal{U}_{0,i-1}} \mathcal{B}(\mathcal{S}_{i|0}) \quad \forall i \in [1, f], \end{cases} \\ \text{s.t. } & \mathcal{S}_{i|0} \in \mathcal{K}_{i|0}^{\mathcal{B}} \quad \forall i \in [1, f]. \end{aligned} \quad (2.14)$$

Since this approach implies an offline resolution of the problem, no computation requirements are to be met ; non-linear and non-convex optimisation techniques can therefore be employed despite their relative computational inefficiency. Stochastic solvers are commonly set-up in order to explore the whole solution space, but other non-linear solvers are also envisioned in cases where local optima are sufficient.

While this approach allows to find initial solutions to the control problem, the requirement for a fast, closed-loop controller is still present to stabilise the dynamics of the system and handle disturbances of small time scale. This approach is therefore employed at relatively high levels of control, such as open-loop motion generation or task regulation (*cf.* figure 2.2).

Time horizon A practical approach to tract the control problem (2.13) in closed-loop is to consider a reduced time horizon ; one of the most widespread approaches consisting in the formulation of the control problem over one single control step. This approach requires the definition of instantaneous objectives $\mathcal{F}_k^d$ at each control time t_k defined such that $\mathcal{F}_f^d \triangleq \mathcal{F}^d$.

$$\begin{aligned} \forall t_k, & \begin{cases} \min_{u_k} \|\mathcal{F}(\mathcal{S}_{k+1}) - \mathcal{F}_{k+1}^d\|^2, \\ \max_{u_k} \mathcal{B}(\mathcal{S}_{k+1}), \end{cases} \\ \text{s.t. } & \mathcal{S}_{k+1} \in \mathcal{K}_{k+1}^{\mathcal{B}}. \end{aligned} \quad (2.15)$$

The set of instantaneous objectives $\{\mathcal{F}_0^d, \dots, \mathcal{F}_f^d\}$ is commonly the output of a planning process (*cf.* figure 2.2).

At the instantaneous level, the dimensionality of the original control problem is evidently reduced. Furthermore, non-linearities in the dynamics model can be handled through linearisation of the equation of motion (2.1). Indeed, considering $\mathbf{M}(\mathbf{q}) \approx \mathbf{M}$, $\mathbf{n}(\mathbf{q}, \boldsymbol{\nu}) \approx \mathbf{n}$ and $\mathbf{g}(\mathbf{q}) \approx \mathbf{g}$ is a practical assumption generally accepted for high control rates. Similarly, non-linearities from the contact dynamics are commonly addressed by considering, at this instantaneous time-scale, the contact regime as fixed and pre-defined by a higher control level.

This extreme reduction of the time horizon to a single time step is nevertheless not the only option to approach the control problem. Indeed, reducing the time horizon would require the definition of local balance constraints which are generally over-conservative with respect to the original problem (2.11) in order to ensure that the final state is balanced, thus producing potentially suboptimal solutions. To address this issue, recent approaches can be found in the literature considering an horizon of several time steps, in the Model Predictive Control framework commonly. Within this framework, the control problem writes

$$\begin{aligned} \forall t_k, \quad & \begin{cases} \min_{\mathcal{U}_{k,k+h}} & \|\mathcal{F}(\mathcal{S}_{k+h|k}) - \mathcal{F}_{k+h}^d\|^2, \\ \max_{\mathcal{U}_{k,i-1}} & \mathcal{B}(\mathcal{S}_{i|k}) \quad \forall i \in [k+1, k+h], \end{cases} \\ \text{s.t.} \quad & \mathcal{S}_{i|k} \in \mathcal{K}_{i|k}^{\mathcal{B}} \quad \forall i \in [k+1, k+h], \end{aligned} \quad (2.16)$$

where h is a finite future time horizon.

However, the computational complexity of such approaches is still an open problem in legged robotics.

Model complexity The central barrier to solving practically the control problem (2.13) actually comes from the complexity of the model itself. This complexity mainly takes the form of dimensionality, non-linearities and non-smoothness. To overcome this obstacle, model reduction and simplification are commonly employed. At the lowest control level, model reductions or simplifications are usually avoided in order to compute actuation inputs with the utmost validity. However, higher control levels can largely benefit from a reduction of the model complexity, the complete dynamics model being enforced at the following control level (*cf.* figure 2.2). A common approach in this perspective is to use this intermediate problem as an online trajectory generator.

Since the contact dynamics are one of the predominant sources of non-linearity and non-smoothness, a relaxed, smooth approximation of these dynamics allows fast resolutions of the control problem [Tassa 2012] over a finite time horizon. Another common approach is to consider reduced dynamics, not subject to non-linearities or non-smoothness. For example, reducing the control problem to the centre of mass and considering contact regimes as predefined allows to intuitively approximate linear models suitable for computationally-efficient control formulations [Kajita 2003, Wieber 2006].

2.2.6 Balance indicators and criteria

Within the control perspective described in problem (2.13), appropriate definitions of the set of balanced states $\mathcal{K}^{\mathcal{B}}$ and of the balance metrics $\mathcal{B}$ are required.

One of the most commonly employed definitions relies on the Zero-Moment Point [Vukobratović 1972] (ZMP), or centre of Pressure (CoP). This point, defined without ambiguity for coplanar contact surfaces solely, allows to capture the net distribution of contact forces. If the CoP is strictly within the boundary of the support polygon, the contact surface cannot rotate around its edges, thus preventing tip-over situations. In other terms, if this condition is respected a torque can be produced at the contact level in order to produce a balancing motion. This condition is widely used as the definition of the set of balanced states $\mathcal{K}^{\mathcal{B}}$ in balance controllers. A definition of a balance metric $\mathcal{B}$ is also generally derived from this condition as the distance between the CoP and the boundaries of the support polygon. However, this condition is not necessary: it is indeed over-conservative as stable dynamic walking motions can be produced with the CoP reaching the edges of the support polygon, with point feet for example. Nevertheless, its efficiency has been largely proved for flat-foot dynamic balance and walking.

A related, yet more general, form of the ZMP is introduced by A. Goswami in [Goswami 1999] as the Foot Rotation Indicator (FRI). Defined as the point on the ground where the net ground reaction force would have to act to keep the foot stationary, it can cross the boundaries of the support polygon if the foot is experiencing rotational accelerations. Therefore, it provides a metric capturing both positive and

negative distances to the stability margin.

The ZMP and FRI indicators however lack genericness as being specifically designed for biped locomotion scenarios. Although generalisations of these indicators can be found in the literature to extend them to more complex multi-contact situations, reverting to the essential definition of balance provides practical solutions to the definition problem of balanced states and metrics.

On the one hand, an instantaneous approach to balance considers the system to be dynamically balanced if there exist admissible contact forces that can support its motion, as illustrated in figure 2.6. Regarding the system as a whole, it essentially states that the Newton-Euler equations of motion (2.2) and contact mechanics are satisfied. A balance stability margin can henceforth be defined as the quantification of either the admissible motions around the current state or the disturbance wrenches that can be supported [Barthélemy 2008].

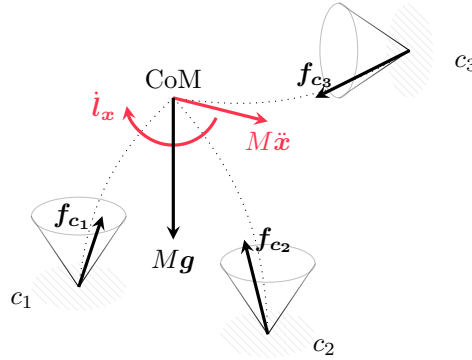


Figure 2.6: The system is instantaneously balanced if there exist admissible contact forces f_{c_i} that can support its motion $(\dot{i}_x, M\ddot{x})$.

On the other hand, a long-term approach to balance requires the consideration of viability. J. Pratt and R. Tedrake propose to this aim to approach viability through the notion of *capturability* in [Pratt 2006]. A state is said to be *capturable* if it is controllable to stable fixed-points, *i.e.* a stable state with a null kinetic energy can be reached from this state. The definition of the Capture Point (CP) is derived from this approach as the point that the CoP should reach in order to bring the system to a capturable state ; the CP must therefore lie within the support polygon in order to be reachable by the CoP. The Capture Point is not unique, and describes a capture region. A stability metric can therefore be derived as the overlap between the capture region and the support polygon, capturing the ability of the system to reach stabilising states.

2.3 Optimising balance at the planning level

Whole-body motion planning consists in generating open-loop joint space trajectories and, potentially, associated control inputs with the overall goal of reaching prescribed objectives³. This motion generation problem is subject to constraints related to the physics of the system both in terms of equation of motion and limits on various meaningful variables (joint position, velocity and torque, distance to obstacles, interaction wrenches, contact points,...). These constraints encapsulate the notion of balance.

³These objectives can be described in the most general way in terms of operational space objectives for some body-part of the robot, *e.g.* its hands.

Looking at this problem, it appears to be strongly non-linear and non convex both in terms of the overall cost function to optimise and the constraints to consider. The computational cost is thus important and highly sensitive to the degree of accuracy of the retained models. One way to address these types of optimisation problem is to resort to stochastic approaches aiming at finding feasible and potentially locally optimal solutions through smart or random sampling. Another approach consists in applying non-linear optimisation techniques leading to locally optimal solution as well but which are generally often influenced by an initial guess. As a matter of fact, the literature in this domain can clearly be divided into what can be called *stochastic* and *non-stochastic* whole-body motion planning even if many contributions mix both approaches. Some essential contributions in both domains are presented hereafter with a focus on the way the computational cost problem is addressed. Indeed, being able to plan fast is a desirable property.

2.3.1 Stochastic whole-body motion planning

Stochastic planning methods generally rely on sampling-based approaches used to explore the configuration space in order to find a feasible (constraint compliant) path from a starting configuration to a final one [LaValle 2006]. When dynamic constraints such as the ones imposed by balance come into play, a first possible approach can be to apply similar exploration techniques in the state space. However, doing so severely increases the dimensionality of the search problem which is already large with humanoids. Alternatively, a decoupled approach can be adopted where the problem is first solved at the kinematic level. The obtained path is then transformed in order to account for dynamic constraints. This is the approach retained in the pioneering work in [Kuffner Jr. 2002] where a collision free path is first computed using a standard randomised planner and then optimised under constraints in order to enforce constraints upon the centre of gravity projection and zero moment point trajectory in order to maintain balance. In this work, only feet contacts are considered and the contact state (stance) remains constant. The problem of finding a sequence of feasible supportive contacts and associated joint space configurations and contact forces is thus not tackled. To tackle this more complex problem, the methodology proposed in [Hauser 2008] also rely on a two-stage approach but the problem separation is of a different nature. Indeed, the first level of search is done at the stance level, trying to find a sequence of kinematically and statically feasible stances (with no restriction on the coplanarity of the feet contacts). This search is biased by a pre-computed motion primitives library aiming at improving the quality of the produced motions. These primitives are also used at the second level where a feasible path is found in configuration space between two consecutive stance transitions. The work in [Bouyarmane 2012] builds on this approach to tackle multi-contact (> 2) problems. At first, similarly to [Kuffner Jr. 2002], a collision free guide path is built using standard configuration space path planning techniques. While this path is collision free, it tries to minimise the distance to obstacles that could serve as potential supportive contacts. Then, a sequence of kinematically and statically feasible stances is searched for using an iterative, best-first algorithm. This algorithm favours stances inducing configurations close to the guide path. Kinematic and static feasibility is checked for using a non-linear constrained optimisation solver outputting an optimal joint space configuration and optimal contact forces under static balance and collision avoidance constraints. The found solution aims at minimising the distance to the pre-planned guide path, the required torque and contact forces as well as some cost favouring transition to the next stance. At run time, an optimisation based multi-objective whole body controller is used to ensure a dynamic execution of the overall motion while stance switching are monitored using a finite-state machine. The work in [Dalibard 2013] follows a similar two-stage logic and proves that any statically balanced, collision-free path for a legged robot sliding on the ground can be approximated by a dynamically balanced, collision-free walk trajectory. Even though it induces a limitation on the type of considered ground (planar), this property is exploited to simplify the search for a dynamically feasible trajectory.

2.3.2 Non-stochastic whole-body motion planning

Similarly to previous stochastic approaches, non-stochastic approaches do not directly tackle the whole-body planning problem without prior simplifications. In the seminal work presented in [Mombaur 2009], periodic and symmetric running motions are studied on an 11 degrees of freedom planar humanoid model which however contains a full description of the hybrid dynamics problem related to contact switching. Contrarily to the general definition of balance used throughout this chapter, balance appears in this work as an objective, together with energy consumption and a cost on the terminal state of the system. This balance objective aims at reaching a stable limit cycle by minimising the spectral radius of the Jacobian of the Poincaré map. The optimisation variable is composed of the evolution of the state, control input and stance switching instants. Robot morphology parameters are also optimised, this work being developed with robot design concern in mind. The problem is solved using what can be seen as a tailored Sequential Quadratic Program (SQP) based on MUSCOD, a multiple shooting algorithm for direct solution of optimal control problem [Bock 1984]. In [El Khoury 2013] and [Lengagne 2013] similar optimisation problems are solved but an initial guess is provided as a collision free path obtained using randomised planning techniques and a sequence of feasible contact stances obtained using the work in [Bouyarmane 2012] respectively. These initial guesses remove the need for considering a full description of the hybrid dynamics and more complex humanoids can be considered, especially as much less dynamic motions are considered. More recently, a similar non-linear optimisation approach is retained in [Kuindersma 2015] where the simplification at the planning level lie in the use of a full model description for kinematics constraints whereas only the free-floating dynamics is considered for the equation of motion. While the evolution of the state, contact locations, contact forces, stances transitions and centre of mass dynamics can be computed, joint level torque inputs cannot be generated using such a technique. However the robot is assumed to have large actuation capabilities and the actual computation of the control inputs is performed at the reactive level using an optimisation based multi-objective whole body controller.

2.4 Optimal regulation of balance tasks

Translating specified objectives into balance-consistent references is a first step towards the conservation of balance during the execution of activities. Nevertheless, unexpected disturbances from modeling errors or external perturbations must be considered in order to maintain balance. State feedback approaches are consequently employed to compute the actuation input tracking these references, while ensuring balance with a view to the current state of the system. The complexity of legged robots dynamics (2.1) however tends to lead to computationally inefficient formulations of the balance problem. A common approach is therefore to introduce an intermediate level refining open-loop references into closed-loop objectives.

To this aim, task regulators are introduced accounting for state feedback, while using reduced models to reduce the computational cost. In the case of balance, the most widespread reduced model capturing the dynamics of the system as a whole is described by the Newton-Euler equations of motion (2.2). Optimisation-based approaches are commonly employed at this level of abstraction to additionally enforce balance constraints in a closed-loop formulation. Key formulations are presented in this section in order to apprehend common techniques to approach the balance problem at this level.

2.4.1 Model Predictive Control

A common optimal approach to balance regulation is Model Predictive Control. MPC employs a model of the system dynamics to preview its behaviour over time and compute an optimal horizon of control inputs maintaining the system into the valid state space.

When writing the balance control problem over a future horizon as described in problem (2.13), several obstacles are met as it requires a time-integration of the local system model (*e.g.* the equations of motion (2.1)). Indeed, the relation between the local model parameters (*e.g.* the inertia matrix $\mathbf{M}$) and the local actuation input u is generally non-linear, leading to non-linear and non-convex optimisation problems. Integrating the local model furthermore implies the resolution of the contact forces which brings additional non-linearities and the dimension of the actuation input further increases the computational cost of the problem. A straightforward approach to bypass these obstacles is to consider a reduced local model of the system in order to decrease its complexity and dimensionality. Additionally, considering the contact regimes as predefined over the problem horizon allows to discard the complexity raised by contact dynamics.

Employing a reduced model, such approaches therefore output optimal control inputs defined at a level of abstraction higher than the actual actuation of the system. These control inputs are thereafter used as refined reference trajectories to be tracked at the whole-body level.

ZMP-based formulations The interest for MPC techniques in the humanoid robotics literature was mainly initiated by S. Kajita *et al.* in the form of Preview Control [Kajita 2003]. In this work, the control problem is reduced to the centre of mass of the system and is written over a finite time horizon as schematised in equation (2.16). Considering the centre of mass jerk as the control input at this level, a linear model can be induced under some assumptions. The main hypotheses of this formulation are that the centre of mass keeps a constant altitude over the preview horizon and that the rate of change of angular momentum of the system is negligible. Balance is approached in this work through the ZMP model and criterion, with its common assumptions: contacts with the environment are coplanar, without any constraints on contact forces. The control problem in this formulation is solely written as the maximisation of balance, defined as a tip-over risk with the ZMP criterion, through the computation of an optimal horizon of centre of mass jerks. While this approach proves efficient in stabilising the dynamics of the centre of mass in walking motions, one of its major flaws is the lack of balance guarantee.

Wieber therefore proposed [Wieber 2006] further developments of this formulation to account for constraints on the ZMP position and subsequently allow the consideration of stability margins.

The original, unconstrained Preview Control was in parallel subject to extensions to account for external disturbances from the environment. S. Kanzaki *et al.* for example propose in [Kanzaki 2005] to directly account for an expected external impacts on the robot in the reduced dynamics model. In [Ibanez 2012] this approach is further developed to simultaneously optimise balance and manipulation tasks in order to maximise their respective performance with respect to known external forces applied to the hand of the robot. In order to include the effects of the interaction of the robot with its environment, required by the manipulation task in the balance control, a distributed preview control is introduced which captures both balance and manipulation behaviours and enables the regulation of the interaction impedance. The initial ZMP preview control is extended to take into account the disturbance resulting from the manipulation task and the preview control of adaptive impedances used to drive the upper limbs. The resulting behaviour is illustrated in a simple scenario. Its aptitude to dynamically extract an optimal control strategy improving tracking performances of both manipulation and balance tasks is also assessed when complex perturbations have to be compensated.

Another approach discards the constant CoM altitude assumption which allows to additionally include friction considerations in the balance problem [Ibanez 2014a]; the extra computational cost being reduced with the formulation of a distributed optimisation problem. More precisely, given that the control of the horizontal dynamics of the centre of mass can withstand limited perturbations, postural stability criteria are specified with respect to the robot centre of mass vertical and horizontal dynamics, and to the angular dynamics of its torso. Formulating the balance problem in a predictive form and distributing it at different

time scales significantly increases the robustness of the system to external disturbances, in terms of both tip-over and slippage risks. This original control architecture is validated through the simulation of an iCub robot performing a walking activity under unknown external actions.

These approaches nevertheless require the predefinition of a reference trajectory of the ZMP, defined through a prior choice of feet positions over the preview horizon ; feet positions being the outputs of a planning process for example. Also, in such formulations the optimisation problem solely considers a balance objective, and no *task* objective is specified. To enhance the flexibility of the balance controller, developments were performed in order to include feet positions as variables of the optimisation problem [Diedam 2008]. These additional degrees of freedom in the control problem allow to consider a task objective in the form of the tracking of a reference centre of mass velocity, in addition to the maximisation of balance. Besides, they can be exploited to recover from larger disturbances.

This approach has been subject to multiple developments. Stephens and Atkeson for example proposed in [Stephens 2010] to extend this formulation to a force-control framework. A. Herdt *et al.* preceded the control problem with an additional optimisation problem aiming at computing optimal orientations of the feet in order to track an angular velocity with the trunk [Herdt 2010], thus improving the capacities of the controller to recover from large disturbances. To further increase the flexibility of the balance controller and reduce the influence of a planning process, developments were also proposed in the recent literature to additionally consider the duration and instants of the steps in the optimisation problem. In [Ibanez 2014b], in order to compute optimal time, duration and position of footsteps along with the centre of mass trajectory of a humanoid, a novel mixed-integer model of the system is presented. The introduction of this model in a predictive control problem brings the definition of a Mixed-Integer Quadratic Program, subject to linear constraints. Simulation results demonstrate the simultaneous adaptation of the gait pattern and posture of the humanoid, in a walking activity under large disturbances, to efficiently compromise between task performance and balance. In addition, a push recovery scenario displays how, using a single balance-performance ratio, distinct behaviours of the humanoid can be specified. The corresponding paper can be found in Appendix B.4. This work is also the object of an experimental validation on the TORO humanoid robot at the DLR. A video summarising the proposed approach can be seen [here](http://pages.isir.upmc.fr/~padois/website/fichiers/videos/2015_ibanez_jnrr.mp4)⁴.

Other criteria MPC formulations based on the ZMP criterion have also been extended to other criteria, such as the capture point in [Krause 2012]. A more generic approach can also be found in [Henze 2014] where the balance problem is directly written as the optimisation problem of contact forces acting on the system, thus allowing the consideration of multi-contacts scenarios. However, this genericness is obtained at the cost of computational impracticability.

Similarly, N. Perrin *et al.* propose in [Perrin 2015] to consider multi-contacts scenarios. More precisely, this work introduces two simple and novel approaches to solve for 3D locomotion with multiple non-coplanar contacts. Both formulations use model predictive control to generate dynamically balanced trajectories with no restrictions on the centre of mass height trajectory. The first formulation treats the balance criterion as an objective function, and solves the control problem using a sequence of alternating convex quadratic programs. The second formulation considers the criterion as constraints, and solves a succession of convex quadratically constrained quadratic programs (QCQPs). These formulations are more general than the classical Inverse Pendulum + ZMP approach, but simple enough to enable fast computations of CoM trajectories through an iterative resolution of convex QPs or convex QCQPs. This claim is supported by the low number of decision variables and constraint equations shown in the problem analysis. The generalisations gained from the proposed model and MPC approach include the ability to allow for multiple non-coplanar contacts and not having to predefine the CoM height trajectory. The results of the two proposed MPC approaches support

⁴http://pages.isir.upmc.fr/~padois/website/fichiers/videos/2015_ibanez_jnrr.mp4

the belief that the proposed 3D model of dynamically balanced locomotion is a good candidate for real-time model predictive control for multi-contact locomotion.

2.4.2 Instantaneous formulations

Optimisation-based approaches are nevertheless not restricted to MPC methods at this level. Open-loop references can also be refined at the instantaneous level and translated into instantaneous acceleration or force objectives. The computational burden being reduced with such formulations over a single step horizon, complex scenarios can be practically considered.

With the objective of distributing a reference value of the balancing net contact wrench over multiple contact points, while respecting friction constraints, C. Ott *et al.* propose in [Ott 2011] an optimisation-based formulation of the balance problem focused on the centre of mass dynamics. With this model, any contact configuration can be considered. The resulting problem is written as a QP, aiming predominantly at minimising the tracking error of the desired net wrench. This intermediate control problem outputs a set of contact forces, considered as objectives to be tracked at a lower-level for each contact points. Lee and Goswami similarly propose to focus on a model of the dynamics of the system reduced to its centre of mass in [Lee 2012b] while considering each individual contact wrenches instead of a net contact wrench. This approach refines at this level desired rates of change in linear and angular momenta into admissible ones, with respect to the foot geometry and contact constraints. In order to keep the computational cost of this problem as low as possible, the authors perform a sequential distribution of the original optimisation problem to obtain two successive QP problems.

2.5 Optimisation-based whole-body control

Desired references and objectives, generally expressed at a relatively high level of abstraction, needs to be translated at the actuation level to compute the corresponding actuation control inputs.

2.5.1 Local optimality

A widespread optimal approach to balance at the actuation level is to solve the inverse kinematics or dynamics problem under balance constraints. At this level, balance constraints are generally restricted to the consideration of contacts stability and coordinated centre of mass accelerations.

On the one hand, this problem has historically been solved using analytical methods based on null-space projection techniques and accounting for constraints as avoidance tasks. Nonetheless, as explained in Chapter 1, this approach is doomed to fail: these constraints-related tasks aim at getting away from the constraints (active avoidance). The number of constraints being potentially higher than the number of DOF, they cannot lead to control solutions that can strictly guarantee the respect of all constraints. On the other hand, whole-body controller are written as a quadratic multi-objective optimisation problem under linear constraints where priorities between the objectives can be dealt with through strict or soft hierarchies. The resulting optimisation problem is therefore to compute optimal actuation inputs which maximise the tracking performance of desired references while respecting balance constraints. The logic behind this choice is straightforward. First, the equation of motion and joint space to task spaces mappings can be written as equalities but they are not sufficient to describe the overall dynamics and physical behaviour of a robot. Indeed, other intrinsic physical constraint have to be accounted for at the joint level as well as in Cartesian space. These constraints do not solely describe relationships between physical quantities but also limits which cannot (control input saturation) or should never be crossed in order to maintain the robot and its environment in proper working conditions. Theses limits translate into inequalities. Assuming a convex solution space, the optimal solution of the control problem lie at the boundary of the feasible (constraint compliant) solution space. Finding the optimal solution thus boils down to finding the active constraint set,

i.e. on which boundary it lies. Optimisation problem solvers are designed to optimally choose this subset of constraints that should be considered when computing the optimal solution of the control problem. The strong mathematical background in convex optimisation is such that optimisation based methods mostly outperform analytical methods attempting to heuristically activate constraints.

In the seminal work in [Faverjon 1987], an obstacle avoidance technique includes a control law structure based on a Quadratic Program (QP). Since then, control approaches relying on optimisation tools such as Linear Quadratic Problem solver have emerged for virtual humans [Abe 2007, Collette 2007]. For real humanoids, to deal with prioritised inequality constraints more easily, hierarchical quadratic programming (HQP) approaches use numerical QP solvers to solve a Hierarchical Quadratic Program [Kanoun 2009]. The idea of HQP is to first solve a QP to obtain a solution for a higher priority task objective; and then to solve another QP for a lower priority task, without increasing the obtained minimum of the previous task objective. This prioritisation process corresponds to solving at best lower-priority tasks in the null-space of higher-priority tasks. The HQP algorithm is applied for solving prioritised inverse dynamics and whole-body motion control under unilateral constraints [Saab 2013]. It requires to solve as many QPs as priority levels, which can be quite time consuming. The computation cost of hierarchical inverse kinematics with inequality constraints is improved by an algorithm developed in [Escande 2014], which permits real time control of the HRP-4 humanoid robot. Similar work is performed in [Herzog 2014] where a reduction of the equation of motion allows for real time control of the SARCOS humanoid robot.

Generally, for an approach based on strict hierarchy, the relative importance of one task with respect to another one of different priority level is parametrised in a binary way: either strictly higher or strictly lower (lexicographic order). However, in many contexts, organising tasks by assigning them a lexicographic order is not generic, *i.e.* can have some limitations. First, a strict priority is just an extreme case of the relations of task importance levels. In fact, a task may not always have a strict priority over another one and it is usually difficult to define a strict hierarchy among some of the tasks. Second, strict priorities can sometimes be too conservative so that they may completely block lower-priority tasks. Third, a change in the task set, such as a swap of task priorities, may lead to discontinuity. An approach to smooth priority rearrangement between two tasks is proposed in [Keith 2011, Petrič 2013] and approaches for continuous and simultaneous transitions of multiple tasks are developed in [Mansard 2009c, Lee 2012a]. A specific inverse operator is proposed in [Mansard 2009c] to ensure continuous inverse in the analytical computation of control laws. The approach presented in [Lee 2012a] is based on intermediate desired values in the task space. When applied to humanoids, the number of tasks and the state dimension are such that the computational cost implied by these approaches is too high for practical use.

Smooth task transitions can be easily achieved within a framework using a weighting strategy [Abe 2007, Collette 2007, Bouyarmane 2011b] by the continuous variation of task weights [Salini 2011]. These control frameworks solve all the constraints and task objectives in one QP and provide a trade-off among task objectives with different importance levels. In a nutshell, the control problem is formulated as follows:

$$\begin{aligned} & \underset{\mathbb{X}}{\operatorname{argmin}} \quad \sum_i \omega_i T_i(\mathbb{X}) \\ & \text{s.t.} \quad \begin{cases} M(q)\dot{\nu} + n(q, \nu) + g(q) = S^T \tau - \sum_j J_{c_j}^T(q) w_{c_j} \\ G\mathbb{X} \preceq h \end{cases} \end{aligned} \quad (2.17)$$

where $\mathbb{X} = (\tau^T, w_c^T, \dot{\nu}^T)^T$. The equality constraint is the equation of motion. The inequality constraint includes the bounds on the joint positions, velocities, and torques (all formulated in terms of τ and $\dot{\nu}$), as

well as the contact existence conditions for each contact point according to the Coulomb friction model:

$$\begin{aligned} \mathbf{C}_{c_j} \mathbf{w}_{c_j} &\leq \mathbf{0} \quad \forall j \\ \mathbf{J}_{c_j}(\mathbf{q}) \dot{\boldsymbol{\nu}} + \dot{\mathbf{J}}_{c_j}(\boldsymbol{\nu}, \mathbf{q}) \boldsymbol{\nu} &= \mathbf{0} \quad \forall j \end{aligned} \quad (2.18)$$

where $\mathbf{C}_{c_j}$ is the linearised friction cone of the j -th contact point.

The objective function is a weighted sum of tasks T_i (weights ω_i) representing the squared error between a desired acceleration or wrench and the system acceleration/wrench. The solution is then a compromise between the different tasks, based on their relative importance. The following types of tasks are generally considered:

- Operational space acceleration $\|\mathbf{J}_i \dot{\boldsymbol{\nu}} + \dot{\mathbf{J}}_i \boldsymbol{\nu} - \ddot{\mathbf{x}}_i^*\|^2$
- Joint space acceleration $\|\dot{\boldsymbol{\nu}} - \dot{\boldsymbol{\nu}}^*\|^2$
- Operational space wrench $\|\mathbf{w}_i - \mathbf{w}_i^*\|^2$
- Joint torque $\|\boldsymbol{\tau} - \boldsymbol{\tau}^*\|^2$

where $\ddot{\mathbf{x}}_i$ is the Cartesian acceleration (time derivative of a twist) of body i , and $\mathbf{w}_i$ the wrench associated with body i . The superscript $*$ refers to the desired acceleration/force. The desired acceleration can be defined by a proportional derivative control⁵:

$$\ddot{\mathbf{z}}^* = \ddot{\mathbf{z}}^{goal} + \mathbf{K}_v(\dot{\mathbf{z}}^{goal} - \dot{\mathbf{z}}) + \mathbf{K}_p(\mathbf{z}^{goal} - \mathbf{z}) \quad (2.19)$$

where $\mathbf{z}$ stands for $\mathbf{x}$ or $\mathbf{q}$, and $\mathbf{K}_p$ and $\mathbf{K}_v$ are the proportional and derivative gains. The superscript *goal* indicates the position, velocity and acceleration wanted for the body or joint (reference trajectory).

As the performances of higher priority tasks cannot be guaranteed by simply adjusting the weights of task objectives, a prioritised control framework is proposed in [Liu 2012b] to ensure the performance of a higher-priority task within a user defined tolerance margin. However, this approach handles priorities of only two levels. The work of M. Liu described in section 1.2 proposes a Generalised Hierarchical Framework allowing to describe both soft and strict priority problems with smooth priority transitions. It is also specifically applied to the humanoid cases [Liu 2016a].

Towards tasks optimisation Recently, the PhD thesis work of R. Lober has questioned the notion of priorities: while some priorities are needed for robustness and safety reasons, trying to achieve incompatible tasks at the reactive level does not make sense. Thus, on-line task optimisation is needed in order to feed reactive whole-body controllers with references which are compatible one with another and more importantly with constraints, among which balance is a very important one. Pursuing this goal, a novel framework is introduced in [Lober 2014] for defining and optimising multiple tasks in order to resolve potential interferences prior to task execution and remove the need for prioritisation. This framework parametrises tasks with Dynamical Movement Primitives, simulates and evaluates their execution, and optimises their parameters based on a general compatibility principle, which is independent of the topology of the robot, tasks or environment. Two test cases on a simulation of a humanoid robot are used to demonstrate the successful optimisation of initially interfering tasks using this framework. The current major drawback is the computational cost of the task execution simulations. Because each update of the task parameters requires a number of simulations to be executed, the average time needed to optimise a set of tasks is too long for a real-world implementation. The on-going work of R. Lober aims at reducing this computational complexity, using efficient dynamic simulation tools⁶ but more notably Bayesian optimisation techniques. R. Lober has

⁵the “—” operator is used here for the sake of simplicity but its meaning is of course related to the physical nature of the corresponding quantity and is a function of the retained parametrisation.

⁶The synthesis of a survey on existing dynamic simulation tools dedicated to Robotics can be found in [Ivaldi 2014a] while the complete results are accessible in [Ivaldi 2014b].

also explored a quite simple idea in [Lober 2015]: using task variability to modulate task priorities during execution, to temporarily deviate certain tasks as needed, in the presence of incompatibilities. The proposed method includes a mapping from task variance to task priority and then provides an approach for computing task variance. Through three common conflict scenarios, it is shown that mapping from task variance to priorities reactively solves a number of task incompatibilities. The corresponding article can be found in Appendix B.5 and the most illustrative video associated to this work can be seen [here](http://pages.isir.upmc.fr/~padois/website/fichiers/videos/lober_IROS2015.mp4)⁷.

2.5.2 Whole-body MPC

While the combination of instantaneous whole-body approaches with reduced MPC task controllers provides the control system with the consideration of both a future horizon and the whole-body dynamics of the system, their sequential layout is necessarily suboptimal. MPC approaches are therefore gradually extended to whole-body control with the consideration of models capturing the system state at the joint level. Computational limitations nevertheless requires the setup of simplifications of the control problem. Indeed, as discussed in Sec. 2.4, time-integration is one of the major challenges of MPC formulations.

A significant improvement in this direction was performed by Y. Tassa *et al.* in [Tassa 2012]. The authors address computational issues in two ways in this work.

First, the resolution of the MPC optimisation problem is considered over the whole duration of the activity. That is, the control problem solely aims at continuously driving the system towards an all-time minimum, rather than finding an optimum at each control step. To this aim a suboptimal solution solely is demanded at each control step, generally obtained through a single iteration of the optimisation algorithm. It therefore allows the consideration of much more complex optimisation problems: explicit formulations of models are indeed not required as an update for a given control input is solely needed. In this particular work, an iterative Linear Quadratic Gaussian optimiser, a variant of Differential Dynamic Programming, is setup to work over the joint actuation space. Computational issues nevertheless arise from the needs to update the model rapidly for an iteration of the optimiser.

To meet this challenge, the second contribution of this work is thus to propose a fast dynamics integrator. Following the remark that frictional contacts are mainly responsible for the dynamics computational burden, the authors consider continuous contact models ignoring the discontinuous contact dynamics and propose a cheap smooth model producing realistic behaviors for the time scales of the MPC problem.

Balance is simply approached in [Tassa 2012] through a static metric on the CoM. Despite the use of this highly conservative balance criterion, dynamic and complex behaviors involving the whole-body of the system are discovered to track a target pose of the torso. However this controller required a time slowdown of $\times 7$ with optimisation iterations of 140ms, thus forbidding a direct real-time implementation. J. Koenemann *et al.* nevertheless described in [Koenemann 2015] a real-time implementation of this controller using a double rate control architecture. In this work, the MPC problem is employed as a trajectory generator running at 10 times slower than the rate of the lower, tracking control loop.

I. Mordatch *et al.* in [Mordatch 2012] also consider regularisation of contact dynamics as a solution to the computational issues it raises. This approach is however fundamentally different ; indeed, in order to avoid the costly computation of contact states from control inputs (performed by the dynamics simulator in the works of Y. Tassa *et al.*), they are explicitly considered in the optimisation problem as control variables. More specifically, the activation and deactivation of contacts is written as thresholds over a continuous variable describing whether a contact should be activated or not. However, the resulting formulation is not computationally-efficient enough to allow real-time implementations.

⁷http://pages.isir.upmc.fr/~padois/website/fichiers/videos/lober_IROS2015.mp4

2.6 Conclusion

Balance is an essential feature of humanoids but, despite a strong understanding of its laws and dynamics, it remains an open problem for control applications. Optimisation-based control approaches to balance explicitly include balance dynamics and constraints in the control problem in order to capture at best the behaviour of the system and fully exploit it to reach complex control objectives. Although theoretically appealing, these approaches intrinsically induce a significant computational burden. In practice, this implies to resort to simplifications on the model and problem complexities, which limits the capacity to actually generate complex behaviours. While whole-body Model Predictive Control stands out as a promising framework, it also suffers from the need for compromises. Its potential is, in the current state of the art, hindered by the difficulty to efficiently integrate the non-linear dynamics of humanoid systems. Future developments in this direction could undoubtedly provide keys to widen the scope of humanoids applications.

Chapter 3

Performance based design and evaluation of robots

Note on this chapter

The contribution presented in this chapter is adapted from an article entitled “Human-oriented design of collaborative robots” and recently submitted to the [International Journal of Industrial Ergonomics](#). This article has been written together with P. Maurice and partly builds up on her PhD thesis work [[Maurice 2015](#)].

“On n’écrase pas une mouche avec un marteau-pilon.”

This short French proverb summarises well the motivations behind the work presented in this chapter: using the right tool renders the task achievement easier. From a control point of view this has an even more specific interpretation: a robot designed with tasks and constraints in mind will be easier to control and is more likely to perform well.

For industrial robots used in traditional manufacturing applications, the notion of performance is mostly characterised by three features:

- precision and repeatability;
- maximum velocity and acceleration capabilities;
- energetic efficiency.

The latest of these three features is clearly antagonistic with the first two and trade-offs have to be made at the design level. However, “standard” robotic applications have been studied for several decades and the morphology and topology of robots for a given conventional application can easily be decided for based on previous experience and well established system design and engineering rules. This unfortunately does not apply to any potential use of robots. There are two reasons for this:

1. the expected workspace of the robot may be so unconventional and/or the nature of the tasks to be executed so diverse that no existing robot type can fit by default. Adapting existing solutions may actually restrain the design process to reach really effective designs.

2. The notion of performance, as described here before, may not be appropriate for the considered application. This is for example the case of collaborative robotics application where the presence of human beings induces a shift in the objectives pursued by introducing a robot in the application. In other words, the notion of performance may have to be redefined.

The first case actually corresponds to a problem that S. Rubrecht has tackled during his PhD thesis [Rubrecht 2011a]. His thesis work was hosted by Bouygues Travaux Publics, within the framework of TELEMACH, an R&D project dedicated to the feasibility of tele-operated maintenance in Tunnel Boring Machines (TBMs). TBMs are widely used for the excavation of small (1m diameter) to large tunnels (15m diameter and more). In this context, the task based design of the morphology of a robotic manipulator evolving in a cluttered environment was one of the central research topics. Indeed, the workspace of an excavation chamber is highly cluttered, so usual design techniques are inefficient and time-consuming. Moreover, the complexity of the TBM environment and the tasks to be accomplished induce a high number of potential design solutions among which the best ones may, given their originality with respect to usual design problems solutions, probably not arise using classical design methods. The use of dedicated CAD tools may help to numerically discard some of the potential solutions, but checking each robot candidate with respect to a representative subset of tasks and environments still remains a complex and time consuming work. Instead, this work proposed to follow an approach where the design process is considered as a multi-objective optimisation problem: tasks and constraints are formulated in terms of functions to optimise and constraints to satisfy. Such a formulation allows the automation of the design process in the preliminary phase. Given a family of automatically obtained solutions, the so-called classical design methods can then be used to converge towards a practical solution. The retained design process is an example of task-based design carried out thanks to an evolutionary process where robot morphologies are generated thanks to a genetic algorithm which evaluation step (fitness function) aims at qualifying the ability of a robot to carry out a maintenance mission in the TBM. The obtained results using such an automatic design approach in this context are presented in [Rubrecht 2011b] as well as in [Rubrecht 2011a] with more details.

The second case is central in the PhD thesis work of P. Maurice [Maurice 2015]. This thesis focuses on the design of collaborative robots with ergonomics as a central concern. In fact, while dimensions related to the nature of the workspace and of the tasks to be executed are present in the design process of collaborative robots, it is really the shift in the notion of performance which renders the design problem for such systems difficult. This chapter presents a generic method for performing detailed ergonomic assessments of co-manipulation activities and its application to the optimal design of collaborative robots. Multiple ergonomic indicators are defined to estimate the different biomechanical demands which occur during manual activities. For any given activity, these indicators are measured through dynamic virtual human simulations, for varying human and robot features. Sensitivity indices are thereby computed to quantify the influence of each parameter of the robot and identify those which should mainly be modified to enhance the ergonomic performance. The sensitivity analysis also allows to extract the indicators which best summarise the overall ergonomic performance of the activity. An evolutionary algorithm is then used to optimise the influential parameters of the robot with respect to the most informative ergonomic indicators, in order to generate an efficient robot design. The whole method is applied to the optimisation of a robot morphology for assisting a drilling activity.

3.1 Context

Work-related musculoskeletal disorders (MSDs) represent a major health problem in developed countries. They account for the majority of reported occupational diseases and affect almost 50 % of industrial workers [Schneider 2010]. Since MSDs result from strenuous biomechanical solicitations [Luttmann 2003], assisting workers with collaborative robots can be a solution when a task is physically demanding yet too complex to be fully automatised (Fig. 3.1); a collaborative robot enables the joint manipulation of objects with the

worker (co-manipulation) and thereby provides a variety of benefits, such as strength amplification, inertia masking and guidance via virtual surfaces and paths [Colgate 2003].

In order to design a robot which decreases at best the risk of developing MSDs, an ergonomic assessment of the robot-worker system must be performed throughout the design process. Though standard ergonomic assessments are based on the observation of a worker performing the task [Li 1999, David 2005], digital evaluations now tend to replace physical evaluations in the design process of workstations; digital evaluations – in which a digital human model (DHM) is used to simulate the worker – indeed present several major advantages [Chaffin 2007]. Firstly, the simulation enables easy access to detailed biomechanical quantities, which otherwise can only be measured on real humans through complex instrumentation, if at all (*e.g.* muscle or joint forces). Secondly, different morphologies of workers can easily be tested without the need for a wide variety of real workers. And thirdly, a virtual – instead of a physical – mock-up of the robot is used for digital assessments, thus removing the need to build a new prototype every time a parameter of the robot is tuned. The overall development time and cost is thereby decreased.



Figure 3.1: A collaborative robot providing strength amplification for tire retreading (developed by RB3D, CEA-LIST, CETIM).

To perform digital ergonomic evaluations, several commercial DHM software for workplace design provide ergonomic analysis tools (*e.g.* Delmia, Jack [Raschke 2004], Ramsis [Seidl 2004], Sammie [Porter 2004]). These software – based on simple rigid-body models of the human body – include standard assessment methods which estimate an absolute level of risk depending on the main MSDs factors [Luttmann 2003] (posture, effort, duration and frequency of the task) and possibly additional factors (*e.g.* RULA [McAtamney 1993], REBA [Hignett 2000] and OWAS [Karhu 1981] methods, OCRA index [Occhipinti 1998], NIOSH equation [Waters 1993]). The resulting ergonomic indicators are, however, either very rough (*e.g.* effect of external load in RULA) and/or task-specific (*e.g.* NIOSH equation for lifting loads), so they do not accurately cover all kinds of manual activities which may be addressed by collaborative robots. Besides, these assessment methods are static, *i.e.* dynamic phenomena are not taken into account; yet fast motions do increase the risk of developing MSDs [Marras 1993]. In collaborative robotics, evaluating the dynamic phases of an activity is even more important because the robot is never perfectly backdrivable and some phenomena cannot be compensated even with a dedicated control law (*e.g.* additional inertia); manipulating the robot might then require extra efforts and cause new MSDs.

Concurrently to DHM software for workplace design, other DHM software provide more accurate musculoskeletal models of the human body, including muscles, tendons, and bones (*e.g.* OpenSim [Delp 2007], AnyBody [Damsgaard 2006], LifeMOD). Beyond classic macroscopic measurements (joint angles, joint forces and moments), these software also provide dynamic measurements (joint velocities and accelerations) and quantities that more accurately account for the biomechanical demands on the human body (muscle force, tendon deformation, muscle fiber length...). The high number of outputs (one for each muscle/tendon/joint) is, however, difficult to interpret without specific biomechanical knowledge, especially when the purpose is to summarise the global ergonomic level of the activity.

The second criticism which can be addressed to both kinds of DHM software concerns the animation of the DHM. The DHM motion is generated through forward or inverse kinematics, pre-defined postures and behaviours (*e.g.* *walk towards*, *reach towards*), or from motion capture data. Apart from motion capture, none of these animation techniques enables to come up with a truly realistic human motion. Kinematic techniques do not take into account the inertial properties of the human body or external load, so the simulated motion is rarely human-like [Chaffin 2007]. Pre-defined behaviours result in more realistic motions since they rely on a pre-recorded motions database, but only a limited number of behaviours can be simulated and they become unrealistic when external conditions are modified (*e.g.* adding a load in a reaching motion). In general, the obtained motion is not even dynamically consistent. For instance, the DHM balance is never considered though it affects the relevance of the evaluation [Lämkkull 2009]. As for motion capture, the human subject and the avatar must experience a similar environment to obtain a realistic simulation. In particular, the interaction forces with the environment are crucial, so the subject must either be provided with a physical mock-up (Fig. 3.2) or be equipped with complex instrumentation (digital mock-up through virtual reality and force feedback devices). Motion capture is therefore highly time and resource consuming.



Figure 3.2: Animation of a DHM using motion capture data, with the Jack software (picture from Jack documentation). The human subject is placed in a physical mock-up of the environment in order to obtain realistic motions.

In order to circumvent the above-mentioned issues, G. De Magistris *et al.* developed an optimisation-based DHM controller to automatically simulate dynamically consistent motions [De Magistris 2013]. The dynamic controller computes DHM joint torques from a combination of anticipatory feed-forward and feed-back control. It has many advantages over kinematics techniques, such as ensuring DHM balance and

generating hand trajectories that are in accordance with some psychophysical principles of voluntary movements. However, though this controller has been successfully used for a virtual ergonomic assessment, the Jacobian-transpose method used in the feedback control does not guarantee the optimality of the solution, because joint torques limits cannot be explicitly included in the optimisation.

Eventually, evaluating the ergonomic benefit provided by a collaborative robot requires that the robot be included in the DHM simulation. Though most DHM software can simulate a DHM within a static environment, they cannot simulate the motion of a collaborative robot which depends on its physical interaction with the DHM, both through its control law and through physical interferences.

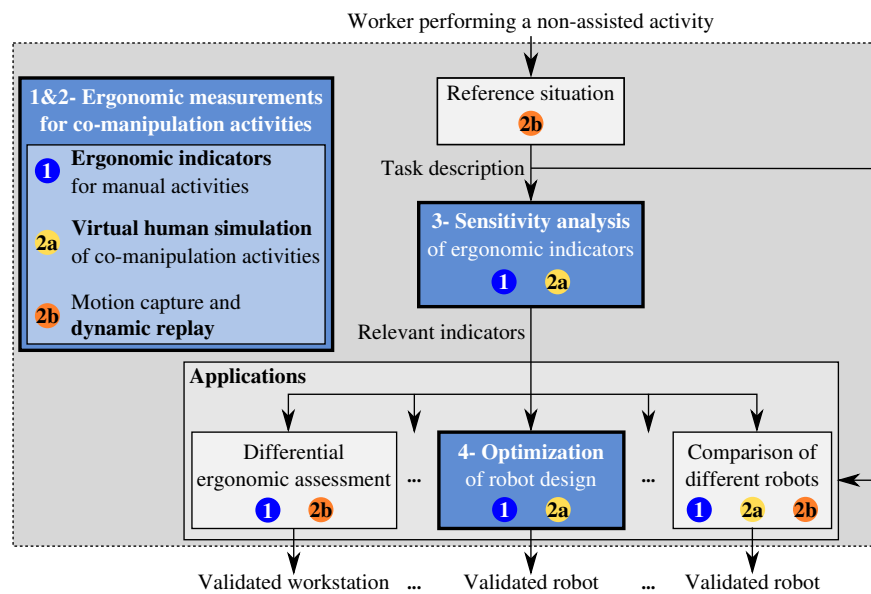


Figure 3.3: Overview of the methodology developed for performing ergonomic assessments of collaborative robots, and its applications. This paper focuses on the optimisation of robot design (4), but other applications are possible.

Thus, despite many available tools for performing virtual ergonomic assessments, none of them is suitable to evaluate co-manipulation activities. This work therefore presents a novel approach for quantitatively comparing the ergonomic benefit provided by different collaborative robots when performing a given activity, and its application to the optimal design of such robots. The proposed method consists in four components (Fig. 3.3):

1. A list of ergonomic indicators defined to accurately account for the different biomechanical demands which occur during manual activities. They cover all kinds of manual activities, without requiring any *a priori* hypotheses on the activity that is performed.
2. A dynamic simulation framework in which a DHM can interact with a controlled collaborative robot. The simulation is used to measure the ergonomic indicators. The DHM is animated through an optimisation-based whole-body controller to ensure the dynamic consistency of the motion. The controller can be used either with high level tasks descriptions (autonomous DHM, 2a), or with motion capture data (2b). 2a enables the evaluation of robots under development without the need for a human subject or physical mock-ups, while 2b allows the replay of a recorded activity to acquire a reference situation (non-assisted gesture) or evaluate existing robots.

3. A sensitivity analysis framework with which the relevance of each ergonomic indicator and its dependence on the robot parameters can be established – for any given activity – without the need for much input data. The analysis enables the identification of the indicators which best summarise the overall ergonomic performance, and of the robot parameters which most affect this performance. The aforementioned simulation framework is used to automatically create and simulate a variety of situations.
4. A framework for optimising design parameters of a collaborative robot with respect to relevant ergonomic indicators, based on a multi-objective evolutionary algorithm.

Thanks to the proposed tools, comparing and optimising the ergonomic benefit provided by collaborative robots is facilitated. The technical gesture is acquired on the initial situation and serves as an input for the sensitivity analysis. The sensitivity analysis enables the identification of a small number of ergonomic indicators relevant for the comparison of robots performances, as well as of the robot parameters to primarily optimise. An optimal robot is then designed with the evolutionary tool by optimising the relevant ergonomic indicators. The initial (non-assisted) and final (with the optimised robot) situations can eventually be compared, to ensure the benefit provided by the robot.

3.2 Method

In digital human modelling, the human body can be represented with different level of detail (rigid bodies, muscles...). The chosen model, however, affects both the biomechanical quantities measured on the model – hence the formulation of ergonomic indicators – and the controller generating the DHM motion (definition of the actuation variables). In this work, the human body is represented with rigid bodies and does not include muscle actuation; each joint is controlled by a single actuator. Even though muscle-related quantities cannot be estimated with such a model, numerous other quantities can be measured to represent the biomechanical demands that occur during whole-body activities (*e.g.* joint loads, joint dynamics, mechanical energy...). Besides, given the high actuation redundancy of the human musculoskeletal system, computing muscle forces requires to solve the muscle recruitment problem (*i.e.* which muscles should be activated – among the infinity of possible activation patterns – to perform a given motion). While musculoskeletal models have proved valid and insightful in specific cases, no general criterion has been established yet for the muscle recruitment problem. The realism of the muscle-related measurements can therefore not be ensured in all possible whole-body situations [Damsgaard 2006, Thelen 2003, Chaffin 2006, Hicks 2015]. The questionable gain of information and the significant computational cost then reduces the interest of musculoskeletal models in the current context.

3.2.1 Ergonomic indicators for collaborative robotics

Ergonomic indicators aim at quantifying exhaustively and concisely the physical demands endured by a worker when executing various manual activities, with or without a collaborative robot. Such indicators should take into account the main MSDs risk factors considered in standard ergonomic assessments (posture, force...), but also phenomena that are usually left aside, such as dynamic demands.

In standard ergonomic assessments, risk factors of different nature are often combined together to form a single and compact ergonomic score. Though the combination of several MSDs factors does increase the risk, the way these various factors interact is, however, not well-established in general [Li 1999]. The different kinds of demands are therefore represented by separate indicators here, so that the formulation of the indicators is not task-dependent. The proposed ergonomic indicators are classified into two families – constraint oriented indicators and goal oriented indicators – detailed hereafter.

3.2.1.1 Constraint oriented indicators

Constraint oriented indicators are local joint measurements – position, velocity, acceleration, torque and power – which directly represent the relative level of joint demands¹. For each one of these five quantities, a global indicator I_s is obtained for the legs, the right arm, the left arm, and the back (plus head), by summing the squared contributions of every joint in the considered limb (similarly to [Pholsiri 2004]). Grouping several joints in one indicator decreases the number of indicators – and thereby the complexity of the ergonomic analysis – while accounting for the situation of the whole body (the evaluation cannot be limited to the joints initially affected by MSDs since an ill-adapted robot may relocate the MSDs risk to other joints). The different limbs of the body can, however, perform very different tasks simultaneously, hence separate indicators for each limb.

The position and torque of each joint are normalised by average physiological limit values before the summing [Chaffin 2006, Holzbaur 2005]; the capacities of the normalised joints are then all equivalent, rendering the summing more meaningful. For velocity, acceleration, and power, however, joint physiological limits are not well-documented in the literature and the normalisation is impossible for now.

3.2.1.2 Goal oriented indicators

Goal oriented indicators are indirect images of the biomechanical demands endured by a worker; they quantify the ability to comfortably perform certain actions (*e.g.* balance, force exertion). Goal oriented indicators are very compact: one indicator accounts for the whole-body situation.

Balance Evaluating the balance quality gives an insight into the effort needed to maintain the posture. Unstable balance indeed requires higher muscular effort since the posture must always be corrected to prevent falling. Balance is quantified through two indicators. *Balance stability margin* represents the capacity to withstand external disturbances; it is evaluated by the sum of the square distances between the centre of Pressure (CoP) and the base of support boundaries [Xiang 2010]. *Dynamic balance* evaluates the dynamic quality of the balance with the inverse (so that all the indicators should be minimised to improve the situation) of the time before the CoP reaches the base of support boundary, assuming its dynamic remains the same.

Force/Movement generation The ability to generate forces and movements is evaluated with manipulability measures [Yoshikawa 1985b], which are global images of the joint demands needed to perform a motion/force [Jacquier-Bret 2012]. This work focuses on skilled technical gestures in which the worker knows the trajectories/forces to follow/exert, therefore directional measures are used; the ability to produce end-effector Cartesian force (resp. velocity) in a given direction is evaluated with the inverse of the *hand(s) force* (resp. *velocity*) *transmission ratio* [Chiu 1987]. The transmission ratio is calculated with the dynamic manipulability [Yoshikawa 1985a] to account for the dynamic effects and the non-homogeneity of the human joint capacities.

Vision Estimating the ability to easily move one's head in various directions gives an insight into the amount of postural change required to follow a visual target (workers tend to look at what they are doing when performing manual activities). The *rotational dexterity of the head* [Yoshikawa 1985b] is therefore used as a vision-related indicator.

Energy The whole-body kinetic energy is a global measure of human energetic performance, since it is directly associated with the power consumed during a movement [Abdel-Malek 2005].

¹See [Maurice 2014a] for a detailed study and validation of some of these indicators.

Indicator definition	Equation
Joint normalised position (RA, LA, B, L)	$\frac{1}{N_j} \sum_{i=1}^{N_j} \left(\frac{q_{j_i} - q_{j_i}^{neutral}}{q_{j_i}^{max} - q_{j_i}^{neutral}} \right)^2$
Joint normalised torque (RA, LA, B, L)	$\frac{1}{N_j} \sum_{i=1}^{N_j} \left(\frac{\tau_i}{\tau_i^{max}(t)} \right)^2$
Joint velocity (RA, LA, B, L)	$\frac{1}{N_j} \sum_{i=1}^{N_j} \dot{q}_{j_i}^2$
Joint acceleration (RA, LA, B, L)	$\frac{1}{N_j} \sum_{i=1}^{N_j} \ddot{q}_{j_i}^2$
Joint power (RA, LA, B, L)	$\frac{1}{N_j} \sum_{i=1}^{N_j} \dot{q}_{j_i} \tau_i $
Kinetic energy	$\frac{1}{2} \boldsymbol{\nu}^T \mathbf{M}(\mathbf{q}) \boldsymbol{\nu}$
Velocity transmission ratio (right hand, left hand)	$\left[\mathbf{u}^T (\mathbf{J} \mathbf{M}^{-1} \mathbf{L}^2 \mathbf{M}^{-1} \mathbf{J}^T)^{-1} \mathbf{u} \right]^{\frac{1}{2}}$
Force transmission ratio (right hand, left hand)	$\left[\mathbf{u}^T (\mathbf{J} \mathbf{M}^{-1} \mathbf{L}^{-2} \mathbf{M}^{-1} \mathbf{J}^T) \mathbf{u} \right]^{\frac{1}{2}}$
Head dexterity	$\frac{\sigma_{min}}{\sigma_{max}}$
Balance stability margin	$\frac{1}{N_b} \sum_{i=1}^{N_b} d_i^2$
Dynamic balance	$\frac{\ \mathbf{v}_{CoP}\ }{d}$

Table 3.1: Ergonomic indicators for evaluating biomechanical demands in manual activities. *RA* stands for *right arm*, *LA* for *left arm*, *B* for *back* and *L* for *legs*. N_j is the number of joints in the considered body-part (RA, LA, B or L), q_{j_i} the angle of joint i , $q_{j_i}^{max}$ the joint limit, $q_{j_i}^{neutral}$ the neutral joint position, $\dot{q}_{j_i}$ the joint velocity, $\ddot{q}_{j_i}$ the joint acceleration, τ_i the joint torque and τ_i^{max} the joint torque capacity, which decreases with fatigue according to the evolution law proposed by L. Ma *et al.* [Ma 2009]. $\boldsymbol{\nu}$ is the generalised velocity, $\mathbf{M}$ the generalised inertia matrix, $\mathbf{J}$ the Jacobian matrix of the considered end-effector, $\mathbf{u}$ the task direction of interest, and $\mathbf{L} = \text{diag}(\tau_i^{max})$ contains the joint torque capacities. σ_{min} (resp. σ_{max}) is the smaller (resp. bigger) singular value of $\mathbf{J}_{rot}^h \mathbf{M}^{-1} \mathbf{L}$, with $\mathbf{J}_{rot}^h$ the rotational part of the head Jacobian matrix. N_b is the number of base of support boundaries, and d_i the distance between the CoP current position and the i th boundary of the base of support. $\mathbf{v}_{CoP}$ is the CoP current velocity, and d the distance between the CoP current position and the base of support boundary along the direction of $\mathbf{v}_{CoP}$.

3.2.1.3 Main features

The aforementioned ergonomic indicators are summarised in Table 3.1. They are relative indicators, *i.e.* they enable to quantitatively compare several situations and identify the most demanding one, but they do not assess an absolute level of risk of developing MSDs.

Besides, all these indicators are instantaneous quantities, *i.e.* they can be measured at each moment of the activity. If the time evolution of the indicators may be interesting, the purpose here is to summarise the whole ergonomic situation with only a limited number of values, to facilitate the comparison of the overall ergonomic performance of different collaborative robots. The instantaneous values of each indicator are therefore time-integrated, so that the whole activity is represented with one single scalar value per indicator.

3.2.2 Simulation of co-manipulation activities

In order to numerically evaluate the ergonomic indicators defined in section 3.2.1, the considered activity must be simulated with a dynamic autonomous DHM, possibly interacting with a collaborative robot. The simulation is run in a dynamic simulation framework based on a physics engine to ensure the physical consistency of the resulting motion and forces.

3.2.2.1 DHM control

The DHM motion is computed by solving an optimisation problem to find the actuation variables (joint torques, accelerations and contact forces) which enable to follow some objectives at best (*e.g.* hand trajectory, centre of mass acceleration), while respecting physical constraints. The general concept of this type of optimisation-based controller are introduced in chapter 1 and the particular structure of this controller for humanoids is recalled in section 2.5.1.

3.2.2.2 Animation modes

The DHM controller presented above is generic and can be used either with motion capture data (replay mode) or with high level tasks descriptions (autonomous mode) (*e.g.* target to reach, place to go)². In both cases, the DHM balance is managed with a high weight centre of mass acceleration task, which reference is computed using a Zero Moment Point preview control [Kajita 2003]. Low weight joint acceleration tasks (postural task) and joint torque tasks are used respectively to define a natural reference posture (standing, arms along the body), and to prevent useless effort.

In autonomous mode, only the body parts that are directly needed to perform the activity – generally one or both hands and the head – are explicitly controlled with an operational acceleration and/or force task (Fig. 3.4). The reference trajectory for the hand task (manipulation task) results from an interpolation between the start and end points specified by the user. The head is controlled with an orientation task, so that the DHM looks at what it is doing (gazing task). In replay mode, on the contrary, the recorded Cartesian positions of markers positioned on the body of a human subject are mapped onto the DHM. An operational acceleration task is created for each marker, and the reference trajectory is the recorded marker trajectory.

The exact values of the tasks weights are manually tuned through trial and error. Though time consuming in the first place, the tuning process does not need to be repeated; the weights obtained are general enough to be used for successfully simulating many different activities.

²See [Maurice 2015] for a detailed description of the tasks included in the controller in autonomous and replay modes.

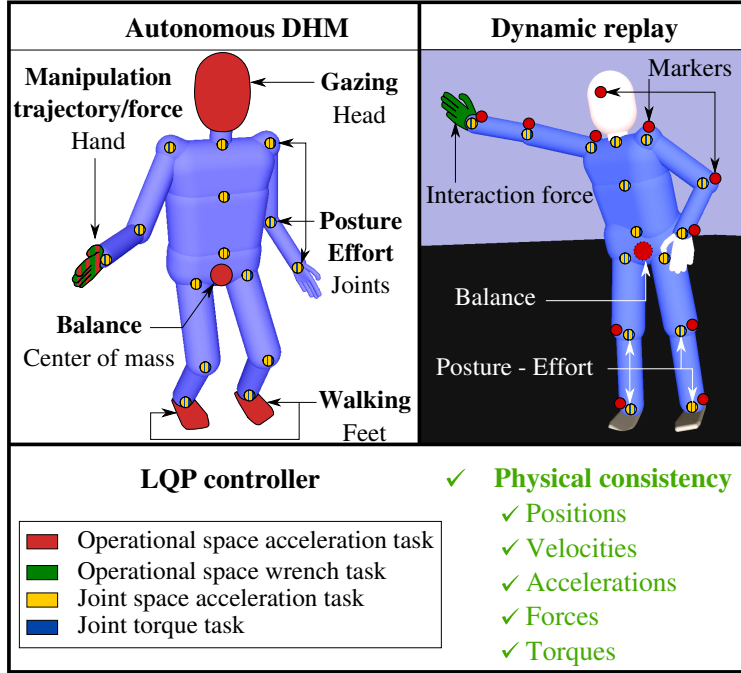


Figure 3.4: Tasks used in the LQP controller for simulating manual activities with an autonomous DHM (left) or for dynamically replaying human motion (right).

3.2.2.3 Robot simulation

This work focuses on collaborative robots which provide strength amplification and are manipulated by the end-effector only (parallel co-manipulation). The simulation method presented in this section is dedicated to such systems specifically.

DHM grasp The DHM fingers are not articulated because grasping requires a complex control of the fingers which is beyond the scope of this work. The human grasp is therefore represented by a 6D spring-damper system between the DHM palm and the robot end-effector.

Control law Strength amplification consists in controlling the robot so that the force it exerts on the manipulated tool (or environment) is an amplified image of the force applied by the worker onto the robot³. Additionally, the weight of the robot and the viscous friction effects are compensated. The inertial effects, on the contrary, are not compensated because such compensation is hard to implement on real robots due to the difficulty/cost to properly measure joint accelerations. The global strength amplification control law is:

$$\boldsymbol{\tau}_r = \alpha \mathbf{J}_{ee,r}^T \mathbf{f}_{vh} + \mathbf{g}_r(\mathbf{q}_r) + \mathbf{B} \dot{\mathbf{q}}_r \quad (3.1)$$

where $\boldsymbol{\tau}_r$ is the vector of robot joint torques, $\mathbf{q}_r$ the robot joint angles, $\dot{\mathbf{q}}_r$ the joint velocities, $\mathbf{g}_r$ the vector of gravity forces, $\mathbf{B}$ the matrix of viscous friction coefficients, $\mathbf{J}_{ee,r}$ the Jacobian matrix of the robot end-effector, $\mathbf{f}_{vh}$ the force applied by the DHM on the robot end-effector, and α the amplification coefficient⁴.

³ In the simulation the interaction force is estimated with the spring-damper system representing the human grasp; on real robots, a force sensor is embedded on the user handle.

⁴ In all generality, α may not be the same for all operational directions. As such, it could be chosen as a diagonal matrix rather than as a scalar.

3.2.3 Sensitivity analysis of the ergonomic performance

The simulation framework described in section 3.2.2 enables to measure multiple ergonomic indicators defined in section 3.2.1. These measurements are, however, not directly useful for the design of collaborative robots. Comparing the overall ergonomic performance of different collaborative robots based on all the ergonomic indicators is indeed not straightforward, because each indicator has a different biomechanical meaning and different indicators may lead to different conclusions. Moreover, the values of the ergonomic indicators *per se* do not provide any information on how to improve the robot design, *i.e.* which parameters should mainly be modified to enhance the overall ergonomic performance. To answer these questions, the most informative indicators and their dependence on the robot parameters must be identified. In most cases, however, no straightforward analytical relation between robot parameters and ergonomic indicators can be established. A statistical sensitivity analysis is therefore conducted [Saltelli 2000]. This section presents an extended version of the work presented in [Maurice 2014b].

3.2.3.1 Method overview

Statistical sensitivity analyses rely on the numerical evaluation of the output (ergonomic indicators here) for numerous values of the input parameters, thus requiring a large number of trials. Having a real subject execute the activity in each situation would be too time consuming, therefore the activity is rather simulated with an autonomous DHM.

The whole process for analysing the relevance of ergonomic indicators regarding the comparison of collaborative robots and the influence of the robot parameters can be summarised as follows (Fig. 3.5):

1. Define the robot parameters which can be altered.
2. Select – among all the possible combinations of parameters values – those that should be tested.
3. Simulate the activity with an autonomous DHM for each selected combination of parameters values, to measure the ergonomic indicators.
4. Compute sensitivity measures for the ergonomic indicators based on their values in all the tested cases.

Steps 1, 2 and 4 are detailed in the following sections. The simulation step 3 is performed with the autonomous DHM as described in section 3.2.2.

3.2.3.2 Parameters selection

The sensitivity analysis aims at estimating – for a given activity – how much each parameter of a collaborative robot affects the ergonomic situation. In early stages of a robot design process, however, the number of possible designs – and hence the list of possible parameters – is infinite and there is *a priori* no reason to choose one over another. In order to be generic, real robot designs are therefore not used. Instead, a robot is modeled by its positive and negative effects on the worker – each effects corresponding to one parameter.

Robot parametrisation In this work, robots are manipulated by the end-effector only (parallel co-manipulation), so the robot is simulated by a 6D mass-spring-damper system attached to the DHM hand (Fig. 3.6). The mass (M_r), stiffness (K_r) and damping (B_r) parameters represent the equivalent dynamics of the robot at the end-effector. The possible geometric interferences between the robot and the DHM are simulated without making hypotheses on the robot design, by limiting the DHM movements (limiting the joints range of motion) and modifying its posture (*e.g.* feet position, joint reference position). External forces are applied on the mass-spring-damper system to simulate the robot actuation (F_{robot}), α being the strength amplification coefficient (Eq. 3.1).

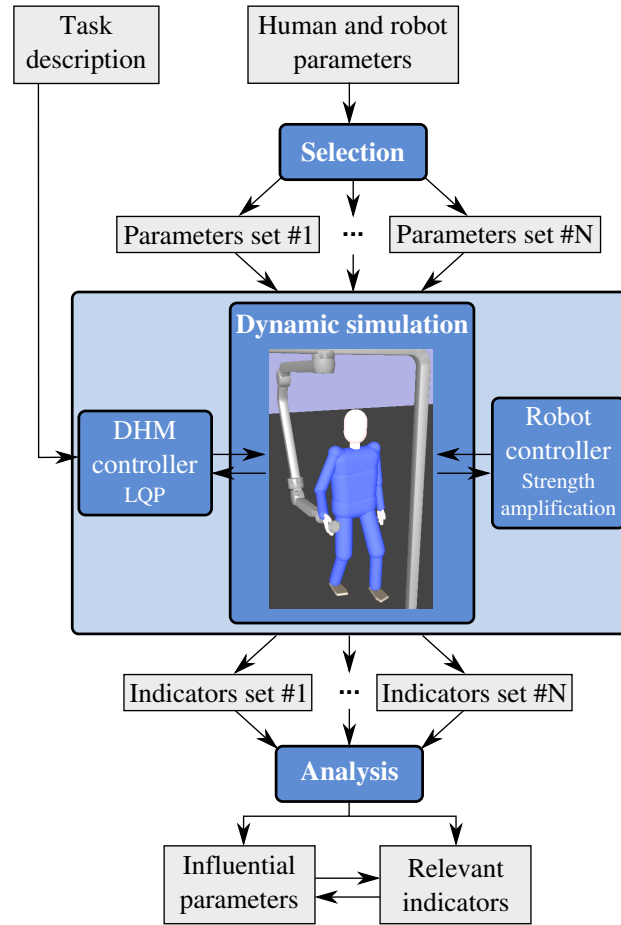


Figure 3.5: Flow chart of the method for identifying informative ergonomic indicators and influential parameters.

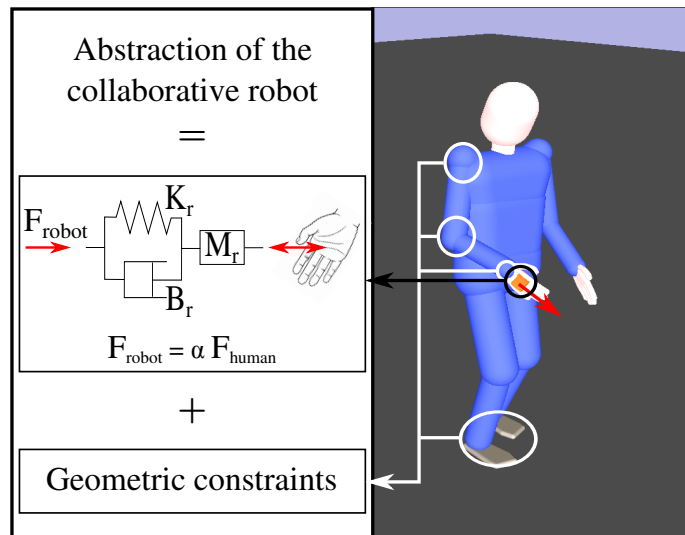


Figure 3.6: Abstraction of a collaborative robot by a mass-spring-damper system attached to the DHM hand and geometric constraints on the DHM motions (only some examples of constraints are displayed here).

Parameters space exploration The robot parameters taking continuous values, they must be discretised to form the different combinations of parameters values to test. But the computational cost of a simulation – though variable – is always expensive (greater or equal than real-time). The number of situations tested is therefore limited and the values of the parameters must be carefully selected.

Optimising the exploration of the parameters space requires a compromise between the number of trials and the precision of the resulting information [Saltelli 2000]. In this work, the analysis aims at quantitatively estimating the influence of each robot parameter on the ergonomic indicators, to identify which parameters should mainly be tuned. The computation of Sobol indices – which relies on the decomposition of the ergonomic indicators variance (functional ANOVA decomposition) – is then appropriate [Hoeffding 1948, Sobol 1993, Homma 1996]. Sobol indices allow a fine ranking of the influence of the different parameters, without requiring specific hypotheses on the ergonomic indicators. Furthermore, their interpretation is quite straightforward – each index measures the percentage of variance of an indicator that is explained by the corresponding parameter(s). Only the first order indices S_i (influence of the parameter X_i alone, with no interaction) and the total indices S_{T_i} (influence of X_i , including all interactions with other parameters) are considered in this work, because they give information on the i -th parameter independently from other parameters. A high S_i means that X_i alone strongly affects the indicator, whereas a small S_{T_i} means that X_i has very little influence on the indicator, even through interactions.

The extended FAST (Fourier amplitude sensitivity testing) spectral method is used for choosing the appropriate parameters values to test (within user-defined bounds) and for computing Sobol indices [Saltelli 1999]. The FAST exploration method is indeed a good compromise between the comprehensiveness of the space exploration and the number of trials.

3.2.3.3 Ergonomic indicators analysis

Once the simulations are performed for all the selected combinations of parameters values, Sobol indices can be computed. But Sobol indices only address single-output models, whereas here each ergonomic indicator is an output. Even though Sobol indices can be computed separately for each indicator, no global sensitivity index can be obtained for a parameter by aggregating indices relative to different indicators – the comparison of indices referring to different indicators being meaningless. Besides, Sobol indices do not help reducing the number of ergonomic indicators to facilitating the comparison of different collaborative robots. The most informative ergonomic indicators must therefore first be identified.

The purpose of this work is not to assess the absolute level of MSDs risks, but to compare collaborative robots. In this context, the relevance of an indicator is not related to its value, but to its variations when the activity is performed with different robots; if the value of an indicator remains unchanged whichever the robot that is used, this indicator is not useful to compare different robots. The most informative indicators are therefore the ones that best explain the disparity of the results when the activity is performed with various robots.

Ranking The problem of reducing the number of indicators to keep only the ones that best explain the disparity is addressed by K. Campbell *et al.* [Campbell 2006] and M. Lamboni *et al.* [Lamboni 2011] in the context of sensitivity analysis for multiple-output models. They propose to decompose the model outputs in a well-chosen basis before applying sensitivity analysis to the most informative components individually, which comes down to a dimensionality reduction problem. Standard dimensionality reduction methods, however, cannot be used here, because they form composite variables (*i.e.* combinations of the initial variables). The ergonomic indicators having different physical meanings, aggregations of various indicators would be meaningless. Moreover, composite variables cannot be used to estimate the global influence of the robot parameters, since the influence of a parameter is likely to be different from one ergonomic indicator

to another. The importance of each ergonomic indicator is therefore represented directly by its variance. The indicators are thus ranked, and the most informative ones (those with the highest variance) are easily identified.

Scaling Before computing their variance, the indicators must be scaled because they have non-homogeneous units, hence different orders of magnitude. Scaling each indicator with a physiological limit value would be ergonomically meaningful, but some indicators do not have well-defined limits (*e.g.* kinetic energy), and even the existing ones may be hard to find (*e.g.* joint acceleration). The order of magnitude (used for the scaling) of an indicator is therefore estimated by measuring the indicator in many different situations with DHM simulations, and taking the average value. Activities of many different kinds (*e.g.* walking, reaching, pushing, carrying) are performed in many different ways, so the range of values of each indicator is assumed to be covered quite exhaustively⁵.

Selection Once the ergonomic indicators are ranked according to their variance, a Scree test [Jolliffe 2002] is performed to decide the number of indicators that are kept; the objective is to limit the number of indicators, while sufficiently accounting for the global ergonomic performance of the activity. Sobol indices are then computed separately for each one of the selected indicators. The indices relative to different indicators still cannot be compared, but the overall number of indices is reduced, making the interpretation of the results easier for the user.

3.2.4 Evolutionary design of a collaborative robot

The evaluation framework presented in the previous sections enables to rank robot candidates with respect to their ergonomic performance, and to identify which design parameters are crucial for improving this performance. Designing and modifying test candidates is, however, left to the robot designer, who has to rely on his/her experience (potentially limited since collaborative robotics is a rather new approach) and preliminary studies. This process is both time and resource intensive. To circumvent these problems, optimisation techniques are used to guide robots design.

Robots are optimised by coupling an evolutionary algorithm (EA) software [Goldberg 1989] with the collaborative robot evaluation framework presented previously. The EA is used for exploring the space of robot designs – *i.e.* providing robot candidates to evaluate – while the simulation tool is used to numerically evaluate the various objectives for each robot candidate (Fig. 3.7) (here the full robot structure – and not its abstraction – is included in the simulation and interacts with the DHM). EAs are well-suited to address the problem of optimal robot design because they enable optimisation over vast and non-continuous search spaces and can handle multi-objectives problems [Doncieux 2011]. Optimal collaborative robot design is indeed a multi-objective problem: the robot must be optimised regarding both the task and the worker, and potentially other aspects such as the cost or complexity of the structure. Moreover, these general objectives are often divided into several specific objectives; the worker-oriented objective, for instance, is evaluated through multiple ergonomic indicators.

The EA used here is the Non-dominated Sorting Genetic Algorithm-II (NSGA-II) [Deb 2002] – based on the Pareto optimality principle – because it efficiently addresses convergence and diversity of the solutions, the two main features that allow to approach the Pareto-optimal front at best. Though NSGA-II (and multi-objective EAs in general) is designed to solve multi-objective problems, the number of objectives affects the convergence of the optimisation; the number of conflicting⁶ objectives should generally be limited to three

⁵The activities used for estimating the indicators order of magnitude are detailed in [Maurice 2014b]. A video is available here: http://pages.isir.upmc.fr/~padois/website/fichiers/videos/maurice_humanoids_2014.mp4

⁶Two objectives are conflicting when it is impossible to satisfy both of them simultaneously.

[Deb 2001]. The ergonomic indicators analysis presented in section 3.2.3 is therefore used beforehand to select a small number of relevant worker-oriented objectives.

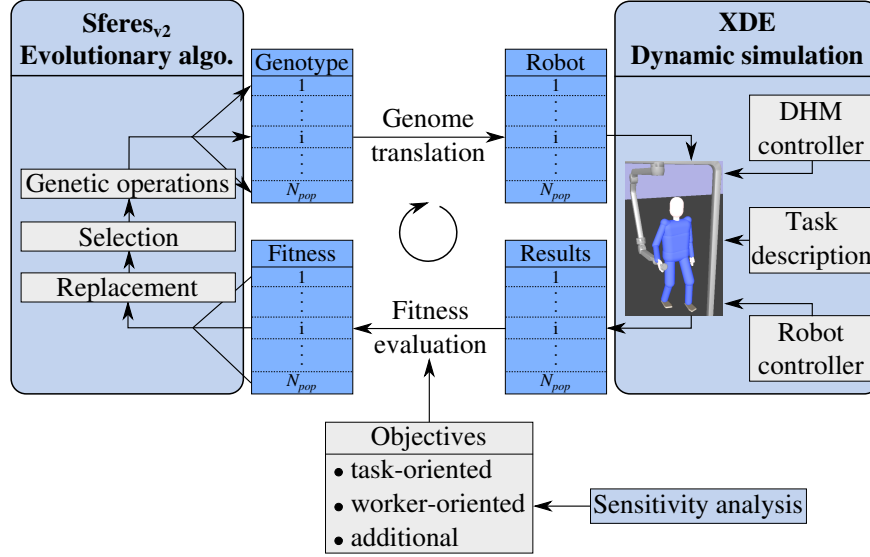


Figure 3.7: Framework for the optimisation of collaborative robot design: XDE-Sferes_{v2} coupling (a single generation loop is represented). The genetic operations are performed by the Sferes_{v2} software [Mouret 2010], which provides a population of robot candidates to evaluate. The simulations are run in the physics-engine based simulation framework XDE developed by CEA-LIST [Merlhiot 2012]. For each candidate, the objectives are measured through a DHM simulation. To limit the number of objectives, only the most informative ergonomic indicators are included in the worker-oriented objectives.

3.3 Results

The whole method for guiding the design of collaborative robots presented in section 3.2 is applied to a real activity. The motion of a human subject performing the considered activity is recorded and replayed, to evaluate the initial situation as a baseline. Autonomous DHM simulations are then run to perform the sensitivity analysis. The indicators which best summarise the overall ergonomic performance of the considered activity are thereby selected, and the robot parameters which should be tuned to enhance this performance are identified. Optimal values of these parameters – with respect to the relevant indicators – are computed using the EA framework. The activity performed with the optimised robot is then compared to the initial (non-assisted) situation.

3.3.1 Acquisition of the initial situation

3.3.1.1 Task description

An industrial manual task requiring significant effort is used as a test case. The activity consists in drilling six holes consecutively in a vertical slab of autoclaved aerated concrete (dimensions: 30 × 60 cm) with a portable electric drill. The locations of the holes are imposed and depicted on Fig. 3.8. The drill weighs 2.1 kg. The average normal force needed to drill a hole is about 40 N. The task duration is not constrained, but it takes about 1 min to perform the whole activity (take the drill, drill the six holes, put the drill down). In the experiment, the drill is held with the right hand only. The subject chooses his feet position but is not allowed to move them during the trial.

3.3.1.2 Motion capture set-up

The subject's motion is recorded with a CodaMotion⁷ system at 100 Hz. The subject is equipped with 25 markers spread all over his body and is standing on a force measurement plate. A 6 axes ATI force sensor⁸ is embedded in the drill handle to measure the drilling forces (Fig. 3.8). The recorded data are filtered with a zero-phase 10 Hz low pass 4th order Butterworth filter.

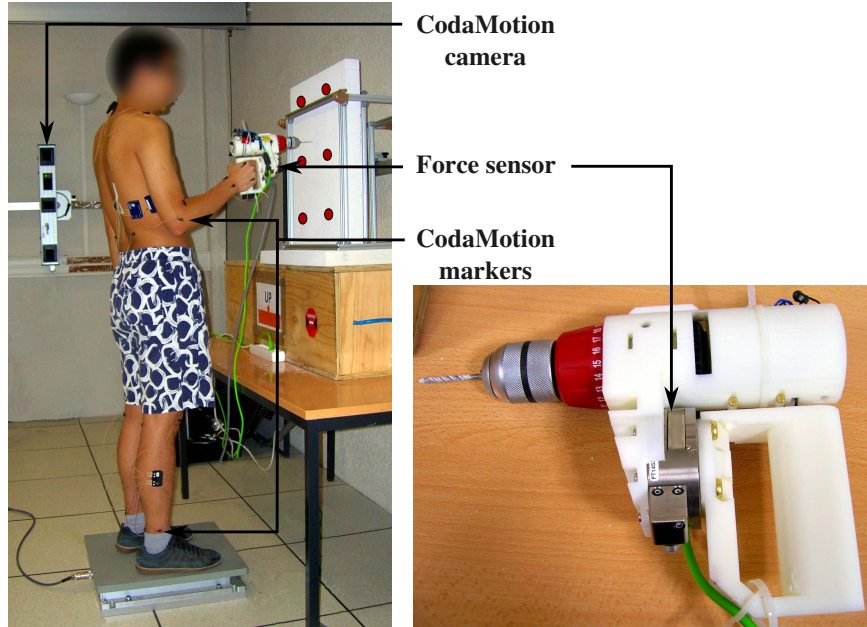


Figure 3.8: Force and motion capture instrumentation for the drilling activity. The subject is standing on a force measurement plate. A commercial drill has been modified to embed a force sensor. The red circles on the slab represent the drilling points.

3.3.1.3 Motion replay

The motion recorded on the human subject is replayed with a DHM⁹, using the dynamic replay method described in section 3.2.2. The simulation is run in the physics-engine based simulation framework XDE developed by CEA-LIST [Merlhiot 2012]. The XDE DHM consists of 21 rigid bodies linked together by 20 compound joints, for a grand total of 45 degrees of freedom (DoFs) plus 6 DoFs for the free-floating base. Each DoF is a revolute joint controlled by a single actuator. Given a subject's stature and mass, the DHM is automatically scaled according to average anthropometric coefficients¹⁰, and each body segment is further manually modified to match the subject morphology when needed.

The replayed motion enables the measurements of the ergonomic indicators in the initial situation. But the technical part of the gesture (*i.e.* the profiles of the tool trajectory and of the drilling force) is also needed to animate the autonomous DHM for the sensitivity analysis simulations¹¹.

⁷www.codamotion.com

⁸www.ati-ia.com/products/ft/ft_models.aspx?id=Gamma

⁹A video is available here: http://pages.isir.upmc.fr/~padois/website/fichiers/videos/maurice_drilling_dyn_replay.mp4

¹⁰segments lengths: <http://www.openlab.psu.edu/tools/calculators/proportionalityConstant>, segments masses: http://biomech.ftvs.cuni.cz/pbpk/kompendium/biomechanika/geometrie_hmotnost_vypocet_en

¹¹The acquisition of the technical gesture in itself only requires markers on the tool or the subject's hand and the drilling force.

3.3.2 Sensitivity analysis

3.3.2.1 Simulations

The drilling activity is simulated in the XDE framework, with the autonomous DHM; only the right hand trajectory and force are explicitly specified through a Cartesian acceleration/force task in the DHM controller (plus balance, postural and effort minimisation tasks). The hand trajectory and drilling force profile recorded on the human subject serve as reference trajectory/force for the hand tasks. The DHM feet do not move during a simulation (*i.e.* no automatic stepping), except if the dynamic balance cannot be maintained and the DHM falls. The drill weight is not included in the simulation because it is supported by the collaborative robot. The abstraction of the collaborative robot (section 3.2.3.2) provides strength amplification during drilling.

3.3.2.2 Input parameters

The input parameters of the sensitivity analysis represent the diversity of potential collaborative robots. In this experiment, only the mass of the robot abstraction (mass-spring-damper system) varies, while the stiffness and damping are kept constant to limit the number of parameters. The geometric interference between the robot and the worker is represented by constraints on the right arm and back (because the robot is manipulated with the right hand) joint limits and joint reference positions, and on the pelvis distance and orientation. The strength amplification coefficient is also included in the parameters. Parameters representing the diversity of workers are added to ensure that the human features do not have a strong impact on the ergonomic situation (otherwise, the robot should include some adjustable parts to adapt to specific workers' morphologies). The worker is defined by his/her stature and body mass index (BMI). The numerical upper and lower bounds of the input parameters are given in Table 3.2.

Parameter	Min.	Max.
DHM stature (m)	1.65	1.80
DHM BMI ($kg.m^{-2}$)	21.0	27.0
angle pelvis - normal to stab ($^{\circ}$)	-30	30
offset distance pelvis - centre of stab (m)	-0.3	0
upper body reference positions ($^{\circ}$)	0, 0, 0, 0	15, 45, 45, 135
upper body joint limits	0.3	1.0
robot mass (kg)	2	10
amplification coefficient	1	3

Table 3.2: Drilling activity parameters definition and limit values. The pelvis position is given in polar coordinates with respect to the centre of the stab. The offset for the pelvis-stab distance is added to the DHM arm length to define the real pelvis-stab distance. The upper-body joint limits are specified as ratio of the regular joint limits and applied to each joint of the back and right arm. The reference positions of the upper-body joints are only modified for the back flexion, shoulder flexion, shoulder abduction, elbow flexion; they are given in the same order and relative to the reference posture (upright, arms along the body).

The R software *sensitivity* toolbox¹² is used to select the parameters values – within the user-defined bounds – that need to be tested for the extended FAST analysis. The sample size and set of frequencies are chosen according to the recommendations of A. Saltelli *et al.* [Saltelli 1999]. They result in a grand total of 8008 simulations. One simulation takes approximately 2 min (real time: 75 s) on one core of a 2.4 GHz Intel R CoreTM i7 laptop, and the simulations can be parallelised.

¹²<http://www.r-project.org>

		Relevant ergonomic indicators				
		Legs position	Right Arm torque	Back torque	FTR drilling direction	Right Arm position
		31 %	19 %	14 %	10 %	7 %
Parameters	DHM stature	10^{-3} 0.03	0.13 0.15	0.19 0.29	0.42 0.52	0.07 0.12
	DHM BMI	10^{-3} 0.02	0.05 0.06	0.02 0.03	0.21 0.23	10^{-5} 0.02
	Pelvis orientation	10^{-4} 0.01	0.10 0.14	0.01 0.03	0.15 0.23	0.15 0.28
	Pelvis distance	10^{-3} 0.01	10^{-4} 0.02	0.01 0.02	0.02 0.03	0.03 0.12
	Upper body ref. position	0.60 0.73	0.20 0.22	0.56 0.69	0.08 0.10	0.23 0.42
	Upper body joint limits	0.26 0.37	0.01 0.03	0.06 0.10	10^{-3} 0.02	0.28 0.43
	Robot mass	10^{-4} 10^{-3}	10^{-6} 10^{-3}	10^{-5} 10^{-3}	10^{-6} 10^{-3}	10^{-5} 0.02
	Amplification coefficient	10^{-4} 10^{-3}	0.46 0.49	10^{-5} 10^{-3}	10^{-4} 10^{-3}	10^{-5} 0.02

Table 3.3: Sobol indices for all five ergonomic indicators identified as relevant for the drilling activity. For each parameter and indicator, the upper value is the first order index, the lower value is the total index. The ergonomic indicators are presented in decreasing order of importance (decreasing variance) from left to right: the percentages below their names correspond to the percentage of the total variance they explain. FTR stands for force transmission ratio. Numbers are coloured from blue (minimum) to red (maximum), to facilitate the reading.

3.3.2.3 Results

The 26 ergonomic indicators defined in section 3.2.1 are analyzed. The velocity and force transmission ratio are computed for the right hand, in the motion direction and in the drilling direction respectively.

Relevant ergonomic indicators Table 3.3 presents the five ergonomic indicators – out of 26 in the initial list – that are identified as relevant according to the sensitivity analysis. The five indicators together represent 81 % of the total variance information, therefore only little information is lost by not taking into the other indicators. The selection of the upper-body torque and position indicators is consistent with the physical demands of the drilling activity (exerting a significant force with the right hand while covering a quite extended area). The absence of any velocity or acceleration indicators is consistent with the fact that the drilling activity does not require fast motions. The presence of the legs joint position indicator as the most discriminating indicator is, however, less expected.

Indicator-Parameter dependence Some parameter-indicator relations represented by Sobol indices in Table 3.3 are strongly expected and confirm the consistency of the proposed analysis (*e.g.* influence of the strength amplification coefficient on the right arm torque indicator, influence of the upper-body geometric parameters on the right arm position indicator). Other relations, however, are less straightforward and could not easily be guessed without the sensitivity analysis (*e.g.* predominant influence of the upper-body geometric parameters on the legs position indicator, absence of influence of the strength amplification coefficient on the back torque indicator).

Conclusion regarding robot design The results of the sensitivity analysis highlights two global trends. Firstly, the robot mass does not significantly affect the overall ergonomic performance, since it has no influence on any of the selected indicators. When designing a collaborative robot for the drilling activity, the robot mass is therefore not a critical parameter (from an ergonomic point of view). Secondly, all the selected indicators are significantly affected by at least one of the parameters representing the geometric interference between the robot and the worker. In the drilling activity, the morphology of the robot is therefore critical to the ergonomic benefit provided by the robot.

3.3.3 Evolutionary design of a robot morphology

The efficiency of a collaborative robot being highly task-dependent, designing a robot specifically for a given activity is often preferable to using a generic robot. Designing a dedicated robot from scratch is, however, costly and therefore not accessible to small companies. An intermediate solution is to use a generic platform including modifiable elements.

3.3.3.1 Optimisation variables

A generic 7 DoFs architecture (similar to a Kuka-LWR) with variable lengths for the first five segments is used (Fig. 3.9); the robot is manipulated by a user handle mounted on the end-effector. The control law of the robot is not optimised (to limit the complexity of the problem and hence the convergence time), so all robot candidates use the same strength amplification control law with $\alpha = 2$ (Eq. 3.1). The optimisation therefore aims at finding optimal values for the segments lengths and for the position and orientation of the robot base¹³.

Though the physical features of the worker do affect the ergonomic performances (Table 3.3), only one average human morphology is used in the optimisation because the purpose is only to make a proof of concept.

¹³Unlike the sensitivity analysis where an abstraction of the robot is used, the drilling activity is simulated here with the full robots candidates.

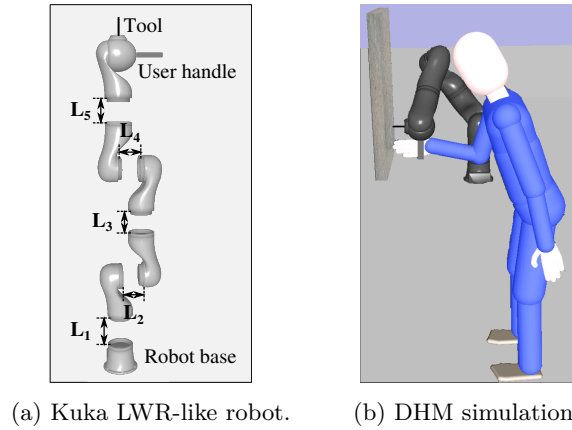


Figure 3.9: DHM simulation of the drilling task, assisted by a 7 DoFs Kuka LWR-like robot with adjustable segments lengths L_i , and providing strength amplification.

3.3.3.2 Objectives

Among the five relevant worker-oriented objectives identified through the sensitivity analysis (Table 3.3), the right hand FTR is removed to decrease the number of objectives. Indeed, the FTR is mainly affected by parameters which are constant in the present optimisation. The right arm and the back torque indicators are gathered into one single indicator, called upper-body torque indicator. Apart from the strength amplification coefficient which is constant here, the most influential parameters are the same (with similar parameter/indicator trends) for both indicators. A total of three worker-oriented objectives are therefore included in the optimisation.

The quality of the drilling task execution (task-oriented objective) is evaluated with the maximal position error of the drill extremity during the drilling phases (one objective). No additional objective is used.

Due to the high number of objectives (four), the probability of a robot belonging to the Pareto front is high, except if a very large population size is used. The population size is, however, constrained by the computation time. Having most of the population in the Pareto front – at least early in the optimisation – is not desirable because it turns the EA into a random search algorithm. The fitness (objective) values are therefore discretised to limit the number of robots in the Pareto front.

3.3.3.3 Evolutionary algorithm parameters

The population size and the number of generations in the EA result from a compromise between the computation time and the convergence of the solution. A population of 100 individuals and 200 generations are used. One generation is entirely evaluated in about 1 hour on a four-core, 2.4 GHz Intel R Core™ i7 laptop¹⁴.

3.3.3.4 Results

The evolution of the four objectives is studied to evaluate the capability of the optimisation to find suitable robot morphologies. The optimised robots are then compared with the non-assisted situation to estimate the ergonomic benefit brought by the robot.

¹⁴The overall optimisation time could however be significantly reduced if a computer with more cores were used because the optimisation framework is implemented so that several XDE simulations can be run simultaneously.

Evolution of the objectives The evolution of the four objectives is displayed in Fig. 3.10 for the whole population. The mean value of each objective decreases over generations, showing that the overall performance of the robots in the population do improve. Except for the upper-body torque indicator, the objectives minimal values stop evolving almost immediately. The convergence of the objectives maximal values, on the contrary, takes between 100 and 150 generations, especially for the three ergonomic indicators. It is therefore easy to find a robot which performs well on one objective, but finding a robot which matches all four objectives is much harder, hence the usefulness of multi-objective optimisation.

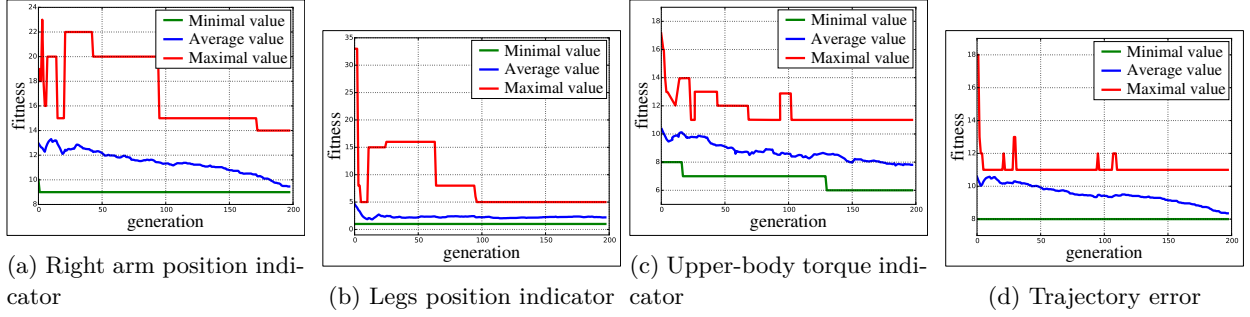


Figure 3.10: Evolution of the minimal, average and maximal values of the four objectives over generations (all the robots in the population are included). The discretised fitness values (no units) – and not the ergonomic indicators/trajectory error values – are plotted.

Comparison with the reference situation During the optimisation, the situation with the robot is never compared with the non-assisted situation. Though the robot performances are optimised, there is no certainty that the use of the robot is indeed beneficial. The five ergonomic indicators relevant for the drilling activity (Table 3.3) are therefore measured in the reference situation (no robot) and with the assistance of two near-optimal robots chosen within the Pareto front of the last generation to represent different solutions (Fig. 3.11).

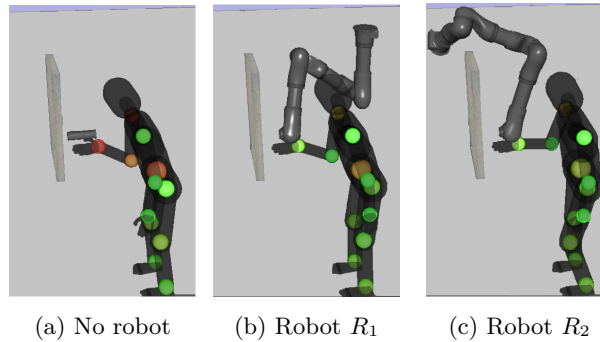


Figure 3.11: Snapshot of the DHM performing the drilling activity without assistance and with the assistance of two near-optimal collaborative robots. The coloured spheres represent the instantaneous level of joint effort.

To make the situations comparable – and as a first validation – all three situations are evaluated with the autonomous DHM (the exact same DHM controller is used). The results are displayed in Table 3.4.

Out of the five relevant indicators, two are significantly improved by both robots (force-related indicators, expected since the robots provide strength amplification), two remain mostly unchanged, and one is worsened. Despite the degradation in the right arm position indicator, the comparatively significant im-

	No robot	R_1	R_2
Right arm position	90	105	125
Legs position	15	25	18
Right arm torque	125	38	47
Back torque	75	43	38
FTR drilling	130	105	112

Table 3.4: Values of the five relevant ergonomic indicators without assistance (*No robot*) and with the assistance of two near-optimal robots (R_1 and R_2). For each indicator, the value displayed is the percentage of the indicator reference value (used for the scaling), so that the comparison is more meaningful (the reference value gives an insight into the average order of magnitude of the indicator, however it does not provide any indication on the absolute level of risk). The indicators in red are worsened by the robot, those in green are improved.

provements in the torque and power indicators demonstrate the benefit of the robots. The two near-optimal robots nevertheless show antagonistic performances (*e.g.* R_1 is better for the right arm torque but worse for the right arm position), so it is hard to say which one is overall the best (even more when all robots in the Pareto front are considered). The choice between different near-optimal robots is then left to the designer or ergonomist, according to his/her main concerns. The optimisation is nevertheless useful, since it performs a pre-selection of the best performing robots. Moreover, the purpose of the optimisation is not to replace the designer, but to provide him/her with interesting preliminary designs to be worked on, for further improving the robot performances.

3.4 Discussion and potential improvements

The physically consistent results and the improvement of the robots performances obtained through the optimisation demonstrate the usefulness of the proposed method. Its application within the design process of collaborative robots for industrial tasks should, however, be considered carefully because of some current limitations which are discussed thereafter.

3.4.1 Limitations of the ergonomic indicators

Though the ergonomic indicators defined in section 3.2.1 cover a wide range of MSDs risk factors, the repetitiveness factor is omitted. Yet, repetitiveness belongs to the main MSDs risk factors [Luttmann 2003]. The comparison of different collaborative robots is therefore conducted on a single work cycle, and the robot which most decreases the physical demands on one work cycle is assumed to be the best overall. But this hypothesis is only valid if the robots do not significantly affect the work rate; this restricts the range of possible applications of the proposed assessment method.

The other time-related risk factor – the duration factor – is taken into account through the time integral value of each ergonomic indicator. This solution comes down to measuring the time spent in different danger zones, each zone being weighed by a danger coefficient equal to the value of the instantaneous demand. But the relation between the time spent in a zone and the risk is very likely not linear. For instance, the same final value can result either from a medium demand all along the task, or from an alternation of strong and light demands. Yet both situations do not have the same biomechanical consequences.

Taking into account the time-frequency aspect of the gesture in the ergonomic evaluation would enable a more accurate assessment, as well as the possibility to extend the possible applications. However, it requires to understand how these time factors affect the human physical capacities, which is closely related to the

open problem of fatigue modelling. Fatigue and its effect on movement variability are the central question of the PhD thesis work of J. Savin. Some preliminary results on the integration of muscular fatigue models have been published in [Savin 2017].

3.4.2 Limitations of the optimisation

The proposed optimisation does improve the performances of the robots, compared to random robots in the initial population. However, the optimisation process is useful only if it outperforms the results a robot designer could achieve (*i.e.* does the optimisation provide robots with better or similar performances, in less or comparable time). The answer to this question is not straightforward – neither in general nor for the drilling activity in particular – but several elements must be underlined. Firstly, given the small number of task-related and environment-related constraints and the high number of DoFs of the robot, the drilling activity is not strongly constrained; a good solution would be much less intuitive in a cluttered environment where the optimisation may be more useful [Rubrecht 2011b]. The success of the optimisation in a cluttered environment, however, strongly depends on the autonomy of the DHM for solving complex problems (*e.g.* anticipating and avoiding collisions while still reaching the task target); this autonomy is currently limited (see section 3.4.3). Secondly, the performances of the optimisation could be improved by tuning the parameters of the evolutionary algorithm. The values used in this work are based on general recommendations, but no comparative studies have been carried out. The tuning of the parameters should nevertheless not be task-dependent, since the purpose is to provide a generic tool and not to address one specific task.

3.4.3 Limitations of the DHM

Musculoskeletal model Unlike the muscular actuation of the human body, the actuation of the DHM (in this work) is at joint level only (joint torques), and each DoF is controlled by a single actuator. The biomechanical quantities measured with such a model are therefore less detailed than what could be achieved with a musculoskeletal model. The DHM joint torques, in particular, do not fully represent the overall physical effort exerted by a person. Due to the redundancy of the human actuation, different combinations of muscle forces can result in a same joint torque. Internal muscle forces (*i.e.* forces which do not generate any joint torque) can thus be generated by a person, but they do not have any equivalent in the DHM model and are therefore not taken into account in the evaluation. Such forces occur during the simultaneous contraction of antagonistic muscles (co-contraction phenomenon) and aim at increasing the joint impedance to withstand perturbations arising from limb dynamics or due to external loads [Gribble 2003]. Though especially important in motions requiring high accuracy, co-contraction occurs in all motions to stabilise the joints and protect joint structures. Not taking co-contraction forces into account therefore leads to an under-estimation of the real human effort. Nevertheless, when comparing several collaborative robots, one can assume that the smaller the effort required to perform the task (not including the co-contraction), the smaller the co-contraction. External efforts (forces to apply on the robot or environment) and gravity-induced efforts (efforts required to maintain a posture) indeed represent a perturbation to the position or force accuracy; if the perturbation is smaller, the stiffness required to resist it is also smaller. A robot which is the best regarding the joint torque indicator without considering the co-contraction is therefore likely to be also the best when including co-contraction.

DHM control Since the sensitivity analysis and the optimisation are both based on DHM simulations, the biomechanical reliability of the results strongly depends on the realism of the autonomous DHM motion. The question of feet – as well as other contacts – placement is essential, since the activities addressed by collaborative robotics often require significant efforts and thus engage the whole-body. Besides, workers may adapt their feet position during the task, if the robot hinders their gestures. Conversely, the DHM currently lacks autonomy regarding contact placement: the feet positions are entirely set by the user and

are therefore not necessarily well-adapted to the task (the DHM can walk or step, but the stepping time and place must be specified beforehand). Solutions for automatic online feet adaptation [Ibanez 2014b] and for optimal contact placement when significant external forces are at play [Liu 2012a] do exist, but they only partly address the problem. The anticipated (*i.e.* not purely reactive) optimal placement of contacts indeed requires complex planning methods [Bouyarmane 2011a], which for now are too computationally expensive to be used in the current context. More generally, simulating highly realistic human motions requires to understand the psychophysical principles that voluntary movements obey. Many studies have been conducted to establish mathematical formulae of these principles, especially for reaching motions (Fitt's law, minimum jerk principle,...). G. De Magistris *et al.* [De Magistris 2013] have successfully implemented some of them within the XDE framework, and adding these features in the controller used in this work is a direction for future work. However, these improvements are currently limited to reaching motions because the driving principles are not yet known for all kinds of motions.

Nevertheless, if the results of the sensitivity analysis and optimisation presented in this paper are affected by the DHM limitations, the method in itself is independent from the DHM control. Thus in the near future an improved control law could be used to animate the DHM, while the analysis and optimisation methods remain the same.

Conclusions and Perspectives

Overview

This dissertation provides a rather complete overview of my research activities. The different contributions cover several topics:

- a generic formulation of reactive control problem (*cf.* Chapter 1) aiming at enforcing the strict respect of constraints over an infinite time horizon while targeting locally optimal control performances;
- a survey of the domain of balance control for balances humanoid behaviours under the prism of optimisation (*cf.* Chapter 2) and with a focus on some my own contributions in that domain ranging from reactive whole-body control and model predictive control to tasks optimisation;
- a methodological approach to performance analysis and optimisation in Robotics (*cf.* Chapter 3), notably applied to ergonomic evaluation for the design of collaborative robots.

Even though it is probably more a kind of wishful thinking rather than a reality, the underlying ingredient of these contributions is genericness. In other words, I have made by best so that the proposed control and performance analysis methodologies we have developed together with my PhD students and colleagues apply with minimum restrictions to all robots and applications.

It is also important to underline the fact that control is central to all these contributions. It may not appear so clearly in the last chapter (*cf.* Chapter 3). Nevertheless, evaluating at best robot morphologies in simulated physical environments requires the most robust and advanced control laws for the robotic systems to be designed but also the capability to simulate at best the behaviours of a virtual human.

Finally, even though this requires a tremendous engineering effort (notably on the work on humanoids), experimental validation is of uttermost importance when pretending to control robots evolving in open-ended environments. I am trying – the way to go is still long – to make it a central concern in all my research and teaching activities.

Challenges

Frédéric Rotella, one of my first Automatic Control professors, once told me: “Come back when you will have found the right question to ask”. He was right: the process that leads us to finding the right research questions is probably more important than the answers themselves. In that respect, my work has been prolific as it raised much more questions than it provided definitive answers. These questions are both short terms and long terms ones and are hopefully taking interesting directions both in terms of general understanding of Robotics and of applicability to real-world use-cases. Some of these questions/challenges are briefly exposed here.

Safe control of collaborative robots

The energy-based control approach for safe human-robot interaction developed together with A. Meguenani provides a very interesting framework as it:

- shifts the notion of safety from a pure geometric problem to a physical one;
- tends to break the classical barrier which consists in making a distinction between robot motion in free-space and robot motion in contact with the environment.

However, some aspects of the problem have been neglected. The first one is actually central: while monitoring the robot energy and constraining it under some limit values is interesting, it is very unlikely that we can guarantee any safety if this limit does not account for the energy introduced by the human operator in the interaction. This energy should be considered and estimated in some way. Before contact, the corresponding kinetic energy could be computed based on an estimation of the person mass as well as an estimation of its velocity. When contact is established, the energy introduced by the operator can be accessed through measurement of the interaction wrenches as well as of the tracking errors induced by these external wrenches. The second limiting factor is related to the absence of a proper theoretical analysis of safety. We tend to believe that our approach can lead to safe behaviours but we have not yet used any theoretical approach to prove it. Looking at this problem in terms of passivity could be one way to go and constitutes a short term perspective.

Realistic virtual human motion

Based on the work of J. Salini and with the contribution of A. Ibanez, P. Maurice has, through her work on virtual ergonomics for the evaluation and design of collaborative robots, demonstrated that virtual humans could be controlled realistically enough and constitute a great tool for fast prototyping of robot morphologies and behaviours. Nonetheless, some developments are still needed to get more realistic behaviours but also provide better estimation of some very important factors such as the energy expenditure and torque demands at the articular level. While introducing musculo-skeletal models seems out of reach as they would induce very large computation times but also would potentially lack realism in some way¹⁵, some improvement are possible, notably with respect to the effects of fatigue. To that end, the on-going work of J. Savin aims at introducing an experimentally validated fatigue model inside the virtual human controller used by P. Maurice and thus obtain more realistic motion of virtual human. However, at this stage the effects of fatigue are just accounted for through a dynamic adaptation of the joint torque limits. This is of course needed but many more potential impacts of fatigue should be considered among which impacts on: the reference posture, tasks control gains, tasks weights, allowed target and trajectory variances, joint limits. A thorough study of the corresponding literature is needed in order to determine the need for considering these impacts of fatigue as well as ways to account for them in the used whole-body controller.

The right models for the right control time scales

The work on constraints compatibility is perfectible and a short term perspective is to extend with A. Meguenani the results of the work of S. Rubrecht at the dynamic level (as mentioned in Section 1.3.3). Unfortunately, as explained in Section 1.3, this will not be enough and, related to the notion of viability, constraints compatibility cannot be guaranteed over an infinite time horizon in open-ended environments and applications. Model predictive control is presented several times in this dissertation, both in Chapter 1 and Chapter 2, as a promising framework for what could be called “realistic” optimal control. However, it raises new challenges. Indeed, the computational complexity of model predictive control can be important when considering non-linear systems and there exist a problem¹⁶ which can be posed as: what is the right level of complexity to be considered for a system when considering its control at different time scales? Ideally, one would like to be able to reactively re-plan a global, rough motion of the system on one end and

¹⁵To the best of our knowledge the underlying principle of whole-body motions are still far from being fully understood.

¹⁶This problem is brilliantly evoked in a recent video of N. Mansard: <https://www.youtube.com/watch?v=5U1Dv-PTXp4>.

reactively compute a local, fine solution to the control problem on the other end. For a fixed computation capacity, one thus has to find the right balance between the complexity of the retained model for the robot and the size of the considered control horizon. This is easier said than done. Indeed too much simplifications at the global level may lead to infeasible solutions at the local level or to highly suboptimal behaviours as explained in Chapter 2.

The work of R. Lober on efficient tasks adaptation in order to maximise their feasibility at the reactive whole-body control level is one step in the direction of adapting global, potentially infeasible, plans for efficient execution at the reactive control level. However, up-to now, there is no efficient tasks compatibility model that can be used for on-line task adaptation and more importantly, no capitalisation is made of the encountered situations. In other words, we keep performing costly computation again and again without accounting for previously found solutions. Introducing incremental learning algorithms could definitely be a solution to learn simplified models of the robot which could efficiently be reused and adapted over time. This differs from the approaches which consider the combination of motion primitives (see for example [Clever 2016]) as the focus here is more one learning models of the systems dynamics rather than learning a catalogue of observed motion, but the pursued goals are in the end quite similar. Overall, this is clearly a longer term and complex perspective but my long-lasting collaboration with O. Sigaud in this domain (see the pioneering PhD work of C. Salaün on model learning for robotics [Salaün 2009, Salaün 2010, Sigaud 2011, Sicard 2011, Droniou 2012]) is an ingredient of interest to achieve novel and impacting contributions. My collaboration with S. Doncieux within the framework of the PhD thesis of J. Pontes is also a promising way towards the use of learning and online adaptation of controllers for the locomotion of legged robots towards different objectives and different terrains [Pontes 2016].

Towards the industry

Traditional industrial companies have recently started to formulate a need for robots out of their cages, with specific use-cases in mind. In my case, this growing interest has been the support of many PhD thesis:

- the PhD thesis of S. Rubrecht with Bouygues Travaux Publics;
- the PhD thesis of P. Maurice with CEA-LIST;
- the PhD thesis of A. Meguenani within the framework of the RTE¹⁷/UPMC chair position on “Intervention Robotics” that I have been holding during the last five years;
- the PhD thesis work of J. Savin together with INRS;
- the PhD thesis work of L. Joseph together with GE Healthcare on the “Design and Control of a robotic manipulator working safely around patients”.

It also creates collaboration opportunities in the domain of collaborative robotics applied to aeronautics for example.

Overall, transferring research to the industry is not the main motivation *per se (sic)*. However, industrial contexts and all their constraints are a perfect way of testing the fact that the control approach that I am attempting to develop are actually well grounded, generic, constraints compliant and robust.

¹⁷RTE is the French system operator for transmission of high-voltage electrical power.

Bibliography

- [Abdel-Malek 2005] K. Abdel-Malek, J. Yang, W. Yu and J. Duncan. *Human Performance Measures: Mathematics*. Technical report, Department of Mechanical Engineering, The University of Iowa, 2005. (Cited on page 45.)
- [Abe 2007] Yeuhi Abe, Marco da Silva and Jovan Popović. *Multiobjective control with frictional contacts*. In Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation, pages 249–258. Eurographics Association, 2007. (Cited on page 35.)
- [Baerlocher 2004] P. Baerlocher and R. Boulic. *An Inverse Kinematic Architecture Enforcing an Arbitrary Number of Strict Priority Levels*. The Visual Computer: International Journal of Computer Graphics, vol. 20, no. 6, pages 402–417, 2004. (Cited on page 12.)
- [Baillieul 1985] J. Baillieul. *Kinematic programming alternatives for redundant manipulators*. In Proceedings of the IEEE International Conference on Robotics and Automation, volume 2, pages 722–728, March 1985. (Cited on page 11.)
- [Barthélemy 2008] Sébastien Barthélemy and Philippe Bidaud. *Stability measure of postural dynamic equilibrium based on residual radius*. In Advances in Robot Kinematics: Analysis and Design, pages 399–407. Springer, 2008. (Cited on page 29.)
- [Ben Israel 2003] A. Ben Israel and T.N.E. Greville. Generalized inverses: Theory and applications, second edition. Springer, 2003. ISBN 0-387-00293-6. (Cited on pages 8 and 79.)
- [Bock 1984] H. G. Bock and K.-J. Plitt. *A Multiple Shooting Algorithm for Direct Solution of Optimal Control Problems*. In Proceedings of the 9th IFAC World Congress, pages 242–247, 1984. (Cited on page 31.)
- [Bouyarmane 2011a] K. Bouyarmane and A. Kheddar. *Multi-contact stances planning for multiple agents*. IEEE International Conference on Robotics and Automation, pages 5246–5253, 2011. (Cited on page 62.)
- [Bouyarmane 2011b] Karim Bouyarmane and Abderrahmane Kheddar. *Using a multi-objective controller to synthesize simulated humanoid robot motion with changing contact configurations*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 4414–4419. IEEE, 2011. (Cited on page 35.)
- [Bouyarmane 2012] Karim Bouyarmane and Abderrahmane Kheddar. *Humanoid robot locomotion and manipulation step planning*. Advanced Robotics, vol. 26, no. 10, pages 1099–1126, 2012. (Cited on pages 30 and 31.)
- [Bruyninckx 2003] Herman Bruyninckx, Peter Soetens and Bob Koninckx. *The Real-Time Motion Control Core of the Orocos Project*. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 2766–2771, 2003. (Cited on page 10.)

- [Campbell 2006] K. Campbell, M.D. McKay and B.J. Williams. *Sensitivity analysis when model outputs are functions*. Reliability Engineering & System Safety, vol. 91, no. 10, pages 1468–1472, 2006. (Cited on page 51.)
- [Chaffin 2006] D.B. Chaffin, G.B.J. Andersson and B.J. Martin. Occupational biomechanics. Wiley, 4th edition, 2006. (Cited on pages 44 and 45.)
- [Chaffin 2007] D.B. Chaffin. *Human motion simulation for vehicle and workplace design*. Human Factors and Ergonomics in Manufacturing & Service Industries, vol. 17, no. 5, pages 475–484, 2007. (Cited on pages 41 and 42.)
- [Chiaverini 1997] S. Chiaverini. *Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators*. IEEE Transactions on Robotics and Automation, vol. 13, no. 3, pages 398–410, Jun 1997. (Cited on page 11.)
- [Chiu 1987] S.L. Chiu. *Control of redundant manipulators for task compatibility*. In Proceedings of the IEEE International Conference on Robotics and Automation, volume 4, pages 1718–1724, 1987. (Cited on page 45.)
- [Clever 2016] D. Clever, M. Harant, K. H. Koch, K. Mombaur and D. M. Endres. *A novel approach for the generation of complex humanoid walking sequences based on a combination of optimal control and learning of movement primitives*. Robotics and Autonomous Systems, 2016. (Cited on page 65.)
- [Colgate 2003] JE Colgate, M Peshkin and SH Klostermeyer. *Intelligent assist devices in industrial applications: a review*. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 2516–2521, 2003. (Cited on page 41.)
- [Collette 2007] Cyrille Collette, Alain Micaelli, Claude Andriot and Pierre Lemerle. *Dynamic balance control of humanoids for multiple grasps and non coplanar frictional contacts*. In Proceedings of the IEEE-RAS International Conference on Humanoid Robots, pages 81–88. IEEE, 2007. (Cited on page 35.)
- [Dalibard 2013] Sébastien Dalibard, Antonio El Khoury, Florent Lamiroux, Alireza Nakhaei, Michel Taïx and Jean-Paul Laumond. *Dynamic walking and whole-body motion planning for humanoid robots: an integrated approach*. The International Journal of Robotics Research, page 0278364913481250, 2013. (Cited on pages 25 and 30.)
- [Damsgaard 2006] M. Damsgaard, J. Rasmussen, S. T. Christensen, E. Surma and M. de Zee. *Analysis of musculoskeletal systems in the AnyBody Modeling System*. Simulation Modelling Practice and Theory, vol. 14, no. 8, pages 1100–1111, 2006. (Cited on pages 42 and 44.)
- [David 2005] G.C. David. *Ergonomic methods for assessing exposure to risk factors for work-related musculoskeletal disorders*. Occupational medicine, vol. 55, no. 3, pages 190–199, 2005. (Cited on page 41.)
- [De Luca 2016] A. De Luca. *Robots with kinematic redundancy*. [Lecture slides](#), March 2016. (Cited on page 11.)
- [De Magistris 2013] G. De Magistris, A. Micaelli, P. Evrard, C. Andriot, J. Savin, C. Gaudes and J. Marsot. *Dynamic control of DHM for ergonomic assessments*. International Journal of Industrial Ergonomics, vol. 43, no. 2, pages 170–180, 2013. (Cited on pages 42 and 62.)
- [Deb 2001] K. Deb. Multi-objective optimization using evolutionary algorithms. John Wiley & Sons, 2001. (Cited on page 53.)
- [Deb 2002] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan. *A fast and elitist multiobjective genetic algorithm: NSGA-II*. IEEE Transactions on Evolutionary Computation, vol. 6, no. 2, pages 182–197, 2002. (Cited on page 52.)

- [Del Prete 2015] Andrea Del Prete, Francesco Nori, Giorgio Metta and Lorenzo Natale. *Prioritized motion-force control of constrained fully-actuated robots: Task Space Inverse Dynamics*. Robotics and Autonomous Systems, vol. 63, pages 150–157, 2015. (Cited on page 20.)
- [Delp 2007] S.L. Delp, F.C. Anderson, A.S. Arnold, P. Loan, A. Habib, C.T. John, E. Guendelman and D.G. Thelen. *OpenSim: open-source software to create and analyze dynamic simulations of movement*. IEEE Transactions on Biomedical Engineering, vol. 54, no. 11, pages 1940–1950, 2007. (Cited on page 42.)
- [Diedam 2008] Holger Diedam, Dimitar Dimitrov, Pierre-Brice Wieber, Katja Mombaur and Moritz Diehl. *Online walking gait generation with adaptive foot positioning through linear model predictive control*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 1121–1126, 2008. (Cited on page 33.)
- [Doncieux 2011] S. Doncieux, J.B. Mouret, N. Bredeche and V. Padois. *Evolutionary robotics: Exploring new horizons*. New Horizons in Evolutionary Robotics, pages 3–25, 2011. (Cited on page 52.)
- [Droniou 2012] A. Droniou, S. Ivaldi, V. Padois and O. Sigaud. *Autonomous Online Learning of Velocity Kinematics on the iCub: a Comparative Study*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems - IROS, pages 3577–3582, Vilamoura, Portugal, October 2012. (Cited on page 65.)
- [El Khoury 2013] Antonio El Khoury, Florent Lamiraux and Michel Taix. *Optimal motion planning for humanoid robots*. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 3136–3141, 2013. (Cited on page 31.)
- [Escande 2014] Adrien Escande, Nicolas Mansard and Pierre-Brice Wieber. *Hierarchical quadratic programming: Fast online humanoid-robot motion generation*. The International Journal of Robotics Research, page 0278364914521306, 2014. (Cited on page 35.)
- [Faverjon 1987] B. Faverjon and P. Tournassoud. *A local based approach for path planning of manipulators with a high number of degrees of freedom*. In Proceedings of the IEEE International Conference on Robotics and Automation, volume 4, pages 1152–1159, 1987. (Cited on page 35.)
- [Ferreau 2014] H.J. Ferreau, C. Kirches, A. Potschka, H.G. Bock and M. Diehl. *qpOASES: A parametric active-set algorithm for quadratic programming*. Mathematical Programming Computation, vol. 6, no. 4, pages 327–363, 2014. (Cited on page 11.)
- [Flacco 2012] F. Flacco, A. De Luca and O. Khatib. *Motion control of redundant robots under joint constraints: Saturation in the Null Space*. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 285–292, May 2012. (Cited on page 11.)
- [Goldberg 1989] D.E. Goldberg. Genetic algorithms in search, optimization, and machine learning. Addison Wesley, 1989. (Cited on page 52.)
- [Gonçalves 2016] V. M. Gonçalves, P. Fraisse, A. Crosnier and B. V. Adorno. *Parsimonious Kinematic Control of Highly Redundant Robots*. IEEE Robotics and Automation Letters, vol. 1, no. 1, pages 65–72, Jan 2016. (Cited on page 8.)
- [Goswami 1999] Ambarish Goswami. *Postural stability of biped robots and the foot-rotation indicator (FRI) point*. The International Journal of Robotics Research, vol. 18, no. 6, pages 523–533, 1999. (Cited on page 28.)
- [Gribble 2003] P.L. Gribble, L.I. Mullin, N. Cothros and A. Mattar. *Role of cocontraction in arm movement accuracy*. Journal of Neurophysiology, vol. 89, no. 5, pages 2396–2405, 2003. (Cited on page 61.)

- [Hauser 2008] Kris Hauser, Timothy Bretl, Jean-Claude Latombe, Kensuke Harada and Brian Wilcox. *Motion planning for legged robots on varied terrain*. The International Journal of Robotics Research, vol. 27, no. 11-12, pages 1325–1349, 2008. (Cited on page 30.)
- [Henze 2014] Bernd Henze, Christian Ott, Maximo Roa *et al.* *Posture and balance control for humanoid robots in multi-contact scenarios based on Model Predictive Control*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 3253–3258, 2014. (Cited on page 33.)
- [Herdt 2010] Andrei Herdt, Nicolas Perrin and Pierre-Brice Wieber. *Walking without thinking about it*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 190–195, 2010. (Cited on page 33.)
- [Herzog 2014] Alexander Herzog, Ludovic Righetti, Felix Grimminger, Peter Pastor and Stefan Schaal. *Balancing experiments on a torque-controlled humanoid with hierarchical inverse dynamics*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 981–988, 2014. (Cited on page 35.)
- [Hicks 2015] J.L. Hicks, T.K. Uchida, A. Seth, A. Rajagopal and S.L. Delp. *Is My Model Good Enough? Best Practices for Verification and Validation of Musculoskeletal Models and Simulations of Movement*. Journal of biomechanical engineering, vol. 137, no. 2, 2015. (Cited on page 44.)
- [Hignett 2000] S. Hignett and L. McAtamney. *Rapid entire body assessment (REBA)*. Applied Ergonomics, vol. 31, no. 2, pages 201–206, 2000. (Cited on page 41.)
- [Hoeffding 1948] W. Hoeffding. *A class of statistics with asymptotically normal distribution*. The annals of mathematical statistics, pages 293–325, 1948. (Cited on page 51.)
- [Holzbaur 2005] K.R.S. Holzbaur, W.M. Murray and S.L. Delp. *A model of the upper extremity for simulating musculoskeletal surgery and analyzing neuromuscular control*. Annals of biomedical engineering, vol. 33, no. 6, pages 829–840, 2005. (Cited on page 45.)
- [Homma 1996] T. Homma and A. Saltelli. *Importance measures in global sensitivity analysis of nonlinear models*. Reliability Engineering & System Safety, vol. 52, no. 1, pages 1–17, 1996. (Cited on page 51.)
- [Ibanez 2012] Aurelien Ibanez, Philippe Bidaud and Vincent Padois. *Unified preview control for humanoid postural stability and upper-limb interaction adaptation*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 1801–1808, 2012. (Cited on page 32.)
- [Ibanez 2014a] Aurelien Ibanez, Philippe Bidaud and Vincent Padois. *A distributed model predictive control approach for robust postural stability of a humanoid robot*. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), pages 202–209, 2014. (Cited on page 32.)
- [Ibanez 2014b] Aurelien Ibanez, Philippe Bidaud and Vincent Padois. *Emergence of humanoid walking behaviors from mixed-integer model predictive control*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 4014–4021, 2014. (Cited on pages 33 and 62.)
- [Ibanez 2015] Aurélien Ibanez. *Emergence of complex behaviors from coordinated predictive control in humanoid robotics*. PhD thesis, Université Pierre et Marie Curie Paris VI, September 2015. (Cited on page 19.)
- [Ivaldi 2014a] S. Ivaldi, J. Peters, V. Padois and F. Nori. *Tools for simulating humanoid robot dynamics: a survey based on user feedback*. In Proceedings of the IEEE-RAS International Conference on Humanoid Robots, pages 842–849, Madrid, Spain, November 2014. (Cited on page 36.)

- [Ivaldi 2014b] Serena Ivaldi, Vincent Padois and Francesco Nori. *Tools for dynamics simulation of robots: a survey based on user feedback*. <http://arxiv.org/abs/1402.7050>, 2014. (Cited on page 36.)
- [Jacquier-Bret 2012] J. Jacquier-Bret, P. Gorce and N. Rezzoug. *The manipulability: a new index for quantifying movement capacities of upper extremity*. *Ergonomics*, vol. 55, no. 1, pages 69–77, 2012. (Cited on page 45.)
- [Jolliffe 2002] I. Jolliffe. *Principal component analysis*. Wiley Online Library, 2002. (Cited on page 52.)
- [Kajita 2003] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi and H. Hirukawa. *Biped walking pattern generation by using preview control of zero-moment point*. *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, pages 1620–1626, 2003. (Cited on pages 28, 32 and 47.)
- [Kajita 2008] Shuuji Kajita and Bernard Espiau. *Legged robots*. In *Springer handbook of robotics*, pages 361–389. Springer, 2008. (Cited on page 21.)
- [Kanoun 2009] Oussama Kanoun, Florent Lamiraux, Pierre-Brice Wieber, Fumio Kanehiro, Eiichi Yoshida and Jean-Paul Laumond. *Prioritizing linear equality and inequality systems: application to local motion planning for redundant robots*. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2939–2944, 2009. (Cited on page 35.)
- [Kanzaki 2005] Shigeru Kanzaki, Kei Okada and Masayuki Inaba. *Bracing behavior in humanoid through preview control of impact disturbance*. In *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*, pages 301–305, 2005. (Cited on page 32.)
- [Karhu 1981] O. Karhu, R. Härkönen, P. Sorvali and P. Vepsäläinen. *Observing working postures in industry: Examples of OWAS application*. *Applied Ergonomics*, vol. 12, no. 1, pages 13–17, 1981. (Cited on page 41.)
- [Keith 2011] F. Keith, N. Wieber P.-B. and Mansard and A. Kheddar. *Analysis of the discontinuities in prioritized tasks-space control under discreet task scheduling operations*. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3887–3892, 2011. (Cited on page 35.)
- [Khatib 1986] Oussama Khatib. *Real-time obstacle avoidance for manipulators and mobile robots*. *The international journal of robotics research*, vol. 5, no. 1, pages 90–98, 1986. (Cited on page 11.)
- [Khatib 2008] Oussama Khatib, Luis Sentis and Jae-Heung Park. *A unified framework for whole-body humanoid robot control with multiple constraints and contacts*. In *European Robotics Symposium 2008*, pages 303–312, 2008. (Cited on page 20.)
- [Koenemann 2015] Jonas Koenemann, Andrea Del Prete, Yuval Tassa, E Todorov, Olivier Stasse, M Bennewitz and Nicolas Mansard. *Whole-body Model-Predictive Control applied to the HRP-2 Humanoid*. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Hamburg, Germany, September 2015. (Cited on page 37.)
- [Krause 2012] Manuel Krause, Johannes Engelsberger, Pierre-Brice Wieber and Christian Ott. *Stabilization of the capture point dynamics for bipedal walking based on model predictive control*. In *Robot Control*, volume 10, pages 165–171, 2012. (Cited on page 33.)
- [Kroger 2010] T. Kroger and F.M. Wahl. *Online Trajectory Generation: Basic Concepts for Instantaneous Reactions to Unforeseen Events*. *IEEE Transactions on Robotics*, vol. 26, no. 1, pages 94–111, Feb 2010. (Cited on page 7.)

- [Kuffner Jr. 2002] James J. Kuffner Jr., Satoshi Kagami, Koichi Nishiwaki, Masayuki Inaba and Hirochika Inoue. *Dynamically-stable motion planning for humanoid robots*. Autonomous Robots, vol. 12, no. 1, pages 105–118, 2002. (Cited on page 30.)
- [Kuindersma 2015] Scott Kuindersma, Robin Deits, Maurice Fallon, Andrés Valenzuela, Hongkai Dai, Frank Permenter, Twan Koolen, Pat Marion and Russ Tedrake. *Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot*. Autonomous Robots, vol. 40, no. 3, pages 429–455, 2015. (Cited on page 31.)
- [Lamboni 2011] M. Lamboni, H. Monod and D. Makowski. *Multivariate sensitivity analysis to measure global contribution of input factors in dynamic models*. Reliability Engineering & System Safety, vol. 96, no. 4, pages 450–459, 2011. (Cited on page 51.)
- [Lämkkull 2009] D. Lämkkull, L. Hanson and R. Örtengren. *A comparative study of digital human modelling simulation results and their outcomes in reality: A case study within manual assembly of automobiles*. International Journal of Industrial Ergonomics, vol. 39, no. 2, pages 428–441, 2009. (Cited on page 42.)
- [LaValle 2006] Steven M LaValle. Planning algorithms. Cambridge university press, 2006. (Cited on pages 7 and 30.)
- [Lee 2012a] J. Lee, N. Mansard and J. Park. *Intermediate Desired Value Approach for Task Transition of Robots in Kinematic Control*. IEEE Transactions on Robotics, vol. 28, no. 6, pages 1260–1277, 2012. (Cited on page 35.)
- [Lee 2012b] Sung-Hee Lee and Ambarish Goswami. *A momentum-based balance controller for humanoid robots on non-level and non-stationary ground*. Autonomous Robots, vol. 33, no. 4, pages 399–414, 2012. (Cited on page 34.)
- [Lengagne 2013] Sébastien Lengagne, Joris Vaillant, Eiichi Yoshida and Abderrahmane Kheddar. *Generation of whole-body optimal dynamic multi-contact motions*. The International Journal of Robotics Research, vol. 32, no. 9-10, pages 1104–1119, 2013. (Cited on page 31.)
- [Li 1999] G. Li and P. Buckle. *Current techniques for assessing physical exposure to work-related musculoskeletal risks, with emphasis on posture-based methods*. Ergonomics, vol. 42, no. 5, pages 674–695, 1999. (Cited on pages 41 and 44.)
- [Liégeois 1977] Alain Liégeois. *Automatic Supervisory Control of the Configuration and Behavior of Multi-body Mechanisms*. IEEE Transactions on Systems, Man and Cybernetics, vol. 7, no. 12, pages 868–871, Dec 1977. (Cited on pages 7 and 11.)
- [Liu 2012a] M. Liu, A. Micaelli, P. Evrard and A. Escande. *Task-driven posture optimization for virtual characters*. Proceedings of the 11th ACM SIGGRAPH/Eurographics conference on Computer Animation, pages 155–164, 2012. (Cited on page 62.)
- [Liu 2012b] Mingxing Liu, Alain Micaelli, Paul Evrard, Adrien Escande and Claude Andriot. *Interactive virtual humans: A two-level prioritized control framework with wrench bounds*. Robotics, IEEE Transactions on, vol. 28, no. 6, pages 1309–1322, 2012. (Cited on page 36.)
- [Liu 2015] M. Liu, S. Hak and V. Padois. *Generalized Projector for Task Priority Transitions During Hierarchical Control*. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 768–773, Seattle, USA, May 2015. (Cited on page 13.)
- [Liu 2016a] M. Liu, R. Lober and V. Padois. *Whole-Body Hierarchical Motion and Force Control for Humanoid Robots*. Autonomous Robots, vol. 40, no. 3, pages 493–504, 2016. (Cited on pages 13 and 36.)

- [Liu 2016b] M. Liu, Y. Tan and V. Padois. *Generalized hierarchical control*. Autonomous Robots, vol. 40, no. 1, pages 17–31, 2016. (Cited on page 13.)
- [Lober 2014] R. Lober, V. Padois and O. Sigaud. *Multiple Task Optimization using Dynamical Movement Primitives for Whole-Body Reactive Control*. In Proceedings of the IEEE-RAS International Conference on Humanoid Robots, pages 193–198, Madrid, Spain, November 2014. (Cited on page 36.)
- [Lober 2015] Ryan Lober, Vincent Padois and Olivier Sigaud. *Variance modulated task prioritization in Whole-Body Control*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 3944–3949, 2015. (Cited on page 37.)
- [Luttmann 2003] A. Luttmann, M. Jäger, B. Griefahn, G. Caffier, F. Liebers and U. Steinberg. *Preventing musculoskeletal disorders in the workplace*. World Health Organization. Protecting Workers’ Health Series, vol. 5, 2003. (Cited on pages 40, 41 and 60.)
- [Ma 2009] L. Ma, W. Zhang, D. Chablat, F. Bennis and F. Guillaume. *Multi-objective optimisation method for posture prediction and analysis with consideration of fatigue effect and its application case*. Computers & Industrial Engineering, vol. 57, no. 4, pages 1235–1246, 2009. (Cited on page 46.)
- [Maciejewski 1985] Anthony A. Maciejewski and Charles A. Klein. *Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments*. The international journal of robotics research, vol. 4, no. 3, pages 109–117, 1985. (Cited on pages 11 and 12.)
- [Mansard 2009a] N. Mansard and F. Chaumette. *Directional Redundancy for Robot Control*. IEEE Transactions on Automatic Control, vol. 54, no. 6, pages 1179–1192, June 2009. (Cited on page 11.)
- [Mansard 2009b] N. Mansard, O. Khatib and A. Kheddar. *A Unified Approach to Integrate Unilateral Constraints in the Stack of Tasks*. IEEE Transactions on Robotics, vol. 25, no. 3, pages 670–685, June 2009. (Cited on page 11.)
- [Mansard 2009c] N. Mansard, A. Remazeilles and F. Chaumette. *Continuity of Varying-Feature-Set Control Laws*. IEEE Transactions on Automatic Control, vol. 54, no. 11, pages 2493–2505, 2009. (Cited on page 35.)
- [Marras 1993] W.S. Marras, S.A. Lavender, S.E. Leurgans, S.L. Rajulu, W.G. Allread, F.A. Fathallah and S.A. Ferguson. *The Role of Dynamic Three-Dimensional Trunk Motion in Occupationally-Related Low Back Disorders: The Effects of Workplace Factors, Trunk Position, and Trunk Motion Characteristics on Risk of Injury*. Spine, vol. 18, no. 5, pages 617–628, 1993. (Cited on page 41.)
- [Maurice 2014a] P. Maurice, Y. Measson, V. Padois and P. Bidaud. *Experimental assessment of the quality of ergonomic indicators for collaborative robotics computed using a digital human model*. In 3rd Digital Human modeling Symposium, Tokyo, Japan, May 2014. (Cited on page 45.)
- [Maurice 2014b] P. Maurice, P. Schlehuber, V. Padois, Y. Measson and P. Bidaud. *Automatic selection of ergonomic indicators for the design of collaborative robots: A virtual-human in the loop approach*. 14th IEEE-RAS International Conference on Humanoid Robots, pages 801–808, 2014. (Cited on pages 49 and 52.)
- [Maurice 2015] P. Maurice. *Virtual ergonomics for the design of collaborative robots*. PhD thesis, Université Pierre et Marie Curie-Paris VI, 2015. (Cited on pages 39, 40 and 47.)
- [McAtamney 1993] L. McAtamney and E.N. Corlett. *RULA: a survey method for the investigation of work-related upper limb disorders*. Applied Ergonomics, vol. 24, no. 2, pages 91–99, 1993. (Cited on page 41.)

- [Meguenani 2015] A. Meguenani, V. Padois and P. Bidaud. *Control of robots sharing their workspace with humans: an energetic approach to safety*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 4678–4684, Hamburg, Germany, September 2015. (Cited on page 16.)
- [Meguenani 2017] A. Meguenani, V. Padois, J. Da Silva, A. Hoarau and P. Bidaud. *Energy-based control for safe Human-robot physical interactions*. In D. Kulic, G. Venture, Y. Nakamura and O. Khatib, editors, Springer Proceedings in Advanced Robotics – The 2016 International Symposium on Experimental Robotics, pages –. Springer International Publishing AG, 2017. (Cited on page 16.)
- [Merlhiot 2012] X. Merlhiot, J. Le Garrec, G. Saupin and C. Andriot. *The XDE mechanical kernel: Efficient and robust simulation of multibody dynamics with intermittent nonsmooth contacts*. Proceedings of the 2nd Joint International Conference on Multibody System Dynamics, 2012. (Cited on pages 53 and 54.)
- [Mombaur 2009] Katja Mombaur. *Using optimization to create self-stable human-like running*. Robotica, vol. 27, no. 03, pages 321–330, 2009. (Cited on page 31.)
- [Mordatch 2012] Igor Mordatch, Emanuel Todorov and Zoran Popović. *Discovery of complex behaviors through contact-invariant optimization*. ACM Transactions on Graphics (TOG), vol. 31, no. 4, page 43, 2012. (Cited on page 37.)
- [Mouret 2010] J.B. Mouret, S. Doncieux et al. *Sferesv2: Evolv'in the multi-core world*. Proceedings of the IEEE Congress on Evolutionary Computation, 2010. (Cited on page 53.)
- [Murray 1994] Richard M Murray, Zexiang Li, S Shankar Sastry and S Shankar Sastry. A mathematical introduction to robotic manipulation. CRC PressI Llc, 1994. (Cited on page 22.)
- [Nocedal 1999] J. Nocedal and S. J. Wright. Numerical optimization. Springer-Verlag, USA, 1999. (Cited on page 10.)
- [Occhipinti 1998] E. Occhipinti. *OCRA: a concise index for the assessment of exposure to repetitive movements of the upper limbs*. Ergonomics, vol. 41, no. 9, pages 1290–1311, 1998. (Cited on page 41.)
- [Ott 2011] Christian Ott, Maximo Roa, Gerd Hirzinger et al. *Posture and balance control for biped robots based on contact force optimization*. In Proceedings of the IEEE-RAS International Conference on Humanoid Robots, pages 26–33. IEEE, 2011. (Cited on page 34.)
- [Padois 2008] V. Padois and R. Smits. *Weighted, damped least-square, inverse velocity kinematics solver*. Part of the open-source OROCOS-KDL Kinematics and Dynamics Library http://docs.ros.org/indigo/api/orocos_kdl/html/classKDL_1_1ChainIkSolverVel__wlds.html, 2008. (Cited on page 10.)
- [Perrin 2015] N. Perrin, D. Lau and V. Padois. *Effective Generation of Dynamically Balanced Locomotion with Multiple Non-coplanar Contacts*. In International Symposium on Robotics Research (ISRR'15), 2015. (Cited on page 33.)
- [Petrič 2013] Tadej Petrič and Leon Žlajpah. *Smooth continuous transition between tasks on a kinematic control level: obstacle avoidance as a control problem*. Robotics and Autonomous Systems, vol. 61, no. 9, pages 948–959, 2013. (Cited on page 35.)
- [Pholsiri 2004] C. Pholsiri. *Task-based decision making and control of robotic manipulators*. PhD thesis, The University of Texas at Austin, 2004. (Cited on page 45.)

- [Pontes 2016] J. Pontes, S. Doncieux, C. Santos and V. Padois. *An Adaptive Approach to Humanoid Locomotion*. In Mohammad O. Tokhi and Gurminder S. Virk, editors, *Advances in Cooperative Robotics – Proceedings of the 19th International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines*, pages 437–444. World Scientific, 2016. (Cited on page 65.)
- [Porter 2004] J.M. Porter, K. Case, R. Marshall and M. Freer. *SAMMIE: A Computer-Aided Ergonomics Design Tool*. In *Working postures and movements – Tools for evaluation and engineering*, pages 431–437. CRC Press, 2004. (Cited on page 41.)
- [Pratt 2006] Jerry E Pratt and Russ Tedrake. *Velocity-based stability margins for fast bipedal walking*. In *Fast Motions in Biomechanics and Robotics*, pages 299–324. Springer, 2006. (Cited on page 29.)
- [Raschke 2004] U. Raschke. *The Jack human simulation tool*. In *Working postures and movements – Tools for evaluation and engineering*, pages 431–437. CRC Press, 2004. (Cited on page 41.)
- [Righetti 2011] Ludovic Righetti, Jonas Buchli, Michael Mistry and Stefan Schaal. *Inverse dynamics control of floating-base robots with external constraints: A unified view*. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1085–1090. IEEE, 2011. (Cited on page 20.)
- [Rubrecht 2011a] S. Rubrecht. *Contributions to the control of constrained robots*. PhD thesis, Université Pierre et Marie Curie - Paris VI, September 2011. (Cited on pages 12, 15 and 40.)
- [Rubrecht 2011b] S. Rubrecht, E. Singla, V. Padois, P. Bidaud and M. de Broissia. *Evolutionary design of a robotic manipulator for a highly constrained environment*. In S. Doncieux, N. Bredèche and J.-B. Mouret, editors, *New Horizons in Evolutionary Robotics*, volume 341 of *Studies in Computational Intelligence*, pages 109–121. Springer, 2011. (Cited on pages 40 and 61.)
- [Rubrecht 2012] S. Rubrecht, V. Padois, P. Bidaud, M. de Broissia and M. Da Silva Simoes. *Motion safety and constraints compatibility for multibody robots*. *Autonomous Robots*, vol. 32, no. 3, pages 333–349, 2012. (Cited on pages 6, 15 and 16.)
- [Saab 2013] Layale Saab, Oscar E Ramos, François Keith, Nicolas Mansard, Philippe Soueres and J Fourquet. *Dynamic whole-body motion generation under rigid contacts and other unilateral constraints*. *IEEE Transactions on Robotics*, vol. 29, no. 2, pages 346–362, 2013. (Cited on page 35.)
- [Salaün 2009] C. Salaün, V. Padois and O. Sigaud. *Control of redundant robots using learned models: an operational space control approach*. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 878–885, Saint-Louis, USA, October 2009. (Cited on page 65.)
- [Salaün 2010] Camille Salaün. *Learning models to control redundancy in Robotics*. PhD thesis, Université Pierre et Marie Curie - Paris VI, August 2010. (Cited on page 65.)
- [Salini 2011] Joseph Salini, Vincent Padois and Philippe Bidaud. *Synthesis of complex humanoid whole-body behavior: a focus on sequencing and tasks transitions*. In *Proceedings of IEEE International Conference on Robotics and Automation*, pages 1283–1290, 2011. (Cited on page 35.)
- [Salini 2012] Joseph Salini. *Dynamic control for the task/posture coordination of humanoids : toward synthesis of complex activities*. PhD thesis, Université Pierre et Marie Curie - Paris VI, June 2012. (Cited on page 13.)
- [Saltelli 1999] A. Saltelli, S. Tarantola and K.P.S. Chan. *A quantitative model-independent method for global sensitivity analysis of model output*. *Technometrics*, vol. 41, no. 1, pages 39–56, 1999. (Cited on pages 51 and 55.)

- [Saltelli 2000] A. Saltelli, K. Chan and E.M. Scott. *Sensitivity analysis*. Wiley, 2000. (Cited on pages 49 and 51.)
- [Savin 2017] J. Savin, M. Gilles, C. Gaudez, V. Padois and P. Bidaud. *Movement variability and digital human models: development of a demonstrator taking the effects of muscular fatigue into account*. In Vincent G. Duffy, editeur, *Advances in Applied Digital Human Modeling and Simulation: Proceedings of the AHFE 2016 International Conference on Digital Human Modeling and Simulation*, pages 169–179. Springer International Publishing, 2017. (Cited on page 61.)
- [Schneider 2010] E Schneider and X Irastorza. *OSH in figures: Work-related musculoskeletal disorders in the EU - Facts and figures*. Technical report, European Agency for Safety and Health at Work, 2010. (Cited on page 40.)
- [Seidl 2004] A. Seidl. *The RAMSIS and ANTHROPOS Human Simulation Tools*. In *Working postures and movements – Tools for evaluation and engineering*, pages 454–462. CRC Press, 2004. (Cited on page 41.)
- [Seraji 1989] H. Seraji. *Configuration control of redundant manipulators: theory and implementation*. IEEE Transactions on Robotics and Automation, vol. 5, no. 4, pages 472–490, Aug 1989. (Cited on page 11.)
- [Shen 2007] Weimin Shen and J. Gu. *Multi-criteria kinematics control for the PA10-7C robot arm with robust singularities*. In *Proceedings of the IEEE International Conference on Robotics and Biomimetics*, pages 1242–1248, Dec 2007. (Cited on pages 11 and 12.)
- [Sicard 2011] G. Sicard, C. Salaün, S. Ivaldi, V. Padois and O. Sigaud. *Learning the velocity kinematics of iCub for model-based control: XCSF versus LWPR*. In *Proceedings of the 11th IEEE-RAS International Conference on Humanoid Robots*, pages 570–575, Bled, Slovenia, October 2011. (Cited on page 65.)
- [Siciliano 1991] B. Siciliano and J.-J.E. Slotine. *A general framework for managing multiple tasks in highly redundant robotic systems*. In *Proceedings of the Fifth International Conference on Advanced Robotics*, pages 1211–1216 vol.2, June 1991. (Cited on page 11.)
- [Sigaud 2011] O. Sigaud, C. Salaün and V. Padois. *On-line regression algorithms for learning mechanical models of robots: a survey*. *Robotics and Autonomous Systems*, vol. 59, no. 12, pages 1115–1129, 2011. (Cited on page 65.)
- [Sobol 1993] I.M. Sobol. *Sensitivity estimates for non linear mathematical models*. *Mathematical Modelling and Computational Experiments*, pages 407–414, 1993. (Cited on page 51.)
- [Stephens 2010] Benjamin J Stephens and Christopher G Atkeson. *Push recovery by stepping for humanoid robots with force controlled joints*. In *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*, pages 52–59. IEEE, 2010. (Cited on page 33.)
- [Tan 2016] Yang Tan. *Continuous Tasks and Constraints Transitions for the Control of robots*. PhD thesis, Université Pierre et Marie Curie, March 2016. (Cited on page 17.)
- [Tassa 2012] Yuval Tassa, Tom Erez and Emanuel Todorov. *Synthesis and stabilization of complex behaviors through online trajectory optimization*. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4906–4913. IEEE, 2012. (Cited on pages 28 and 37.)
- [Thelen 2003] D.G. Thelen, F.C. Anderson and S.L. Delp. *Generating dynamic simulations of movement using computed muscle control*. *Journal of biomechanics*, vol. 36, no. 3, pages 321–328, 2003. (Cited on page 44.)

- [Vukobratović 1972] Miomir Vukobratović and J Stepanenko. *On the stability of anthropomorphic systems*. Mathematical biosciences, vol. 15, no. 1, pages 1–37, 1972. (Cited on page 28.)
- [Waters 1993] T.R. Waters, V. Putz-Anderson, A. Garg and L.J. Fine. *Revised NIOSH equation for the design and evaluation of manual lifting tasks*. Ergonomics, vol. 36, no. 7, pages 749–776, 1993. (Cited on page 41.)
- [Whitney 1969] D.E. Whitney. *Resolved Motion Rate Control of Manipulators and Human Prostheses*. IEEE Transactions on Man-Machine Systems, vol. 10, no. 2, pages 47–53, June 1969. (Cited on page 7.)
- [Wieber 2002] Pierre-Brice Wieber. *On the stability of walking systems*. In Proceedings of the international workshop on humanoid and human friendly robotics, 2002. (Cited on page 26.)
- [Wieber 2006] Pierre-Brice Wieber. *Trajectory free linear model predictive control for stable walking in the presence of strong perturbations*. In Proceedings of the IEEE-RAS International Conference on Humanoid Robots, pages 137–142. IEEE, 2006. (Cited on pages 28 and 32.)
- [Wieber 2015] P.-B. Wieber, R. Tedrake and S. Kuindersma. *Modeling and control of legged robots*. In B. Siciliano and O. Khatib, editors, Springer Handbook of Robotics. Springer, 2nd edition, 2015. (Cited on pages 21 and 24.)
- [Xiang 2010] Y. Xiang, J.S. Arora, S. Rahmatalla, T. Marler, R. Bhatt and K. Abdel-Malek. *Human lifting simulation using a multi-objective optimization approach*. Multibody System Dynamics, vol. 23, no. 4, pages 431–451, 2010. (Cited on page 45.)
- [Yoshikawa 1985a] T. Yoshikawa. *Dynamic manipulability of robot manipulators*. In Proceedings of the IEEE International Conference on Robotics and Automation, volume 2, pages 1033–1038. IEEE, 1985. (Cited on page 45.)
- [Yoshikawa 1985b] T. Yoshikawa. *Manipulability of robotic mechanisms*. The International Journal of Robotics Research, vol. 4, no. 2, pages 3–9, 1985. (Cited on page 45.)

Appendix A

Linear system inversion

The derivation of the solutions provided in this appendix are mostly based on the work of A. Ben Israel and T. Greville [Ben Israel 2003].

A.1 Weighted Euclidean norm solutions

In chapter 1, section 1.1.1, the following optimisation problem is posed

$$\mathbf{x}_{sol} = \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{x}\|_{\mathbf{W}_x}^2 \quad (\text{A.1a})$$

$$\text{s.t. } \mathbf{C}\mathbf{x} = \arg \min_{\mathbf{b}} \frac{1}{2} \|\mathbf{F}\mathbf{b} - \mathbf{y}\|_{\mathbf{W}_y}^2 \quad (\text{A.1b})$$

where:

- $\mathbf{W}_x \in \mathbb{R}^{n_x}$ and $\mathbf{W}_y \in \mathbb{R}^{n_y}$ are symmetric, positive definite matrices;
- $\mathbf{A} = \mathbf{F}\mathbf{C}$ is a full rank decomposition of $\mathbf{A}$;
- $\mathbf{F} \in \mathbb{R}^{n_y \times n_r}$ and $\mathbf{C} \in \mathbb{R}^{n_r \times n_x}$ are respectively full-column rank and full-row rank;
- $\mathbf{b} = \mathbf{C}\mathbf{x}$.

The Lagrangian associated to the problem (A.1b) is

$$L_1(\mathbf{b}) = \frac{1}{2} \left(\mathbf{b}^T \mathbf{F}^T \mathbf{W}_y \mathbf{F} \mathbf{b} - 2 \mathbf{y}^T \mathbf{W}_y \mathbf{F} \mathbf{b} + \mathbf{y}^T \mathbf{W}_y \mathbf{y} \right) \quad (\text{A.2})$$

and the nullity of its gradient provides the optimal solution

$$\mathbf{b}_{sol} = \left(\mathbf{F}^T \mathbf{W}_y \mathbf{F} \right)^{-1} \mathbf{F}^T \mathbf{W}_y \mathbf{y}. \quad (\text{A.3})$$

From there, the second problem (A.1a) can be written

$$\mathbf{x}_{sol} = \arg \min_{\mathbf{x}} \underbrace{\frac{1}{2} \mathbf{x}^T \mathbf{W}_x \mathbf{x}}_{L_2(\mathbf{x})} \quad (\text{A.4a})$$

$$\text{s.t. } \underbrace{\mathbf{C}\mathbf{x} - \left(\mathbf{F}^T \mathbf{W}_y \mathbf{F} \right)^{-1} \mathbf{F}^T \mathbf{W}_y \mathbf{y}}_{\mathbf{c}_2(\mathbf{x})} = \mathbf{0}. \quad (\text{A.4b})$$

The Karush-Kuhn-Tucker (KKT) optimality conditions for this problem are

$$\mathbf{c}_2(\mathbf{x}_{sol}) = \mathbf{0} : \quad \mathbf{C}\mathbf{x}_{sol} = \left(\mathbf{F}^T \mathbf{W}_y \mathbf{F}\right)^{-1} \mathbf{F}^T \mathbf{W}_y \mathbf{y} \quad (\text{A.5a})$$

$$\nabla L_2(\mathbf{x}_{sol}) - \boldsymbol{\lambda}^{T, sol} \nabla \mathbf{c}_2(\mathbf{x}_{sol}) = \mathbf{0} : \quad \mathbf{W}_x \mathbf{x}_{sol} - \mathbf{C}^T \boldsymbol{\lambda}_{sol} = \mathbf{0} \quad (\text{A.5b})$$

where $\boldsymbol{\lambda}$ is the vector of Lagrange multipliers associated to the constraints. This yields the following linear system of equations

$$\begin{bmatrix} \mathbf{W}_x & -\mathbf{C}^T \\ \mathbf{C} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x}_{sol} \\ \boldsymbol{\lambda}_{sol} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \left(\mathbf{F}^T \mathbf{W}_y \mathbf{F}\right)^{-1} \mathbf{F}^T \mathbf{W}_y \mathbf{y} \end{bmatrix}. \quad (\text{A.6})$$

This problem has a unique solution which can be computed using block-wise inversion, giving the solution

$$\begin{bmatrix} \mathbf{x}_{sol} \\ \boldsymbol{\lambda}_{sol} \end{bmatrix} = \begin{bmatrix} \mathbf{W}_x^{-1} \mathbf{C}^T \left(\mathbf{C} \mathbf{W}_x^{-1} \mathbf{C}^T\right)^{-1} \left(\mathbf{F}^T \mathbf{W}_y \mathbf{F}\right)^{-1} \mathbf{F}^T \mathbf{W}_y \mathbf{y} \\ \left(\mathbf{C} \mathbf{W}_x^{-1} \mathbf{C}^T\right)^{-1} \left(\mathbf{F}^T \mathbf{W}_y \mathbf{F}\right)^{-1} \mathbf{F}^T \mathbf{W}_y \mathbf{y} \end{bmatrix}. \quad (\text{A.7})$$

The solution can thus be written

$$\mathbf{x}_{sol} = \mathbf{A}_{\mathbf{W}_x, \mathbf{W}_y}^+ \mathbf{y} \quad (\text{A.8})$$

where

$$\boxed{\mathbf{A}_{\mathbf{W}_x, \mathbf{W}_y}^+ = \mathbf{W}_x^{-1} \mathbf{C}^T \left(\mathbf{C} \mathbf{W}_x^{-1} \mathbf{C}^T\right)^{-1} \left(\mathbf{F}^T \mathbf{W}_y \mathbf{F}\right)^{-1} \mathbf{F}^T \mathbf{W}_y.} \quad (\text{A.9})$$

■

A.2 Regularised, Weighted Euclidean norm solutions

In chapter 1, section 1.1.1, the following, three stages, optimisation problem is posed

$$\mathbf{x}_{sol} = \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{x}\|_{\mathbf{W}_x}^2 \quad (\text{A.10a})$$

$$\text{s.t. } \underbrace{\mathbf{V}' \mathbf{x} = \arg \min_{\mathbf{b}} \frac{1}{2} \left(\left\| \boldsymbol{\Sigma} \mathbf{b} - \underbrace{\arg \min_{\mathbf{c}} \frac{1}{2} \|\mathbf{U}' \mathbf{c} - \mathbf{y}\|_{\mathbf{W}_y}^2}_{\mathbf{c}_{sol}} \right\|^2 + \|\mathbf{b}\|_{\boldsymbol{\Lambda}}^2 \right)}_{\mathbf{b}_{sol}} \quad (\text{A.10b})$$

Similarly to problem (A.1b), the first stage of the problem is written

$$\mathbf{c}_{sol} = \arg \min_{\mathbf{c}} \frac{1}{2} \|\mathbf{U}' \mathbf{c} - \mathbf{y}\|_{\mathbf{W}_y}^2 \quad (\text{A.11})$$

and yields the solution

$$\mathbf{c}_{sol} = \left(\mathbf{U}'^T \mathbf{W}_y \mathbf{U}'\right)^{-1} \mathbf{U}'^T \mathbf{W}_y \mathbf{y}. \quad (\text{A.12})$$

This leads to the second stage of the problem

$$\mathbf{b}_{sol} = \arg \min_{\mathbf{b}} \frac{1}{2} \left(\|\boldsymbol{\Sigma} \mathbf{b} - \mathbf{c}_{sol}\|^2 + \|\mathbf{b}\|_{\boldsymbol{\Lambda}}^2 \right). \quad (\text{A.13})$$

The Lagrangian associated to this problem is given by

$$L_3 = \frac{1}{2} \left(\mathbf{b}^T \boldsymbol{\Sigma}^T \boldsymbol{\Sigma} \mathbf{b} - 2 \mathbf{c}_{sol}^T \boldsymbol{\Sigma} \mathbf{b} + \mathbf{c}_{sol}^T \mathbf{c}_{sol} + \mathbf{b}^T \boldsymbol{\Lambda} \mathbf{b} \right) \quad (\text{A.14})$$

and the optimality condition $\nabla L_3(\mathbf{b}_{sol}) = \mathbf{0}$ leads to the intermediate solution

$$\mathbf{b}_{sol} = \left(\Sigma^T \Sigma + \Lambda \right)^{-1} \Sigma^T \mathbf{c}_{sol}. \quad (\text{A.15})$$

Finally, the last sub-problem

$$\mathbf{x}_{sol} = \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{x}\|_{\mathbf{W}_x}^2 \quad (\text{A.16a})$$

$$\text{s.t. } \mathbf{V}' \mathbf{x} = \mathbf{b}_{sol}, \quad (\text{A.16b})$$

is solved similarly to problem (A.1a) and the solution can be written

$$\mathbf{x}_{sol} = \mathbf{W}_x^{-1} \mathbf{V}'^T \left(\mathbf{V}' \mathbf{W}_x^{-1} \mathbf{V}'^T \right)^{-1} \mathbf{b}_{sol} \quad (\text{A.17})$$

and the regularised, weighted pseudo-inverse of $\mathbf{A}$ is given by

$$\mathbf{A}_{\mathbf{W}_x, \mathbf{W}_y, \Lambda}^+ = \mathbf{W}_x^{-1} \mathbf{V}'^T \left(\mathbf{V}' \mathbf{W}_x^{-1} \mathbf{V}'^T \right)^{-1} \left(\Sigma^T \Sigma + \Lambda \right)^{-1} \Sigma^T \left(\mathbf{F}^T \mathbf{W}_y \mathbf{F} \right)^{-1} \mathbf{F}^T \mathbf{W}_y. \quad (\text{A.18})$$

■

A.3 Linear systems inversion under constraints

In chapter 1, section 1.1.2, the following optimisation problem is posed

$$\mathbf{x}_{sol} = \arg \min_{\mathbf{x}} \frac{1}{2} \left(\|\mathbf{A} \mathbf{x} - \mathbf{y}\|^2 \right) \quad (\text{A.19a})$$

$$\text{s.t. } \mathbf{C}_a \mathbf{x} = \mathbf{c}_a. \quad (\text{A.19b})$$

Assuming that $\mathbf{C}_a$ is full-row rank, there exist at least one exact solution to this optimisation problem which can be written

$$\mathbf{x} = \mathbf{C}_a^+ \mathbf{c}_a + \mathbf{P}_{N(\mathbf{C}_a)} \mathbf{x}_0. \quad (\text{A.20})$$

An optimal choice for $\mathbf{x}_0$ is given by solving

$$\mathbf{x}_{0,sol} = \arg \min_{\mathbf{x}_0} \frac{1}{2} \left(\|\mathbf{A} (\mathbf{C}_a^+ \mathbf{c}_a + \mathbf{P}_{N(\mathbf{C}_a)} \mathbf{x}_0) - \mathbf{y}\|^2 \right) \quad (\text{A.21})$$

leading to

$$\mathbf{x}_{0,sol} = (\mathbf{A} \mathbf{P}_{N(\mathbf{C}_a)})^+ (\mathbf{y} - \mathbf{A} \mathbf{C}_a^+ \mathbf{c}_a). \quad (\text{A.22})$$

The solution $\mathbf{x}_{sol}$ thus writes

$$\mathbf{x}_{sol} = \mathbf{C}_a^+ \mathbf{c}_a + \mathbf{P}_{N(\mathbf{C}_a)} (\mathbf{A} \mathbf{P}_{N(\mathbf{C}_a)})^+ (\mathbf{y} - \mathbf{A} \mathbf{C}_a^+ \mathbf{c}_a). \quad (\text{A.23})$$

■

The same solution can be obtained by writing the KKT conditions related to problem (A.19) however the demonstration is left to the reader.

Appendix B

Selected publications

B.1 Generalised Hierarchical Control

Generalized hierarchical control

Mingxing Liu^{1,2} · Yang Tan^{1,2} · Vincent Padois^{1,2}

Received: 4 December 2014 / Accepted: 27 April 2015 / Published online: 13 May 2015
© Springer Science+Business Media New York 2015

Abstract Multi-objective control systems for complex robots usually have to handle multiple prioritized tasks. Most existing hierarchical control techniques handle either strict task priorities by using null-space projectors or a sequence of quadratic programs; or non strict task priorities by using a weighting strategy. This paper proposes a novel approach to handle both strict and non-strict priorities of an arbitrary number of tasks. It can achieve multiple priority rearrangements simultaneously. A generalized projector, which makes it possible to completely project a task into the null-space of a set of tasks, while partially projecting it into the null-space of some other tasks, is developed. This projector can be used to perform priority transitions and task insertion or deletion. The control input is computed by solving one quadratic programming problem, where generalized projectors are adopted to maintain a task hierarchy, and equality or inequality constraints can be implemented. The effectiveness of this approach is demonstrated on a simulated robotic manipulator in a dynamic environment.

Electronic supplementary material The online version of this article (doi:[10.1007/s10514-015-9436-1](https://doi.org/10.1007/s10514-015-9436-1)) contains supplementary material, which is available to authorized users.

✉ Mingxing Liu
liu@isir.upmc.fr

Yang Tan
tan@isir.upmc.fr

Vincent Padois
padois@isir.upmc.fr

¹ Sorbonne Universités, UPMC Univ Paris 06, UMR 7222, Institut des Systèmes Intelligents et de Robotique (ISIR), F-75005 Paris, France

² CNRS, UMR 7222, ISIR, F-75005 Paris, France

Keywords Redundant robots · Task hierarchy · Priority switching · Dynamics · Torque-based control

1 Introduction

Redundant robots are nowadays expected to perform complex missions involving the simultaneous performance of multiple tasks. Even though robot redundancy makes it possible for these robots to perform multiple tasks simultaneously, task conflicts may still occur when all the task objectives cannot be satisfied at the same time. In order to handle conflicts, tasks are usually assigned with different priority levels. Therefore, controllers for complex robots must be able to handle multiple prioritized tasks and respect various constraints imposed by the robot body and the environment.

A large number of hierarchical control frameworks are presented in the robotics literature for the management of multiple operational task objectives. Some of them deal with *strict task hierarchies*, which ensure that critical tasks are fulfilled with higher priorities and lower-priority tasks are performed only in the null-space of higher priority tasks. Other approaches handle *non-strict task hierarchies*. The solution of these approaches is a compromise among task objectives of different weights. In a non-strict task hierarchy, a lower priority task is not restricted in the null-space of higher priority tasks, thus it may still affect their performances.

In a more general context, the robot may need to deal with both strict and non-strict hierarchies. Moreover, for robots acting in dynamically changing contexts, non-strict priorities between tasks may become strict ones and task priorities may have to be switched in order to cope with changing situations.

With the aim of handling both strict and non-strict hierarchies simultaneously, a novel control framework called

generalized hierarchical control (GHC) is presented in this paper. The contributions of this work are as follows. (i) The development of a generalized projector, which can regulate to what extent a lower-priority task is projected into the null-space of a higher-priority task. In other words, this generalized projector allows a task to be completely, partially, or not at all projected into the null-space of some other tasks. (ii) The development of a generic dynamic hierarchical control framework, which solves a single quadratic program (QP) and uses generalized projectors as a mechanism to account for an arbitrary number of strict and non-strict task priorities. It can achieve desired priority transitions, as well as an elegant way of inserting and deleting tasks among those to be performed. Task hierarchies are handled by the modulation of a priority matrix, without the necessity of modifying the control problem formulation each time the hierarchies change.

2 Related works

This section reviews some classical hierarchical control frameworks and priority transitions within them.

2.1 Approaches for handling a strict hierarchy

Analytical methods based on null-space projections can ensure that lower priority tasks are executed only in the null-space of higher-priority tasks. The idea is based on the use of the limited Jacobian of a lower-priority task, which is projected into the null-space of higher priority tasks by the application of a null-space projector (Liégeois 1977). Such an idea is applied in prioritized inverse kinematics (Mistry et al. 2007, 2008), in acceleration based control (Hsu et al. 1989; Flacco et al. 2012b), and in joint torque based control (Khatib 1987; Sentis and Khatib 2004a; Sentis et al. 2010). A generic framework, from which several existing control laws can be derived, is presented in Peters et al. (2008). Projected inverse dynamics schemes are developed for constrained systems in Aghili (2005), Khatib et al. (2008), where the dynamics equation is projected into the null-space of the Jacobian of constraint equations. As the limited Jacobians mentioned above could be rank deficient, task priority strategies involving their pseudo-inverses may lead to algorithmic singularities. To overcome the effects of such singularities, a technique based on damped least-squares and extended Jacobian (Egeland et al. 1991) is proposed, which requires to choose the damping factor carefully in order to guarantee good behaviors near singularities.

Inequality constraints are usually difficult to be directly dealt with in analytical approaches using pseudo-inverses and projection matrices. A common method is to transform inequality constraints into task objectives by apply-

ing artificial potential fields (Khatib 1986), from which repulsive forces are derived to prevent the robot from entering into activation zones of the inequality constraints (Sentis and Khatib 2004b, 2005; Khatib 1986; Stasse et al. 2008; Padois et al. 2007; Saab et al. 2009). However, performing these tasks cannot guarantee that these inequality constraints are actually met. The approach presented in Mansard et al. (2009) integrates unilateral constraints at any priority level, albeit time consuming. The algorithm introduced in Flacco et al. (2012a, b) proposes to disable the most critical joint and redistribute joint motion commands to guarantee the satisfaction of some hard bounds of joint variables. However, this algorithm deals with inequality constraints only at the joint level. Furthermore, the optimal solution satisfying the control problem may require the movement of a joint which has unfortunately been disabled.

To deal with prioritized inequality constraints more easily, hierarchical quadratic programming (HQP) approaches use numerical QP solvers to solve a hierarchical quadratic program (Kanoun et al. 2009; Saab et al. 2011, 2013; Escande et al. 2014). The idea of HQP is to first solve a QP to obtain a solution for a higher priority task objective; and then to solve another QP for a lower priority task, without increasing the obtained minimum of the previous task objective. This prioritization process corresponds to solving lower-priority tasks in the null-space of higher-priority tasks while trying to satisfy lower-priority tasks at best.

Generally, approaches that handling strict hierarchies parameterize the relative priority of one task with respect to another one of a different importance level in a lexicographic way (Saab et al. 2013): either strictly higher or strictly lower. However, in many contexts, organizing tasks by assigning them with strict priorities is not generic, i.e. can have some limitations. First, a strict priority is just an extreme case of the relative priority of tasks. In fact, a task may not always have a strict priority over another one and it is usually difficult to define a strict hierarchy among a set of tasks. Second, strict priorities can sometimes be too conservative so that they may completely block lower-priority tasks. Compared with a discrete parameterization of strict task priorities, a continuous parameterization of both strict and non-strict task priorities is richer and more informative. Therefore, this work uses a continuous priority parameterization. Moreover, priorities are defined here by pairs of tasks and are encoded by a priority matrix. This choice of priority representation can handle not only a single standard lexicographic hierarchy as HQP does, but also a complex priority network. For example, it can represent two lexicographic hierarchies $1 \succ 2 \succ 3^1$ and $4 \succ 5 \succ 6$, with an additional relationship $2 \succ 5$, leaving the relationships among all the other pairs of tasks free.

¹ The notation $i \succ j$ indicates that task i has a strict higher priority over task j .

2.2 Approaches for handling a non-strict hierarchy

Non-strict priorities are usually handled by approaches using weighting strategies (Abe et al. 2007; Collette et al. 2007; Liu et al. 2011; Salini et al. 2011; Bouyarmane and Kheddar 2011). These control frameworks solve all the constraints and task objectives in one QP and provide a trade-off among task objectives with different importance levels. As the performances of higher priority tasks cannot be guaranteed by simply adjusting the weights of task objectives, a prioritized control framework is proposed in Liu et al. (2012) to ensure the performance of a higher-priority task. However, this approach handles priorities of only two levels. In approaches based on weighting strategies, task priorities can be parameterized continuously. Nonetheless, even though the work in Kerrigan and Maciejowski (2000) on soft constraints in model predictive control could probably be adapted to provide a way to reach the extreme case of strict priorities, the existing robotic applications of these frameworks do not extend to strict hierarchies. The control framework proposed in this paper outperforms weighting strategies by permitting priorities to change gradually from a non-strict case to a strict case.

2.3 Task transitions

Recently, different methods have been developed to handle task transition problems. An approach to smooth priority rearrangement between only two levels of tasks is proposed in Keith et al. (2011), Petrič and Žlajpah (2013). A specific inverse operator is proposed in Mansard et al. (2009) to ensure continuous inverse in the analytical computation of control laws. The approach presented in Lee et al. (2012) is based on intermediate desired values in the task space. When the number of task transitions increases, this approach suggests to apply an approximation to reduce the computational cost. An approach of hierarchical control with continuous null-space projections is presented in Dietrich et al. (2012). However, the design of an activator used by this approach makes it difficult to be implemented for the separate handling of different task directions. On the other hand, task transitions can be easily achieved within a non-strict hierarchy by the continuous variation of task weights (Salini et al. 2011). This method is used in HQP approaches to swap the priorities of tasks coming from two consecutive priority levels (Jarquin et al. 2013). This is achieved by tuning the weights of the tasks, which are merged in the same priority level, to comply with their priorities before and after the transition phase. However this strategy may require a set of swaps before bringing a task to the desired priority level. A novelty of the control framework proposed in this paper is that it allows the simultaneous priority rearrangements for an arbitrary number of pairs of tasks, and it requires only one

swap to switch priorities between each pair of tasks at two non-consecutive levels.

3 Modeling

Consider a robot as an articulated mechanism with n degrees of freedom (DoF) including n_a actuated DoF. The dynamics of the robot in terms of its generalized coordinates $\mathbf{q} \in \mathbb{R}^n$ is written as follows

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{J}_c(\mathbf{q})^T \boldsymbol{\chi}, \quad (1)$$

where $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n \times n}$ is the generalized inertia matrix; $\dot{\mathbf{q}} \in \mathbb{R}^n$ and $\ddot{\mathbf{q}} \in \mathbb{R}^n$ are the vector of velocity and the vector of acceleration in generalized coordinates, respectively; $\mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^n$ is the vector of Coriolis, centrifugal and gravity induced joint torques; $\boldsymbol{\chi} = [\mathbf{w}_c^T \boldsymbol{\tau}^T]^T$ is the vector of the actuation torques ($\boldsymbol{\tau} \in \mathbb{R}^{n_a}$) and the external contact wrenches applied to the robot ($\mathbf{w}_c = [\mathbf{w}_{c,1}^T \dots \mathbf{w}_{c,n_c}^T]^T$), with n_c , the number of contact points; $\mathbf{J}_c(\mathbf{q})^T = [\mathbf{J}_{c,1}(\mathbf{q})^T \dots \mathbf{J}_{c,n_c}(\mathbf{q})^T \mathbf{S}(\mathbf{q}, \dot{\mathbf{q}})^T]$ is the transpose of a Jacobian matrix, with $\mathbf{J}_{c,n_\beta}(\mathbf{q})$, the Jacobian matrix associated to a contact point β and $\mathbf{S}(\mathbf{q}, \dot{\mathbf{q}})^T \in \mathbb{R}^{n \times n_a}$, a selection matrix for the actuated DoF. In the control problem considered in this paper, the vector $\boldsymbol{\chi}$ is called the action variable.

A task i of a physical frame attached to the robot body can be defined by the following characteristics:

- (I) A task variable $\xi_i \in \mathbb{R}^{m_i}$ expressed in terms of some goals to be achieved by the task frame in the task space, such as a desired position or orientation of dimension m_i . The second order derivative of ξ_i can be linearly related to that of $\mathbf{q}$

$$\ddot{\xi}_i = \mathbf{J}_i(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{J}}_i(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} \quad (2)$$

where $\mathbf{J}_i(\mathbf{q})$ is the Jacobian matrix representing the differential kinematics mapping from joint space to task space, and $\dot{\mathbf{J}}_i(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}$ is the task space drift vector.

- (II) A local controller $\ddot{\xi}_i^d$, the goal of which is to correct task errors and ensure the convergence of the task variable ξ_i towards its desired trajectory. For task motion control, $\ddot{\xi}_i^d$ can take the form of a proportional-integral-derivative controller with a feed-forward term. For task wrench control, $\ddot{\xi}_i^d$ can take the form of a proportional-integral controller with a feed-forward term. The wrench task can be expressed as a motion task using the inverse of the operational space inertia matrix $\mathbf{A}_i(\mathbf{q}) = [\mathbf{J}_i(\mathbf{q})\mathbf{M}(\mathbf{q})^{-1}\mathbf{J}_i(\mathbf{q})^T]^{-1}$ (Khatib 1995; Chang and Khatib 1999)

$$\ddot{\xi}_i^d = \mathbf{A}_i(\mathbf{q})^{-1}\mathbf{w}_i^d \quad (3)$$

which maps the desired task wrench $\mathbf{w}_i^d$ to a desired acceleration $\ddot{\mathbf{x}}_i^d$ at the task frame.

- (III) A set of relative importance levels with respect to n_t tasks, including task i , characterized by a priority matrix

$$\mathbf{A}_i = \text{diag}(\alpha_{i1}\mathbf{I}_{m_1}, \dots, \alpha_{ij}\mathbf{I}_{m_j}, \dots, \alpha_{in_t}\mathbf{I}_{m_{n_t}}) \quad (4)$$

where $\mathbf{A}_i$ is a diagonal matrix, the main diagonal blocks of which are square matrices: $\alpha_{ij}\mathbf{I}_{m_j}$. $\mathbf{I}_{m_j}$ is the $m_j \times m_j$ identity matrix, and $\alpha_{ij} \in [0, 1]$. By convention, the coefficient α_{ij} indicates the priority of task j with respect to task i .

- $\alpha_{ij} = 0$ corresponds to the case where task j has strict lower priority with respect to task i .
- $0 < \alpha_{ij} < 1$ corresponds to a non-strict priority between the two tasks: the greater the value of α_{ij} , the higher the importance level of task j with respect to task i .
- $\alpha_{ij} = 1$ corresponds to the case where task j has a strict higher priority with respect to task i .

The role of the particular element α_{ii} is given and explained in detail in Sect. 4.2.1.

4 Generalized projector for hierarchical control

The hierarchical control proposed in this paper is based on a new generalized projector, which can precisely regulate how much a task is affected by other tasks. The following part of this subsection first looks at several forms of projectors, then the analysis of which leads to the development of the generalized projector.

4.1 Review of projectors for hierarchical control

Strict priorities can be handled by analytical methods using a null-space projector $\mathbf{N}_j = \mathbf{I} - \mathbf{J}_j^\dagger \mathbf{J}_j$, where $\mathbf{J}_j^\dagger$ is the Moore–Penrose pseudo-inverse of the Jacobian $\mathbf{J}_j$.² The projection of a task i into the null-space of another task j can ensure that the lower-priority task i is performed without producing any motion for the higher-priority task j . To handle priorities between one task i and a set of other tasks with higher priorities, task i is projected into the null-space of an augmented Jacobian of all the higher priority tasks (Siciliano and Slotine 1991; Baerlocher and Boulic 1998).

To achieve smooth priority transitions, the null-space projector is replaced by the following matrix in Keith et al. (2011), Petrić and Žlajpah (2013)

$$\mathbf{N}'_j(\alpha_{ij}) = \mathbf{I} - \alpha_{ij}\mathbf{J}_j^\dagger \mathbf{J}_j, \quad (5)$$

where a scalar parameter $\alpha_{ij} \in [0, 1]$ is used to regulate the priority between two tasks i and j . The greater α_{ij} is, the more task i is projected into the null-space of task j . This method can handle priority transitions between only two levels of tasks, and it can hardly be extended to the case of simultaneous transitions among multiple priority levels.

The following matrix $\mathbf{N}''$ is proposed in Dietrich et al. (2012) for continuous null-space projections

$$\mathbf{N}'' = \mathbf{I} - \mathbf{V}\mathbf{\Theta}\mathbf{V}^T, \quad (6)$$

where $\mathbf{V} \in \mathbb{R}^{n \times n}$ is the right singular vectors of $\mathbf{J}_j$, the Jacobian of a higher priority task, and $\mathbf{\Theta} \in \mathbb{R}^{n \times n}$ is a diagonal activation matrix. The k th diagonal element of $\mathbf{\Theta}$, $\theta_{kk} \in [0, 1]$, refers to the k th column vector in $\mathbf{V}$: when $\theta_{kk} = 1$, the k th direction in $\mathbf{V}$ is activated in $\mathbf{N}''$; when $0 < \theta_{kk} < 1$, the k th direction in $\mathbf{V}$ is partially deactivated; when $\theta_{kk} = 0$, the k th direction in $\mathbf{V}$ is deactivated. As mentioned in Dietrich et al. (2012), for any one-dimensional task j ($\mathbf{J}_j \in \mathbb{R}^{1 \times n}$), the matrix (6) becomes

$$\mathbf{N}''_j = \mathbf{I} - \theta_{11} \frac{\mathbf{J}_j^T}{\|\mathbf{J}_j\|} \frac{\mathbf{J}_j}{\|\mathbf{J}_j\|}, \quad (7)$$

where only the first element θ_{11} of $\mathbf{\Theta}$ is relevant. $\mathbf{N}''_j$ can be applied to achieve activation or deactivation of task j direction in the projection matrix by the variation of the scalar θ_{11} . When extended to a task (or a set of tasks) of m directions ($\mathbf{J}_j \in \mathbb{R}^{m \times n}$), this method allows one to apply the same transition to all the m directions of $\mathbf{J}_j$, but its application for achieving the separate regulation of each task direction is not easy. This is because each activator θ_{kk} is directly referred to the k th direction in the right singular vectors of $\mathbf{J}_j$, but not directly referred to a specific direction in $\mathbf{J}_j$.

4.2 Generalized projector

In order to achieve variations of multiple task priorities simultaneously among an arbitrary number of tasks, and to be able to ensure a priority network with both strict and non-strict priorities, an approach to the computation of a generalized projector $\mathbf{P}_i(\mathbf{A}_i) \in \mathbb{R}^{n \times n}$ is developed in this section. Here the subscript i in $\mathbf{P}_i$ indicates that the projector takes into account the priorities of a set of tasks with respect to task i . The dependence of $\mathbf{P}_i$ to $\mathbf{A}_i$ is sometimes omitted hereafter for clarity reasons. Similarly to the form of the matrix $\mathbf{N}''$ in the case of considering a one-dimensional task (7), the form of $\mathbf{P}_i$ is obtained without the necessity of the computation of pseudo-inverse matrices. Moreover, the new projector allows one to regulate the activation of each task directions in a more

² The dependence to $\mathbf{q}$ is omitted for clarity reasons.

intuitive way, by regulating the priority matrix $\mathbf{A}_i$ that is more closely related to task directions than the activator Θ in (6).

First, look at the following matrix, which extends $\mathbf{N}'_j$ defined by (7) from the handling of one task direction to the handling of the directions of n_t tasks

$$\mathbf{N}''' = \mathbf{I} - \sum_{j=1}^{n_t} \alpha_{ij} \frac{\mathbf{J}_j^T}{\|\mathbf{J}_j\|} \frac{\mathbf{J}_j}{\|\mathbf{J}_j\|}, \quad (8)$$

where, without the loss of generality, each task dimension is supposed to be 1, and α_{ij} with $j = 1, 2, 3, \dots, n_t$ parameterizes the priority of each of the n_t tasks with respect to a certain task i . For any task k among the n_t tasks with $\alpha_{ik} = 1$, which means that task k is of the highest priority, the product of $\mathbf{N}'''$ with $\mathbf{J}_k$ leads to

$$\begin{aligned} \mathbf{J}_k \mathbf{N}''' &= \mathbf{J}_k - \mathbf{J}_k \frac{\mathbf{J}_j^T}{\|\mathbf{J}_j\|} \frac{\mathbf{J}_j}{\|\mathbf{J}_j\|} - \sum_{j \neq k} \mathbf{J}_k \alpha_{ij} \frac{\mathbf{J}_j^T}{\|\mathbf{J}_j\|} \frac{\mathbf{J}_j}{\|\mathbf{J}_j\|} \\ &= - \sum_{j \neq k} \mathbf{J}_k \alpha_{ij} \frac{\mathbf{J}_j^T}{\|\mathbf{J}_j\|} \frac{\mathbf{J}_j}{\|\mathbf{J}_j\|}. \end{aligned} \quad (9)$$

In (9), $\mathbf{J}_k \mathbf{N}''' = 0$ if $\mathbf{J}_k \mathbf{J}_j^T = 0$ for each $j \neq k$ and $\alpha_{ij} > 0$. This means that the highest priority of task k may not be satisfied if it is interfered by a lower priority task j , which has a component along task k direction. On the contrary, task priorities can be maintained if such task interferences disappear, or in other words, if all the lower priority task directions become orthogonal to all the higher priority task directions. Based on this observation, the computation of the generalized projector $\mathbf{P}_i$ is divided into three steps.

Step 1 is a preliminary processing of the matrices $\mathbf{J}$ and $\mathbf{A}_i$, where

$$\mathbf{J} = [\mathbf{J}_1^T \dots \mathbf{J}_j^T \dots \mathbf{J}_{n_t}^T]^T \quad (10)$$

is the augmented Jacobian concatenating the Jacobian matrices of all the n_t tasks. The processing of $\mathbf{J}$ and $\mathbf{A}_i$ is carried out according to the priorities of the n_t tasks with respect to task i . As each row of $\mathbf{J}$ is associated to α_{ij} , the rows of $\mathbf{J}$ can be sorted in descending order with respect to the values of the diagonal elements in $\mathbf{A}_i$. The resulting matrix $\mathbf{J}_{s_i}$ is thus constructed so that tasks which should be the least influenced by task i appear in its first rows, while tasks which can be the most influenced by task i appear in its last rows. The values in $\mathbf{A}_i$ are sorted accordingly, leading to $\mathbf{A}_i^s$, the diagonal elements of which are organized in descending order starting from the first row.

Step 2 consists in the computation of a matrix $\mathbf{B}_i(\mathbf{J}_{s_i}) \in \mathbb{R}^{r \times n}$ by using $\mathbf{J}_{s_i}$, where r is the rank of $\mathbf{J}_{s_i}$. The rows of

$\mathbf{B}_i(\mathbf{J}_{s_i})$ form an orthonormal basis of the joint space obtained using elementary row transformations on $\mathbf{J}_{s_i}$. Algorithm (1) describes this computation. As in any numerical scheme, tolerances are used here for numerical comparison, such as ϵ in line #11 of Algorithm (1), which is defined as a small positive value. As the use of ϵ may lead to rank jumps in $\mathbf{B}_i$, it is suggested to assign the smallest value greater than zero to ϵ to avoid large variation of $\mathbf{B}_i$.

Step 3 is to compute the generalized projector, which is given by

$$\mathbf{P}_i(\mathbf{A}_i) = \mathbf{I}_n - \mathbf{B}_i(\mathbf{J}_{s_i})^T \mathbf{A}_i^{r,s}(\mathbf{A}_i, \text{origin}) \mathbf{B}_i(\mathbf{J}_{s_i}), \quad (11)$$

where $\mathbf{A}_i^{r,s}$ is a diagonal matrix of degree r . The vector *origin* $\in \mathbb{R}^r$ is a vector of the row indexes of $\mathbf{J}_{s_i}$ selected during the construction of the orthonormal basis $\mathbf{B}_i$. Each of these r rows in $\mathbf{J}_{s_i}$ is linearly independent to all the previously selected ones. The diagonal elements of $\mathbf{A}_i^{r,s}$ are restricted to the r diagonal elements of $\mathbf{A}_i^s$, which correspond to the r rows of $\mathbf{J}_{s_i}$, the row indexes of which belong to *origin*. Algorithm (2) summarizes the construction of the generalized projector.

Note that the interference of lower priority tasks with higher priority tasks, which exists in (8) if two task directions of different priorities are not orthogonal ($\mathbf{J}_k \mathbf{J}_j^T \neq 0$), is avoided in $\mathbf{P}_i$. Indeed, each row in $\mathbf{B}_i$ corresponds to the component of a task direction that is effectively accounted for by the projector $\mathbf{P}_i$. The row sorting in step one ensures that higher priority task directions are accounted for in $\mathbf{B}_i$ prior to any lower priority task direction, and the orthonormalization process in step two ensures that each direction (or row) of $\mathbf{B}_i$ is orthogonal to previous rows associated to all the higher priority task directions.

By varying the value of each α_{ij} in $\mathbf{A}_i$, one can regulate the priority of each task j with respect to task i separately. Indeed, during the execution of task i , the projector $\mathbf{P}_i$ can be configured such that

- for tasks having strict priority over task i , the movement along their task directions is completely forbidden by setting corresponding $\alpha_{i\bullet}$ to 1;
- for tasks over which task i has a strict priority, the movement along their directions is completely allowed by setting corresponding $\alpha_{i\bullet}$ to 0;
- and for tasks with non strict priorities, the movement along their task directions is partially allowed according to the value of their priority parameters. The increase of the values of corresponding $\alpha_{i\bullet} \in (0, 1)$ leads to the increase of the priorities of the associated tasks with respect to task i , and thus stronger restriction of task i movements along their task directions.

Algorithm 1: Orthonormal basis computation - *GetOrthBasis(J)*

Data: $\mathbf{J}, \epsilon$
Result: $\mathbf{B}, \text{origin}, r$

```

1 begin
2    $n \leftarrow \text{GetNbCol}(\mathbf{J})$ 
3    $m \leftarrow \text{GetNbRow}(\mathbf{J})$ 
4    $i \leftarrow 0$ 
5   for  $k \leftarrow 0$  to  $m - 1$  do
6     if  $i \geq n$  then
7       break
8      $\mathbf{B}[i, :] \leftarrow \mathbf{J}[k, :]$ 
9     for  $j \leftarrow 0$  to  $i - 1$  do
10       $\mathbf{B}[i, :] \leftarrow \mathbf{B}[i, :] - (\mathbf{B}[i, :] \mathbf{B}[j, :]^T) \mathbf{B}[j, :]$ 
11    if  $\text{norm}(\mathbf{B}[i, :]) > \epsilon$  then
12       $\mathbf{B}[i, :] \leftarrow \mathbf{B}[i, :] / \text{norm}(\mathbf{B}[i, :])$ 
13       $\text{origin}[i] \leftarrow k$ 
14       $i \leftarrow i + 1$ 
15   $r \leftarrow i$ 
16  return  $\mathbf{B}, \text{origin}, r$ 
17 end
```

Algorithm 2: Generalized projector computation - task i

Data: $\mathbf{A}_i, \mathbf{J}$
Result: $\mathbf{P}_i$

```

1 begin
2    $n \leftarrow \text{GetNbCol}(\mathbf{J})$ 
3    $\text{index} \leftarrow \text{GetRowsIndexDescOrder}(\mathbf{A}_i)$ 
4    $\mathbf{A}_i^s \leftarrow \text{SortRows}(\mathbf{A}_i, \text{index})$ 
5    $\mathbf{J}_{s_i} \leftarrow \text{SortRows}(\mathbf{J}, \text{index})$ 
6    $\mathbf{B}_i, \text{origin}, r \leftarrow \text{GetOrthBasis}(\mathbf{J}_{s_i}) \triangleright \text{Alg. (1)}$ 
7    $\mathbf{A}_i^{r,s} \leftarrow \text{GetSubDiagMatrix}(\mathbf{A}_i^s, \text{origin})$ 
8    $\mathbf{P}_i \leftarrow \mathbf{I}_n - \mathbf{B}_i^T \mathbf{A}_i^{r,s} \mathbf{B}_i$ 
9  return  $\mathbf{P}_i$ 
10 end
```

4.2.1 Task insertion and deletion

There is a particular case induced by the proposed formulation and corresponding to the influence of task i on itself. Even though not intuitive, this self-influence has to be interpreted in terms of task existence, modulated by α_{ii} . If $\alpha_{ii} = 1$ then task i is projected into its own null-space, i.e. it is basically canceled out. Decreasing α_{ii} continuously to 0 activates task i gradually. Conversely, increasing α_{ii} continuously from 0 to 1 deactivates the task gradually.

5 Generalized hierarchical control framework

This paper handles task hierarchies subject to linear constraints. This multi-objective control problem is formulated as a Linear Quadratic Programming (LQP) problem here, where all the task objectives and constraints are solved simultaneously in one LQP. Constraints are formulated in terms of

priority consistent joint accelerations by applying generalized projectors.

This section first briefly reviews the LQP control framework that is commonly used by weighting strategies, then explains the implementation of generalized projectors in such a framework to achieve generalized hierarchical control.

5.1 LQP control framework for weighting strategies

When only non-strict task hierarchies are considered, weighting strategies, such as those proposed in Collette et al. (2007), Liu et al. (2011), Salini et al. (2011), Bouyarmane and Kheddar (2011), can be applied to handle the relative priorities of multiple elementary tasks. In this case, the control problem can be formulated as a LQP problem as

$$\arg \min_{\ddot{\mathbf{q}}, \chi} \sum_{i=1}^{n_t} \left\| \mathbf{f}_i(\ddot{\mathbf{q}}, \ddot{\xi}_i^d) \right\|_{\mathbf{Q}_i}^2 + \left\| \begin{bmatrix} \ddot{\mathbf{q}} \\ \chi \end{bmatrix} \right\|_{\mathbf{Q}_r}^2 \quad (12a)$$

$$\text{subject to} \quad \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{J}_c(\mathbf{q})^T \chi \quad (12b)$$

$$\mathbf{G}(\mathbf{q}, \dot{\mathbf{q}}) \begin{pmatrix} \ddot{\mathbf{q}} \\ \chi \end{pmatrix} \leq \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) \quad (12c)$$

where $\mathbf{f}_i(\ddot{\mathbf{q}}, \ddot{\xi}_i^d) = \mathbf{J}_i(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{J}}_i(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - \ddot{\xi}_i^d$ is the objective function which measures the error of task i . The regulation term minimizes the norm of joint accelerations and action variables. For a redundant robot with many solutions satisfying the same task objective, the regulation term is useful for ensuring the uniqueness of the solution (Salini et al. 2011). In (12), $\mathbf{Q}_i = \omega_i \mathbf{I}_{m_i}$ is a weighting matrix to regulate the importance level of task i , with ω_i denoting the weight of task objective i . $\mathbf{Q}_r = \omega_r \mathbf{I}_{n+n_a+3n_c}$ is the weighting matrix of the regularization term, with ω_r denoting the weight value. As the regulation term may increase task error, ω_r is usually very small compared to ω_i . The equation of motion (12b) constitutes an equality constraint to ensure physical realism. The matrix $\mathbf{G}$ and the vector $\mathbf{h}$ express some other equality or inequality constraints, such as actuation capabilities (maximum actuator torques and velocities), geometrical limits (joint limits, Cartesian space obstacles), and contact wrenches (contact existence conditions, bounds on the norms of contact wrenches).

By solving (12), the solution of $\ddot{\mathbf{q}}$ and χ can be obtained, from which the solution of joint torques is extracted.

5.2 Generalized hierarchical control using generalized projectors

The control framework based on weighting strategy (12) can qualitatively regulate the relative priorities of tasks by weighting task objectives, but it cannot ensure strict task priorities. The GHC framework proposed here extends framework (12) through the implementation of generalized

projectors defined by (11) to handle a priority network with both strict and non-strict task priorities.

Consider the control problem for solving n_t tasks. The operating principle of GHC is summarized by the following LQP problem, which takes into account the desired task priorities parameterized by the priority matrix $\mathbf{A}_i$.

$$\arg \min_{\dot{\mathbf{q}}', \chi} \sum_{i=1}^{n_t} \left\| \mathbf{f}_i \left(\ddot{\mathbf{q}}'_i, \ddot{\xi}_i^d \right) \right\|^2 + \left\| \begin{bmatrix} \ddot{\mathbf{q}}' \\ \chi \end{bmatrix} \right\|_{\mathbf{Q}_r}^2 \quad (13a)$$

$$\text{subject to } \mathbf{J}_c(\mathbf{q})^T \chi = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) \quad (13b)$$

$$\mathbf{G}(\mathbf{q}, \dot{\mathbf{q}}) \begin{pmatrix} \ddot{\mathbf{q}} \\ \chi \end{pmatrix} \leq \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) \quad (13c)$$

$$\ddot{\mathbf{q}} = \mathbf{P}\ddot{\mathbf{q}}' = \sum_{i=1}^{n_t} \mathbf{P}_i(\mathbf{A}_i)\ddot{\mathbf{q}}'_i \quad (13d)$$

with $\ddot{\mathbf{q}}' = \begin{bmatrix} \ddot{\mathbf{q}}'_1 \\ \vdots \\ \ddot{\mathbf{q}}'_{n_t} \end{bmatrix}$ and $\mathbf{P} = [\mathbf{P}_1(\mathbf{A}_1) \dots \mathbf{P}_{n_t}(\mathbf{A}_{n_t})]$. Each

$\ddot{\mathbf{q}}'_i$ in (13) is an intermediate joint acceleration variable associated to each task i and $\ddot{\mathbf{q}}$ is the overall joint accelerations accounting for the sets of desired task priorities $(\mathbf{A}_1, \dots, \mathbf{A}_{n_t})$. Assuming a perfect model, $\ddot{\mathbf{q}}$ is the joint accelerations resulting from the application of the joint torques computed by solving (13).

This optimization problem minimizes the objective function of each task as well as the magnitude of the control input, subject to constraints. Each task objective function is expressed in terms of the intermediate joint acceleration variable $\ddot{\mathbf{q}}'_i$. Note that in GHC, task priorities are handled by using the generalized projectors $\mathbf{P}_i$ in (13d) instead of task weights ω_i . Therefore, here the task weighting matrix $\mathbf{Q}_i$ is set to the identity matrix, which is omitted in (13a).

A solution to the equation of motion (13b) can be ensured as long as there exists a highest priority task i such that $\mathbf{P}_i(\mathbf{A}_i) = \mathbf{I}_n$ (with $\mathbf{A}_i$ being the zero matrix), which means that this task is not projected in the null-space of any other task. Indeed, (13b) can be expressed in terms of intermediate joint accelerations as

$$\mathbf{J}_c(\mathbf{q})^T \chi = \mathbf{M}(\mathbf{q})\mathbf{P}\ddot{\mathbf{q}}' + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}), \quad (14)$$

with $\mathbf{P} = [\mathbf{P}_1(\mathbf{A}_1) \dots \mathbf{P}_{n_t}(\mathbf{A}_{n_t})]$. As the inertia matrix $\mathbf{M}$ is positive definite, a solution to (14), and thus (13b), can be ensured if $\mathbf{P}$ has full row rank. A sufficient condition to ensure this property of $\mathbf{P}$ is that there exists at least one $\mathbf{P}_i$ which equals the identity matrix, and this is the case for the highest priority task in a hierarchy.

Since the constraints have a higher priority than the objectives in LQP, and in (13) the constraints are expressed in terms of the overall joint accelerations $\ddot{\mathbf{q}}$, it is ensured that the solution accounting for desired task hierarchies satisfies the constraints. Or in other words, the GHC framework ensures

that the constraints, such as the equation of motion (13b) and the other constraints of physical limitations (13c), have a higher priority over task hierarchies. Moreover, this GHC framework can handle strict task hierarchies represented by standard lexicographic orders. The proof is provided in Appendix.

Another property of GHC is that it is robust to both kinematic and algorithmic singularities. In this framework based on LQP, tasks are expressed in a forward way and most LQP solvers do not require the explicit inversion of Jacobian matrices. Therefore, GHC does not have problems of numerical singularities due to kinematic singularities. Moreover, unlike approaches using the pseudo-inverse of limited Jacobians $(\mathbf{J}_i \mathbf{N}_j)$, which requires special treatment for handling algorithmic singularities when the limited Jacobians drop rank (Sadeghian et al. 2013), GHC does not require the inversion of priority consistent Jacobians. Therefore, the framework does not have to handle such kind of algorithmic singularities.

6 Results

The proposed GHC framework (13) is applied to the control of a 7-DoF KUKA LWR robot. The experiments are conducted in the Arboris-Python simulator (Salini 2013), which is a rigid multibody dynamics and contacts simulator written in Python. The LQP problem is solved by a QP solver included in CasADi-Python (Andersson et al. 2012), which is a symbolic framework for dynamic optimization.

In the experiments, three tasks are defined: task 1 for the control of the three dimensional position (or position and force) of the end-effector, task 2 for the control of the three dimensional position of the elbow, and task 3 for the control of the 7-DoF posture. The elbow task target is a static target position and the posture task target is a static posture. For each task i , an optimization variable $\ddot{\mathbf{q}}'_i \in \mathbb{R}^7$ is defined. A local proportional-derivative controller $\ddot{\xi}_i^d$ is used to ensure the convergence of each task variable towards its target. When a task target is static, $\ddot{\xi}_i^d = k_p \mathbf{e}_i + k_d \dot{\mathbf{e}}_i$ with $k_p = 30s^{-2}$ and $k_d = 20s^{-1}$. When tracking a desired trajectory $\ddot{\xi}_i^*$, $\ddot{\xi}_i^d = \ddot{\xi}_i^* + k_p \mathbf{e}_i + k_d \dot{\mathbf{e}}_i$ with $k_p = 100s^{-2}$ and $k_d = 20s^{-1}$. The priority parameter matrices associated with the three tasks are: $\mathbf{A}_i = \text{diag}(\alpha_{i1}\mathbf{I}_3, \alpha_{i2}\mathbf{I}_3, \alpha_{i3}\mathbf{I}_7)$ with $i = 1, 2, 3$. The regularization weight ω_r is chosen as 0.01. The following function is used for the smooth variation of α_{ij} (conversely α_{ji}) from 0 to 1 during the transition time period $([t_1, t_2])$

$$\alpha_{ij}(t) = 0.5 - 0.5 \cos \left(\frac{t - t_1}{t_2 - t_1} \pi \right), \quad t \in [t_1, t_2], \quad (15)$$

$$\alpha_{ji}(t) = 1 - \alpha_{ij}(t).$$

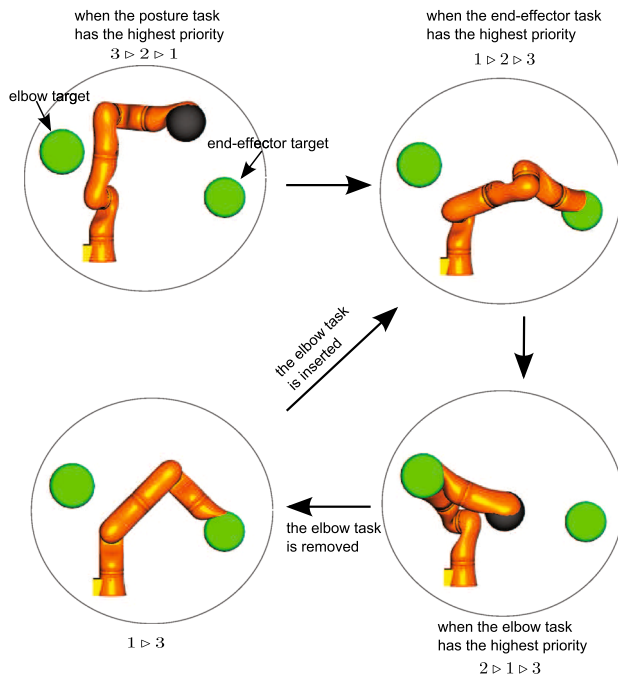


Fig. 1 Experiment of priority switching

6.1 Hierarchical control with priority transitions, task insertions, and task deletions

In the experiments, task hierarchy is changed four times (see Fig. 1), and the equality constraint (13b) as well as inequality constraints, such as joint velocity and joint torque limits, are imposed. The evolution of the task hierarchy is $3 \triangleright 2 \triangleright 1 \Rightarrow 1 \triangleright 2 \triangleright 3 \Rightarrow 2 \triangleright 1 \triangleright 3 \Rightarrow 1 \triangleright 3 \Rightarrow 1 \triangleright 2 \triangleright 3$. In the beginning, the tasks, in the priority level decreasing order, are the posture task, the elbow task, and the end-effector task. Then the end-effector task priority increases and the posture task priority decreases simultaneously. Afterward, the priorities of the end-effector task and the elbow task are switched. Then the elbow task is removed. Finally, the elbow task is inserted with its priority level between those of the end-effector task and the posture task.

The experiment is carried out first using static task targets for steady state error analysis, then using a dynamic end-effector trajectory of a lemniscate shape. Moreover, the performance of GHC is compared with the HQP approach (Kanoun et al. 2011).

The results corresponding to the use of static task targets are presented in Figs. 2, 3, 4, and 5. Task errors by using HQP (Fig. 2) as well as those by using GHC with different hierarchy rearrangement durations (Figs. 3 and 4) are shown. The hierarchy rearrangement duration is 0.005 s in Fig. 3 and 2 s in Fig. 4. Figure 5 shows the integration of the absolute values of the resulting joint jerks $\sum_{i=1}^n \left(\int_0^t \left| \frac{d^3 q_i}{dt^3} \right| dt \right)$ by using HQP that performs instantaneous hierarchy rearrangements,

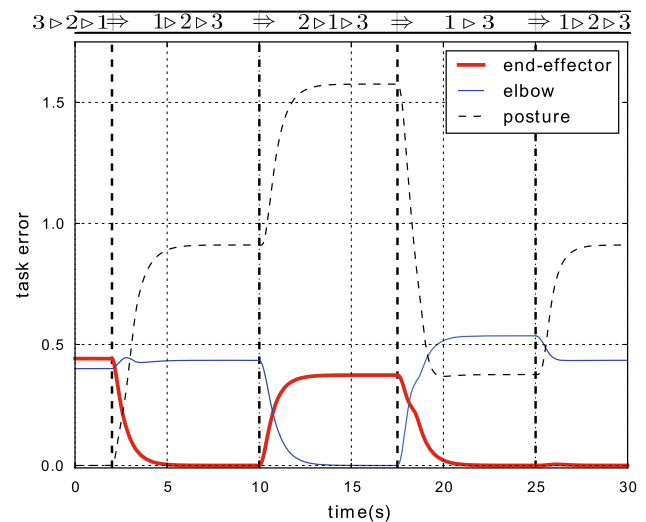


Fig. 2 Task errors using HQP, with fixed task targets. Priority transitions as well as the insertion and deletion of the elbow task are performed. The hierarchy rearrangement is instantaneous

as well as by using GHC with faster and slower hierarchy rearrangements. Steady state task errors for each task hierarchy configuration are shown in Table 1, where the results using GHC and HQP are included.

GHC provides similar results in terms of task errors compared with HQP, as can be observed in Figs. 2 and 3. The results of task errors in Table 1 show that both GHC and HQP can ensure strict priority. When controlled by either GHC or HQP, errors of the tasks with the highest priority are very small. Moreover, GHC can perform slower and smoother hierarchy rearrangements that require less joint jerks. This can be seen in Fig. 5, which shows that GHC can generate smaller joint jerks than HQP does.

When a lemniscate-shaped end-effector trajectory is used, the end-effector task is to move along this lemniscate orbit periodically, with an orbital period of 2π s. The desired and the resulting end-effector trajectory is shown in Fig. 6. The resulting task errors using GHC is presented in Fig. 7. The resulting joint velocities and joint torques are shown in Fig. 8. A video of this experiment that presents the main features of GHC (priority transitions, the insertion and deletion of tasks) is attached to this paper.

Figure 7 shows that when the end-effector task has the highest priority, it can track its desired trajectory precisely. Moreover, Fig. 8 shows that joint velocity and joint torque limits are respected, which demonstrate that GHC can maintain desired task hierarchies while satisfying constraints.

6.2 Hierarchical control with a force task

In this experiment, the end-effector task is to move towards a plane, and then to apply a desired contact force against

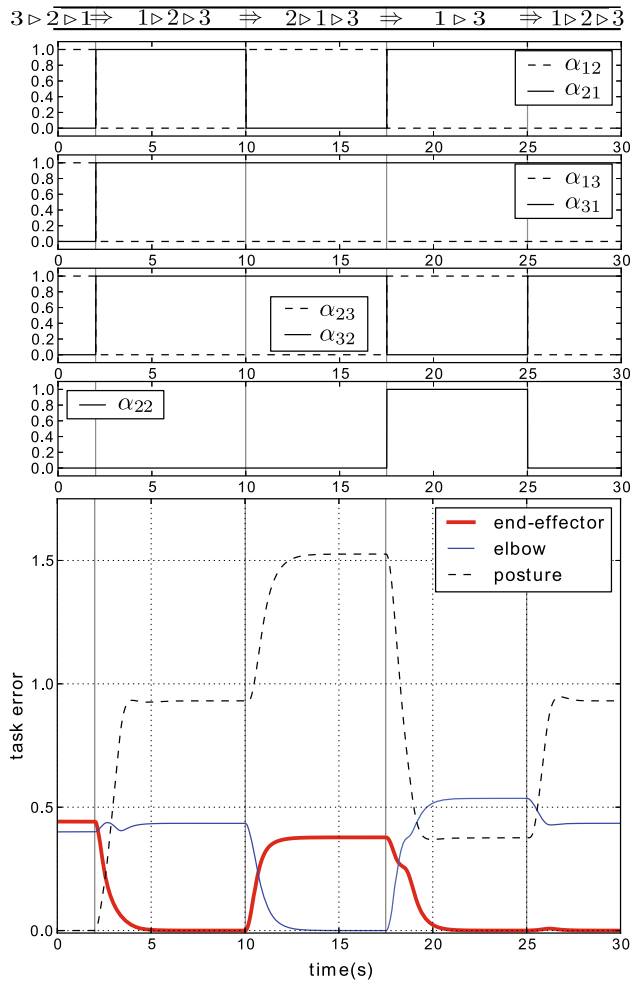


Fig. 3 Evolution of α s (top) and task errors (bottom) using GHC, with fixed task targets. Priority transitions as well as the insertion and deletion of the elbow task are performed. The hierarchy rearrangement duration is 0.005 s

the plane in the vertical direction (see Fig. 9). Before the establishment of the contact with the plane, the end-effector task (task 1) is a motion task. Once the contact is established, the end-effector task is a composition of a position subtask in the horizontal plane (task 1a) and a force subtask in the vertical direction (task 1b). The force task is transformed into a motion task by applying (3). The evolution of task hierarchy is $2 \triangleright 1 \triangleright 3 \Rightarrow 1 \triangleright 2 \triangleright 3$. At the beginning of this experiment, the elbow task has the highest priority, and the initial position of the end-effector is above its target position but pointing upwards. Then the priorities between the elbow task and the end-effector task switches. The increase of the priority level of the end-effector task allows the end-effector to point downwards, move towards its target position on the surface of the plane, and then push against the plane.

The change of α s, the errors of the elbow position and the end-effector horizontal position, as well as the error of the end-effector contact force are shown in Fig. 10. After the

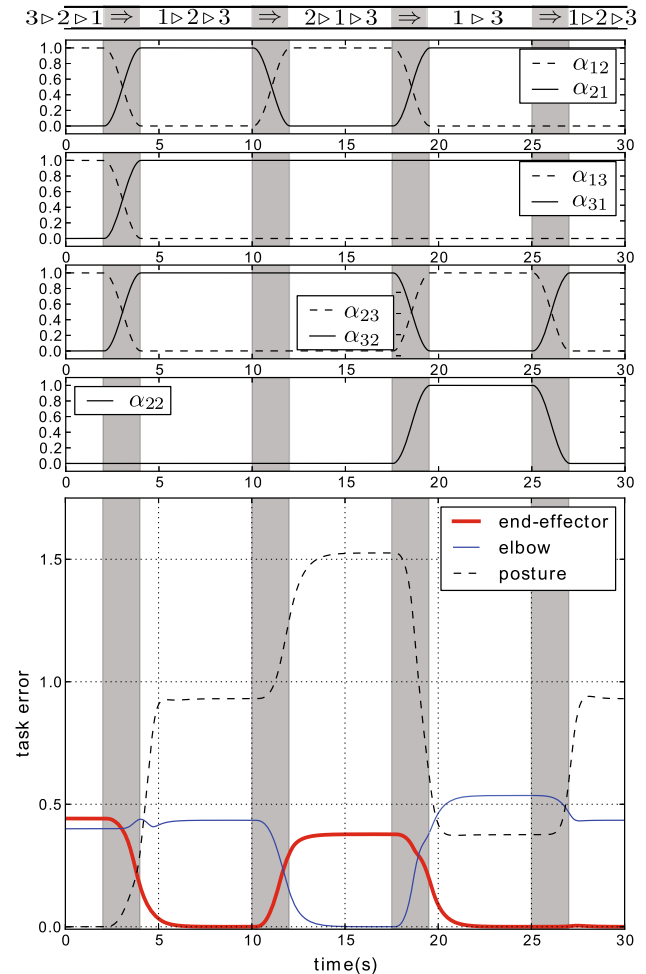


Fig. 4 Evolution of α s (top) and task errors (bottom) using GHC, with fixed task targets. Priority transitions as well as the insertion and deletion of the elbow task are performed. The hierarchy rearrangement duration is 2 s

priority switch, the highest priority of the end-effector task allows the controlled frame to achieve its target position and to follow its contact force reference after the contact establishment, except for the impact peak at the moment when the contact is established between the two rigid bodies. This result illustrates the fact that the highest priority of the end-effector task, including both the horizontal position control component and the vertical force control component, is maintained.

6.3 Control with a non-strict hierarchy

In the previous experiments, non-strict priorities are used only in the transition phase. Experiments here handle constant non-strict priorities by using GHC. The elbow task and the end-effector task are considered, with α_{12} being set to a value between 0 and 1. Figure 11 shows the task errors with respect to different values of α_{12} . It can be seen in this figure

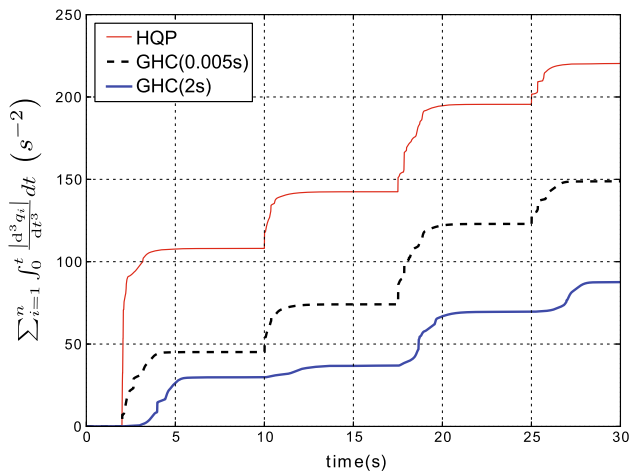


Fig. 5 Integration of the absolute values of joint jerks using GHC and HQP, with fixed task targets. The value is increased each time the task hierarchy is changed. GHC generates less amount of joint jerks by performing slower hierarchy rearrangements; while HQP, which perform instantaneous hierarchy rearrangements generates larger joint jerks

Table 1 Steady state task errors for each task hierarchy configuration

Priority	3 ▷ 2 ▷ 1		
Task	1	2	3
GHC	0.46	0.40	2.2e−30
HQP	0.46	0.40	2.8e−10
Priority	1 ▷ 2 ▷ 3		
Task	1	2	3
GHC	1.0e−6	0.46	1.8
HQP	4.5e−7	0.46	1.8
Priority	2 ▷ 1 ▷ 3		
Task	1	2	3
GHC	0.42	2.6e−6	3.0
HQP	0.42	2.7e−6	3.1
Priority	1 ▷ 3		
Task	1	2	3
GHC	3.9e−6	0.55	0.79
HQP	4.5e−6	0.55	0.79

that by setting α_{12} to certain values between 0 and 1, no task is completely satisfied and non-strict hierarchies are achieved. When $\alpha_{12} = 0.5$, the errors of the two tasks show more or less an equal compromise between them. When α_{12} is increased, the performance of the end-effector task is reduced in order to improve the performance of the elbow task. In fact, the increase of α_{12} corresponds to the increase of the elbow task weight with respect to that of the end-effector when using a weighting strategy.

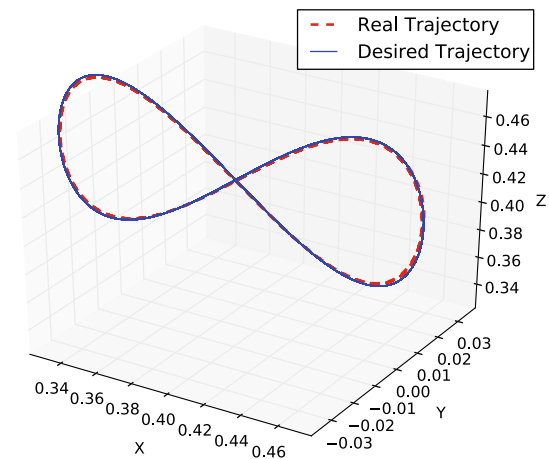


Fig. 6 The desired and the resulting end-effector trajectory provided by GHC, when the end-effector task has the highest priority. The end-effector moves along the lemniscate-shaped trajectory with an orbital period of 2π s

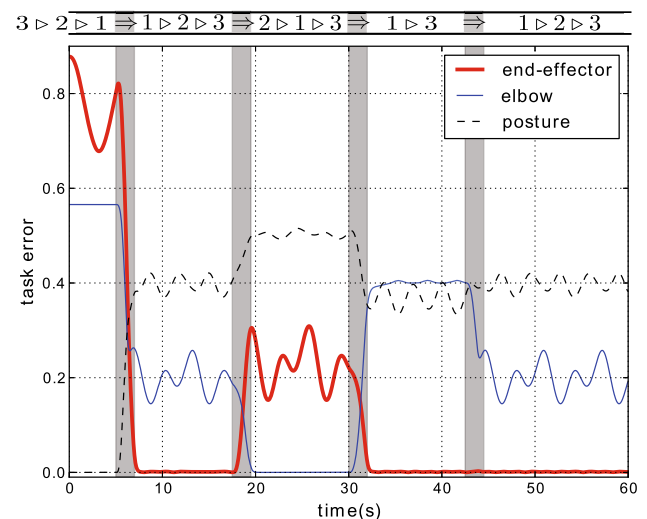


Fig. 7 Task errors using GHC, with the end-effector tracking a lemniscate-shaped trajectory. Desired priority transitions as well as the insertion and deletion of the elbow task are achieved. Strict priorities are maintained

7 Discussion

In this section, the computation cost of GHC is analyzed, which shows that the computation time tends to increase with the number of DoF of the robot and the number of tasks. For a robot of n DoF performing k tasks of different priority levels with a total task dimension of m , the computation cost by using the HQP solver (Escande et al. 2013) is dominated by the hierarchical complete orthogonal decomposition, whose cost is equivalent to $n^2m + nm^2 + \sum_{i=1}^k (m_i - r_i)m_i^2$, with m_i and r_i being respectively the task dimension and the rank of task Jacobian in the i th hierarchy. By using the GHC strategy, the magnitude order of optimization variables is kn , since an

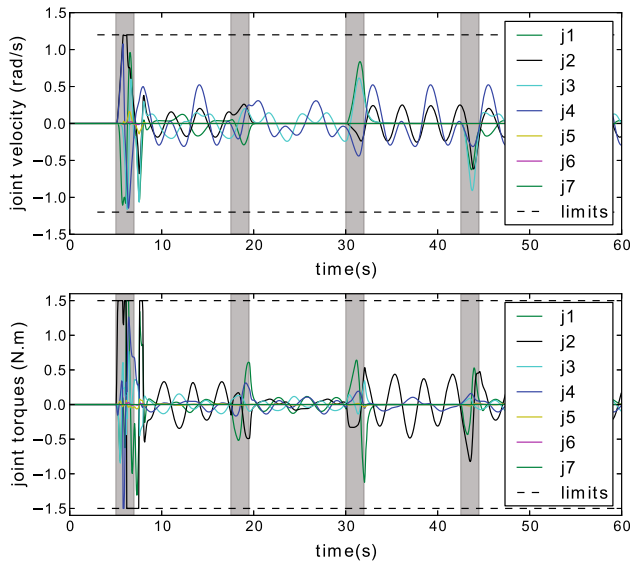


Fig. 8 Evolution of joint velocities and joint torques. The upper and lower bounds of $\dot{\mathbf{q}}$ are 1.2 and -1.2 rad/s, respectively. The upper and lower bounds of $\boldsymbol{\tau}$ are 1.5 and -1.5 Nm, respectively. These bounds are voluntarily set low in order to easily illustrate the fact that they are respected



Fig. 9 The target end-effector position is on the plane. The end-effector starts from an initial position, which is above the target position and pointing upwards, then it should move towards the target position, and then apply desired forces to the plane

intermediate joint acceleration variable $\ddot{\mathbf{q}}'_i \in \mathbb{R}^n$ is created for each task i . In this case, one level of QP (13) needs to be solved, so the computation cost is in $O((kn)^2m + knm^2 + (m - r)m^2)$, with r being the rank of the augmented task Jacobian.

The computational cost of the current GHC strategy is sensitive to the number of DoF of the robot and the number of tasks. For a fixed-based KUKA robot with 7 DoFs performing n_1 motion tasks of different priority levels, a set of intermediate joint acceleration variables $\ddot{\mathbf{q}}' \in \mathbb{R}^{7n_1}$ and the joint torques $\boldsymbol{\tau} \in \mathbb{R}^7$ needs to be solved for. For a fixed-based humanoid robot iCub with 32 DoF performing n_2 tasks, the number of variables would be $32(n_2 + 1)$. Figure 12 shows the computation time of using GHC to solve randomly selected hierarchical control problems for the KUKA robot and the

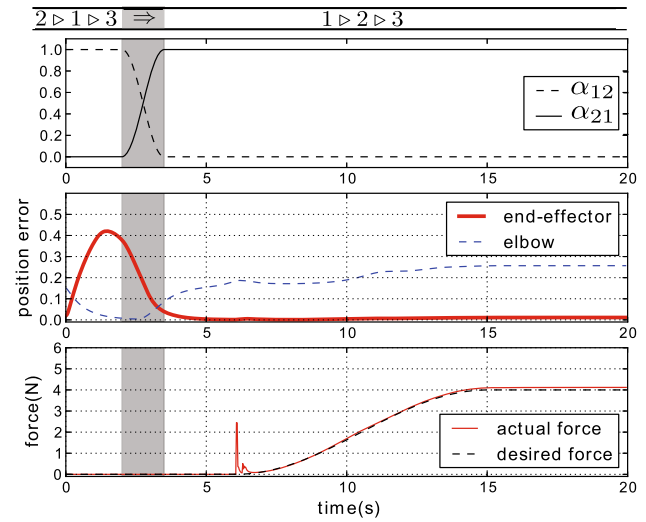


Fig. 10 Results of contact force control. The *top figure* shows the change of α s. The *figure in the middle* shows the end-effector position error in the horizontal plane as well as the elbow position error. The *bottom figure* represents the desired and resulting contact forces between the end-effector and the plane

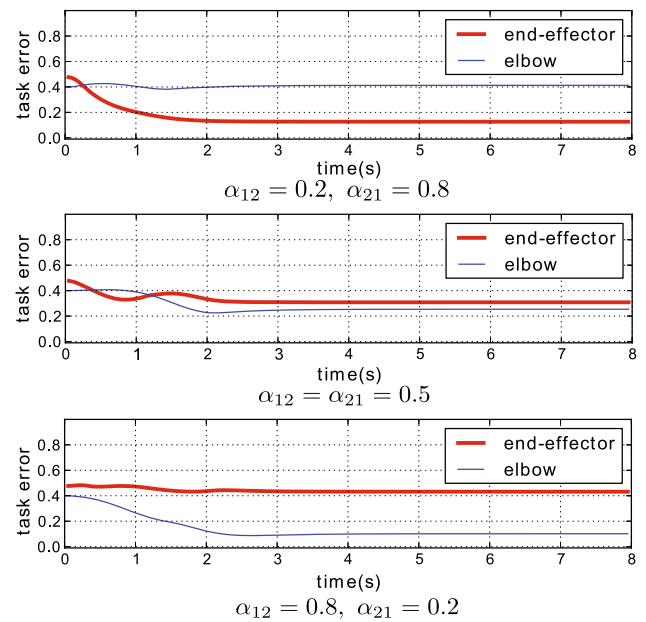


Fig. 11 Errors of the elbow task and the end-effector task with respect to their different relative priorities. Three priority values are applied: $\alpha_{12} = 0.2$, $\alpha_{12} = 0.5$, and $\alpha_{12} = 0.8$

iCub robot performing different numbers of tasks. Each control problem consists of the constraint (13b), a posture task with random joint goal positions, and a set of 3-dimensional Cartesian motion tasks with random goal positions. For the KUKA robot performing totally 5 tasks, the mean computation time per iteration is 2.7 ms; for the iCub robot performing the same number of tasks, the mean computation time is 88

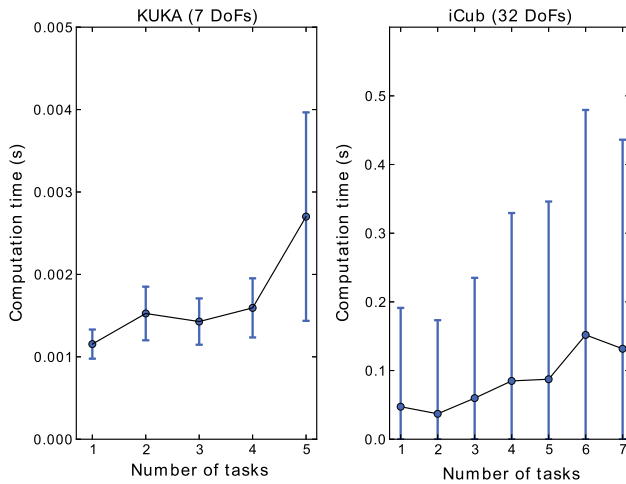


Fig. 12 Mean and standard deviation of the computation time per iteration, when using GHC to solve randomly selected hierarchical control problems for a fixed-based KUKA robot and a fixed-based iCub robot. Each control problem consists of a posture task and a set of 3D Cartesian motion tasks (0–4 motion tasks for KUKA and 0–6 motion tasks for iCub), subject to the whole-body equilibrium constraint (13b)

ms. These results correspond to a C++ implementation of the controller on a standard Linux PC.

8 Conclusions and future works

This paper proposes a generalized hierarchical control approach for handling tasks with both strict and non strict priorities. A generalized projector is developed. It can precisely regulate how much a task can influence or be influenced by other tasks through the modulation of a priority matrix: a task can be completely, partially, or not at all projected into the null-space of other tasks. Multiple simultaneous changes of task priorities can be achieved by using this generalized projector and, using the same mechanism, tasks can be easily inserted or deleted. Moreover, the GHC approach can maintain and switch task priorities while respecting a set of equality and inequality constraints.

In this work, the GHC approach is illustrated at the dynamic level; however, the generalized projector introduced here is not restricted to this case. In fact, it can also be used in other types of controllers, such as a velocity kinematics controller or a quasi-static controller. The idea is to associate each task with an intermediate task variable in joint space ($\dot{q}_i'$, $\ddot{q}_i'$, τ_i' , etc.), then to apply generalized projectors to these task variables, and finally the global joint space variable is the sum of each projected task variables ($\mathbf{P}_i(\mathbf{A}_i)\dot{q}_i'$, $\mathbf{P}_i(\mathbf{A}_i)\ddot{q}_i'$, $\mathbf{P}_i(\mathbf{A}_i)\tau_i'$, etc.).

Immediate future work includes the reduction of the computational cost of GHC to achieve real-time control of complex robots with a high number of DoF. Moreover, the use of robot learning techniques to incrementally learn and improve the tuning of the relative influence of each task with respect to others is of great interest.

Acknowledgments We would like to thank the reviewers for their insightful comments on the paper. This work was partially supported by the European Commission, within the CoDyCo project (FP7-ICT-2011-9, No. 600716) and by the RTE company through the RTE/UPMC chair Robotics Systems for field intervention in constrained environments held by Vincent Padois.

Proof of the maintenance of strict hierarchies represented by standard lexicographic orders subject to constraints

This section proves that the proposed GHC approach (13) can maintain strict task hierarchies represented by standard lexicographic orders while accounting for linear constraints.

Suppose there are n_t tasks that should be organized in a way such that each task i has a strict lower priority than task $i - 1$ with $i = 2, \dots, n_t$. In this case, the generalized projector $\mathbf{P}_i$ for a task i is in fact a null-space projector, which projects a task Jacobian into the null-space of all the previous $i - 1$ tasks, and each $\mathbf{A}_i$ is an identity matrix. Let each task objective function be $f_i = \mathbf{J}_i \mathbf{x}_i' - \mathbf{x}_i^d$, with $\mathbf{x}_i'$ being a joint space task variable, such as $\dot{q}_i'$, $\ddot{q}_i'$, or τ_i' , etc. Moreover, the global variable $\mathbf{x} = \sum_i \mathbf{P}_i \mathbf{x}_i'$ should satisfy linear equality or inequality constraints $\mathbf{G}\mathbf{x} \leq \mathbf{h}$.

At the first stage, the regulation term is neglected, and the optimization problem can be written as follows

$$\arg \min_{\mathbf{x}'_{(n_t)}} \sum_{i=1}^{n_t} \left\| \mathbf{J}_i \mathbf{x}_i' - \mathbf{x}_i^d \right\|^2 \quad (16)$$

$$\text{subject to } \mathbf{G} \sum_{i=1}^{n_t} \mathbf{P}_i \mathbf{x}_i' \leq \mathbf{h}$$

where $\mathbf{x}'_{(n_t)} = \{\mathbf{x}_1', \mathbf{x}_2', \dots, \mathbf{x}_{n_t}'\}$, and the solution to (16) is denoted as $\mathbf{x}^*_{(n_t)} = \{\mathbf{x}_1^*, \mathbf{x}_2^*, \dots, \mathbf{x}_{n_t}^*\}$.

When $n_t = 1$, the optimization problem can be written as

$$\arg \min_{\mathbf{x}'_{(1)}} \left\| \mathbf{J}_1 \mathbf{x}'_{(1)} - \mathbf{x}_1^d \right\|^2 \quad (17)$$

$$\text{subject to } \mathbf{G}\mathbf{x}'_{(1)} \leq \mathbf{h}.$$

The solution to this problem $\mathbf{x}^*_{(1)}$ is the same as the one to the problem formulated by HQP.

When $n_t = k$, the optimization problem is formulated as

$$\begin{aligned} & \arg \min_{\mathbf{x}'_{(k)}} \sum_{i=1}^k \left\| \mathbf{J}_i \mathbf{x}'_i - \mathbf{x}_i^d \right\|^2 \\ & \text{subject to } \mathbf{G} \sum_{i=1}^k \mathbf{P}_i \mathbf{x}'_i \leq \mathbf{h}. \end{aligned} \quad (18)$$

Suppose the solution $\mathbf{x}_{(k)}^*$ can maintain the strict task hierarchy: if a task $k+1$ is inserted with lowest priority with respect to the set of k tasks, then the optimization problem with the $k+1$ tasks can be written as

$$\begin{aligned} & \arg \min_{\mathbf{x}'_{(k+1)}} \sum_{i=1}^k \left\| \mathbf{J}_i \mathbf{x}'_i - \mathbf{x}_i^d \right\|^2 + \left\| \mathbf{J}_{k+1} \mathbf{x}'_{k+1} - \mathbf{x}_{k+1}^d \right\|^2 \\ & \text{subject to } \mathbf{G} \left(\sum_{i=1}^k \mathbf{P}_i \mathbf{x}'_i + \mathbf{P}_{k+1} \mathbf{x}'_{k+1} \right) \leq \mathbf{h}. \end{aligned} \quad (19)$$

As $\mathbf{P}_k \mathbf{P}_{k+1} = \mathbf{P}_{k+1}$, the term $\sum_{i=1}^k \mathbf{P}_i \mathbf{x}'_i + \mathbf{P}_{k+1} \mathbf{x}'_{k+1}$ in the constraint in (19) is equivalent to $\sum_{i=1}^{k-1} \mathbf{P}_i \mathbf{x}'_i + \mathbf{P}_k \varsigma_k$, with

$$\varsigma_k = \mathbf{x}'_k + \mathbf{P}_{k+1} \mathbf{x}'_{k+1}. \quad (20)$$

Then problem (19) can be written as

$$\begin{aligned} & \arg \min_{\mathbf{x}'_{(k)}, \varsigma_k, \mathbf{x}_{k+1}} \sum_{i=1}^{k-1} \left\| \mathbf{J}_i \mathbf{x}'_i - \mathbf{x}_i^d \right\|^2 + \left\| \mathbf{J}_k \varsigma_k - \mathbf{x}_k^d \right\|^2 \\ & \quad + \left\| \mathbf{J}_{k+1} \mathbf{x}'_{k+1} - \mathbf{x}_{k+1}^d \right\|^2 \\ & \text{subject to } \mathbf{G} \left(\sum_{i=1}^{k-1} \mathbf{P}_i \mathbf{x}'_i + \mathbf{P}_k \varsigma_k \right) \leq \mathbf{h} \\ & \quad \varsigma_k = \mathbf{x}'_k + \mathbf{P}_{k+1} \mathbf{x}'_{k+1}. \end{aligned} \quad (21)$$

$\mathbf{x}'_k$ in (21) is a free variable, and this problem can be separated into two sub-problems. The first sub-problem is

$$\begin{aligned} & \arg \min_{\mathbf{x}'_{(k-1)}, \varsigma_k} \sum_{i=1}^{k-1} \left\| \mathbf{J}_i \mathbf{x}'_i - \mathbf{x}_i^d \right\|^2 + \left\| \mathbf{J}_k \varsigma_k - \mathbf{x}_k^d \right\|^2 \\ & \text{subject to } \mathbf{G} \left(\sum_{i=1}^{k-1} \mathbf{P}_i \mathbf{x}'_i + \mathbf{P}_k \varsigma_k \right) \leq \mathbf{h}. \end{aligned} \quad (22)$$

The optimal solution $\sum_{i=1}^{k-1} \mathbf{P}_i \mathbf{x}_i^{*,'} + \mathbf{P}_k \varsigma_k^*$ to this problem is equivalent to the one of (18). Indeed, these two solutions have the same effect on task k

$$\mathbf{J}_k \sum_{i=1}^k \mathbf{P}_i \mathbf{x}_i^{*,'} = \mathbf{J}_k \left(\sum_{i=1}^{k-1} \mathbf{P}_i \mathbf{x}_i^{*,'} + \mathbf{P}_k \varsigma_k^* \right). \quad (23)$$

To prove (23), one needs to notice that $\mathbf{J}_i \mathbf{P}_j = \mathbf{0}$ with $j \geq i$. The second sub-problem is given by

$$\arg \min_{\mathbf{x}_{k+1}} \left\| \mathbf{J}_{k+1} \mathbf{x}'_{k+1} - \mathbf{x}_{k+1}^d \right\|^2. \quad (24)$$

Therefore, the insertion of a lower priority task $k+1$ does not change the optima of the k previous task objectives. In other words, the strict task hierarchy of an arbitrary number of tasks subject to linear constraints can be maintained.

We have proved that each lower priority task will not increase the obtained optima of all the previous tasks. The rest of this proof explains the roles of the regulation term. As mentioned in Sect. 5, the use of a regulation term, which minimizes the norm of each task variable, helps to ensure the uniqueness of the solution. As each task objective i is assigned with the weight $\omega_i = 1$, which is much greater than the weight of the regulation term ($\omega_r \ll 1$), the task variables are optimized to mainly satisfy task objectives. Moreover, in GHC, this regulation term also helps to improve the performance of lower priority tasks. Consider $k+1$ levels of tasks to handle, as $\mathbf{J}_i \mathbf{P}_j = \mathbf{0}$ with $j \geq i$, the final solution is $\sum_{i=1}^k \mathbf{P}_i \mathbf{x}_i^* + \mathbf{P}_{k+1} \mathbf{x}_{k+1}^*$. Denoting the elements required by task i as $\mathbf{x}_i^{i,*}$ and the rest elements that are not effectively handled by task objective i as $\mathbf{x}_i^{f,*}$, the final solution can be rewritten as $S = \sum_{i=1}^k \mathbf{P}_i^i \mathbf{x}_i^{i,*} + \sum_{i=1}^k \mathbf{P}_i^f \mathbf{x}_i^{f,*} + \mathbf{P}_{k+1} \mathbf{x}_{k+1}^*$, with $\mathbf{P}_i^i$ and $\mathbf{P}_i^f$ the columns in $\mathbf{P}_i$ that correspond to $\mathbf{x}_i^{i,*}$ and $\mathbf{x}_i^{f,*}$ respectively. The term $\sum_{i=1}^k \mathbf{P}_i^f \mathbf{x}_i^{f,*}$ that is not required by the k previous tasks may contribute to task $k+1$ and affect its task performance. The minimization of the norm of $\mathbf{x}_i^f$ in the regulation term improves the performance of task $k+1$ by making S closer to $\sum_{i=1}^k \mathbf{P}_i^i \mathbf{x}_i^{i,*} + \mathbf{P}_{k+1} \mathbf{x}_{k+1}^*$, where $\mathbf{P}_i^i \mathbf{x}_i^{i,*}$ are used to perform the k previous tasks and $\mathbf{P}_{k+1} \mathbf{x}_{k+1}^*$ is used to perform the $(k+1)$ th task in the null-space of all the higher priority tasks.

References

- Abe, Y., da Silva, M., & Popović, J. (2007). Multiobjective control with frictional contacts. In *Proceedings of the ACM SIGGRAPH/eurographics symposium on computer animation* (pp. 249–258).
- Aghili, F. (2005). A unified approach for inverse and direct dynamics of constrained multibody systems based on linear projection opera-

- tor: Applications to control and simulation. *IEEE Transactions on Robotics*, 21(5), 834–849. doi:[10.1109/TRO.2005.851380](https://doi.org/10.1109/TRO.2005.851380).
- Andersson, J., Kesson, J., & Diehl, M. (2012). Casadi—A symbolic package for automatic differentiation and optimal control. In *Recent advances in algorithmic differentiation*.
- Baerlocher, P., & Boulic, R. (1998). Task-priority formulations for the kinematic control of highly redundant articulated structures. In *IEEE/RSJ international conference on intelligent robots and systems* (Vol. 1, pp. 323–329). doi:[10.1109/IROS.1998.724639](https://doi.org/10.1109/IROS.1998.724639).
- Bouyarmane, K., & Kheddar, A. (2011). Using a multi-objective controller to synthesize simulated humanoid robot motion with changing contact configurations. In *IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 4414–4419). doi:[10.1109/IROS.2011.6094483](https://doi.org/10.1109/IROS.2011.6094483).
- Chang, K. S., & Khatib, O. (1999). Efficient algorithm for extended operational space inertia matrix. In *IEEE/RSJ international conference on intelligent robots and systems, IEEE* (Vol. 1, pp. 350–355).
- Collette, C., Micaelli, A., Andriot, C., & Lemerle, P. (2007). Dynamic balance control of humanoids for multiple grasps and non coplanar frictional contacts. In *7th IEEE-RAS International Conference on Humanoid Robots* (pp. 81–88).
- Dietrich, A., Albu-Schaffer, A., & Hirzinger, G. (2012). On continuous null space projections for torque-based, hierarchical, multi-objective manipulation. In *IEEE international conference on robotics and automation (ICRA)* (pp. 2978–2985). doi:[10.1109/ICRA.2012.6224571](https://doi.org/10.1109/ICRA.2012.6224571).
- Egeland, O., Sagli, J., Spangelo, I., & Chiaverini, S. (1991). A damped least-squares solution to redundancy resolution. In *IEEE international conference on robotics and automation* (Vol. 1, pp. 945–950). doi:[10.1109/ROBOT.1991.131710](https://doi.org/10.1109/ROBOT.1991.131710).
- Escande, A., Mansard, N., & Wieber, P. B. (2013). Hierarchical quadratic programming: Companion report. Tech. Rep. <http://hal.archives-ouvertes.fr/hal-00970816>.
- Escande, A., Mansard, N., & Wieber, P. B. (2014). Hierarchical quadratic programming: Fast online humanoid-robot motion generation. *The International Journal of Robotics Research*. doi:[10.1177/0278364914521306](https://doi.org/10.1177/0278364914521306).
- Flacco, F., De Luca, A., & Khatib, O. (2012a). Motion control of redundant robots under joint constraints: Saturation in the null space. In *IEEE International Conference on Robotics and Automation (ICRA)* (pp. 285–292). doi:[10.1109/ICRA.2012.6225376](https://doi.org/10.1109/ICRA.2012.6225376).
- Flacco, F., De Luca, A., & Khatib, O. (2012b). Prioritized multi-task motion control of redundant robots under hard joint constraints. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 3970–3977). doi:[10.1109/IROS.2012.6385619](https://doi.org/10.1109/IROS.2012.6385619).
- Hsu, P., Mauser, J., & Sastry, S. (1989). Dynamic control of redundant manipulators. *Journal of Robotic Systems*, 6(2), 133–148. doi:[10.1002/rob.4620060203](https://doi.org/10.1002/rob.4620060203).
- Jarquin, G., Escande, A., Arechavala, G., Moulard, T., Yoshida, E., & Parra-Vega, V. (2013). Real-time smooth task transitions for hierarchical inverse kinematics. In *13th IEEE-RAS international conference on humanoid robots* (pp. 528–533). doi:[10.1109/HUMANOIDS.2013.7030024](https://doi.org/10.1109/HUMANOIDS.2013.7030024).
- Kanoun, O., Lamiroux, F., Wieber, P. B., Kanehiro, F., Yoshida, E., & Laumond, J. P. (2009). Prioritizing linear equality and inequality systems: Application to local motion planning for redundant robots. In *IEEE international conference on robotics and automation* (pp. 2939–2944).
- Kanoun, O., Lamiroux, F., & Wieber, P. B. (2011). Kinematic control of redundant manipulators: Generalizing the task-priority framework to inequality task. *IEEE Transactions on Robotics*, 27(4), 785–792.
- Keith, F., Wieber, P. B., Mansard, N., & Kheddar, A. (2011). Analysis of the discontinuities in prioritized tasks-space control under discrete task scheduling operations. In *IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 3887–3892). doi:[10.1109/IROS.2011.6094706](https://doi.org/10.1109/IROS.2011.6094706).
- Kerrigan, E. C., & Maciejowski, J. M. (2000). Soft constraints and exact penalty functions in model predictive control. In *Proceedings of the UKACC international conference*. Cambridge, UK.
- Khatib, O. (1986). Real-time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research*, 5(1), 90–98.
- Khatib, O. (1987). A unified approach for motion and force control of robot manipulators: The operational space formulation. *IEEE Journal of Robotics and Automation*, 3(1), 43–53.
- Khatib, O. (1995). Inertial properties in robotic manipulation: An object-level framework. *The International Journal of Robotics Research*, 14(1), 19–36.
- Khatib, O., Sentis, L., & Park, J. H. (2008). A unified framework for whole-body humanoid robot control with multiple constraints and contacts. *European robotics symposium 2008, springer tracts in advanced robotics* (Vol. 44, pp. 303–312). Berlin: Springer.
- Lee, J., Mansard, N., & Park, J. (2012). Intermediate desired value approach for task transition of robots in kinematic control. *IEEE Transactions on Robotics*, 28(6), 1260–1277. doi:[10.1109/TRO.2012.2210293](https://doi.org/10.1109/TRO.2012.2210293).
- Liégeois, A. (1977). Automatic supervisory control of the configuration and behavior of multibody mechanisms. *IEEE Transactions on Systems, Man and Cybernetics*, 7(12), 868–871.
- Liu, M., Micaelli, A., Evrard, P., Escande, A., & Andriot, C. (2011). Interactive dynamics and balance of a virtual character during manipulation tasks. In *IEEE international conference on robotics and automation (ICRA)* (pp. 1676–1682).
- Liu, M., Micaelli, A., Evrard, P., Escande, A., & Andriot, C. (2012). Interactive virtual humans: A two-level prioritized control framework with wrench bounds. *IEEE Transactions on Robotics*, 28(6), 1309–1322. doi:[10.1109/TRO.2012.2208829](https://doi.org/10.1109/TRO.2012.2208829).
- Mansard, N., Khatib, O., & Kheddar, A. (2009a). A unified approach to integrate unilateral constraints in the stack of tasks. *IEEE Transactions on Robotics*, 25(3), 670–685.
- Mansard, N., Remazeilles, A., & Chaumette, F. (2009). Continuity of varying-feature-set control laws. *IEEE Transactions on Automatic Control*, 54(11), 2493–2505. doi:[10.1109/TAC.2009.2031202](https://doi.org/10.1109/TAC.2009.2031202).
- Mistry, M., Nakanishi, J., & Schaal, S. (2007). Task space control with prioritization for balance and locomotion. In *IEEE/RSJ international conference on intelligent robots and systems* (pp. 331–338). doi:[10.1109/IROS.2007.4399595](https://doi.org/10.1109/IROS.2007.4399595).
- Mistry, M., Nakanishi, J., Cheng, G., & Schaal, S. (2008). Inverse kinematics with floating base and constraints for full body humanoid robot control. In *8th IEEE-RAS international conference on humanoid robots* (pp. 22–27). doi:[10.1109/ICHR.2008.4755926](https://doi.org/10.1109/ICHR.2008.4755926).
- Padois, V., Fourquet, J. Y., Chiron, P., et al. (2007). Kinematic and dynamic model-based control of wheeled mobile manipulators: A unified framework for reactive approaches. *Robotica*, 25(2), 157.
- Peters, J., Mistry, M., Udwadia, F., Nakanishi, J., & Schaal, S. (2008). A unifying framework for robot control with redundant dofs. *Autonomous Robots*, 24(1), 1–12. doi:[10.1007/s10514-007-9051-x](https://doi.org/10.1007/s10514-007-9051-x).
- Petrič, T., & Žlajpah, L. (2013). Smooth continuous transition between tasks on a kinematic control level: Obstacle avoidance as a control problem. *Robotics and Autonomous Systems*, 61(9), 948–959. doi:[10.1016/j.robot.2013.04.019](https://doi.org/10.1016/j.robot.2013.04.019).
- Saab, L., Soueres, P., & Fourquet, J. Y. (2009). Coupling manipulation and locomotion tasks for a humanoid robot. In *International conference on advances in computational tools for engineering applications (ACTEA)* (pp. 84–89).
- Saab, L., Mansard, N., Keith, F., Fourquet, J. Y., & Soueres, P. (2011). Generation of dynamic motion for anthropomorphic systems under prioritized equality and inequality constraints. In *IEEE international conference on robotics and automation (ICRA)* (pp. 1091–1096).

- Saab, L., Ramos, O., Keith, F., Mansard, N., Soueres, P., & Fourquet, J. Y. (2013). Dynamic whole-body motion generation under rigid contacts and other unilateral constraints. *IEEE Transactions on Robotics*, 29(2), 346–362. doi:10.1109/TRO.2012.2234351.
- Sadeghian, H., Villani, L., Keshmiri, M., & Siciliano, B. (2013). Dynamic multi-priority control in redundant robotic systems. *Robotica*, 31, 1155–1167. doi:10.1017/S0263574713000416.
- Salini, J. (2013). *Arboris-Python: A rigid body dynamics and contacts simulator written in python*. <https://github.com/salini/arboris-python>.
- Salini, J., Padois, V., & Bidaud, P. (2011). Synthesis of complex humanoid whole-body behavior: A focus on sequencing and tasks transitions. In *IEEE international conference on robotics and automation (ICRA)* (pp. 1283–1290). doi:10.1109/ICRA.2011.5980202.
- Sentis, L., & Khatib, O. (2004a). Prioritized multi-objective dynamics and control of robots in human environments. In *4th IEEE/RAS international conference on humanoid robots*. (Vol. 2, pp. 764–780). doi:10.1109/ICHR.2004.1442684.
- Sentis, L., & Khatib, O. (2004b). Task-oriented control of humanoid robots through prioritization. In *IEEE RAS/RSJ international conference on humanoid robots*.
- Sentis, L., & Khatib, O. (2005). Synthesis of whole-body behaviors through hierarchical control of behavioral primitives. *International Journal of Humanoid Robotics*, 02(04), 505–518. doi:10.1142/S0219843605000594.
- Sentis, L., Park, J., & Khatib, O. (2010). Compliant control of multi-contact and center of mass behaviors in humanoid robots. *IEEE Transactions on Robotics*, 26(3), 483–501. doi:10.1109/TRO.2010.2043757.
- Siciliano, B., & Slotine, J. J. (1991). A general framework for managing multiple tasks in highly redundant robotic systems. In *Fifth international conference on advanced robotics* (Vol. 2, pp. 1211–1216). doi:10.1109/ICAR.1991.240390.
- Stasse, O., Escande, A., Mansard, N., Miossec, S., Evrard, P., & Kheddar, A. (2008). Real-time (self)-collision avoidance task on a HRP-2 humanoid robot. In *IEEE international conference on robotics and automation (ICRA)* (pp. 3200–3205).



Mingxing Liu is a Post-Doctoral Fellow at the Institut des Systèmes Intelligents et de Robotique (ISIR, UMR CNRS 7222) at Université Pierre et Marie Curie (UPMC), Paris, France. In 2009, she received both her engineering degree from Ecole Centrale Paris, France and her master's degree in Automatic Control from Shanghai Jiao Tong University, China. From 2009 to 2012, she is a PhD student at the Systems and Technologies Integration Laboratory of the French

Atomic Energy Commission (CEA-LIST). She received the Ph.D. degree in Robotics in 2013 from UPMC. Her research interests include automatic control, whole-body control for redundant robots such as industrial manipulators and humanoid robots, as well as human-robot interaction.



Yang Tan received the M.S. degree in Mechanical Engineering from Tianjin University, Tianjin, China in June 2012. He is pursuing the Ph.D. degree at the Institute of Intelligent Systems and Robotics (ISIR, UMR CNRS 7222), University of Pierre and Marie Curie, Paris, France. His current research interests include robotics and control.



Vincent Padois is an Associated Professor of Robotics and Computer Science and a member of the Institut des Systèmes Intelligents et de Robotique (ISIR, UMR CNRS 7222) at Université Pierre et Marie Curie in Paris, France. In 2001, he receives both an engineering degree from the Ecole Nationale d'Ingénieurs de Tarbes (ENIT), France and his master's degree in Automatic Control from the Institut National Polytechnique de Toulouse (INPT), France. From

2001 to 2005, he is a Ph.D. student in Robotics of the ENIT/INPT Laboratoire Genie de Production. In 2006 and 2007, he is a Post-Doctoral Fellow in the Stanford Artificial Intelligence Laboratory and more specifically in the group of Professor O. Khatib. Since 2007, his research activities at ISIR are mainly focused on the automatic design, the modeling and the control of redundant and complex systems such as wheeled mobile manipulators, humanoid robots as well as standard manipulators evolving under constraints in complex environments. He is also involved in research activities that aim at bridging the gap between adaptation and decision making techniques provided by Artificial Intelligence and low-level, reactive control. Since 2011, he holds the "Intervention Robotics" RTE/UPMC chair position.

B.2 Motion safety and constraints compatibility for multi-body robots

Motion safety and constraints compatibility for multibody robots

Sébastien Rubrecht · Vincent Padois · Philippe Bidaud ·
Michel de Broissia · Max Da Silva Simoes

Received: 15 January 2011 / Accepted: 28 November 2011 / Published online: 7 January 2012
© Springer Science+Business Media, LLC 2011

Abstract In this paper we propose a methodology to ensure safe behaviors of multibody robots in reactive control frameworks. The permanent satisfaction of constraints being insufficient to ensure safety, this approach focuses on the constraints expression: the compatibility between these constraints is studied, and safe alternatives are ensured when compatibility cannot be established. Case studies involving obstacles, joint position, velocity and acceleration limits illustrates the approach. A particular method is developed to take full advantage of a smooth state of the art avoidance techniques (Faverjon and Tournassoud in Proceedings of the 1987 IEEE international conference on robotics and automation, pp. 1152–1159, 1987) while maintaining safety. Experiments involving a 6-DOF manipulator operating in a cluttered environment illustrate the reliability of the approach and validate the expected performances.

S. Rubrecht (✉) · M. de Broissia
Bouygues Travaux Publics, St Quentin en Yvelines, France
e-mail: s.rubrecht@bouygues-construction.com

M. de Broissia
e-mail: m.debroissia@bouygues-construction.com

V. Padois · P. Bidaud
Institut des Systèmes Intelligents et de Robotique, Université
Pierre et Marie Curie, Paris, France

V. Padois
e-mail: vincent.padois@isir.upmc.fr

P. Bidaud
e-mail: philippe.bidaud@isir.upmc.fr

M. Da Silva Simoes
Commissariat à l’Energie Atomique et aux Energies Alternatives,
CEA, LIST, Interactive Robotics Laboratory, Fontenay aux
Roses 92265, France
e-mail: max.simoes@cea.fr

Keywords Robotic constraints · Constraints compliant control · Motion safety · Multibody robot safety · Safe reactive control

1 Introduction

The starting point of the present work deals with the use of a robotic arm in charge of maintenance tasks in the inner part of the excavation room of a tunnel boring machine (Fig. 1). The robot is teleoperated to perform inspections and cleaning tasks in a static and cluttered environment. The control of such a robot is reactive (teleoperation is not compatible with offline planning) and subject to real time constraints (force feedback requires control loop frequencies higher than 500 Hz). Although the features of this problem are common in the literature, safety issues remain. For example, most of the collisions avoidance methods do not take the system dynamics into account, which may cause collisions in tight environments and high speed motions. As the robot is subject to kinematic and dynamic constraints that cannot be ignored, it is important to take the physical properties of the system and its environment into account to ensure motion safety.

1.1 Safety criterions for control

The notion of *safety* for a system is a principle applied at various levels. At the design level, safety is often integrated directly in the system (Ikuta et al. 2003; Zinn et al. 2004; Haddadin et al. 2010). At the control level, the work related to offline optimal trajectory planning is closely linked to joint constraints management (Brady 1982; Biagiotti and

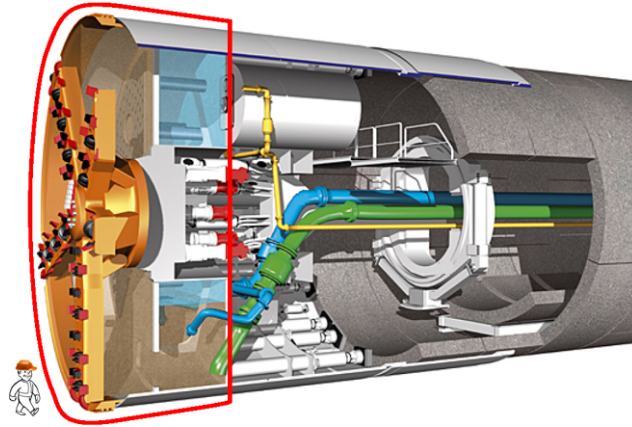


Fig. 1 Tunnel Boring Machine. The red line encircles the cutter head and the excavation room, which is the manipulator's working area

Melchiorri 2008); in spite of the context differences,¹ their recent adaptations to online frameworks (Kröger 2010) exhibit some similarities with reactive control techniques. In a strictly reactive context, safety has been neglected for a long time. Recently, Fraichard (2007) proposed 3 criteria to ensure safety:

1. to decide its future motion, a robotic system should consider its own dynamics;
2. to decide its future motion, a robotic system should consider the environment objects future behavior;
3. to decide its future motion, a robotic system should reason over an infinite time-horizon.

In case of a static environment (not known *a priori*), only the first criterion stands: the future behavior of the objects in the environment remains identical to the current one. The third criterion can thus be integrated in the first one if the consideration of the system's own dynamics is done over an infinite time-horizon (referred later as the *extended criterion 1*).

1.2 Safety of common approaches for collisions avoidance with multibody robots

The number of constraints being potentially higher than the number of DOF, the usual *active avoidance techniques* (approaches for which the avoidance requires a motion) involved in multi-objectives frameworks (issued from Khatib et al. (1986) and Maciejewski et al. (1985)) cannot lead to safety. Moreover, they do not involve dynamics in the avoidance magnitude computation.

Faverjon and Tournassoud (1987) proposed an avoidance technique included in a Quadratic Programming (QP) control law structure. This method limits the velocities toward obstacles by inequalities (*passive avoidance*), which is more

likely to avoid the collisions whatever the number of obstacles. QP are now widely used in manipulators or humanoids control (Decre et al. 2009; Escande et al. 2010), but the avoidance methods still do not include dynamics on an infinite time-horizon. It results that, to our knowledge, no control law for multibody robot passes the extended criterion 1.

In fact, most of the research work related to safety at the control level is led in the field of mobile robotics, i.e. single body mobile robots avoiding collisions: models are simpler, and the operational capabilities predictions are easier (operational deceleration limits do not depend on the robot configuration for example). As an example, the Dynamic Window Approach (DWA) (Fox et al. 1997) involves the acceleration limits of a mobile robot and ensures its safety in a fix environment (extended criterion 1). More recent developments in this domain are part of the framework based on the notion of Inevitable Collision State proposed by Fraichard and Asama (2004): e.g. Martinez-Gomez and Fraichard (2009), Althoff et al. (2010), Bautin et al. (2010).

To the best of our knowledge, although this framework could be used to assess the safety of a wider scope of applications, (1) it has never been applied to multibody robots; (2) it is limited to collisions avoidance with respect to dynamics, which can be formulated as the compatibility between the constraint of geometric collisions avoidance and acceleration limits. However, these are just two constraints among the many constraints that have to be faced in robotics: joint position, velocity, acceleration and torque limits (joint space), collisions with obstacles and forbidden regions (Cartesian space), contacts conservation constraints (Park and Khatib 2008), comanipulation and cooperation (Khatib et al. 2001), actuators temperature limits (Guilbert et al. 2008), etc. We can conclude that there is still a lack regarding robots safety (in particular for multibody robots) when considering a large variety of constraints.

1.3 Constraints compatibility

All these constraints can be considered at the velocity kinematics level for example, where the model is traditionally formulated as

$$\dot{\mathbf{X}}_{des}(t) = \frac{\partial \mathbf{X}_{des}(t)}{\partial \mathbf{q}} \dot{\mathbf{q}}(t) = \mathbf{J}_T(\mathbf{q}(t)) \dot{\mathbf{q}}(t) \quad (1)$$

where t is the current instant, $\mathbf{J}_T(\mathbf{q})$, $\dot{\mathbf{X}}_{des}$, $\mathbf{q}$ and $\dot{\mathbf{q}}$ are respectively the operational task Jacobian matrix (size (m, n)), the operational desired velocity vector (size m), the robot configuration (size n) and the joint space velocity vector (size n). The QP formulation of the control problem has the advantage to explicit the constraints that are supposed to be

¹Most of these approaches are exclusively concerned with joint physical limits as operational constraints are managed by path planning.

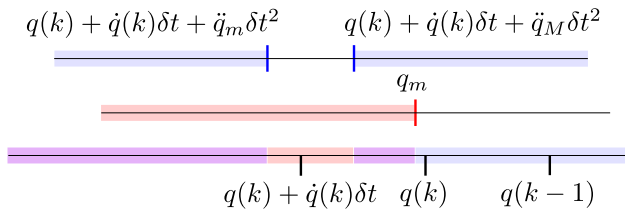


Fig. 2 Incompatibility between constraints represented in the joint space. The terms $q(k) + \dot{q}(k)\delta t + \ddot{q}_m(k)\delta t^2$ and $q(k) + \dot{q}(k)\delta t + \ddot{q}_M(k)\delta t^2$ are respectively the configurations induced by a full acceleration and full deceleration (approximation based on finite differences). The subscripts m and M denote respectively the minimum (negative value) and the maximum (positive value) limit of the considered variable. The control admissibility space of joint acceleration limits (*up*) and minimum joint position limit (*middle*) cannot be satisfied simultaneously: $q(k+1) \approx q(k) + \dot{q}(k)\delta t$ will inevitably violate one of the constraints (*bottom*)

satisfied by the robot

$$\min_{\dot{q}(k+1) \in \mathbb{R}^n} \|\dot{X}_{des}(k+1) - J_T(q(k))\dot{q}(k+1)\| \quad (2)$$

$$\text{subject to } J_C(q(k))\dot{q}(k+1) - b(k) \leq 0 \quad (3)$$

where k is the current time step, $\dot{q}(k+1)$ is the velocity vector chosen for the next time step (control input vector), $J_C(q(k))$ is the Jacobian of constraints (size (p, n)) and $b(k)$ is the constraints limit vector (size p). The feasibility of the problem (i.e. the existence of $\dot{q}(k+1)$ such that $J_C(q(k))\dot{q}(k+1) - b(k) \leq 0$) is usually taken for granted; however, if the constraints expressions have not been carefully set-up, incompatibilities may occur. The typical case is the joint position limit violation because of limited accelerations illustrated by Fig. 2. If a joint gets close to one of its position limits with a high velocity, its deceleration capabilities may not be sufficient to avoid the collision with the boundary (joint position limit). As an example, a maximum deceleration 2 rad/s^2 imposed on a joint moving at 1.0 rad/s requires 0.5 s to actually stop; then, distance travelled is 0.25 rad . This example illustrates the fact that satisfying at each time the joint position and the joint acceleration limits does not prevent from a constraint violation due to incompatibility. Usually, virtual envelopes are set up around the physical limits to absorb such violations. These envelopes do not guarantee safety and often artificially limit the performances of the robot. Relying on a safe approach taking dynamics into account would enable to reduce significantly those envelopes.

The contribution of this paper is to propose a methodology to ensure safety at the control level. The control problem resolution is out of the scope of this paper: it is assumed that once the control problem is feasible, a control law algorithm such as the one proposed in Rubrecht et al. (2010a, 2010b), solves it appropriately (Constraints Compliant Control law). The present work focuses on the formulation of the control problem.

The proposed methodology is applied to the case of a multibody robot in a static environment (extended criterion 1). Section 2 exposes the description retained for the robotic system and its constraints and proposes a definition of safety at the control level. Section 3 is dedicated to the methodology description, whereas Sect. 4 details case studies dealing with static obstacles, joint positions, velocities and accelerations constraints. Finally, a set of experiments illustrates the approach in Sect. 5.

2 Description for safety

In this section, an appropriate description of the robot and its constraints is introduced and a resulting definition of safety is proposed. In this work the control problem is formulated at the velocity kinematic level. The assumptions of this study are an exact perception of the system and the environment, an exact knowledge of the model and the system real capabilities and an exact execution when the desired joint input satisfies the constraints: $u(k) = \dot{q}_{des}(k+1)$ for time step k is exactly carried out at the next time step ($\dot{q}(k+1) = \dot{q}_{des}(k+1)$).

2.1 E-state

First, it appears that the description of the behavior of a robotic system Σ and its constraints through its state s as defined in the State Representation formalism is insufficient. As a matter of fact, an extended state vector (e-state) is defined and denoted σ ; it gathers all the variables which allow to describe Σ and its constraints. The e-state is defined over continuous time ($t \in \mathbb{R}_+$) since it contains variables used to describe the physical system. For example, a n -DOF manipulator controlled at the velocity kinematic level and constrained by collisions avoidance and joint position, velocity and acceleration limits has the following e-state

$$\sigma = [q^T \dot{q}^T \ddot{q}^T d^T]^T \quad (4)$$

where d^T is a vector of distances to obstacles. In the same example, the state of Σ would be $s = q$. Conversely to σ , the control vector $u(k) = \dot{q}(k+1)$ belongs to $\mathbb{R}^n$ and it is defined over the discrete time ($k \in \mathbb{N}$). The e-state space is denoted $\mathcal{S}$ and the control space ($\mathbb{R}^n$) is denoted $\mathcal{C}$.

2.2 E-state constraints

The notion of constraint usually refers to both a test on the system (“Is the joint boundary exceeded?”—denoted by *e-state constraint*) and a prerequisite to motion (“The control input sent to the actuator should not lead to exceed the joint boundary.”—denoted by *control constraint*). The

e-state constraints describe if Σ satisfies safety at the current time, i.e. when not considering any time horizon. They can be expressed through Boolean functions such as

$$\begin{aligned} \mathcal{S} &\rightarrow \mathcal{B} \\ f : \sigma &\mapsto 1 \quad \text{if the constraint is satisfied} \\ &0 \quad \text{else,} \end{aligned} \quad (5)$$

where $\mathcal{S}$ is the e-state space and $\mathcal{B}$ the Boolean space.

As an example of e-state constraint, $f_{PM,3}$ (P for Position limit, M for Maximum) describes the superior position limit of the third joint

$$f_{PM,3}: \sigma(t) \mapsto q_3(t) - q_{M,3} \leq 0 \quad (6)$$

where $q_3(t)$ is the joint position of joint 3 at time t and $q_{M,3}$ is the maximum joint boundary value.

Any e-state satisfying the p e-state constraints imposed to Σ satisfies the property $\bigwedge_{i=1}^p (f_i(\sigma)) = 1$, where $\bigwedge$ is the logical conjunction operator (AND). This means that all e-state constraints are simultaneously true for the e-state σ . In this case, σ is called an *instant-safe e-state*.

2.3 Subspaces of the e-state space and definition of safety

The e-state space $\mathcal{S}$ is composed of subspaces that can be identified. The subspace of $\mathcal{S}$ gathering all the instant-safe e-states is denoted $\mathcal{S}_A$. Conversely, the complementary subspace gathers the e-states violating an e-state constraint; it is denoted $\mathcal{S}_{VS}$ (VS for Violation e-State). As illustrated by Fig. 2, maintaining at each time step σ in $\mathcal{S}_A$ for the next time step is not sufficient to prevent an inevitable e-state constraint violation in the future. As a consequence, a part of $\mathcal{S}_A$ must never be reached to guarantee safety.

An e-state leading inevitably to an e-state constraint violation is called an Inevitable Violation e-State (IVS). It is an extension of the notion of Inevitable Collision State (ICS) defined by Fraichard and Asama (2004) which denotes a state from which, whatever the sequence of control inputs sent, a collision finally occurs. Once an IVS is reached, the system can be considered as not safe anymore as an e-state constraint violation is going to happen. The space of IVS is a subspace of $\mathcal{S}_A$ denoted $\mathcal{S}_{IVS}$. The union of $\mathcal{S}_{IVS}$ and $\mathcal{S}_{VS}$ is denoted $\mathcal{S}_V$ and gathers all the e-states that should be avoided to ensure safety. The complement of $\mathcal{S}_V$ in $\mathcal{S}$ is denoted $\mathcal{S}_{\bar{V}}$. These subspaces are illustrated on Fig. 3. A definition of safety is then

Definition 1 The **safety** of a robotic system Σ is ensured at the control level if its e-state σ cannot reach $\mathcal{S}_V$.

This definition enlightens the role of the constraints expression to limit the evolution of the system toward dangerous areas.

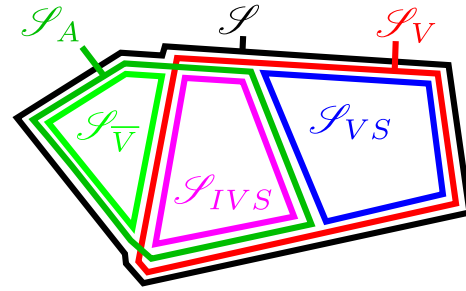


Fig. 3 Partitioning of the e-state space. To be safe, a system should not be able to reach $\mathcal{S}_V$

The last subspace to define in this section regards the space that can be reached by a system. Given an initial e-state σ_0 , $\mathcal{R}(\sigma_0)$ denotes the space of all the reachable e-states on an infinite time horizon through all the possible constraints compliant control sequences.

3 Methodology to ensure safety

This section exposes the methodology to ensure safety. The proposed methodology must be carried out offline, upstream from any robotic mission in a new constraints context. The equivalent of e-state constraints, the *control constraints*, should be formulated at the control level. Once their *validity* is ensured, they should either be proved *compatible*, or the permanent availability of an *alternative safe behavior* must be ensured on an infinite time horizon.

3.1 Step 1: control constraints definition

The controller cannot act directly on the e-state σ ; it modifies it indirectly through the control vector u . Reciprocally, at each time step, by imposing conditions on the e-state, each e-state constraint forbids an area of the control vector space $\mathcal{C}$. Hence, to each e-state constraint f is associated a control constraint F which can be defined as the function returning the space of *admissible* control vectors $\mathcal{C}_A(\sigma)$, i.e. the control vectors leading to instant safe e-states at the next time step. A control constraint can be expressed as a Boolean function returning whether a given control vector belongs to $\mathcal{C}_A(\sigma)$ or not

$$\begin{aligned} (\mathcal{S}, \mathcal{C}) &\mapsto \mathcal{B} \\ F : (\sigma, u) &\mapsto 1 \quad \text{if } u \in \mathcal{C}_A(\sigma) \\ &0 \quad \text{else.} \end{aligned} \quad (7)$$

The control input being discrete, control constraints are defined over discrete time ($k \in \mathbb{N}$). As an extension of the notation $\sigma(t)$ ($t \in \mathbb{R}_+$), $\sigma(k)$ ($k \in \mathbb{N}$) denotes the e-state at time step k .

In the example of the third joint superior position limit, if the control is done at the velocity kinematic level, a possible control constraint is

$F_{PM,3}$:

$$(\sigma(k), \dot{q}(k+1)) \mapsto \dot{q}_3(k+1) - \frac{q_{M,3} - q_3(k)}{\delta t} \leq 0. \quad (8)$$

It can be mentioned that from a practical point of view, the inequalities imposed on the system at each time step in the QP control law structure are an example of control constraints. In this case, at a given time step k , these terms are gathered in

$$J_C(q(k))\dot{q}(k+1) - b(k) \leq 0. \quad (9)$$

3.2 Step 2: validity

In order to ensure safety, the first stage is to check that control constraints are *valid*.

Definition 2 Validity. Let σ be an instant-safe e-state at time step k , a control constraint F is said valid if its satisfaction implies the satisfaction of its associated e-state constraint f at next time step $k+1$ and for all time between k and $k+1$.

$$\sigma \in \mathcal{S}_A, k \in \mathbb{N}, \forall t \in [k\delta t; (k+1)\delta t]:$$

$$F(\sigma(k), u(k)) = 1 \Rightarrow f(\sigma(t)) = 1.$$

The validity of constraints is most of the time an assumption rather than a formally proved property. For example, constraints at various physical levels (position, velocity, acceleration, etc.) must be converted to the (single) control physical level, which is often done thanks to first order approximations (finite differences). The control being in discrete time, the approximations induced by finite differences generate errors between the discrete ideal behavior and the real one. However, it is assumed that the sampling period is appropriately chosen to ensure that these errors remain acceptable with respect to the various usual sources of errors (model approximations, sensors precision, etc.). As a remark, reducing the space of admissible control vectors associated to a control constraint generally enables to ensure validity.

3.3 Step 3: compatibility

A second stage to ensure safety is to check that the set of control constraints is *compatible*.

Definition 3 Compatibility. Given an initial e-state σ_0 in $\mathcal{S}_V$, a set of p control constraints is compatible if for all e-states σ in $\mathcal{R}(\sigma_0)$, there exists u in $\mathcal{C}$ such that $\bigwedge_{i=1}^p (F(\sigma, u)) = 1$.

The following proposition establishes that validity and compatibility ensure safety.

Proposition 1 Let σ_0 in $\mathcal{S}_V$ be the e-state of Σ , a robotic system constrained by p e-state constraints. If the p control constraints of Σ are valid and compatible, then safety is ensured.

Proof Let Σ be a robotic system in an initial (time step 0) e-state σ_0 belonging to $\mathcal{S}_V$. As the control constraints are compatible, there exists u in $\mathcal{C}$ such that $\bigwedge_{i=1}^p (F(\sigma_0, u)) = 1$. Thus the control problem is feasible and as all the constraints are valid, σ is an instant-safe e-state at time step 1 and for all time between time steps 0 and 1. This reasoning can be extended by recursion for all time steps. As a consequence, σ is maintained in $\mathcal{S}_A$ on an infinite time-horizon, which means that it is maintained in $\mathcal{S}_V$; as a consequence, it cannot reach $\mathcal{S}_V$ and safety is ensured. $\square$

3.4 Step 4: design of alternative safe behaviors

The study of compatibility between control constraints is complex: an exhaustive method would consist to, given an initial e-state σ_0 , evaluate all the control constraints for all the e-states σ reachable from σ_0 to detect empty intersections between control admissibility spaces $\mathcal{C}_A(\sigma)$. Given the diversity of constraints, it seems vain to look for generic methods to detect incompatibilities and modify control constraints appropriately to eradicate them. Moreover, sometimes incompatibilities cannot be resolved: when variables cannot be measured accurately, or when there is no model available, another method should be used to ensure safety.

A second way to guarantee safety is to ensure the permanent availability of a sequence of control solutions leading to instant-safe e-states on an infinite time horizon. At each time step, it is ensured that the controller will be able at next time step to switch to an infinite sequence of controls leading to exclusively instant-safe e-states. Similarly to the proof of Proposition 1, σ is maintained in $\mathcal{S}_A$ on an infinite time-horizon, which means that it is maintained in $\mathcal{S}_V$; as a consequence, it cannot reach $\mathcal{S}_V$ and safety is ensured. This control sequence is called an Alternative Safe Behavior (ASB—referred as evasive manoeuvres by Parthasarathi et al. (2007)). Dedicated ASBs are exposed in Sect. 4.5 according to the specifications of the proposed case studies.

3.5 Summary and methodology

To describe the physical system Σ and its constraints in continuous time, the e-state σ is proposed, and the status of the system with respect to its constraints is given by the e-state constraints f . Based on this description, a definition of safety at the control level is proposed: Σ is safe if its e-state

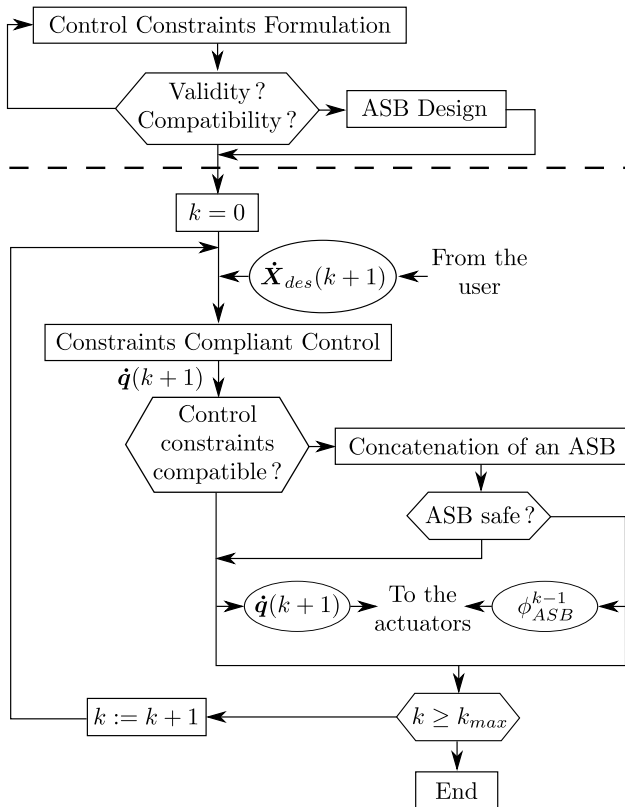


Fig. 4 Safe controller algorithm. The part of the algorithm above the dashed line is offline and is concerned with the control problem formulation; the part of the algorithm under the dashed line is online and is concerned with the control problem resolution. From each identified e-state constraint (physical limit or induced by the mission), a control constraint is formulated offline. If the validity of the constraints cannot be proved, a new formulation of the control constraints must be expressed. It is always possible to find valid control constraints by reducing their space of admissible control vector $\mathcal{C}_A$. If the compatibility of the control constraints cannot be proved, a new formulation can be expressed and evaluated, or an ASB must be established. Once this is done, the reactive control loop is launched. At each time step, the controller is fed with operational inputs and solves the control problem thanks to any Constraints Compliant Control algorithm. In particular, it can include usual constraints avoidance techniques (e.g. the one of Maciejewski and Klein 1985). If the compatibility of the control constraints defined offline could not be proved, an ASB sequence is concatenated to the desired joint motion: if the resulting behavior is not safe, then the first control input of the ASB is sent (represented by ϕ_{ASB}^{k-1}), which safety has been proved at a previous time step; else, the control solution is sent to the actuators

σ is not able to reach the forbidden e-states $\mathcal{S}_V$. In order to prevent this, the control constraints F are defined, and the validity property links control constraints in discrete time and e-state constraints in continuous time. To ensure safety, either the compatibility property must be proved for the set of control constraints, or the availability of an Alternative Safe Behavior must be ensured on an infinite time horizon.

As an outcome, the following methodology is proposed to obtain a safe behavior for a system subject to constraints in a reactive framework:

1. Based on e-state constraints, formulate associated control constraints;
2. Prove the validity of control constraints. If not possible, go back to 1;
3. Prove the compatibility of control constraints. If not possible, either go back to 1 and modify the control constraints expression or define permanently available ASBs. These ASBs will then be computed at each time step to ensure that the controller is able to switch at the following time step to an infinite sequence of controls leading to exclusively instant-safe e-states.

Based on this methodology, a safe controller algorithm can be set up, as shown on Fig. 4.

4 Case studies

To illustrate the approach described previously, three case studies are proposed, based on combinations of the following constraints: {joint position limits—joint velocity limits—joint acceleration limits—collisions avoidance} applied to a n -DOFs serial manipulator Σ controlled at the velocity kinematic level. The first case study considers compatible constraints with intuitive expressions. The second case study involves constraints that must be modified to be proved compatible. The third case study deals with constraints that cannot be proved to be always compatible; as a result, the permanent availability of an ASB is required.

4.1 E-state constraints expression

This first section exposes the e-state constraints expressions. These expressions describe if, for a given time t in $\mathbb{R}_+$, the e-state σ of Σ is instant-safe. The subscripts m and M denote respectively the minimum (negative value by convention) and the maximum (positive value by convention) limit of the considered variable.

Joint position limit

$$f_{PM}: \sigma(t) \mapsto q(t) \leq q_M \quad (10)$$

$$f_{Pm}: \sigma(t) \mapsto q_m \leq q(t) \quad (11)$$

Joint velocity limit

$$f_{VM}: \sigma(t) \mapsto \dot{q}(t) \leq \dot{q}_M \quad (12)$$

$$f_{Vm}: \sigma(t) \mapsto \dot{q}_m \leq \dot{q}(t) \quad (13)$$

Joint acceleration limit

$$f_{AM}: \sigma(t) \mapsto \ddot{q}(t) \leq \ddot{q}_M \quad (14)$$

$$f_{Am}: \sigma(t) \mapsto \ddot{q}_m \leq \ddot{q}(t) \quad (15)$$

Collisions avoidance A collision is characterized by

$$\Sigma \cap \Omega \neq \emptyset \quad (16)$$

where Σ is the system (meant here as the set of all the system points) and Ω is the set of all the obstacles points. The e-state constraint expression is then

$$f_O: \sigma(t) \mapsto \forall A \in \Sigma, \quad G_A(q(t)) \notin \Omega \quad (17)$$

where $G_A(q)$ is the geometric model of point A belonging to the robot.

4.2 Case study 1: joint position limits, joint velocity limits, collisions avoidance

This case study involves three constraints: joint position limits, joint velocity limits and collisions avoidance. The following control constraints (assumed to be valid, see Sect. 3.2) are derived from (10)–(15) thanks to finite differences. They are given for a time step k in $\mathbb{N}$.

Joint position limit (joint i) F_P

$$F_{PM,i}: (\sigma(k), u(k)) \mapsto J_{c_i^+} \dot{q}(k+1) \leq \frac{q_{M,i} - q_i(k)}{\delta t} \quad (18)$$

$$F_{Pm,i}: (\sigma(k), u(k)) \mapsto J_{c_i^-} \dot{q}(k+1) \leq \frac{q_i(k) - q_{m,i}}{\delta t} \quad (19)$$

where $J_{c_i^+} = [0, \dots, 0, 1, 0, \dots, 0]$ (the term i being 1) and $J_{c_i^-} = [0, \dots, 0, -1, 0, \dots, 0]$.

Joint velocity limit (joint i) F_V

$$F_{VM,i}: (\sigma(k), u(k)) \mapsto J_{c_i^+} \dot{q}(k+1) \leq \dot{q}_{M,i} \quad (20)$$

$$F_{Vm,i}: (\sigma(k), u(k)) \mapsto J_{c_i^-} \dot{q}(k+1) \leq -\dot{q}_{m,i} \quad (21)$$

Collisions avoidance F_O

$$F_O: (\sigma(k), u(k)) \mapsto J_{A,B}(q(k)) \dot{q}(k+1) \leq \frac{d_{A,B}(k)}{\delta t} \quad (22)$$

for all pairs of points (A, B) , where A belongs to the robot and B to the obstacles; $d_{A,B}$ is the distance between A and B ; $J_{A,B}(q(k))$ is the (line) Jacobian of point A along the direction $A \rightarrow B$. For practical reasons, this infinite set of constraints is reduced to one constraint per segment of the robot (shortest distance). This assumption is frequently made despite its limits in some cases (as shown by Kanehiro et al. 2008). It is considered sufficient in the present study.

The space of admissible control vectors for control constraints F_P , F_V and F_O are respectively denoted $\mathcal{C}_A^P$, $\mathcal{C}_A^V$ and $\mathcal{C}_A^O$.

Validity being assumed (cf. Sect. 3.2), the compatibility is checked.

Proposition 2 The set $\{F_P, F_V, F_O\}$ is compatible

Proof Let $\dot{q}_0$ be the null control vector ($\dot{q}_0 = \mathbf{0}$) and let σ_0 be in $\mathcal{S}_V$. For any $\sigma \in \mathcal{R}(\sigma_0)$, $\dot{q}(k+1) = \dot{q}_0$ belongs to $\mathcal{C}_A^P$, $\mathcal{C}_A^V$ and $\mathcal{C}_A^O$. As a result, for all $\sigma(k)$ in $\mathcal{R}(\sigma_0)$, $\dot{q}_0$ is solution of the control problem and thus $F_P(\sigma, \dot{q}_0) \wedge F_V(\sigma, \dot{q}_0) \wedge F_O(\sigma, \dot{q}_0) = 1$. $\square$

As all the control constraints are proved to be always compatible, safety is ensured without modification or ASB required.

4.3 Case study 2: joint position limits, joint velocity limits, joint acceleration limits

As in the previous case, this case study involves three constraints but collisions avoidance is replaced by joint acceleration limits. The control constraints for joint position and velocity limits are taken from case study 1 (18)–(21); the control constraints of joint acceleration limits are derived from (14) and (15) thanks to finite differences (joint i)

$$F_{AM,i}: (\sigma(k), u(k)) \mapsto J_{c_i^+} \ddot{q}(k+1) \leq \ddot{q}_{M,i} \delta t + \dot{q}_i(k) \quad (23)$$

$$F_{Am,i}: (\sigma(k), u(k)) \mapsto J_{c_i^-} \ddot{q}(k+1) \leq -\ddot{q}_{m,i} \delta t - \dot{q}_i(k) \quad (24)$$

The space of admissible control vectors for control constraint F_A is denoted $\mathcal{C}_A^A$.

Proposition 3 The sets of control constraints generated by $\{F_P, F_A\}$ are incompatible.

Proof Let σ_0 be in $\mathcal{S}_V$. From σ_0 , any $\sigma_V \in \mathcal{R}(\sigma_0)$ for which a given joint satisfies

$$\dot{q}(k) > \frac{q_M - q(k)}{\delta t} - \ddot{q}_m \delta t \quad (25)$$

is such that $F_{PM,i}(\sigma_V)$ and $F_{Am,i}(\sigma_V)$ are not compatible, which traduces that $f_{PM,i}$ and $f_{Am,i}$ cannot be satisfied simultaneously. As there is no assumption or constraint preventing from reaching σ_V , then F_P and F_A are incompatible. $\square$

This incompatibility is illustrated on Fig. 2. It has been locally treated by Decre et al. (2009), but as shown in Rubrecht et al. (2010b), the proposed method is tight and can be smoothed by imposing that the joint distance to the joint position limit at next time step should remain superior to the current joint distance needed to decelerate. As a result, a modified expression of F_P is proposed (joint i)

$F_{PM',i}:$

$(\sigma(k), u(k))$

$$\mapsto J_{c_i^+} \dot{q}(k+1) \leq \frac{(q_M - q(k)) - \frac{1}{2}(s_1^2 - s_1)\ddot{q}_m \delta t^2}{(s_1 + 1)\delta t} \quad (26)$$

$F_{PM',i}:$

$(\sigma(k), u(k))$

$$\mapsto J_{c_i^-} \dot{q}(k+1) \leq \frac{(q_m - q(k)) - \frac{1}{2}(s_2^2 - s_2)\ddot{q}_M \delta t^2}{(s_2 + 1)\delta t} \quad (27)$$

with

$$s_1 = -\frac{\sqrt{-2\ddot{q}_m(q_M - q(k))}}{\ddot{q}_m \delta t} \quad (28)$$

$$s_2 = \frac{\sqrt{-2\ddot{q}_M(q_m - q(k))}}{\ddot{q}_M \delta t} \quad (29)$$

The choice of (26) and (27) as the joint position limits constraints provoke a small reduction of the reachable positions. Actually, the resolution of $\dot{q}(k+1) = 0$ in (26) induces

$$q_M - q(k) = \frac{-\ddot{q}_m \delta t^2}{8} \quad (30)$$

which means that the asymptotic value of the joint position according to this constraint is no longer q_M but $q_{M'} = q_M - \frac{-\ddot{q}_m \delta t^2}{8}$. The order of magnitude of this reduction is $\sim \delta t^2$, which can be considered negligible. However, it is a reduction of the space of reachable e-state, and all the compatibility studies involving this control constraint must then be checked over the joint position space $\mathcal{S}_A^{P'} = [q_{m'}; q_{M'}]$ where $q_{m'}$ and $q_{M'}$ are the vectors of general term respectively $q_m + \frac{\ddot{q}_M \delta t^2}{8}$ and $q_M - \frac{-\ddot{q}_m \delta t^2}{8}$.

Proposition 4 The set $\{F_{P'}, F_V, F_A\}$ is compatible.

Proof Let σ_0 be in $\mathcal{S}_V$, the current time step k be in $\mathbb{N}$ and the current e-state $\sigma(k)$ be in $\mathcal{R}(\sigma_0)$. The design of (26) and (27) is based on the condition

$$\Delta q(k+1) > d_{\mathbb{R},dec}(k) \quad (31)$$

where $\Delta q(k+1)$ is the joint distance to the position limit at the next time step and $d_{\mathbb{R},dec}(k)$ is a vector of upper bounds of the joint distances needed to stop at current time step (cf. Rubrecht et al. 2010b). This condition implies that the vector of maximum deceleration velocity $\dot{q}_{dec}(k+1)$ which general term is

$$\dot{q}_{dec}(k+1) = \begin{cases} \dot{q}(k) + \ddot{q}_m \delta t & \text{if } \dot{q}(k) \geq -\ddot{q}_m \delta t \\ \dot{q}(k) + \ddot{q}_M \delta t & \text{if } \dot{q}(k) \leq -\ddot{q}_M \delta t \\ 0 & \text{else} \end{cases} \quad (32)$$

belongs to $\mathcal{C}_A^{P'}(\sigma)$. Then, by definition, it belongs to $\mathcal{C}_A^A(\sigma)$. Finally, as it reduces the velocity magnitude, it belongs to $\mathcal{C}_A^V(\sigma)$. As a result, for all $\sigma(k)$ in $\mathcal{R}(\sigma_0)$, $\dot{q}(k+1) = \dot{q}_{dec}$ is solution of the control problem and thus $F_{P'}(\sigma, \dot{q}_0) \wedge F_V(\sigma, \dot{q}_0) \wedge F_A(\sigma, \dot{q}_0) = 1$. $\square$

This incompatibility between control constraints being resolved, the control constraints are ensured to be always compatible, which ensures safety.

4.4 Case study 3: joint position limits, joint velocity limits, joint acceleration limits, collisions avoidance

This case study involves four constraints, gathering the two previous case studies: joint position limits, joint velocity limits, joint acceleration limits and collisions avoidance. The considered control constraints are (20)–(24), (26) and (27).

Proposition 5 The sets generated by $\{F_O, F_A\}$ are incompatible.

Proof Let σ_0 be in $\mathcal{S}_V$. From σ_0 , any $\sigma_V \in \mathcal{R}(\sigma_0)$ for which

$$J_{A,B}(q(k))\dot{q}(k) > \frac{d_{A,B}(k)}{\delta t} - J_{A,B}(q(k))\ddot{q}_m \delta t \quad (33)$$

shows that $F_O(\sigma)$ and $F_{Am}(\sigma)$ are not compatible, which shows that $f_O(\sigma)$ and $f_{Am}(\sigma)$ cannot be satisfied simultaneously. As there is no assumption or constraint preventing from reaching σ_V , then F_O and F_A are incompatible. $\square$

As mentioned in Rubrecht et al. (2010b), the incompatibility induced by the simultaneous presence of F_O and F_A is complex. Actually, the operational acceleration depends on the robot configuration (derived from (22)).

$$\ddot{X}_{A,B} = J_A(q)\ddot{q} + \dot{J}_A(q)\dot{q} \quad (34)$$

which does not enable to rely on any value for the operational acceleration capabilities along a trajectory. In the worst cases, these capabilities may fall down to zero, which prevents to take a lower bound on which to rely for the deceleration capabilities estimate. As a result, ensuring compatibility between joint acceleration limits and collisions avoidance seems impossible without an exploration in the neighborhood of the current system e-state, which may turn time-consuming and thus not acceptable in a real-time reactive control framework. In this case, the permanent availability of an Alternative Safe Behavior is required.

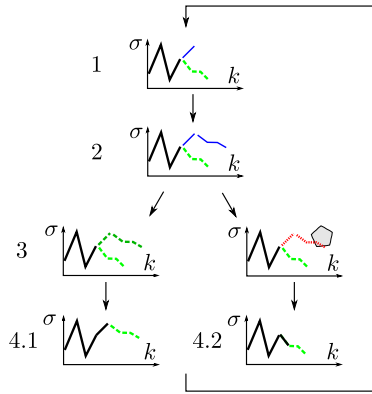


Fig. 5 Algorithm of maximum deceleration based ASB. *Thin plain* is for non validated motion, *dots* is for non admissible motion and *dashes* for safe motion. (1) Control solution computation $\dot{q}(k+1)$; (2) ASB1 profile computation ϕ_{ASB1}^k ; (3) Admissibility check; (4.1 & 4.2) Send appropriate output

4.5 Alternative safe behavior

When the control compatibility cannot be proved, the permanent availability of an Alternative Safe Behavior is required, to be triggered in case of critical situation. As mentioned in Sect. 3.4, an ASB is a sequence of control solutions u leading to instant-safe e-states on an infinite time horizon. It must be computed at each time step and the safety of the resulting e-states must be checked, thus it should be fast to compute. To clarify the following descriptions, let ϕ denotes an infinite constraint compliant control input, i.e. an infinite sequence of controls u satisfying the control constraints at each time step.

4.5.1 Algorithm based on maximum joint deceleration ASB

As a preliminary observation, as the environment is assumed to be static, once an instant-safe e-state is static (no variation with respect to time), it remains safe until the end of time. Consequently, the first ASB proposed ϕ_{ASB1} is a full deceleration at the joint level. This deceleration is the most efficient way to stop the robot: it is fast as no Jacobian has to be recomputed at each time step, and the number of time steps necessary to obtain a static robot is minimized. As the control constraints of joint position, velocity and acceleration limits are compatible, the only remaining constraint to check on all e-states resulting from ϕ_{ASB1} is F_O , that is an intersection between the robot bodies and the environment. The method is detailed on algorithm 1. It is assumed that for $k=0$, the initial e-state σ_0 belongs to $\mathcal{S}_V$. The algorithm is illustrated on Fig. 5.

4.5.2 Algorithm based on mixable joint deceleration ASB

As the robot may oscillate between two behaviors (between $\dot{q}(k+1)$ issued from the task, and ϕ_{ASB1}^{k-1} issued from the

ASB) a safe but rough behavior is expected from Algorithm 1 when moving near obstacles. The problem lies in the *maximal* deceleration toward the static e-state; when ϕ_{ASB1} is chosen at one time step, it is likely to be retained until the robot stops. As shown on a simple example in Fig. 6, in most cases there is no available space during ϕ_{ASB1} for another motion than full deceleration. To have a small margin in the intersection on the control admissibility space, it is proposed to add a prediction ϕ_{ASB2} with reduced accelerations capabilities. At each time step, both predictions are tested (ϕ_{ASB1} based on $\ddot{q}_{min/max}$; ϕ_{ASB2} based on $\alpha\ddot{q}_{min/max}$, with $\alpha \lesssim 1$). Once one of these behavior leads the robot in intersection with the environment, the robot adopts a ϕ_{ASB1}^{k-1} (maximal deceleration). For both behaviors, in case of violation, $\ddot{q}_{min/max}$ is applied. If ϕ_{ASB2} is violated (in most cases), the control will then have a margin in the control admissibility space during its deceleration, where another behavior can be inserted.

The method established by Faverjon and Tournassoud (1987) is proposed to illustrate this approach. This method is referred as the Smooth Avoidance Technique (SAT). Briefly, this method limits the operational velocity of each point of the robot bodies that gets close to an obstacle. The velocity limitation is done through an inequality constraint in the QP framework ((2) and (3)). This limitation involves 2 parameters

$$\dot{d} = -a \frac{d - d_s}{d_i - d_s} \quad \text{for } d \leq d_i \quad (35)$$

where $\dot{d}$ is the temporal derivative of the distance d between a given point of the robot and the obstacle, a is a positive coefficient for adjusting convergence speed, d_s is the security distance and d_i is the distance influence, i.e. the distance under which the constraint is active. The control constraint associated to (35) is

F_C :

$(\sigma(k), u(k))$

$$\mapsto J_A \dot{q}(k) \leq -a \frac{d_{A,B}(k) - d_s}{d_i - d_s} \quad \text{for } d_{A,B}(k) \leq d_i. \quad (36)$$

Including the expression of this control constraint at a given time step does not necessarily yield a feasible control problem. However, it is acceptable to violate it as it does not involve security but rather a desired behavior. Checking the compatibility at each time step is not trivial: knowing if a set of linear constraints is compatible may require the resolution of the associated linear system. An approximate answer is given by checking whether the configuration of maximum deceleration is admissible. It is not a requirement for compatibility (there may be cases for which this configuration is not admissible whereas the constraints are compatible) but it is a sufficient condition. As a result, at each time step the compatibility between the SAT control constraint and the

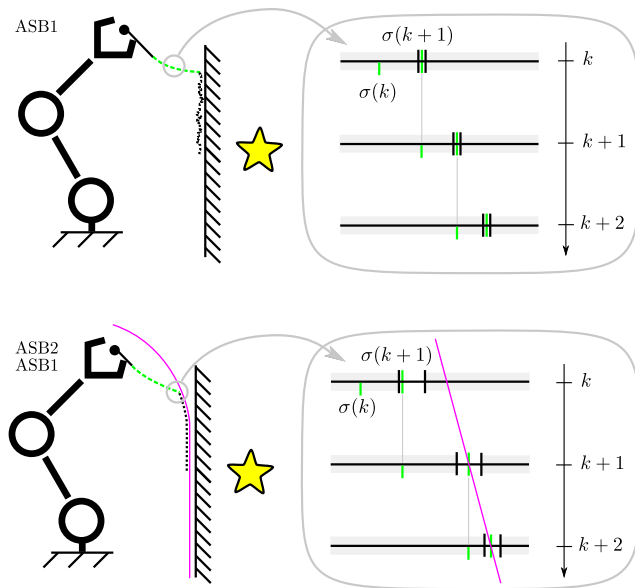


Fig. 6 Comparative behaviors of robots trying to reach a keypoint (star) behind a wall. On the right, the schemes are representations of the e-states projected on the joint space of the second DOF of the system during 3 time steps. *Top*: maximum joint deceleration (ASB1). The motion of the robot is decomposed in three parts. *Dark plain path*: the motion is computed through the control law, and at each time step the controller concatenates the control vectors to be sent with a full deceleration, to check if a collision occurs and decide if the control vector should be sent or not. *Fair dashed path*: a collision with the predicted full deceleration being detected, it is applied before sending the control law computed input; during the ASB deceleration, only the full deceleration control solution is admissible (*top right*). However, when the robot stops, it is close to the obstacle. *Dark dots path*: once near the obstacles, the controller oscillates between the control law solutions and ASB. *Bottom*: mixable joint deceleration (ASB1 & ASB2). As in the scheme at the top, the motion of the robot is decomposed in three parts. *Dark plain path*: control law based motion; it is shorter than the upper one, because deceleration predictions are based on under-estimated capabilities. *Fair dashed path*: the ASB is done with maximal deceleration capabilities, but as it has been triggered before, the control law solutions can be chosen in a small (but not reduced to a point) interval (*bottom right*). *Dark dots on the fair plain curve*: as a result, the robot progression toward the wall can be damped by a smooth path constraint. *Fair plain curve*: representation of the smooth path trajectory

other constraints is checked: if the SAT is not compatible, it is not considered.

The final method is detailed on Algorithm 2. As for Algorithm 1, it is assumed that for $k = 0$, the initial e-state σ_0 belongs to $\mathcal{S}_V$.

The algorithm is illustrated on Fig. 7.

5 Results

The following part details the results obtained with a 6-DOF manipulator. The results are composed of 3 experiments showing:

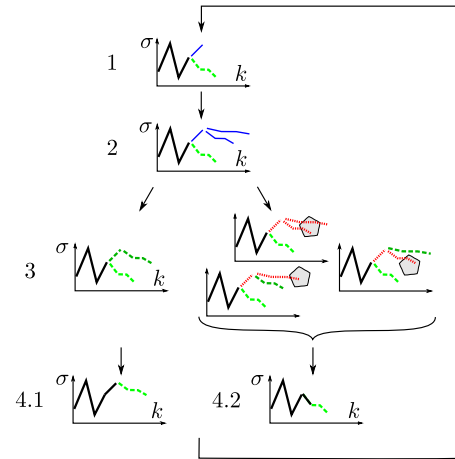


Fig. 7 Algorithm of mixable deceleration based ASB. *Thin plain* is for non validated motion, *dots* is for non admissible motion and *dashes* for safe motion. (1) Control solution computation $\dot{q}(k+1)$; (2) ASB1 (ϕ_{ASB1}^k) and ASB2 (ϕ_{ASB2}^k) profile computation; (3) Admissibility check; (4.1 & 4.2) Send appropriate output

Algorithm 1 Maximum joint deceleration ASB

for all $k \in \mathbb{N}$ **do**

(1) Compute an admissible solution $\dot{q}(k+1)$

(2) Compute deceleration trajectory ϕ_{ASB1}^k

if (3) for all σ resulting from ϕ_{ASB1}^k , σ is an instant-safe e-state **then**

(4.1) Send $\dot{q}(k+1)$

else

(4.2) Send the first element of ϕ_{ASB1}^{k-1}

end if

end for

Algorithm 2 Mixable joint deceleration ASB

for all $k \in \mathbb{N}$ **do**

Control constraints: $\{F_{P'}, F_V, F_A, F_O, F_C\}$

if $\bigcap_{i=P', V, A, O, C} \mathcal{C}_A^i \neq 1$ **then**

Control constraints: $\{F_{P'}, F_V, F_A, F_O\}$

end if

(1) Compute an admissible solution $\dot{q}(k+1)$

(2) Compute deceleration trajectories $\phi_{ASB1}^k, \phi_{ASB2}^k$

if (3) for all σ_1 resulting from ϕ_{ASB1}^k , σ_1 is a safe e-state AND for all σ_2 resulting from ϕ_{ASB2}^k , σ_2 is a safe e-state **then**

(4.1) Send $\dot{q}(k+1)$

else

(4.2) Send the first element of ϕ_{ASB1}^{k-1}

end if

end for

- the safe behavior obtained thanks to the resolution of the joint constraints compatibility;
- the safe behavior obtained thanks to the resolution of the joint constraints compatibility and the maximum joint deceleration ASB;
- the safe behavior obtained thanks to the resolution of the joint constraints compatibility and the mixable joint deceleration ASB with the SAT.

5.1 Experiments presentation

The experiments were performed in a facility of the French Alternative Energies and Atomic Energy Commission (CEA), a government-funded technological research organization. The 6-DOF arm used in these experiments is a 100daN advanced remote hydraulic manipulator with force feedback capabilities, the Maestro (David et al. 2007), designed by CEA and transferred to Cybernetix.² It is usually used in various applications where remote handling with high strength and dexterity are needed, e.g. in nuclear or offshore hostile environments.

5.1.1 Experimental equipment

The robot's controller uses a generic hard real-time application, TAO2000 (Gicquel et al. 2001), developed by CEA for Computer Aided Teleoperation Systems (teleoperators) and coming from its experience for objects remote manipulation in hazardous environment. It can address both masters and slaves robots, whatever their kinematics and actuation technologies, providing them a whole generic set of useful features with nearly no specific development. This application provides, via a standard Ethernet link, a high level communication interface to control the robot and a low level real-time tuning and spying interface.

The Maestro works in front of a tunnel boring machine cutting wheel mock-up (Fig. 8). At each time step, the operational input sent is a desired velocity issued from a 3-DOF desired point (position only, no orientation). It induces a Degree Of Redundancy (DOR) of 3.

5.1.2 Initial assumptions versus experimental conditions

Despite the work carried out on safety, the assumptions enounced in Sect. 2 induce approximations which may provoke minor incompatibilities. These incompatibilities are localized and do not have a big impact on the robot behavior: as shown on the following results, the envelope needed to absorb them could be small with respect to what would be needed without the compatibility study. However, at the control level, an incompatibility provokes the impossibility to

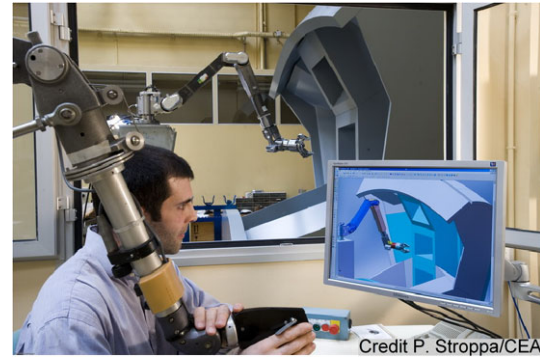


Fig. 8 Teleoperated Maestro operating in front of the tunnel boring machine cutting wheel mockup

solve the problem. For practical reasons, the occurring incompatibilities are *denied* at the control level: for example, if the current position of the joint parameter $q_3(k)$ is inferior to the artificial minimum joint position $q_{m,3}^{art}$, then the inferior joint position limit is taken as the minimum between the current joint position and theoretical minimum joint position: $q_{m,3}$ gets $\min(q_3(k), q_{m,3}^{art})$.

From a practical point of view, the envelopes around the joint position limits e_j and around the environment e_c are unknown from the controller and considered as an origin offset: for example, the controller considers that a collision occurs if the distance between the robot and the environment is lower than e_c .

5.2 Safe behavior with compatible constraints

This first experiment³ illustrates the behavior of a multi-body robot subject to control constraints modified to become compatible.

Task presentation The robot is subject to a brutal fold up from a configuration of extended robot to a configuration in which the robot has reached its joint position limits (Fig. 9). During a first period (5.0 s), the desired operational velocities are maintained constant and maximum toward a point at the left infinite; then, the desired operational velocity brings back the robot toward the initial Cartesian point at lower velocity (the aim is to check that the deceleration toward the joint position limit is safe). The considered e-state constraints are joint position, velocity and acceleration limits ((10)–(15)), treated in case study 2, Sect. 4.3. For the sake of clarity, the e-state constraints limits are the same for each joint: respectively ± 1.0 rad, ± 1.5 rad/s, ± 2.0 rad/s². The acceleration limits have been taken voluntarily low (lower than the robot actual capabilities) in order to better illustrate the results. For this particular experiment, the trajectory is

²<http://www.cybernetix.fr/Hydraulic-arms>.

³<http://www.isir.upmc.fr/UserFiles/File/...VpadoiS/Medias/JointPosLim.avi>.



Fig. 9 Views of the robot in initial position (extended) and at $t = 5.0$ s (fold up). The white arrow is the constant operational desired velocity from $t = 0.0$ s to $t = 5.0$ s

considered in a (z, y) -plane (2 DOF desired velocity) and only 3 DOFs are used (see Fig. 9), which brings the DOR to 1.

Control law The control framework is the one proposed on Fig. 4. The control constraints used to enforce the considered e-state constraints are F_V , F_A and $F_{P'}$ (respectively (20), (21), (23), (24), (26) and (27)), which are proved to be compatible (Sect. 4.3). The control problem is expressed as a QP, and the solver is an efficient open source algorithm.⁴

Given the limits of the robot, the joint position overshoot could reach 0.56 rad without the proposed methodology (by taking (18) and (19) as the joint position control constraints for example). As a benefit of our approach, the envelope retained on joint position limits for this experiment is $e_j = 0.1$ rad.

Results and analysis The results are presented on Fig. 10. The $\dot{q}_{des}$ values are the joint velocity sent to the actuators ($\dot{q}(k+1)$) and $\dot{q}_{actual}$ are the velocities actually carried out by the actuators. Only the 3 DOFs concerned by the planar trajectory are represented (the other are excluded from the model, so they remain static). During the first second, each joint contributes to the operational motion at its best: accelerations are maximal for each joint. Joint 5 is the first to undergo a deceleration (before reaching its maximum velocity) due to the initial proximity to its position limit. Joint 3 reaches its velocity limit for a short time. Joint 2 does not perform high accelerations due to the fact that the operational velocity is sufficiently high thanks to the other joints. At $t = 4.0$ s, the small motion of joint 2 is induced by a disturbance which, given the system configuration, leads to the tracking of the desired Cartesian velocity. At $t = 5.0$ s, the operational desired velocity is inverted (the robot goes back to its initial operational position), and the robot gets away from its boundaries without any difficulty. At the end

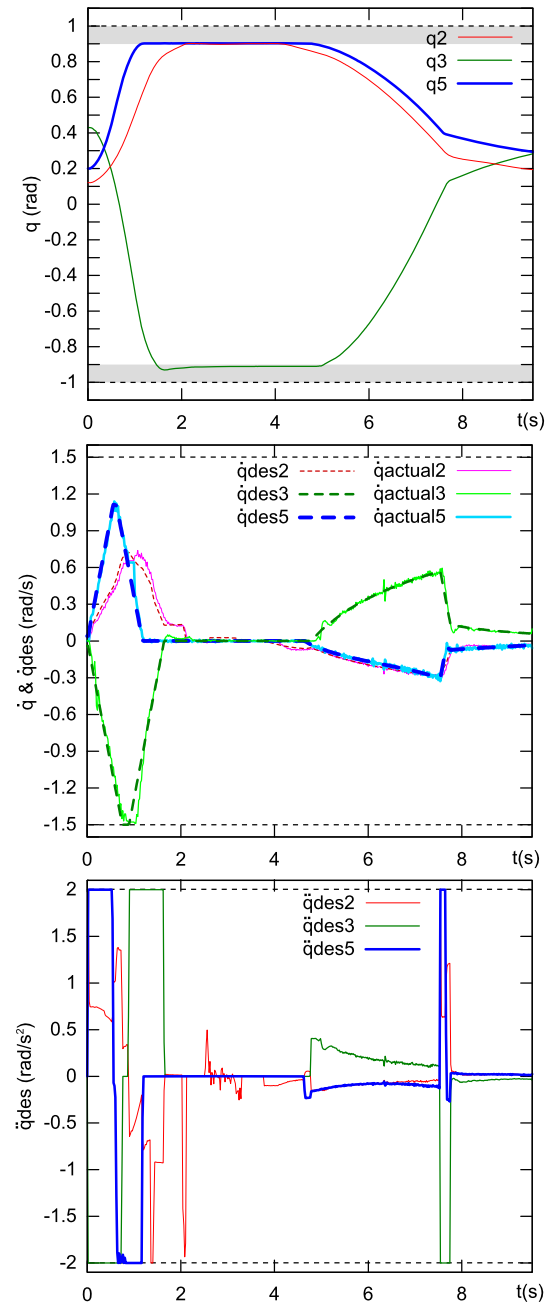


Fig. 10 Position, velocity and acceleration of the joint 2, 3, and 5 during experiment 1. The position is directly measured on the robot, the velocity $\dot{q}_{des}$ is the input sent to actuators and $\dot{q}_{actual}$ is the measured one. The acceleration is computed from $\dot{q}_{des}$. All the variables remain between their limits. The control constraints modification imposes appropriate decelerations to satisfy the joint position limits

of the experiment, ($t = 7.5$ s), the deceleration is provoked by a reduction of the operational desired velocity; it is not provoked by any constraint. The envelope violation of joint 3 occurring at the beginning of the experiment ($t = 1.5$ s) is attributed to the approximations discussed in Sect. 5.1.2 (especially the exact execution of the desired joint input). However, the envelope is hardly violated, which tends to

⁴QuadProg++: <http://sourceforge.net/projects/quadprog/>.

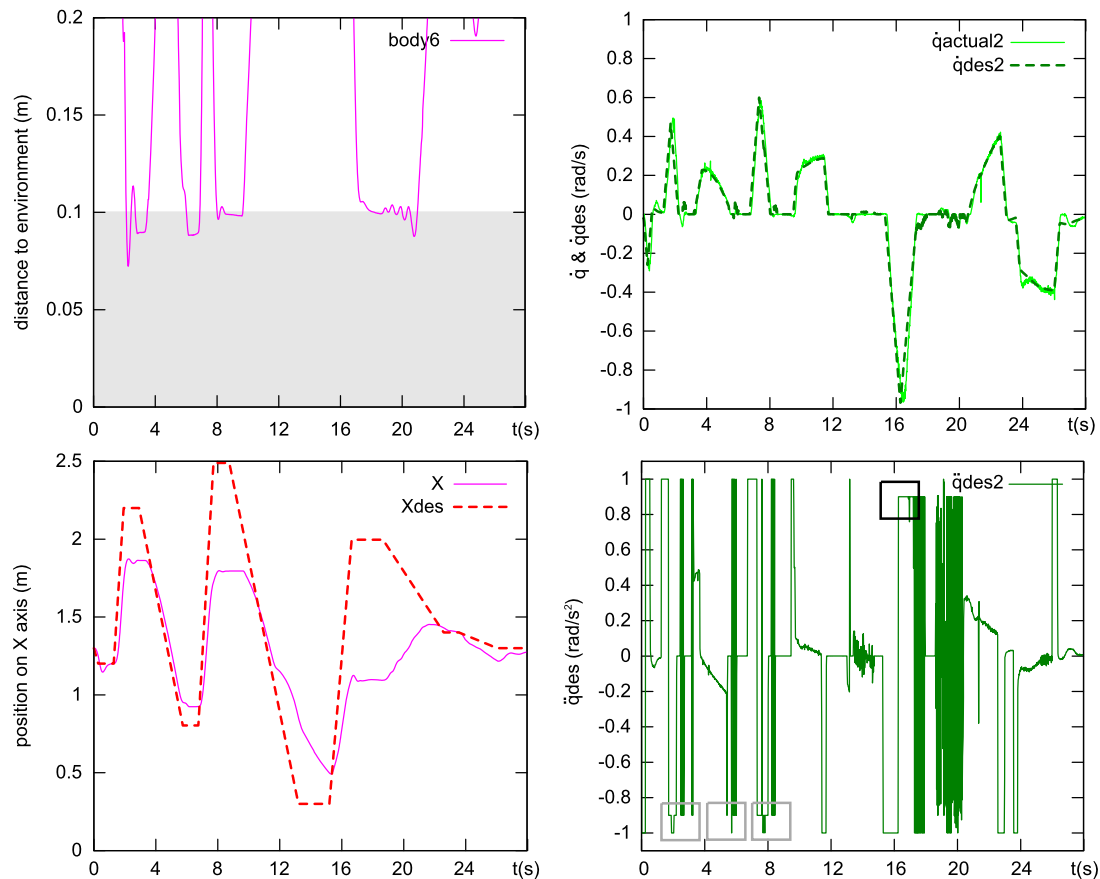


Fig. 11 Results of experiment 2. *Left column*: shortest distance between body 6 and the environment, desired and actual position along Cartesian axis X; *right column*: desired and actual velocity of joint 2, accelerations of joint 2

show that it can reliably be reduced (maximum overshoot is $3.03E^{-2}$ rad).

5.3 Safe behavior with ASB

The second experiment⁵ illustrates the behavior of a multi-body robot subject to incompatible control constraints; at each time step, the computed control input is sent if a consecutive deceleration toward a static e-state is admissible (see Sect. 4.5).

Task presentation The robot is subject to various motions in the cluttered environment of the cutting wheel mock-up (Fig. 8). The desired operational velocity is issued from a 3D trajectory involving unreachable points. The considered e-state constraints are joint position, velocity and acceleration limits and collisions avoidance ((10)–(15) and (17)). The trajectory involves motions close to joint position limits. The joint position limits are ± 1.0 rad. The joint velocity limits are not reached during this experiment; the joint accelerations limits are set to ± 1.0 rad/s². The distances are

computed in real-time using a CAD model of the environment (Fig. 8).

Control law The control law is similar to the previous experiment. To deal with incompatible constraints (joint acceleration limits and collisions avoidance, as shown in Sect. 4.4), the control uses Algorithm 1. To ease the analysis and differentiate accelerations due to the trajectory tracking and accelerations issued from ASB1, the acceleration value for prediction and alternative behavior is lower than the one retained for the control constraint: ± 0.9 rad/s². This modification has no major impact on the results but makes them clearer.

Given the limits of the robot, the joint position overshoot could reach 0.5 rad without the proposed methodology. In the same conditions, given the dimensions of the robot, the potential collision without ASB would have required an envelope ~ 1 m to be avoided. As a benefit of our approach, the envelope retained on joint position limits is $e_j = 0.1$ rad and the envelope around the environment is $e_c = 0.1$ m.

Results and analysis The results are presented on Fig. 11. The 4 motions getting close to obstacles, easily identifiable at $t = 2.0$ s, $t = 5.0$ s, $t = 8.0$ s and $t = 17.0$ s on the graph

⁵<http://www.isir.upmc.fr/UserFiles/File/...VpadoiS/Medias/ObstAvoidASB1.avi>.

of distance to environment, end-up in the envelope e_c . The 3 first motions ($t = 2.0$ s, $t = 5.0$ s and $t = 8.0$ s) get close to obstacles with a reasonable velocity, but as there is no compatibility between collisions avoidance and acceleration limits, resort to the alternative behavior is needed (fair squares). The fourth motion toward obstacles is done at higher speed; the deceleration begins nearly 1.0 s before the impact (dark square). Finally a motion in the neighborhood of the obstacle generates high frequency oscillations on the acceleration ($t = 20.0$ s). Actually, as the robot remains close to the obstacles, a deceleration at the joint level tends to maintain the robot close to the environment. As a result, oscillations between the trajectory tracking and the alternative behavior occur. Thanks to the envelope e_c taken, safety is preserved.

5.4 Integration of smooth avoidance technique into the mixable joint deceleration ASB

This third experiment⁶ illustrates the possibility to introduce usual collisions avoidance methods into a safe framework for a multibody robot. The resulting behavior remains safe and takes full advantage of the avoidance method.

Task presentation The robot is in charge of reaching a setpoint from which it is separated by an infinite horizontal plane (Fig. 12). As in the first experiment, the trajectory is considered in a plane (2 DOF desired velocity) and only 3 DOFs are used (the same as in Fig. 9), which gets the DOR to 1. The magnitude of the desired velocity is maintained constant toward the desired point. At the end of the experiment, a second setpoint is given to get far from the obstacle. The considered e-state constraints are joint position, velocity and acceleration limits and collisions avoidance ((10)–(15) and (17)). The trajectory does not involve motions close to joint position limits. The joint velocity limits are not reached along this motion. The joint accelerations limits are set to ± 1.0 rad/s².

Control law The control law is based on the one of experiment 2, but uses Algorithm 2 instead of Algorithm 1. Actually, the approach used to preserve safety in the previous experiment has the severe drawback to generate oscillations on the accelerations when the robot is moving along obstacles. Actually, the control alternates between the trajectory tracking and the alternative behavior at nearly each time step. As detailed in Sect. 4.5, the control constraint induced by the SAT (36) is added to the set of considered constraints. The following values have been used: $d_i = 0.15$ m (area I), $d_s = 0.07$ m (area II) and the envelope around obstacles is $e_c = 0.05$ m (area III). Taking different values for d_s and e_c eases the interpretation of the results.

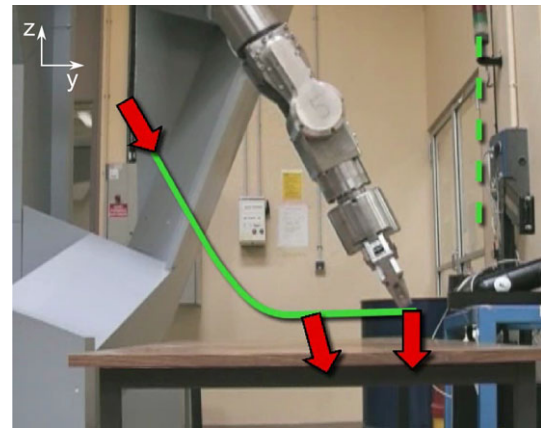


Fig. 12 Views of the robot during the trajectory. The arrows show the desired operational velocity input along the robot trajectory (curve)

Results and analysis The results are presented on Fig. 13. As in experiment 2, the arrival on the obstacle causes the maximum overshoot in the area II. The robot never enters the security envelope (area III) as it is managed by the SAT. The distance to the obstacle stabilizes during the sliding motion (see Fig. 12) until $t = 6.0$ s when another objective is given to the effector. The transition time can be detected on the acceleration (square), when it switches from 1.0 rad/s² (deceleration coming from the alternative behavior) to approximately 0.93 rad/s². At that time ($t = 1.62$ s), the distance to the obstacle is 10.9 cm, and the avoidance method begins to limit the robot motion along direction z . The acceleration is then smooth, the collision management being ensured by the SAT. During the motion along the obstacle (between $t = 2.0$ s and $t = 6.0$ s), the velocity of joint 2 contributes to the motion, but the velocity is small as the setpoint is far under the table, increasing the angle between the desired velocity vector and the infinite plane toward orthogonality.

6 Conclusion

The work presented in this paper exposes a methodology to ensure safety of multibody robots behaviors. Satisfying the constraints at each time step may turn out insufficient because of constraints incompatibilities; as a consequence, to obtain a safe behavior, the control problem can be considered as a problem of constraints formulation. The proposed approach enables to study the compatibility of constraints and establishes the link between constraints compatibility and safety. It also proposes alternatives if constraints compatibility cannot be established.

Case studies illustrate the approach. The constraints expressions are modified to ensure compatibility when possible; if not, the permanent possible resort to a safe behavior

⁶<http://www.isir.upmc.fr/UserFiles/File/...VpadoiS/Medias/ObstAvoidASB2SAT.avi>.

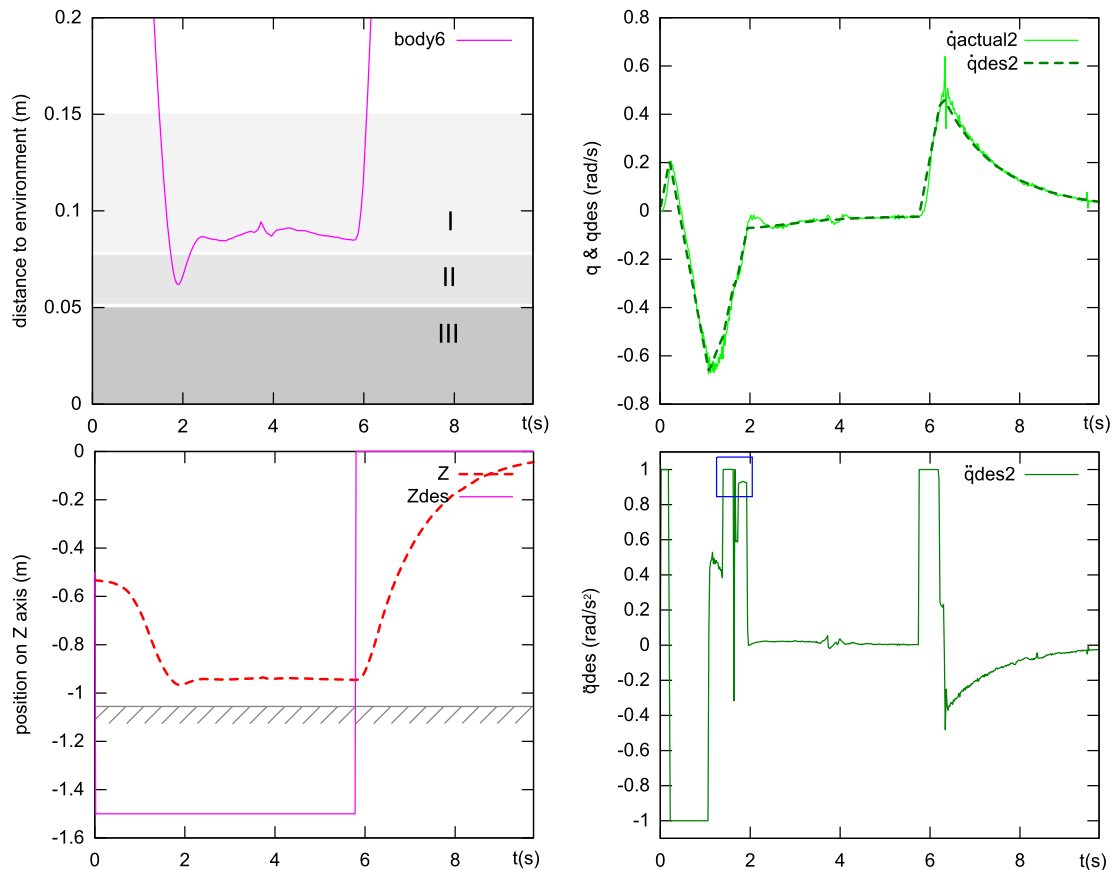


Fig. 13 Results of experiment 3. *Left column*: shortest distance between body 6 and the environment, desired and actual position along Cartesian axis Z; *right column*: desired and actual velocity of joint 2, accelerations of joint 2

is ensured. A particular method is developed to take full advantage of the usual avoidance techniques while maintaining safety. These works have been applied on a 6-DOF manipulator operating in a cluttered environment. The results obtained confirm the reliability of the approach and validates the expected performances.

Future works will address new applications and extensions of the proposed methodology. This approach can be applied to other control levels (e.g. torque control) and include other types of constraints related to physical limits (torque limits, jerk limits, power limits, etc.) or user specifications (contact persistence, comanipulation, etc.). The extensions of the presented methodology include the adaptations of the work carried out on ICS to IVS. The concept of ICS has generated a significant amount of works in the field of mobile robotics: ICS-checker in the 2D case (Martinez-Gomez and Fraichard 2008), solutions to approximate the ICS set (Parthasarathi and Fraichard 2007), probabilistic approaches (Althoff et al. 2010; Bautin et al. 2010), etc. These works offer many perspectives to increase the use of safe multibody robots.

Acknowledgements This work is involved in the *Telemach* project; it has been supported by the French National Research Agency (ANR), Interactive Systems and Robotics Program 2007 (PSIROB07).

References

- Althoff, D., Althoff, M., Wollherr, D., & Buss, M. (2010). Probabilistic collision state checker for crowded environments. In *Proceedings of the 2010 IEEE international conference on robotics and automation* (pp. 1492–1498).
- Bautin, A., Martinez-Gomez, L., & Fraichard, T. (2010). Inevitable collision states: A probabilistic perspective. In *Proceedings of the 2010 IEEE international conference on robotics and automation* (pp. 4022–4027).
- Biagiotti, L., & Melchiorri, C. (2008). *Trajectory planning for automatic machines and robots*. Berlin: Springer.
- Brady, M. (1982). *Robot motion: planning and control*. Cambridge: The MIT Press.
- David, O., Measson, Y., Bidard, C., Rotinat-Libersa, C., & Russotto, F.-X. (2007). Maestro: a hydraulic manipulator for maintenance and decommissioning application. In *Transaction of the European nuclear conference*.
- Decre, W., Smits, R., Bruyninckx, H., & De Schutter, J. (2009). Extending iTaSC to support inequality constraints and non-instantaneous task specification. In *Proceedings of the 2009 IEEE international conference on robotics and automation* (pp. 964–971).

- Escande, A., Mansard, N., & Wieber, P.-B. (2010). Fast resolution of hierarchized inverse kinematics with inequality constraints. In *Proceedings of the 2010 IEEE international conference on robotics and automation* (pp. 3733–3738).
- Faverjon, B., & Tournassoud, P. (1987). A local based approach for path planning of manipulators with a high number of degrees of freedom. In *Proceedings of the 1987 IEEE international conference on robotics and automation* (pp. 1152–1159).
- Fox, D., Burgard, W., & Thrun, S. (1997). The dynamic window approach to collision avoidance. *IEEE Robotics & Automation Magazine*, 4(1), 23–33.
- Fraichard, T. (2007). A short paper about motion safety. In *Proceedings of the 2007 IEEE international conference on robotics and automation* (pp. 1140–1145).
- Fraichard, T., & Asama, H. (2004). Inevitable collision states. A step towards safer robots? *Advanced Robotics*, 18(10), 1001–1024.
- Gicquel, P., Andriot, C., Lauture, F., Measson, Y., & Desbats, P. (2001). TAO2000: a generic control architecture for advanced computer aided teleoperation systems. In *Proceedings of the 9th ANS topical meeting on robotics and remote systems*.
- Guilbert, M., Joly, L., & Wieber, P.-B. (2008). Optimization of complex robot applications under real physical limitations. *The International Journal of Robotics Research*, 27(5), 629–644.
- Haddadin, S., Albu-Schffer, A., Eiberger, O., & Hirzinger, G. (2010). New insights concerning intrinsic joint elasticity for safety. In *Proceedings of the 2010 IEEE-RSJ international conference on intelligent robots and systems* (pp. 2181–2187).
- Ikuta, K., Ishii, H., & Nokata, M. (2003). Safety evaluation method of design and control for human-care robots. *The International Journal of Robotics Research*, 22(5), 281.
- Kanehiro, F., Lamiroux, F., Kanoun, O., Yoshida, E., & Laumond, J.-P. (2008). A local collision avoidance method for non-strictly convex polyhedra. In *Proceedings of robotics: science and systems IV*, Zurich, Switzerland.
- Khatib, O. (1986). Real-time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research*, 5(1), 90–98.
- Khatib, O., Yokoi, K., Brock, O., Chang, K., & Casal, A. (2001). Robots in human environments. *Archives of Control Sciences*, 1.11(3–4), 123–138. Special Issue on Recent Developments in Robotics.
- Kröger, T. (2010). *Springer tracts in advanced robotics: Vol. 58. On-line trajectory generation in robotic systems*. Berlin: Springer.
- Maciejewski, A., & Klein, C. (1985). Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments. *The International Journal of Robotics Research*, 4(3), 109–117.
- Martinez-Gomez, L., & Fraichard, T. (2008). An efficient and generic 2D inevitable collision state-checker. In *Proceedings of the 2008 IEEE-RSJ international conference on intelligent robots and systems* (pp. 234–241).
- Martinez-Gomez, L., & Fraichard, T. (2009). Collision avoidance in dynamic environments: an ics-based solution and its comparative evaluation. In *Proceedings of the 2009 IEEE international conference on robotics and automation* (pp. 100–105).
- Park, J., & Khatib, O. (2008). Robot multiple contact control. *Robotica*, 26(5), 667–677.
- Parthasarathi, R., & Fraichard, T. (2007). An inevitable collision state-checker for a car-like vehicle. In *Proceedings of the 2007 IEEE international conference on robotics and automation* (pp. 3068–3073).
- Rubrecht, S., Padois, V., Bidaud, P., & Broissia, M. (2010a). Constraint compliant control for a redundant manipulator in a cluttered environment. In *Advances in Robot Kinematics: Motion in Man and Machine* (pp. 367–376).

Rubrecht, S., Padois, V., Bidaud, P., & de Broissia, M. (2010b). Constraints Compliant Control: constraints compatibility and the displaced configuration approach. In *Proceedings of the 2010 IEEE-RSJ international conference on intelligent robots and systems* (pp. 677–684).

Zinn, M., Khatib, O., Roth, B., & Salisbury, J. (2004). Playing it safe. *IEEE Robotics & Automation Magazine*, 11(2), 12–21.



belongs to the R&D department of Bouygues Travaux Publics, where his activities focus on robotics in construction.



RTE/UPMC Chair of “Robotics Systems for field intervention in constrained environments”. His research activities mainly focus on the design, modeling and control of redundant and complex systems such as wheeled mobile manipulators and humanoid robots. He is also interested in the combination of model-based control theory in Robotics with incremental model learning techniques.



Scientifique). He became full Professor at UPMC in 1998. Since 1981,

Sébastien Rubrecht graduated from Ecole Nationale Supérieure d’Arts et Métiers (ENSAM, France) and received his master’s degree in Robotics from University Pierre et Marie Curie (UPMC—Paris, France) in 2007. He received his Ph.D. in Robotics at Institut des Systèmes Intelligents et de Robotique (ISIR) at UPMC in 2011 where he was co-financed by Bouygues Travaux Publics. His research interests include task based design with evolutionary algorithms and reactive control of manipulators. He

Vincent Padois is an Associate Professor of Robotics and Computer Science and a member of the Institut des Systèmes Intelligents et de Robotique (ISIR) at University Pierre et Marie Curie (UPMC—Paris, France). He received his master’s degree in 2001 and his Ph.D. in 2005 both in Automatic Control from the Institut National Polytechnique de Toulouse, France. In 2006 and 2007, he was a post-doctoral fellow in the group of Professor O. Khatib at Stanford University. Since 2011, he holds the

Philippe Bidaud is the director of the Institut des Systèmes Intelligents et de Robotique at University Pierre et Marie Curie (UPMC—Paris, France). He received his PhD degree in Mechanical Eng./Robotics from University of Poitiers (France) in 1984 and the habilitation for directing research from UPMC in 1996 for his work on “Design and Control of Complex Robotics Systems”. For 15 years he was as a researcher in the section of Physical Sciences for Engineers at CNRS (Centre National de la Recherche

his research interests are the field of the design of complex robotics systems, articulated hands, locomotion systems, assembly systems, haptic devices from the view point of modeling and analysis of mechanical systems and of their associated control. He also works in complex interaction control and modeling, control of smart materials, micro-robotics systems, design and optimization of compliant structures, and high mobility systems. In 1997–98, Philippe Bidaud was a visiting Professor at the Field and Space Laboratory of the Department of Mechanical Engineering at MIT. He has published more than 150 papers including contributions to books, journals, international conferences. He holds several patents and has been very active in technology transfer and valorisation through the Centre de Robotique Intégrée Ile de France.



Michel de Broissia has obtained a first engineering degree in France before carrying out graduate studies in Canada where he acquired a Ph.D. in applied sciences in 1977. He was then appointed to a professor at the Chemical Engineering department of the Sherbrooke University where he was teaching Computer Science and numerical analysis. His research activities were oriented to environmental topics: pollutant dispersion in air and water, oil slick motion and free surface modeling. He was an associate professor

when he left the Sherbrooke University in 1984. He then entered the Bouygues Group where he joined an R&D structure mostly involved to large scale civil engineering projects. His main activities concerned the following topics: CAD solutions adapted to civil engineering, Tunnel boring machine guidance, Development of specific instrumentation, Robotics in construction. He participated and coordinated several research projects within the European R&D framework and French research programs. He retired in January 2011.



Max Da Silva Simoes obtained both his Electrical engineering and his Automatics and Applied Computer Science degrees in 1996. He has been working as consultant for more than 10 years in industry on embedded systems, with an expertise in real-time SW, in fields such as Automotive, Robotics or Defense. During this period he practiced for 4 years system engineering, translating customers needs into systems requirements. Since 2008, he is involved at CEA in activities related to conception of “Safe” controllers, supervision of remote handling with 3D models, control and modeling of robots.

B.3 Control of robots sharing their workspace with humans: an energetic approach to safety

Control of robots sharing their workspace with humans: an energetic approach to safety

Anis Meguenani¹, Vincent Padois¹ and Philippe Bidaud^{1,2}

Abstract— In this paper, we propose a physically meaningful energy-related safety indicator for robots sharing their workspace with humans. Based on this indicator, a safety criterion accounting for the breaking capabilities of the robot is included as a quadratic constraint in the control algorithm. This constraint is modulated by the distance between the human operator and the end-effector of the robot. The control algorithm is formulated as an optimization problem and computes the actuation torque of a robotic manipulator given some task to be performed and physical constraints to respect. The overall framework is validated in a physics simulation software on a Kuka LWR4 and different behaviours of the robot towards a considered obstacle in its environment are evaluated and discussed.

I. INTRODUCTION

Service and intervention robotics, as well as more traditional industrial robotics applications, are evolving in a direction where the workspace of the robot is very likely to be shared with humans. This may induce deliberate¹ and non-intentional physical interactions. Safety in this context becomes a critical issue to be dealt with [1].

To ensure safe human-robot interactions, several approaches have been explored in the robotics literature. At the hardware level, the mechanical design can be optimized to reduce the apparent inertia of the robot [2] and compliant components can be introduced to allow smoother contacts and less severe impacts [3]. Torque sensing at the joint level also provides a way to actively control the impedance of the robot. The Kuka-DLR lightweight robot [4], [5], [6] has been specifically designed to meet these challenges.

Different control approaches using internal and external force/torque sensors have been developed to handle safety during pre and post impact/contact phases [7], [8], [9]. Haddadin in [10] and De Luca in [11] present different strategies to reduce the effect of undesired impacts. A collision detection parameter based on the estimated external torque is introduced and used to scale down the link inertia obtaining a “lighter” robot that “flees” from the collision area. An other strategy is the use of the disturbance input to slow the robot until zero velocity then pushing it back along its original path. Heizmann and Zelinsky in [12] propose a



Fig. 1. View of a user sharing its workspace with the KUKA LWR manipulator. The kinetic energy of the system is modulated as a function of the distance between the human operator and the end-effector of the robot in order to best perform the task of the robot while ensuring safety.

safety criterion based on the *potential impact force* to filter the control torque of the system. The introduced controller scheme allows one to consider two potential contact points at the same time for a real-time implementation. As the degree of potential injury is directly related to the mass and velocity of the colliding objects, the controller proposed in [13] takes into account the reflected robot inertia along a collision direction to decide about the maximum operational point velocity. The bounds on this velocity are based on experimental results relating mass, velocity, geometry and medically observable soft tissue injury by systematic drop-testing experiments with pig abdominal wall sample. By making use of the redundancy property of a KUKA/DLR lightweight arm, [14] proposes a physical interaction strategy that is able to react safely to collisions while continuing to execute as much possible of the original task.

Kinetic energy has already been discussed in [10] and [13] as a good representation of the risk of injury. It is used in the work presented in this paper to synthesize a physically meaningful safety indicator. This indicator can also include elastic potential energy associated with the controller in phases where the robots physically interacts with its environment. The kinetic energy part of the proposed criterion is used to constrain the dynamic behaviour of a Kuka LWR4 serial robot in the direction of a considered obstacle². The imposed constraint accounts for the breaking

¹Anis Meguenani, Vincent Padois and Philippe Bidaud are with:
-Sorbonne Université, UPMC Univ Paris 06, UMR 7222, Institut des Systèmes Intelligents et de Robotique, F-75005, Paris, France
-CNRS Centre National de la Recherche Scientifique, UMR 7222, Institut des Systèmes Intelligents et de Robotique, F-75005, Paris, France
Email:{meguenani,bidaud,padois}@isir.upmc.fr
²Philippe Bidaud is with ONERA, 91123 Palaiseau, France
Email:philippe.bidaud@onera.fr

¹For example if the robot can be both used in an autonomous mode or in a comanipulation mode.

²All along the paper, “obstacle” is used as a generic term for any external element of the environment, *e. g.* a human operator.

capabilities of the robot and is modulated as a function of the distance between the robot and the human operator.

In order to properly account for the safety constraint, the control problem is expressed as a Quadratically Constrained Quadratic Program (QCQP) [15]. The computation of the adequate actuation torque needed to perform a trajectory tracking in operational space is subject to several linear inequality constraints accounting for the physical limitations of the robot (joint limits, joint velocity and torque saturations) as well as for a limit value on the quadratic, energy-based safety indicator. The proposed control framework is expected to decrease impact forces due to collisions by constraining the kinetic energy of the robot while contact forces induced by deliberate physical interactions can be limited through some constraint on the elastic potential energy. Using the same framework, contact with the environment can be enabled, modulated and disabled by a straightforward modification of physically meaningful control parameters. Fig. 1 illustrates a typical workspace-sharing scenario for the proposed controller.

This paper is organised as follows. In section II, the proposed safety indicator and associated safety criterion are formulated and expressed as a function of the control input of the system, *i.e.* the actuation torque. In Section III, the controller is derived: tasks related objectives are formulated and the expression of the inequality constraints acting on the system is provided. In Section IV, an experimental scenario is introduced based on which the possibilities offered by the proposed controller are illustrated and discussed in several cases in simulation. Finally, Section V summarizes the contribution and provides an overview of the future work.

II. SAFETY CRITERION

In this section, a safety indicator quantifying the degree of danger (risk induced by a collision) represented by the robot towards a nearby human operator is introduced. This indicator has to be physically meaningful, related to the control input and computable in real-time.

During a collision phase, the risk of injury for a human operator depends mainly on the shape of the robot and on the generated impact force. For a given shape, to ensure safety, an indicator whose value is related to the impact force is proposed. A safety criterion, namely a bound on the maximum value of the safety indicator, is then derived.

A. Energy dissipation model and safety indicator

The generated impact force during a collision phase can be written as a function of the dissipated energy and the shock absorption distance:

$$\begin{aligned} \int_u F_{impact} du &= E_{dissipated} \\ &= E_c^{hum} + E_c^{rob} + E_p^{hum} + E_p^{rob}, \end{aligned} \quad (1)$$

With F_{impact} the generated impact force during the collision, u the shock absorption distance and $E_{dissipated}$ the dissipated energy which is equal to the sum of the kinetic

E_c and potential E_p energy of both of the human operator and the robot.

On the one hand, the left side parameters of the shock absorption equation (1) are not directly related to the actuation torque. Moreover, it is impossible to have an accurate model of the human body-robot impedance³. As a matter of fact, the use of the impact force or of the shock absorption distance as a safety indicator is neither desirable nor possible. On the other hand, the dissipated energy is closely related to the impact force and can be directly related to the actuation torque and thus controlled in order to reduce the impact of a collision.

At a given time, very few assumptions can be made on the current level of energy of the human operator and on its future evolution. As a consequence, the retained safety indicator S is robot-centered:

$$\begin{aligned} S &= E_c^{ij} + E_p^{ij} \\ &= \frac{1}{2} m(\mathbf{q})_{ij}^{eq} v_{i/j}^2 + \frac{1}{2} K(\mathbf{q})_{ij}^{eq} e^T e, \end{aligned} \quad (2)$$

where $1/m(\mathbf{q})_{ij}^{eq} = J(\mathbf{q})_C^{i,j} M(\mathbf{q})^{-1} J(\mathbf{q})_C^{i,jT}$. $m(\mathbf{q})_{ij}^{eq}$ is the equivalent mass of the robot segment i in the direction of obstacle j expressed in the cartesian space [16]. $M(\mathbf{q})$ is the joint space inertia matrix of the robot and $\mathbf{q}$ its joint space configuration. $v_{i/j} = J(\mathbf{q})_C^{i,j} \dot{\mathbf{q}}$ is the relative velocity of the closest point C belonging to the robot segment i in the direction of obstacle j , with respect to obstacle j . $J(\mathbf{q})_C^{i,j}$ is the Jacobian of the robot segment i expressed at C and projected along the distance vector towards obstacle j . $K(\mathbf{q})_{ij}^{eq}$ is the equivalent controller stiffness⁴ at point C projected along the distance vector towards obstacle j . When in contact, e is the error induced by the contact on the position and orientation of point C . This error is 0 when there is no contact.

To ensure safety for both the robot and any nearby obstacle, the introduced indicator has to be considered for each (robot segment i , considered obstacle j) pair, *i.e.* for n_o obstacles and a robot composed of n_b mobile bodies, $n_o \times n_b$ safety indicators. Within the framework of this paper, and without loss of generality, a single obstacle O is considered and the only mobile body of the robot considered for safety is the end-effector (EE). Indeed, it is the last segment of the fixed base serial robot (Kuka LWR4) that holds the practical load and consequently deploys the maximum kinetic energy. Also, at this stage of the work, only kinetic energy is considered. The safety indicator can thus be written:

$$\begin{aligned} S &= E_c^{EE,O} \\ &= \frac{1}{2} m(\mathbf{q})^{eq} v^2, \end{aligned} \quad (3)$$

where $m(\mathbf{q})^{eq} = m(\mathbf{q})_{EE,O}^{eq}$ and $v = v_{EE/O}$. This indicator represents the energy that would have to be dissipated by the

³This model would have to be individual and body-part specific.

⁴In this work, the robot is supposed to be rigid with respect to the controller stiffness capabilities but it would be possible to integrate the compliance of the robot in the safety indicator if, for example, series elastic actuator were used.

end-effector of the robot and the human operator in case of an immediate collision.

B. Safety limit value

Given E_{limit} , some limit value of the energy that can be dissipated by a human and the robot during an impact, the safety criterion can be written $S \leq E_{limit}$ and must always be satisfied. Given the nature of S , such a constraint if imposed at the control level, may have two consequences: a limitation of the velocity of the end-effector in the direction of the obstacle and a modification of its apparent mass in the same direction. However, when no human operator is present at a close distance from the robot, it is not necessary to saturate the developed kinetic energy. E_{limit} should then depend on the amount of kinetic energy that is considered to be *safe* just before the occurrence of a contact/collision but also be a function f of the distance d between the end-effector of the robot and the considered human operator:

$$S \leq E_{limit} = E_{safe} + f(d). \quad (4)$$

The value of E_{safe} depends on the nature of the obstacle and of the tool carried by the end-effector. It also depends on the nature of the interaction that can be allowed between the robot and its surrounding environment. Thus, if any contact between the robot and the considered obstacle is forbidden: $E_{safe} = 0 \rightarrow v = 0$. When contact is allowed: $E_{safe} > 0$ is the maximum value of the kinetic energy allowed for the robot just before the collision. $f(d)$ is a weighting function depending on the distance d between the end-effector and the considered obstacle.

Based on the previous statements, three working zones, illustrated on Figure 2, are defined for the dynamic behaviour of the robot:

- 1) a safe zone for $d < d_{safe}$ in which the kinetic energy must be lower than E_{safe} ;
- 2) a working zone for $d_{safe} < d < d_{max}$ where the kinetic energy is constrained when the robot is moving toward the obstacle;
- 3) and a third zone for $d > d_{max}$ in which maximum dynamic performances are allowed for the system.

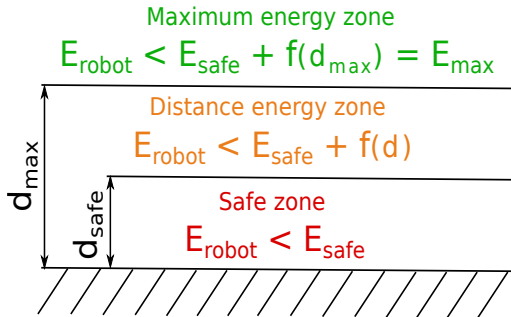


Fig. 2. Energy zones for the dynamic behaviour of the robot.

In the case where the considered obstacle is approaching the robot, the later must be able to develop sufficient breaking

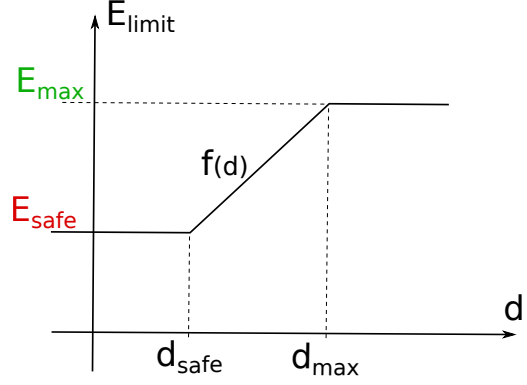


Fig. 3. Evolution of the kinetic energy constraint depending on the distance d between the end-effector and the obstacle.

capacities to satisfy the imposed constraint on the kinetic energy. The weighting function $f(d)$ must therefore account for the dynamics of the robot at every time-step. From the Work-Energy theorem, the amount of work exerted on the robot during the breaking phase is equal to the variation of its kinetic energy. Moreover, this work can be expressed as a product between the *equivalent breaking force* F_{eq} applied on the end-effector and the breaking distance:

$$\begin{aligned} W &= \Delta E_c \\ &= F_{eq}(d - d_{safe}) \\ &= E_{limit}(d) - E_{limit}(d_{safe}) \\ &= f(d) - f(d_{safe}). \end{aligned} \quad (5)$$

The term f represents the maximum energy that can be dissipated during the breaking phase. By choosing this function to be linear inside the distance energy working zone (see Figure 3), it can be written:

$$f(d) = k(d - d_{safe}). \quad (6)$$

The slope coefficient k of the weighting function $f(d)$ represents the equivalent breaking force applied on the end-effector in the direction of the obstacle. It depends on the available breaking torques $\tau_{breaking}$ and the Jacobian of the end-effector in direction of the considered obstacle $J(q)_C$:

$$\tau_{breaking} = J(q)_C^T k. \quad (7)$$

Sufficient breaking capacities have to be guaranteed over the distance d . However, $J(q)_C$ can only be considered constant locally and k is thus a function of the future configurations of the robot. Given the non linear nature of robotic manipulators, predicting the evolution of k is a complex problem. In the worst cases, this value is very close to 0 and to ensure safety E_{limit} should always be equal to E_{safe} , strongly limiting the dynamic performances of the robot when $d < d_{max}$. Given the global objectives of this work, this is not satisfactory and an average value of k (> 0) is considered all over the workspace of the robot. As demonstrated in the work of Rubrecht *et al.* [17] this is a reasonable working assumption as safe alternative behaviours can be constructed on-line based on the knowledge of the

joint space breaking capabilities which are constant and can be guaranteed over an infinite time horizon.

C. Safety criterion extension

The safety criterion previously introduced considers the squared relative velocity between the end-effector and a nearby obstacle. Thus, there is no differentiation between the case where the robot is going towards the obstacle and where it is moving away from it. In a forbidden contact situation ($E_{safe} = 0$), $v = 0$ is imposed which forbids the robot from going towards the obstacle but also from moving away from it. To avoid constraining the motion of the robot in the opposite direction of the obstacle, the safety indicator can be signed:

$$S = \frac{1}{2} \text{sign}(v) m(\mathbf{q})^{eq} v^2, \quad (8)$$

with $\text{sign}(v) = 1$ when the end-effector is getting closer to the considered obstacle.

The safety criterion is thus finally written:

$$S \leq E_{safe} + k(d - d_{safe}), \quad (9)$$

with S defined by (8).

III. SAFE DYNAMIC CONTROLLER

In this section a dynamic control strategy that ensures safety for both of the human operator and the robot is proposed. The objective is to compute the control torque τ in order to perform a trajectory tracking task while respecting a number of constraints at every time-step:

- respect the introduced safety criterion to prevent damaging collisions,
- respect the physical limits of the system.

A. Task formulation

The objective function of the controller is defined as an error function to be minimized. It could be for example an acceleration task if the robot has to perform a trajectory tracking, or a wrench task if the wrench applied on the environment has to be controlled.

In this work, a trajectory tracking performance is considered. A cartesian acceleration task is then defined as an error between the expected acceleration $\ddot{\mathbf{X}}^c$ and the real acceleration $\ddot{\mathbf{X}}$ of the robot-end effector. Considering $\ddot{\mathbf{X}} = J(\mathbf{q})\ddot{\mathbf{q}} + \dot{J}(\mathbf{q})\dot{\mathbf{q}}$ (where $J(\mathbf{q})$ is the Jacobian of the end-effector), it can be written as function of the control input using the equation of motion of the system:

$$\ddot{\mathbf{X}} = J(\mathbf{q})M(\mathbf{q})^{-1}(\tau - \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}})) + \dot{J}(\mathbf{q})\dot{\mathbf{q}}, \quad (10)$$

where $\mathbf{b}(\mathbf{q}, \dot{\mathbf{q}})$ are the non linear terms of the equation of motion, namely gravity, Coriolis and centrifugal induced generalized forces. $\ddot{\mathbf{X}}^c$ can be computed with a PD controller with feed-forward term in order to track some desired trajectory $\mathbf{X}(t)^*$. The acceleration task function to minimize can then be written:

$$g(\tau, \ddot{\mathbf{X}}^c) = \ddot{\mathbf{X}}^c - (J(\mathbf{q})M(\mathbf{q})^{-1}(\tau - \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}})) + \dot{J}(\mathbf{q})\dot{\mathbf{q}}). \quad (11)$$

B. Constraints formulation

The physical limits of the system have to be accounted for when solving the control problem. The computed control input $\tau_{|k}$ at instant k must be such that these limits are not violated at the next time step $k + 1$. They can naturally be written as inequality constraints:

$$\begin{cases} \mathbf{q}_{min} \leq \mathbf{q}_{|k+1} \leq \mathbf{q}_{max}, \\ \dot{\mathbf{q}}_{min} \leq \dot{\mathbf{q}}_{|k+1} \leq \dot{\mathbf{q}}_{max}, \\ \tau_{min} \leq \tau_{|k} \leq \tau_{max}. \end{cases} \quad (12)$$

To be easily accounted for, these constraints have to be expressed as a function of the control variable $\tau_{|k}$. This can be done based on the state at instant k and on a local discrete linear approximation of the behaviour of the robot in joint space with a time step δt :

$$\begin{cases} \ddot{\mathbf{q}}_{|k} = M(\mathbf{q}_{|k})^{-1}(\tau_{|k} - \mathbf{b}(\mathbf{q}_{|k}, \dot{\mathbf{q}}_{|k})), \\ \dot{\mathbf{q}}_{|k+1} = \dot{\mathbf{q}}_{|k} + \delta t \ddot{\mathbf{q}}_{|k}, \\ \mathbf{q}_{|k+1} = \mathbf{q}_{|k} + \delta t \dot{\mathbf{q}}_{|k} + \frac{\delta t^2}{2} \ddot{\mathbf{q}}_{|k}. \end{cases} \quad (13)$$

In an equivalent way, the safety indicator $S_{|k+1}$ can be expressed as a function of the control variable $\tau_{|k}$. Expressing v as a function of the joint space velocity: $v = J(\mathbf{q})_C \dot{\mathbf{q}}$, $v_{|k+1}$ is given by:

$$v_{|k+1} = J(\mathbf{q})_C (\dot{\mathbf{q}}_{|k} + \delta t \ddot{\mathbf{q}}_{|k}). \quad (14)$$

From (14), the quadratic constraint related to the safety criterion is written:

$$\frac{1}{2} \text{sign}(v_{|k+1}) m(\mathbf{q}_{|k})^{eq} v_{|k+1}^2 \leq E_{safe} + k(d - d_{safe}). \quad (15)$$

C. Controller formulation

The proposed control strategy computes the control torque by minimizing the norm of the cartesian acceleration task function expressed in the following quadratic form:

$$\arg \min_{\tau} \left\| g(\tau, \ddot{\mathbf{X}}^c) \right\|_{Q_t}^2 + \epsilon \|\tau\|_{Q_r}^2, \quad (16)$$

subject to (12) and (15).

Q_t and Q_r are positive semidefinite weighting matrices and $\|a\|_Q$ is the Q -weighted euclidean norm of a . $\epsilon \|\tau\|_{Q_r}^2$ with $\epsilon \ll 1$ serves as a regularization task in order to ensure the uniqueness of the control solution and minimize the norm of the computed control torque. It can be shown that the quadratic forms composing the tasks and constraints expression (16), (12) and (15) can be written as functions of positive semidefinite matrices. This QCQP optimization problem is thus convex and admits a unique global minimum.

IV. EXPERIMENTAL RESULTS

The controller described in Section III is implemented as a C++ Orocos component [16] on a virtual model of the Kuka LWR4 serial robot using XDE, a robotics-oriented physics simulation engine [18].

In this section, different behaviours that can be induced using different values of the algorithm parameters are presented and discussed. First, a test case scenario used as

a basis for all the different controller configurations is presented. An obstacle is introduced in the workspace of the robot and different interaction modes are simulated. Non physical interactions and collision tests are performed with and without a kinetic energy constraint on the robot end effector in the direction of the considered obstacle.

A. Test case scenario

As a main activity, the robot performs a repetitive pick and place movement where it tracks a desired position and orientation in the cartesian space (see Fig. 4). The controller is implemented without any constraint on the kinetic energy and the QP problem is solved at every time-step to compute the needed control torque. The QP is solved in real time using Gurobi, a commercial optimization software [19].

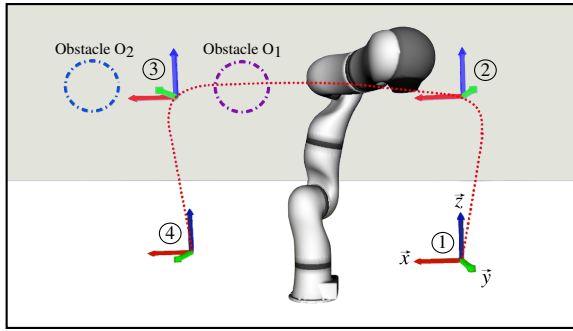


Fig. 4. Kuka LWR4 serial robot within the XDE simulator near its considered obstacle. Case O_1 is when the obstacle intersects with the robot trajectory. Case O_2 is when the obstacle is nearby the robot but does not intersect with its trajectory

The controller described by (16) is implemented only with the linear constraints on the physical limitations of the system. The robots movement is then as dynamic as possible and the pick and place task is performed with the maximum needed kinetic energy to satisfy the desired X^* , $\dot{X}^*$ and $\ddot{X}^*$. The kinetic energy of the end-effector in the direction of the nearby considered obstacle (case O_2 in Fig. 4) is shown in Fig. 5.a.

The maximum tracking errors in the cartesian space are $5 \times 10^{-3} m$ for the position and $2.3 \times 10^{-2} rad$ for orientation. One of the main advantages of using a QP to compute the robot control torque is the possibility to take the physical constraints of the system into account. From Fig. 6 it can be seen that the limits on the articular position, velocity and torque are respected whenever the robot reaches the considered constraints.

B. Obstacle intersecting with the robot trajectory and no constraint on the kinetic energy

In this scenario, the obstacle intersects with the ②-③ segment of the pick and place movement trajectory (case O_1 in Fig. 4). When a collision occurs between the robot and the rigid object, most of the kinetic energy is dissipated. Fig. 7 shows the dissipation of the kinetic energy of the end-effector during a collision phase with the considered obstacle.

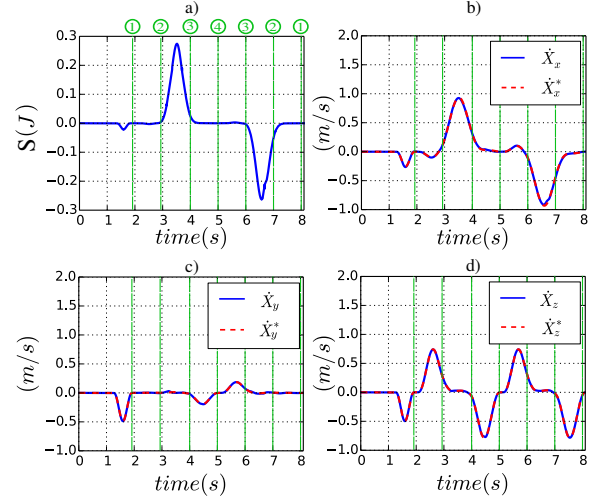


Fig. 5. a): Unconstrained kinetic energy of the end-effector in the direction of a nearby obstacle (case O_2 in Fig. 4). b), c) and d): Velocity performance for the pick and place movement nearby the considered obstacle.

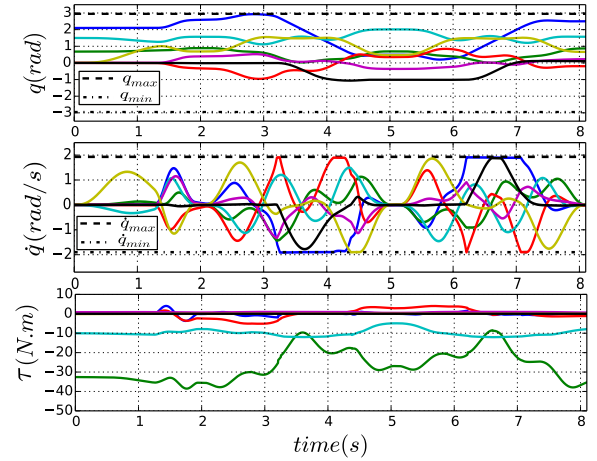


Fig. 6. Articular positions, velocity and torque of the pick and place movement without constraint on the kinetic energy of the end-effector.

According to (1) this fast dissipation of the kinetic energy induces a large impact force. This force can generate damages to the objects during the collision phase. Thus, the controller can be considered unsafe.

C. Nearby obstacle and constraint on the kinetic energy

In this case, a constraint on the kinetic energy of the end-effector is added to safely account for the presence of a considered obstacle. This constraint limits the actuation torque and, accordingly to (8), has a direct impact on the velocity of the end-effector. Depending on how the controller parameters values d_{safe} , E_{safe} and k are chosen, physical contact can be enabled or disabled.

1) *Obstacle not intersecting the robot trajectory:* In this scenario, the obstacle does not intersect with the path of the pick and place movement and the controller parameters are chosen as $E_{safe} = 0.01 J$, $k = 0.44 N.m$, $d_{safe} = 0.8 m$ and $d_{max} = 1.5 m$.

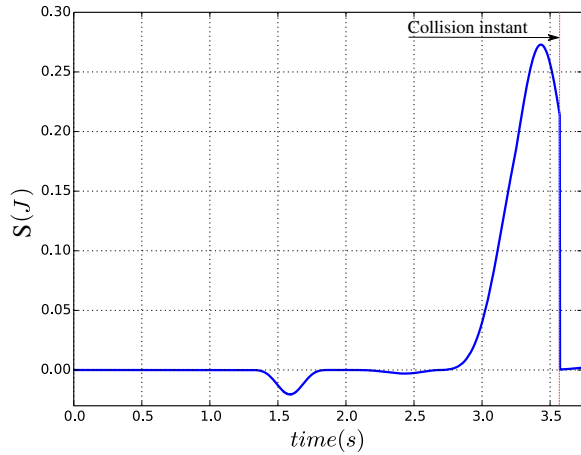


Fig. 7. Dissipation of the unconstrained kinetic energy of the robot end-effector in the direction of the considered obstacle during a collision phase.

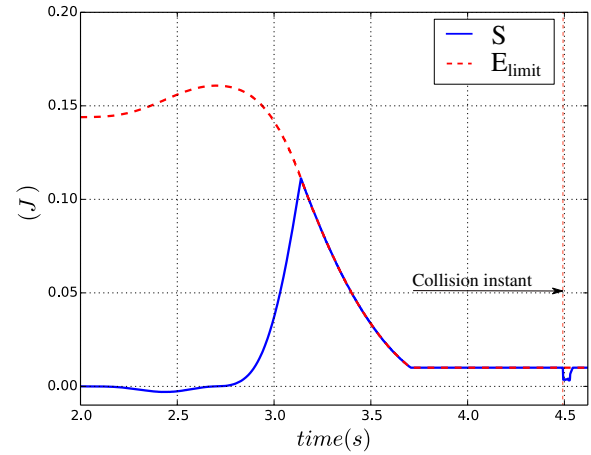


Fig. 9. Dissipation of the constrained kinetic energy of the end-effector in the direction of the considered obstacle during a collision phase.

In this particular case, the robot succeeds in achieving the pick and place movement but with a diminished dynamic performance compared to the unconstrained kinetic energy behaviour (see Fig. 8). Indeed, the constraint on the kinetic energy in the direction of the obstacle directly influences the velocity and the apparent inertia of the robot end-effector (2).

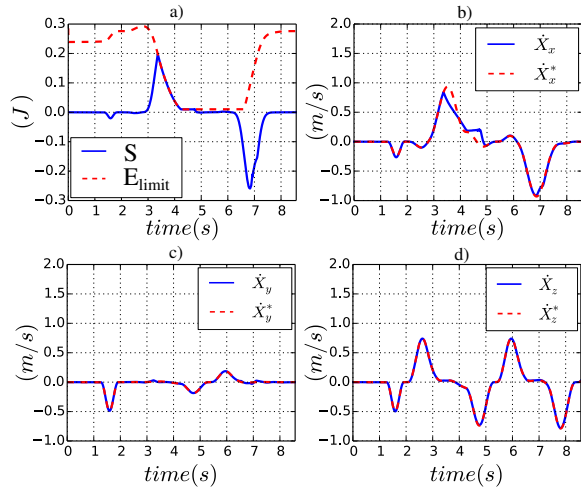


Fig. 8. a): Constrained kinetic energy of the end-effector in the direction of a nearby obstacle (case O_2 in Fig. 4). b), c) and d): Influence of the constrained kinetic energy on the velocity performance for the pick and place movement nearby the considered obstacle.

The constraint on the kinetic energy of the end-effector in the direction of the considered obstacle (Fig. 8.a) is respected at every time-step and a drop in the velocity can be observed in the $\dot{X}_x$ component (Fig. 8.b).

2) *Obstacle intersecting the robot movement:* In this scenario, the obstacle intersects with the ②-③ segment of the pick and place movement trajectory and the controller parameters are taken as $E_{safe} = 0.01 J$, $k = 0.44 N.m$, $d_{safe} = 0.8 m$ and $d_{max} = 1.5 m$. The kinetic energy of the end-effector during the collision phase is shown in Fig. 9.

The kinetic energy profiles of the two collision phases in Fig. 9 and 7 show the benefit of using the safety criterion introduced in (3). Indeed, the dissipated energy when the kinetic energy of the end-effector is initially constrained is less than the dissipation without any constraint. This particular property of the presented controller allows safer physical interactions between the robot and its environment.

3) *Stopping behaviour when the obstacle intersects with the trajectory of the robot:* An other behaviour that can be induced using the same controller with different parameters values is the collision avoidance performance. Indeed, specifying $E_{safe} = 0 J$ at a desired distance d_{safe} will force the robot to stop and prevents it from getting in contact with the considered obstacle (case O_1 in Fig. 4). Fig. 10 shows the robot stopping performance with the following parameters values $E_{safe} = 0 J$, $k = 0.44 N.m$, $d_{safe} = 0.2 m$ and $d_{max} = 1.5 m$. The end-effector reaches exactly the desired kinetic energy at the desired distance from the considered obstacle.

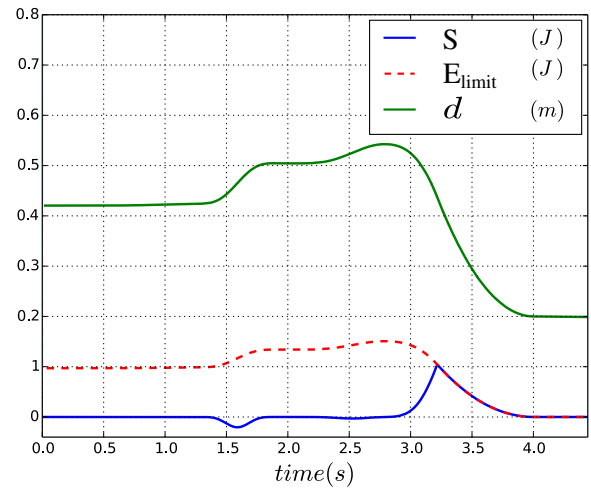


Fig. 10. Distance between the end-effector and the considered obstacle for a collision avoidance behaviour

V. CONCLUSION

The energy based safety indicator proposed and validated in this paper holds a great potential for human/robot collaboration tasks. Indeed, energy is a universal component that can describe several physical phenomena linked to the physical interaction process. Velocity, inertia and also contact forces can all be expressed and modulated with this same quantity. Using the presented control framework and the introduced energy based criterion, the robot has been proven capable of producing different behaviours towards a nearby considered obstacle just by acting on physically meaningful control parameters. During its motion, at every time-step, the kinetic energy of the end-effector is controlled. If a collision occurs or contact with the environment is desired, the dissipated energy is modulated to smooth the interaction process and guarantee safety for both the robot and the obstacle. Enabling/disabling contact and stopping the robot at a desired distance from the obstacle are different behaviours that can be obtained using the same controller.

On-going work focuses on the hardware integration of the presented control framework and safety criterion on a Kuka LWR4 serial robot. The distance between the end-effector and the human operator is acquired with a 3D visual system, here a Microsoft Kinect, and encouraging preliminary results have been obtained as illustrated on Fig. 1. The reliability and continuity of the measured distance is still to be improved and the velocity of the human operator must be considered. The Gurobi QCQP solver is running as an Orocos component on a Linux operating system patched with Xenomai to ensure proper real-time constraints at 1 kHz. Given the computational load induced by the QCQP, a 1 kHz sampling frequency cannot be guaranteed yet and the overall performances have to be improved.

Besides the improvement of the computational aspects of the control problem, future work will focus on the potential energy part of the safety criterion. The (*kinetic* + *potential*) energy exchange between the robot and its environment still has to be studied, validated in simulation and integrated on the real robot.

VI. ACKNOWLEDGMENTS

This work was partially supported by the RTE company through the RTE/UPMC chair Robotics Systems for field intervention in constrained environments held by Vincent Padois. The authors also would like to thank Antoine Seeleuthner for his initial contribution to this work as well as Antoine Hoarau and Sovannara Hak for their help with the implementation.

REFERENCES

- [1] R. Alami, A. Albu-Schaeffer, A. Bicchi, R. Bischoff, R. Chatila, A. De Luca, A. De Santis, G. Giralt, J. Guiochet, G. Hirzinger, *et al.*, "Safe and dependable physical human-robot interaction in anthropic domains: State of the art and challenges," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*. IEEE, 2006, pp. 1–16.
- [2] M. Zinn, O. Khatib, B. Roth, and J. K. Salisbury, "Playing it safe [human-friendly robots]," *Robotics & Automation Magazine, IEEE*, vol. 11, no. 2, pp. 12–21, 2004.
- [3] S. Haddadin, N. Mansfield, and A. Albu-Schäffer, "Rigid vs. elastic actuation: Requirements & performance," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 5097–5104.
- [4] R. Bischoff, J. Kurth, G. Schreiber, R. Koeppel, A. Albu-Schäffer, A. Beyer, O. Eiberger, S. Haddadin, A. Stemmer, G. Grunwald, *et al.*, "The kuka-dlr lightweight robot arm-a new reference platform for robotics research and manufacturing," in *41st International Symposium on Robotics*, 2010, pp. 1–8.
- [5] C. Loughlin, A. Albu-Schäffer, S. Haddadin, C. Ott, A. Stemmer, T. Wimböck, and G. Hirzinger, "The dlr lightweight robot: design and control concepts for robots in human environments," *Industrial Robot*, vol. 34, no. 5, pp. 376–385, 2007.
- [6] G. Hirzinger, A. Albu-Schäffer, M. Hahnle, I. Schaefer, and N. Sporer, "On a new generation of torque controlled light-weight robots," in *IEEE International Conference on Robotics and Automation*, vol. 4, 2001, pp. 3356–3363.
- [7] D. M. Ebert and D. D. Henrich, "Safe human-robot-cooperation: Image-based collision detection for industrial robots," in *IEEE/RSJ International Conference On Intelligent Robots and Systems*, 2002, pp. 1826–1831.
- [8] V. J. Lumelsky and E. Cheung, "Real-time collision avoidance in tele-operated whole-sensitive robot arm manipulators," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 23, no. 1, pp. 194–203, 1993.
- [9] K. Ikuta, H. Ishii, and M. Nokata, "Safety evaluation method of design and control for human-care robots," *The International Journal of Robotics Research*, vol. 22, no. 5, pp. 281–297, 2003.
- [10] S. Haddadin, A. Albu-Schäffer, A. De Luca, and G. Hirzinger, "Collision detection and reaction: A contribution to safe physical human-robot interaction," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 3356–3363.
- [11] A. De Luca, A. Albu-Schäffer, S. Haddadin, and G. Hirzinger, "Collision detection and safe reaction with the dlr-iii lightweight manipulator arm," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 1623–1630.
- [12] J. Heinzmann and A. Zelinsky, "Quantitative safety guarantees for physical human-robot interaction," *The International Journal of Robotics Research*, vol. 22, no. 7-8, pp. 479–504, 2003.
- [13] S. Haddadin, A. Khoury, T. Rokahr, S. Parusel, R. Burgkart, A. Bicchi, and A. Albu-Schäffer, "A truly safely moving robot has to know what injury it may cause," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 5406–5413.
- [14] A. De Luca and L. Ferrajoli, "Exploiting robot redundancy in collision detection and reaction," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 3299–3305.
- [15] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.
- [16] O. Khatib, "Inertial properties in robotic manipulation: An object-level framework," *The International Journal of Robotics Research*, vol. 14, no. 1, pp. 19–36, 1995.
- [17] S. Rubrecht, V. Padois, P. Bidaud, M. De Broissia, and M. Da Silva Simoes, "Motion safety and constraints compatibility for multibody robots," *Autonomous Robots*, vol. 32, no. 3, pp. 333–349, 2012.
- [18] X. Merhiot, J. Garrec, G. Saupin, and C. Andriot, "The xde mechanical kernel: efficient and robust simulation of multibody dynamics with intermittent nonsmooth contacts," in *International Conference on Multibody System Dynamics*, Stuttgart, Germany, 2012.
- [19] Gurobi Optimization Inc., "Gurobi optimizer reference manual," 2015. [Online]. Available: <http://www.gurobi.com>

B.4 Emergence of humanoid walking behaviours from mixed-integer model predictive control

Emergence of humanoid walking behaviors from Mixed-Integer Model Predictive Control

Aurelien Ibanez¹, Philippe Bidaud^{1,2} and Vincent Padois¹

Abstract— Balance strategies range from continuous postural adjustments to discrete changes in contacts: their simultaneous execution is required to maintain postural stability while considering the engaged walking activity. In order to compute optimal time, duration and position of footsteps along with the center of mass trajectory of a humanoid, a novel mixed-integer model of the system is presented. The introduction of this model in a predictive control problem brings the definition of a Mixed-Integer Quadratic Program, subject to linear constraints. Simulation results demonstrate the simultaneous adaptation of the gait pattern and posture of the humanoid, in a walking activity under large disturbances, to efficiently compromise between task performance and balance. In addition, a push recovery scenario displays how, using a single balance-performance ratio, distinct behaviors of the humanoid can be specified.

I. INTRODUCTION

Humanoid robots employ locomotion systems which are essentially of hybrid nature, combining discrete supports and multi-body dynamics. Legged locomotion couples these two sub-systems in different ways, and distinct behaviors emerge depending on the desired performance and constraints [1]. The choice of balance strategy is all the more important in situations where the robot undergoes large physical disturbances. Appropriate strategies in such cases generally lie between whole-body balancing and changes in the base of support (BoS). The robotics literature presents various solutions in this range, involving torque compensation [2], bracing behaviors [3] at the center of mass (CoM), angular momentum rejection [4] and shifts in foot placement [4], [5].

Changes in the BoS, however, have the potential to provide a greater degree of stabilization than whole-body adjustments in a fixed-support configuration [6], and may, on one hand, even be necessary depending on the activity and the constraints the system is subject to. On the other hand control of these changes may be more challenging. It requires indeed the determination of the adequate time, duration, distance and direction of the shift, while regarding both constraints on the system and the desired motion of the robot. With the Capture Region approach [4], Pratt *et al.* propose to solve the problem of when and where to take a step, along with suitable CoM and angular momentum behaviors. Although it has largely demonstrated its efficiency in push-recovery

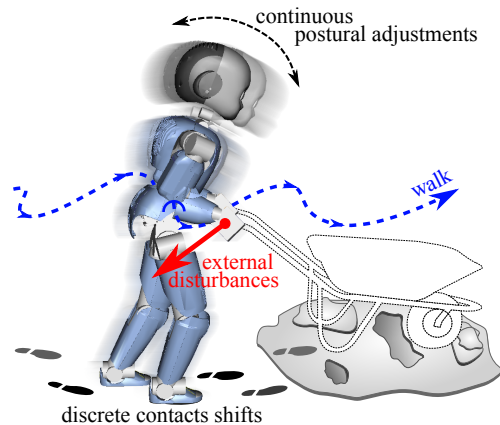


Fig. 1. Illustration of the hybrid nature of safe biped walking: the walking task induces discrete changes in the base of support, while balance is maintained through continuous postural adjustments and adaptation of the contacts position.

cases, this method lacks consideration of the constraints the robot is subject to and of the engaged activity, which could potentially affect the feasibility of the expected step. Furthermore, little insight on the suitable duration of the step is given.

Predictive approaches are appropriate to preview the influence of the duration and placement of the step taken, and can be applied to the Capture Region method as Krause *et al.* [7] propose in a Model Predictive Control (MPC) framework. MPC indeed provides a future time window to estimate the evolution of the system's state, and the formulation of an optimization problem is favorable to the consideration of constraints and objectives the robot must comply with. However, the additional complexity generally requires the use of reduced models, constraints approximations and predefined entities or heuristics in order to obtain computationally-efficient formulations. Moreover the hybrid nature of biped walking, involving continuous evolution of the system's motion and discrete changes in constraints and forces acting on it as illustrated in Figure 1, tends to prevent straightforward formulations. In order to confine the resolution to the continuous, smooth part of the problem, Herdt *et al.* [5] propose to exploit a previously given horizon of contacts activation to simultaneously optimize the CoM trajectory with contacts positions. Another method consists in regularizing the problem: Mordatch *et al.* [8] make for example use of hard-constraints smoothing for rigid contacts, in order to ensure regularity of the model.

¹Aurelien Ibanez, Philippe Bidaud and Vincent Padois are with:
- Sorbonne Universités, UPMC Univ Paris 06, UMR 7222, Institut des Systèmes Intelligents et de Robotique, F-75005, Paris, France
- CNRS, UMR 7222, Institut des Systèmes Intelligents et de Robotique, F-75005, Paris, France
{ibanez,bidaud,padois}@isir.upmc.fr

²Philippe Bidaud is with the ONERA, 91123 Palaiseau, France
philippe.bidaud@onera.fr

This paper proposes an original approach to the capture, in a predictive framework, of the influence of both CoM dynamics and changes in the BoS with respect to constraints on the system and the ongoing walking activity. No regularization of the hybrid nature of biped walking is carried out, and discrete events are described using a computationally favorable, redundant set of highly-constrained integer variables. Behavior of the system is specified at a high level, as a sole ratio between balance and performance of the walking activity. The resulting MPC problem takes the form of a quadratic, linearly constrained mixed-integer program (MIQP) which allows to determine, over a preview horizon, an optimal strategy between changes in the BoS and CoM behavior, subject to multiple constraints, maximizing balance and performance of a walking activity.

This work is presented as follows. A linearly-constrained, mixed-integer set of variables is first introduced in Section II to characterize admissible changes in the BoS. The Zero-Moment Point (ZMP) approach is employed to capture, in terms of balance, the state of the system with respect to the BoS. The resulting model allows the linear expression of several constraints related to the walking activity. Section III employs this model of the system in an MPC framework, and defines the control objective as a compromise between balance and performance of the walking task, which is formulated as the tracking of a desired CoM trajectory. An optimal horizon of changes in the BoS and CoM trajectory is computed from a MIQP, without the use of pre-defined gait patterns or heuristics. Simulations results in Section IV validates this approach in various scenarios. Performance of the tracking task is demonstrated in a sinusoidal trajectory tracking case, and the introduction of large physical disturbances on the robot exhibits the conjoint adaptation of gait pattern and CoM trajectory. Last, the influence of the balance and walking weights is exhibited in a push recovery case where the system behavior, as a response to an unknown external action, significantly varies for different values of the weights ratio.

II. ZERO-MOMENT POINT, MIXED-INTEGER BIPED MODEL

In order to adopt the appropriate balance strategy, control algorithms should exploit a model of the system describing the effect of various quantities of interest on the postural stability of the robot. Indeed, employing this model to evaluate the performance of different evolutions of each of these quantities may lead to the identification of the best combination to select.

Under several hypotheses, the Zero-Moment Point [9] approach captures the balance state of the system by relating its CoM dynamics to the base of support. However, the hybrid nature of biped walking clearly differentiates CoM dynamics changes from BoS shifts: the former is continuous in essence, while the latter is restrained to discrete events.

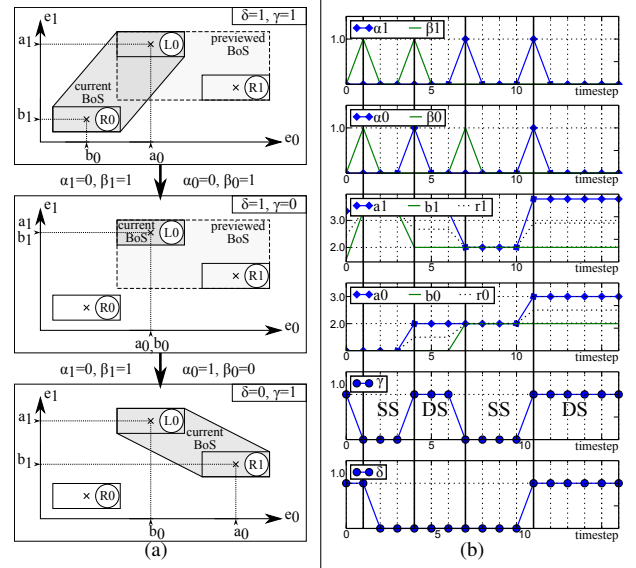


Fig. 2. Mixed-integer contact state description —2(a): Real valued $\mathbf{a}$ and $\mathbf{b}$ describe bounds of the position of the feet in contact, with their respective binary rising/falling edges α and β . Binary variable γ differentiates single and double support phases, and δ restrains the evolution of bounds $\mathbf{a}, \mathbf{b}$ during transitions from double to single support. Current BoS is linearly expressed (dark gray) and previewed BoSs are approximated by their bounding box (light gray). —2(b): typical evolution of linearly-constrained variables ($\mathbf{a}, \mathbf{b}, \alpha, \beta, \delta, \gamma$), related to the base of support. Variable $\mathbf{r} = (\mathbf{a} + \mathbf{b})/2$ is displayed here to describe the center of the supporting surface.

MPC methods, writing the control problem as an optimization problem, generally consider the continuous members of the ZMP model solely as degrees of freedom (DoF). Kajita *et al.* [10] and Wieber [11] for example adjusted the CoM dynamics to maintain balance; Herdt *et al.* [5] isolated the discrete aspect of changes in the BoS in a priorly-defined activation matrix, allowing to add the determination of the footsteps location to the MPC problem.

However, the time of activation of discrete changes in the BoS might noticeably affect the performance of both balance and the walking task. This section therefore describes a model considering simultaneously the CoM dynamics, amplitude and instants of changes in the BoS. Aiming at a computationally-efficient formulation, the use of a redundant set of integer and real variables to describe the contact state of the robot allows to keep a linear form of the various constraints the system is subject to.

A. Mixed-Integer biped model

To capture the discrete nature of changes in the BoS, with a view to future optimization, a choice of linearly-constrained, redundant descriptors of the contact state is proposed. Additional details can be found in [12].

The amplitude and position of the BoS are described by its bounding box, as illustrated in Figure 2(a): real-valued variables $(\mathbf{a}, \mathbf{b}) \in \mathbb{R}^2 \times \mathbb{R}^2$ are defined as the upper and lower bounds, respectively, of the position in the two horizontal directions of the feet in contact. The discrete essence of changes in the BoS leads to the definition of several con-

straints, called shape constraints, restraining variables (a, b) . Essentially, (a, b) must be piecewise-constant. This constraint can be linearly expressed with the introduction of binary variables $(\alpha, \beta) \in \{0, 1\}^2 \times \{0, 1\}^2$ as rising and falling edges of a and b , respectively.

Bounds (a, b) are furthermore implicitly related to feet positions: thus there exists couplings between the two directions of a and b , called admissibility constraints.

First, single support phases (SS) impose that a equals b , as the BoS is reduced to one foot (*cf.* Figure 2(a)). SS phases must hence be differentiated from double support ones (DS): the binary variable $\gamma \in \{0, 1\}$ is introduced to this purpose, and allows to define this constraint linearly.

Second, each change in the pair (a, b) leads to the alternation of γ , as any shift in the BoS corresponds to either a transition SS $\rightarrow$ DS or DS $\rightarrow$ SS. Note that, as a result, γ is fully defined by (a, b) and (α, β) assuming that an initial value of γ is known.

Last, potential changes in a and b from DS to SS depend on the configuration of the feet, relatively to bounds a and b , in the previous DS phase. Indeed, as illustrated in Figure 2(a), only the pairs (a_1, a_0) or (b_1, b_0) can change when leaving the first DS configuration. On the contrary, leaving the last DS configuration can only lead to changes in the pairs (b_1, a_0) or (a_1, b_0) . This coupling can be linearly expressed with the introduction of the binary variable $\delta \in \{0, 1\}$: the first configuration in Figure 2(a) corresponds to $\delta = 1$ and the last to $\delta = 0$. Note that this relation is bilateral: when leaving SS to DS, changes in (a, b) set the value of δ , and when switching back to SS, the value of δ restrains potential changes in (a, b) .

Figure 2(b) proposes typical evolutions of this set of BoS variables.

Contact state of the robot can therefore be described by the linearly-constrained, mixed-integer set of variables $(a, b, \alpha, \beta, \delta, \gamma)$. The ZMP model, neglecting rotational effects, relates CoM dynamics to the BoS; state ξ of the system in a balance perspective can thus be reduced to

$$\xi = [a \ b \ \alpha \ \beta \ \delta \ \gamma \ c \ \dot{c} \ \ddot{c}]^T, \quad (1)$$

where $c \in \mathbb{R}^3$ is the position of the CoM in the world frame. In the rest of this paper, time is sampled at discrete control instants t_i , and notation v_j for function v of time t denotes the value $v(t_j)$, and $v_{k|j}$ the value $v(t_j)$ estimated from control time t_k . As stated earlier, the major contribution of such a description of the contact state is that, in discrete time, shape and admissibility constraints can be put in the linear form

$$\forall k \in \mathbb{N}, \quad \mathbf{A}_{cl}\xi_k + \mathbf{A}_{cr}\xi_{k+1} \preceq \mathbf{f}_c, \quad (2)$$

where $\mathbf{A}_{cl}$ and $\mathbf{A}_{cr}$ are $n_c \times 19$ matrices, and $\mathbf{f}_c$ a vector in $\mathbb{R}^{n_c}$.

Biped postural stability, in non-sliding cases, can be obtained by avoiding tip-over of the humanoid: with coplanar contacts, the center of pressure (CoP) must stay away from

the edges of the BoS. The ZMP approach [9] allows the computation of the position p on the ground of the CoP from the dynamics of the system. Ignoring rotational effects compared to translational ones, and neglecting vertical acceleration of the CoM compared to its horizontal acceleration, p writes

$$p = h - \frac{c \cdot e_2}{g} \ddot{h} \quad \text{where} \quad h = c - (c \cdot e_2)e_2, \quad (3)$$

with e_2 the ascendant vertical direction, g the gravity amplitude and h the horizontal position of the CoM. In order to have a linear expression of p with respect to ξ and considering the approximation on the vertical acceleration of the CoM, the altitude of the CoM is considered as constant in the rest of this paper. The relevant CoM dynamics are thus the horizontal ones h , therefore system state ξ , matrices $\mathbf{A}_{cl}$ and $\mathbf{A}_{cr}$ and vector $\mathbf{f}_c$ are modified accordingly. Also, the balance constraint is linear with respect to the horizontal dynamics of the CoM, but changes in the supporting surface being reflected by changes in state variables (a, b) , the inclusion constraint is quadratic with respect to the state ξ of the system. Nevertheless, as shown in [5], overestimating the convex hull of this surface by its bounding box in forward and lateral directions brings the definition of a set of linear inequality constraints with respect to ξ . Note that this approximation is solely made for future potential DS phases: constraints for an established BoS can be fully implemented without overestimation (*cf.* Figure 2(a)). Remarks on this approximation are provided in Section IV-D. Under this overestimation, the CoP constraints write

$$\forall k \in \mathbb{N}, \quad \mathbf{A}_p \xi_k \preceq \mathbf{f}_p, \quad (4)$$

where $\mathbf{A}_p$ is a $n_p \times 16$ matrix, and $\mathbf{f}_p$ a vector in $\mathbb{R}^{n_p}$.

The previous description of the contact state, with the ZMP model, brings the resulting mixed-integer biped model at time t_k

$$\xi_k = [a_k \ b_k \ \alpha_k \ \beta_k \ \delta_k \ \gamma_k \ h_k \ \dot{h}_k \ \ddot{h}_k]^T$$

$$\text{s.t.} \quad \begin{cases} \mathbf{A}_{c,r}\xi_k \preceq \mathbf{f}_c - \mathbf{A}_{c,l}\xi_{k-1}, \\ \mathbf{A}_p \xi_k \preceq \mathbf{f}_p, \\ (\alpha_k, \beta_k) \in \{0, 1\}^2 \times \{0, 1\}^2, \\ (\delta_k, \gamma_k) \in \{0, 1\} \times \{0, 1\}, \\ (a_k, b_k) \in \mathbb{R}^2 \times \mathbb{R}^2, \\ (h_k, \dot{h}_k, \ddot{h}_k) \in \mathbb{R}^2 \times \mathbb{R}^2 \times \mathbb{R}^2, \end{cases} \quad (5)$$

which describes and restrains the balance state of the system.

B. Walking motion constraints

The model described in (5) allows the consideration of a large variety of linear constraints, inherent to the control problem of walking humanoids. Relevant constraints regarded in this paper are maximum leg span, maximum swinging foot average velocity and minimal SS/DS phases durations. Such constraints will ban strategies involving too large or fast steps, and hinder solutions such as fast series of small steps. Note that additional constraints such as maximal SS/DS durations, bounds on the CoM velocity and

acceleration, or bounds on positions of the feet can also be put in a linear form with respect to the system state ξ .

Let $\bar{v}$ denote the maximal swinging foot velocity, $\bar{s}$ the maximal step length and $\bar{t}$ the largest of minimum durations of SS and DS phases. Constraints on leg span, foot velocity and SS/DS durations write

$$\sum_{t_{k+j}-t_k \leq \max(\bar{s}/\bar{v}, \bar{t})} \mathbf{A}_{w,j} \xi_{k+j} \preceq \mathbf{f}_w, \quad j \in \mathbb{N}, \quad (6)$$

where $\mathbf{A}_{w,j}$ is a $n_w \times 19$ matrix, and $\mathbf{f}_w$ a vector in $\mathbb{R}^{n_w}$. It can be shown that $\mathbf{f}_w$ depends on a history of states ξ prior to t_k , and can be written in the form

$$\mathbf{f}_w = \mathbf{f}_{w_0} + \sum_{t_k - t_{k-j} < \max(\bar{s}/\bar{v}, \bar{t})} \mathbf{B}_{w,j} \xi_{k-j}, \quad j \in \mathbb{N}^*. \quad (7)$$

Condition $t_{k+j} - t_k \leq \max(\bar{s}/\bar{v}, \bar{t})$ (*resp.* $t_k - t_{k-j}$) states that the forward (*resp.* backward) history influence is irrelevant beyond the time step where both minimum duration and maximum average velocity constraints are necessarily respected. A constraint to avoid the overlapping of feet is added, in the form of a linearized collision avoidance constraint.

This model is employed in the following section to evaluate and optimize the performance, in terms of walking activity and balance, of admissible evolutions of the system state.

III. QUADRATIC, MIXED-INTEGER MPC

The walking control problem, in a predictive framework, consists in finding an optimal and admissible horizon of future system states $\xi_{k|k+j}$ that ensure the performance of an ongoing walking activity while maximizing balance of the system. The descriptors of the system state presented in Section II being taken as variables of an optimization problem, these objectives can be written in the form of quadratic functions, leading to the formulation of a MPC problem as a mixed-integer quadratic program.

Considering approximations exploited in the ZMP model, a minimal set of parameters can be obtained to describe an horizon of system states $\xi_{k|k+j}$. Indeed, a discrete integration scheme brings, with piecewise constant CoM jerks $\mathbf{u} = \partial^3 \mathbf{h} / \partial t^3$

$$\forall j \in \mathbb{N}^*, \quad \hat{\mathbf{h}}_{k|k+j+1} = \mathbf{A}_h \hat{\mathbf{h}}_{k|k+j} + \mathbf{B}_h \mathbf{u}_{k|k+j+1}, \quad (8)$$

where $\hat{\mathbf{h}} = [\mathbf{h} \ \dot{\mathbf{h}} \ \ddot{\mathbf{h}}]^T$ is the CoM horizontal dynamics, and $\mathbf{A}_h$, $\mathbf{B}_h$ integration matrices. The minimal set of variables required to preview $\xi_{k|k+j+1}$ from $\xi_{k|k+j}$ is hence denoted $\chi_{k|k+j+1}$, defined as

$$\chi = [\mathbf{a} \ \mathbf{b} \ \boldsymbol{\alpha} \ \boldsymbol{\beta} \ \boldsymbol{\delta} \ \boldsymbol{\gamma} \ \mathbf{u}]^T.$$

The model (5) can thus be written as the Linear Time-Invariant process

$$\forall j \in \mathbb{N}^*, \quad \xi_{k|k+j+1} = \mathbf{Q} \xi_{k|k+j} + \mathbf{T} \chi_{k|k+j+1}, \quad (9)$$

where $\mathbf{Q}$ and $\mathbf{T}$ are state description matrices derived from (8). Relation (9) allows to preview an horizon $\mathbf{C}_{k,N}$ of N future states $\xi_{k|k+j}$ from an horizon $\mathbf{X}_{k,N}$ of N future inputs $\chi_{k|k+j}$ and the actual state $\xi_{k|k} = \xi_k$. The preview writes, with $\mathbf{P}$ and $\mathbf{R}$ combinations of $\mathbf{Q}$ and $\mathbf{T}$

$$\mathbf{C}_{k,N} = \begin{bmatrix} \xi_{k|k+1} \\ \vdots \\ \xi_{k|k+N} \end{bmatrix} = \mathbf{P} \xi_k + \mathbf{R} \begin{bmatrix} \chi_{k|k+1} \\ \vdots \\ \chi_{k|k+N} \end{bmatrix},$$

denoted as

$$\mathbf{C}_{k,N} = \mathbf{P} \xi_k + \mathbf{R} \mathbf{X}_{k,N}. \quad (10)$$

Linear equation (10) allows to reformulate the linear equalities (2), (4) and (6) as

$$\mathbf{A} \mathbf{X}_{k,N} \preceq \mathbf{f} \quad (11)$$

where $\mathbf{f}$ depends on a history of actual and previous states ξ , as shown in (2), (6) and (7).

A walking activity can be interpreted as reaching a target position with a desired horizontal velocity, which can be expressed at the level of the CoM. It is in such terms a tracking task, whose performance can be evaluated as a tracking error. Let $\hat{\mathbf{h}}^{ref}$ denote the desired trajectory of the CoM. Objective J_w of the walking activity can be written, over a preview horizon, as the minimization of

$$J_{wk} = \sum_{j=1}^N \left\| \mathbf{S} \left(\hat{\mathbf{h}}_{k|k+j} - \hat{\mathbf{h}}_{k|k+j}^{ref} \right) \right\|^2, \quad (12)$$

where $\mathbf{S}$ is a 6×6 weighting selection matrix, diagonal, defining whether position, velocity and/or acceleration are tracked in each of the two horizontal directions. For example, a standstill activity aiming at $\dot{\mathbf{h}} \rightarrow \mathbf{0}$ can be expressed with $\hat{\mathbf{h}}^{ref} = \mathbf{0}$ and the only non-null terms of $\mathbf{S}$ corresponding to velocity in both horizontal directions; that is, $J_{wk} = \sum_{j=1}^N \|\dot{\mathbf{h}}_{k|k+j}\|^2$.

While performance of the walking activity is essential, robustness of the posture of the robot is also a major objective in walking motions. Balance is guaranteed, under the assumptions of the model, by the CoP constraint (4) in (11). Nevertheless, as a tip-over situation occurs when the CoP reaches the edges of the BoS, robustness of the balance state of the robot can be captured as a distance to the edges. The balance maximization objective can thus be written as the minimization of J_b

$$J_{bk} = \sum_{j=1}^N \left\| \mathbf{p}_{k|k+j} - \mathbf{r}_{k|k+j} \right\|^2, \quad (13)$$

where $\mathbf{r} = (\mathbf{a} + \mathbf{b})/2$ is the center of the BoS, *i.e.* the point at the greatest distance from the edges of the supporting surface.

Secondary objectives are added to the control problem in these works for regularization purposes. They aim at minimizing CoM jerks, avoid excessive changes in solutions from one control step to another, keep track of the previous BoS

size and prefer DS phases over SS ones. These regularization objectives are written in the form of a quadratic cost q , which can be used to reward desired behaviors in order to favor short and frequent stepping over slower gaits, for example.

Objectives (12) and (13) are quadratic with respect to $\mathbf{X}_{k,N}$, and are considered in the global cost function J

$$J_k = \omega_b J_{b_k} + \omega_w J_{w_k} + q = \mathbf{X}_{k,N}^T \mathbf{H} \mathbf{X}_{k,N} + \mathbf{d}^T \mathbf{X}_{k,N}, \quad (14)$$

where $\mathbf{H}$ is a positive definite matrix, $\mathbf{d}$ a vector and (ω_b, ω_w) scalar weights defining a compromise between balance robustness and tracking performance. The MPC problem finally writes

$$\min_{\mathbf{X}_{k,N}} \mathbf{X}_{k,N}^T \mathbf{H} \mathbf{X}_{k,N} + \mathbf{d}^T \mathbf{X}_{k,N} \quad \text{s.t.} \quad \begin{cases} \mathbf{A} \mathbf{X}_{k,N} \preceq \mathbf{f} \\ \boldsymbol{\xi}_{k|k} = \boldsymbol{\xi}_k \\ \boldsymbol{\xi}_{k|k+j+1} = \mathbf{Q} \boldsymbol{\xi}_{k|k+j} + \mathbf{T} \mathbf{x}_{k|k+j+1}, \\ (\boldsymbol{\alpha}_{k|k+j}, \boldsymbol{\beta}_{k|k+j}) \in \mathbb{R}^2 \times \mathbb{R}^2, \\ (\boldsymbol{\alpha}_{k|k+j}, \boldsymbol{\beta}_{k|k+j}) \in \{0, 1\}^2 \times \{0, 1\}^2, \\ (\delta_{k|k+j}, \gamma_{k|k+j}) \in \{0, 1\} \times \{0, 1\}, \\ \mathbf{u}_{k|k+j} \in \mathbb{R}^2, \end{cases} \quad (15)$$

which is a Mixed-Integer Quadratic Program in canonical form. The computational complexity of problem (15) is strongly related to the number of non-real variables. Nevertheless its QP form allows the use of fast algorithms and binary variables can be sampled at a lower frequency, as discussed in Section IV-D.

IV. RESULTS

Contribution of the MPC formulation (15) is exposed with several simulation results, demonstrating the variety of behaviors generated in diverse scenarios.

Opening results illustrate that, without the use of any heuristics other than motion constraints and controller weights, the MPC (15) automatically generates an intuitive gait pattern to follow a varying reference velocity. A second set of results exhibits optimal modifications of the gait pattern to accommodate an ongoing walking activity to large physical disturbances on the robot. Last, results in a push-recovery scenario demonstrate how weights in (14) allows to implicitly define, at a high level, different behaviors of the humanoid.

Simulations are performed using the ARBORIS-PYTHON simulator [13] developed at ISIR, and whole-body motion from optimal outputs of the MPC (15) is ensured by an LQP-based controller [14] for an iCub [15] robot model. The system state is previewed over an horizon of $1.0s$, and simulations are run with a time step of $dt = 1.e^{-2}s$. The humanoid weighs $\approx 27kg$ with a height of $\approx 1m$.

A. Gait generation: sinusoidal velocity tracking

A first simulation scenario is performed to illustrate the behavior of the MIQP problem in nominal conditions. A walking activity is specified as the tracking of a sinusoidal CoM velocity in the forward direction, and a null velocity in the lateral direction. Major weights (ω_b, ω_w) in the cost

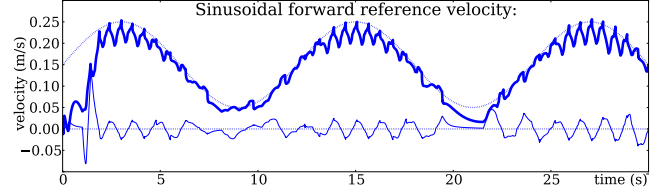


Fig. 3. CoM velocity tracking performance — *thick line*: forward — *thin line*: lateral — *dotted lines*: reference velocities

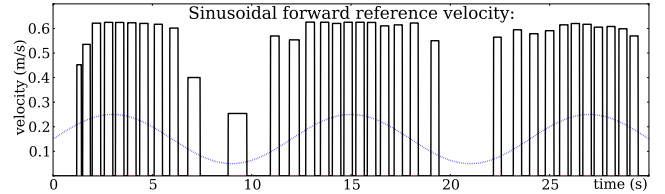


Fig. 4. Generated gait pattern from CoM velocity tracking — *solid*: average foot velocity during single support — *dotted*: CoM reference velocities

function (14) are equal, and forward and lateral directions have the same importance in the tracking error function (12), *i.e.* $S_{ij} \in \{0, 1\}$.

Performance of the tracking task is depicted in Figure 3: both forward and lateral CoM velocities tend towards the reference $\dot{\mathbf{h}}^{ref}$. However, oscillations are still visible as the result of the alternation between SS and DS phases. Indeed, velocity of the CoM is restrained by the CoP constraints (4) its acceleration is subject to, constraints that are all the more restrictive in single support phases. Consequently, velocity tracking is hindered during SS phases, and DS phases *release* DoFs of the CoM, allowing the recovery of the tracking task. Nevertheless, in periods of fast reference velocity $\dot{\mathbf{h}}^{ref}$, a new SS phase is rapidly required and the robot do not have time, due to regularizing terms q , to fully reach the required velocity.

This last effect is illustrated in Figure 4 which displays SS/DS alternations with the average foot velocity during each SS phase. Indeed, time periods of high reference velocity such as $t = 15s$ require steps of high velocity and frequency, while periods of medium reference velocity such as $t \in [10s, 13s]$ lead to slower, sparser steps and a better CoM velocity tracking.

From a more distant point of view, Figure 4 exposes an intuitive strategy computed from (15): steps taken automatically tend to be longer, faster and more frequent as the desired walking velocity grows higher.

Note that equal weights $\omega_b = \omega_w$ in (14) tend to give a

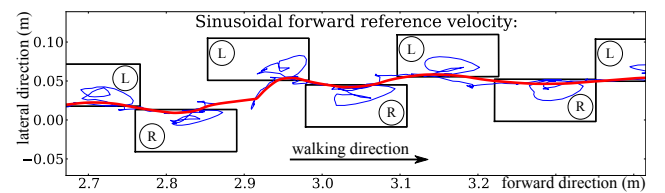


Fig. 5. Evolution of the actual CoP and CoM with generated footsteps — *thick line*: CoM position — *thin line*: actual CoP position

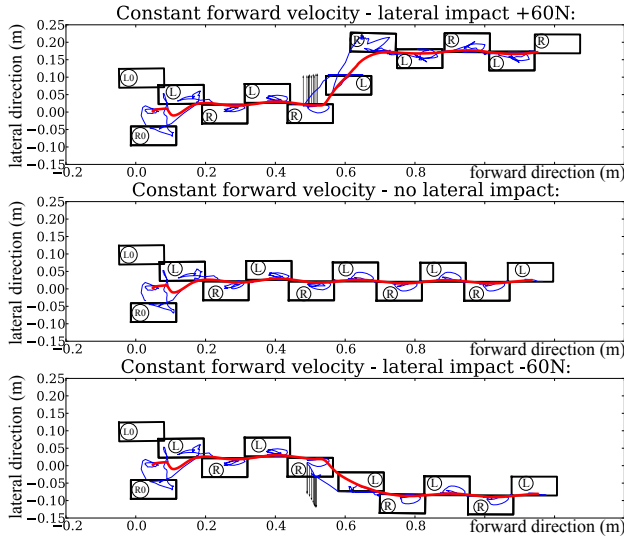


Fig. 7. Evolution of the actual CoP and CoM with the adaptation of generated footsteps to an unknown impact — *thick line*: CoM position — *thin line*: actual CoP position

higher priority to the walking task, particularly in cases of high velocity tracking. This is mainly due to objectives J_w and J_b not being normalized, hence letting differences in magnitude have an influence. This effect can be observed in Figure 5 which presents an extract of CoM, CoP and feet positions. Although balance is still ensured, the tracking task slightly prevents, especially in SS phases, the CoP from staying in the middle of the BoS.

Figure 5 shows that the feet tend to be aligned in the forward direction as a result of the null velocity tracking in the lateral direction, in order to minimize the lateral velocity of the CoM during footsteps. However, cases of low forward reference velocities require small steps: the linearized non-overlapping constraint on the feet forces a lateral gap between them, hence inducing a loss of performance of the tracking task in the lateral direction, as shown in Figure 3 around $t = 22s$ for example.

B. Activity adjustments: walking under large disturbances

This simulation aims at exhibiting, for an identical set of weights in the cost function (14), differences in the optimal behavior against unknown disturbances. The humanoid has to perform a walking task with a target forward velocity of $0.20m.s^{-1}$, and a null desired lateral velocity. Three cases are compared: an unknown lateral impact of $+60N$ (more than a fifth of the total weight of the robot) is applied to the head of the humanoid during a period of $0.1s$, then no effort is applied to the robot and, last, an opposite impact of $-60N$ is considered. Note that applying the external action to the head brings additional disturbances as large rotational effects are induced, effects which are not taken into account in the ZMP model (3). Snapshots of the different cases are provided in Figure 6.

Figure 7 shows that changes in the system state from the impact lead to an intuitive modification of the steps taken:

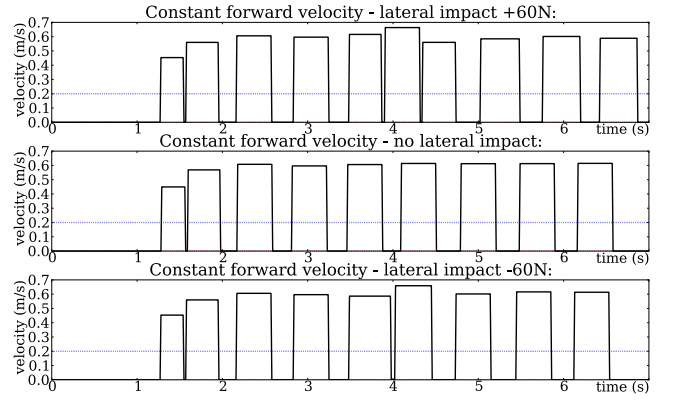


Fig. 8. Generated gait pattern from CoM velocity tracking as a response to different impacts — *solid*: average foot velocity during single support — *dotted*: CoM reference velocities

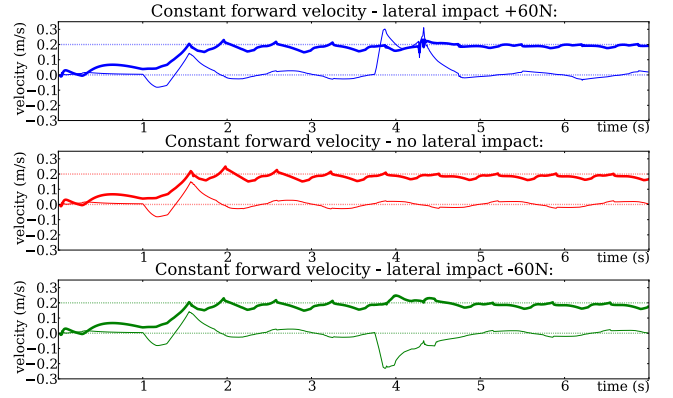


Fig. 9. CoM velocity tracking performance against the impact — *thick line*: forward — *thin line*: lateral — *dotted lines*: reference velocities

balance is ensured with a lateral drift of the robot in the direction of the impact. It can be noted that for the impact of $+60N$ the right foot crosses the left one: such a solution can be excluded by a set of linear constraints in (15), or self-collisions should be handled at the whole-body control level. The impact occurring during an ongoing step, Figure 8 shows that the speed and duration of this step is slightly altered as a response to the disturbance. Moreover, a major aspect of the recovery strategy is illustrated: the DS phase following the impact is noticeably shorter than in the undisturbed case. A lateral step is indeed taken almost immediately in order to improve balance.

These modifications in the gait pattern as a response to the impact seem *natural* from the balance point of view. Nevertheless, as illustrated in Figure 9, although a null lateral velocity obviously cannot be reached during the impact, the desired CoM velocities are rapidly recovered. Also, it can be noted that the tracking task in the forward direction is almost unaffected by the impact. The recovery steps taken are hence compatible with the ongoing walking activity, as objective function J_w is still part of the optimization problem.

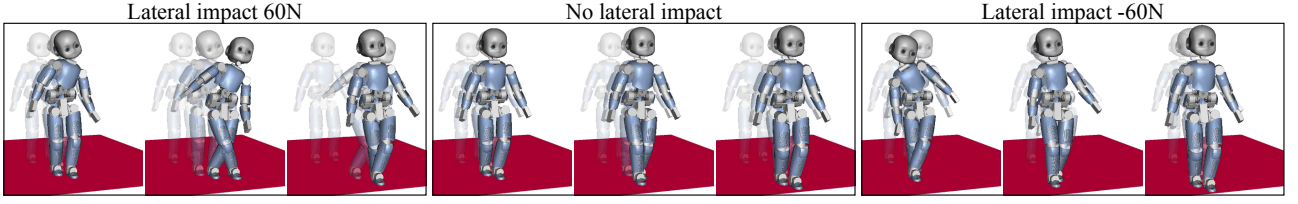


Fig. 6. Snapshots of a walking activity under different unknown external impacts, applied to the head of the humanoid

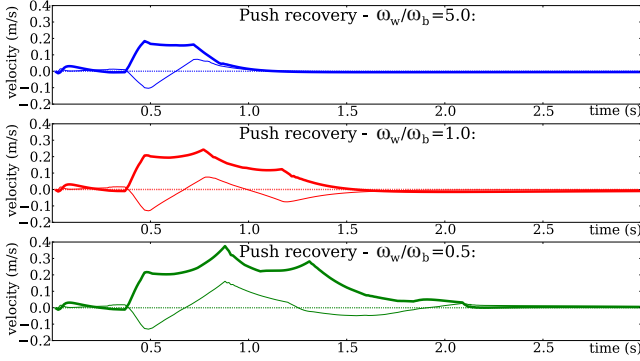


Fig. 10. Influence of the objective weights on the CoM velocity tracking performance — *thick line*: forward — *thin line*: lateral — *dotted lines*: reference velocities

C. Behavior specification: push-recovery

A last simulation scenario demonstrates how the balance behavior of the humanoid can be specified with the sole ratio ω_w/ω_b . Differences in behavior are illustrated in a push recovery case. The humanoid must stand still, *i.e.* $\dot{\mathbf{h}}^{ref} = \mathbf{0}$, and experiences an unknown impact of $50N$ in the forward direction and $-25N$ in the lateral one, during a period of $0.1s$. The standing task is incrementally relaxed in three cases. A first set of weights targets the best activity performance with $\omega_w/\omega_b = 5.0$, a second is placed as reference with $\omega_w/\omega_b = 1.0$ and a last set gives a higher priority to the balance objective J_b with $\omega_w/\omega_b = 0.5$. The difference in the resulting behavior of the humanoid is illustrated in figures 10, 11 and 12 for the three cases.

As expected, results provided in Figure 10 show that priority of the tracking objective J_w , relatively to J_b , has a direct influence on the evolution of the CoM velocity. Although the humanoid eventually reaches the null reference velocity in all cases, control with $\omega_w/\omega_b = 5.0$ attains the standstill objective around three times faster than with $\omega_w/\omega_b = 0.5$.

This gain in performance is achieved with a noticeably different gait pattern for the three cases. Indeed, as shown in Figure 11 the first controller stops the CoM with one step, the second with two and the third recovers a standstill state after three steps. It can be observed that the duration of the first recovery step grows with the decrease of the ratio ω_w/ω_b , which can be interpreted as the concurrence of two effects. During SS phases, the dynamics of the CoM are conditioned by the CoP constraint (4), hence potentially affecting the tracking objective J_w as shown in Figure 10; on the other hand, a longer step duration allows to place

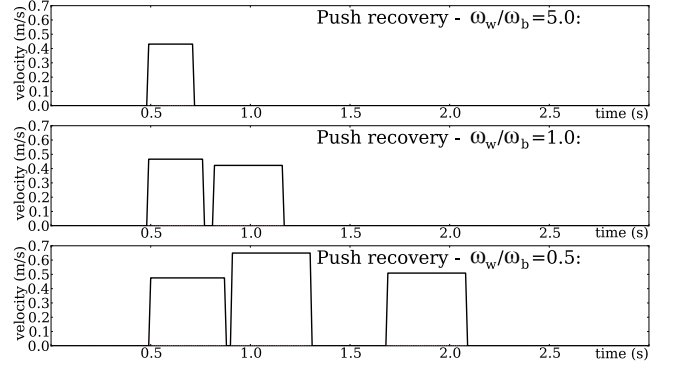


Fig. 11. Generated gait pattern from CoM velocity tracking for different objective weights in a push recovery scenario — *solid*: average foot velocity during single support

the BoS as desired, in favor of the balance objective J_b . Essentially, DS phases provides the CoM with a greater degree of freedom, while SS phases are required to adapt the BoS in regard to the balance objective.

Figure 12 indeed confirms these remarks. The first recovery step tends to be further away from the initial feet configuration as the ratio ω_w/ω_b decreases, which leads, in average, to a better placement of the CoP with respect to the BoS.

D. Remarks and discussions

As observed in figures 5, 7 and 12, the CoP happens to reach the edges of the BoS, thus putting the system at tip-over risk although constraints on the CoP (4) are specified with a safety margin of $\approx 30\%$, and are guaranteed in problem (15). The ZMP model (3) used as an approximation of the CoP ignores, among others, rotational effects. However such effects may yet be of large magnitude in the cases studied in this section: fast reference velocities and sudden disturbances are considered, leading to rapid steps and, as a consequence, fast movements of the swinging leg. Moreover, the application of the external force to the head of the robot induces rotational effects from the entire upper body of the humanoid (*cf.* figure 6), thus increasing the gap between the actual CoP and estimated ZMP (3). Note that this gap might be partially reduced by setting a lower maximal foot velocity. Another approximation regarding the CoP is the overestimation of the convex hull of the contact points by its bounding box, in order to write linear constraints with respect to state ξ . Nevertheless, outputs from (15) can be validated at each control step to ensure that the previewed CoP remains inside the convex hull of the BoS. If not, a fast QP program

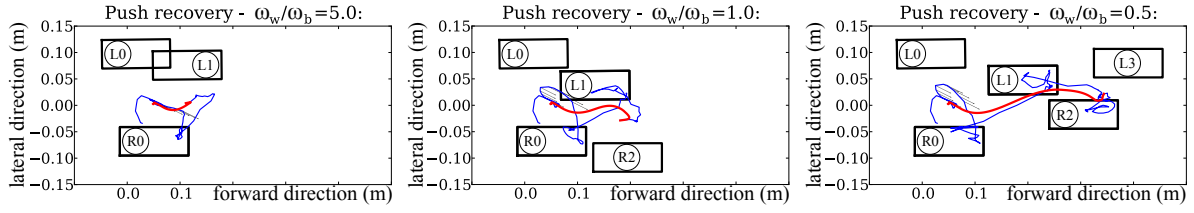


Fig. 12. Evolution of the actual CoP and CoM with generated footsteps for an identical impact and different objective weights (ω_b, ω_w) — *thick line*: CoM position — *thin line*: actual CoP position

as described in [11] can be set up to recompute a valid CoM trajectory (possibly with new feet positions as in [5]), taking as input the changes in BoS provided by (15). Nevertheless, such a safety setup was not necessary in the presented simulations.

Results introduced in this section were obtained with the GUROBI OPTIMIZER [16]. A non uniform time sampling scale was used, in order to keep a reduced size of the optimization vector while avoiding the accumulation of integration errors in the preview. Furthermore, variables associated to changes in the BoS were sampled at a lower frequency than variables describing the CoM dynamics. Indeed, the complexity of the MIQP (15) is strongly related to the number of integer variables, and it can arbitrarily be considered that changes in the BoS occur at a lower frequency than CoM adjustments. Solving the MIQP (15) took an average of $40ms^1$, and note that since MIQP solvers generally employ branch-and-bound based algorithms, computation time can largely be influenced by the order of variables and the use of heuristics, helping at reaching the best nodes faster.

Lastly, it can be noted that the model of the BoS presented in Section II omits DoFs from the rotations of the feet. Nevertheless, orienting the BoS in the motion or disturbance direction may be beneficial to balance.

V. CONCLUSIONS

A novel predictive formulation of the walking control problem is validated in this paper. The consideration of all admissible changes in the BoS adds degrees of freedom to the behavior of the system, behavior which can be specified with a high-level lever in the form of a weight ratio between balance and walking. Simulation results show a large range of balance strategies adopted by the humanoid, independently from prior gait patterns and moderated by the imposed compromise between balance and walking performance, as a response to the ongoing activity and unknown disturbances on the system.

Future works may consider rotations of the feet to generate a wider range of motions, and investigate in further details the computational efficiency of the problem. Additional

developments may also consider configurations with non-coplanar contacts.

ACKNOWLEDGMENT

This work was partially supported by the French Ministry of Higher Education and Research, by the RTE company through its chair “Robotics Systems for field intervention in constrained environments” held by Vincent Padois and by the European Commission within the CoDyCo project (FP7-ICT-2011-9, No. 600716).

REFERENCES

- [1] A. F. Miguel, “The emergence of design in pedestrian dynamics: Locomotion, self-organization, walking paths and construal law,” *Physics of life reviews*, 2013.
- [2] V. Prahlad, G. Dip, and C. Meng-Hwee, “Disturbance rejection by online zmp compensation,” *Robotica*, vol. 26, no. 1, p. 9, 2008.
- [3] S. Kanzaki, K. Okada, and M. Inaba, “Bracing behavior in humanoid through preview control of impact disturbance,” in *Proc. of the IEEE-RAS Int. Conf. on Humanoid Rob.* IEEE, 2005, pp. 301–305.
- [4] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, “Capture point: A step toward humanoid push recovery,” in *Proc. of the IEEE-RAS Int. Conf. on Humanoid Rob.* IEEE, 2006, pp. 200–207.
- [5] A. Herdt, H. Diedam, P.-B. Wieber, D. Dimitrov, K. Mombaur, and M. Diehl, “Online walking motion generation with automatic footstep placement,” *Advanced Robotics*, vol. 24, no. 5-6, pp. 719–737, 2010.
- [6] B. E. Maki, W. E. Mcilroy, and G. R. Fernie, “Change-in-support reactions for balance recovery,” *Engineering in Medicine and Biology Magazine, IEEE*, vol. 22, no. 2, pp. 20–26, 2003.
- [7] M. Krause, J. Engelsberger, P.-B. Wieber, and C. Ott, “Stabilization of the capture point dynamics for bipedal walking based on model predictive control,” in *Robot Control*, vol. 10, no. 1, 2012, pp. 165–171.
- [8] I. Mordatch, E. Todorov, and Z. Popović, “Discovery of complex behaviors through contact-invariant optimization,” *ACM Transactions on Graphics (TOG)*, vol. 31, no. 4, p. 43, 2012.
- [9] M. Vukobratović and J. Stepanenko, “On the stability of anthropomorphic systems,” in *Mathematical Biosciences*, vol. 15, no. 1, 1972, pp. 1–37.
- [10] S. Kajita, F. Kanehiro, K. Kaneko, K. Kajiura, K. Harada, K. Yokoi, and H. Hirukawa, “Biped walking pattern generation by using preview control of zero-moment point,” in *Proc. of the IEEE ICRA*, 2003.
- [11] P.-B. Wieber, “Trajectory free linear model predictive control for stable walking in the presence of strong perturbations,” in *Proc. of the IEEE-RAS Int. Conf. on Humanoid Rob.*, 2006, pp. 137–142.
- [12] A. Ibanez, P. Bidaud, and V. Padois, “Automatic optimal biped walking as a mixed-integer quadratic program,” in *Submitted to Advances in Robot Kinematics*, J. Lenarcic and B. Roth, Eds. Springer, 2014.
- [13] S. Barthelemy, J. Salini, and A. Micelli, “Arboris-python.” [Online]. Available: <https://github.com/salini/arboris-python>
- [14] J. Salini, V. Padois, and P. Bidaud, “Synthesis of complex humanoid whole-body behavior: a focus on sequencing and tasks transitions,” in *Proc. of the IEEE ICRA*. IEEE, 2011, pp. 1283–1290.
- [15] G. Sandini, G. Metta, and D. Vernon, “The icub cognitive humanoid robot: An open-system research platform for enactive cognition,” in *50 Years of Artificial Intelligence*, ser. Lecture Notes in Computer Science. Springer, 2007, ch. 32, pp. 358–369.
- [16] I. Gurobi Optimization. [Online]. Available: <http://www.gurobi.com>

¹on a dual-core i7-2620M for 74 DoFs and 275 constraints in average. The computation time being greater than the timestep, simulation was slowed down in this work. Future works will investigate the implementation of a slower control loop for gait adaptation.

B.5 Variance modulated task prioritization in whole-body control

Variance Modulated Task Prioritization in Whole-Body Control

Ryan Lober¹, Vincent Padois¹ and Olivier Sigaud¹

Abstract—Whole-Body Control methods offer the potential to execute several tasks on highly redundant robots, such as humanoids. Unfortunately, task combinations often result in incompatibilities which generate undesirable behaviors. Prioritization techniques can prevent tasks from perturbing one another but often to the detriment of the lower precedence tasks. For many tasks, static prioritization is not necessary or even appropriate because tasks can often be achieved in variable ways, as in reaching. In this paper, we show that such task variability can be used to modulate task priorities during execution, to temporarily deviate certain tasks as needed, in the presence of incompatibilities. We first present a method for mapping from task variance to task priority and then provide an approach for computing task variance. Through three common conflict scenarios, we demonstrate that mapping from task variance to priorities reactively solves a number of task incompatibilities.

I. INTRODUCTION

Highly redundant robots, such as humanoids or anthropomorphic platforms, provide the capability of executing several tasks simultaneously. Unfortunately, this versatility comes at the cost of control difficulty due to the high dimensionality and undetermined nature of the inverse control problem. Over the past few decades, Whole-Body Control, or WBC, techniques have emerged as effective means of controlling these systems, by allowing multiple tasks to be specified simultaneously and utilizing their full capacity [1], [2], [3], [4].

The execution of multiple tasks can induce unwanted behaviors due to incompatibilities between them. Typically, priorities are used to ensure that safety-critical tasks, such as balancing, remain unperturbed by incompatibilities with uncritical tasks like reaching. These priorities may be strict [4] or soft [3] hierarchies.

While distinctions between tasks deemed safety-critical or not are made with relative ease, discriminating between uncritical tasks is less trivial. In many situations it is impossible to analytically justify the prioritization of one task over another and as a result, tasks priorities are commonly subject to arbitrary manual tuning. In many cases, such static prioritization is too restrictive and can engender additional incompatibilities that could be otherwise avoided. For anthropomorphic or humanoid robots, generating tasks for the end-effectors (EE) is crucial for manipulation and interaction with the robot's environment; however, manipulation tasks are

generally not considered safety-critical and are consequently difficult to prioritize.

EE tasks generally require that the robot pass through one or more waypoints, as in goal reaching. These tasks possess the property of *task redundancy* [5], which implies that there exist infinitely many ways of passing from one waypoint to the next. In practice, trajectories passing through the waypoints are generated and fed to the controller. However, these trajectories do not need to be followed with the same precision as near the waypoints. Works from the field of imitation learning have approached incompatibility resolution for single EE tasks, i.e. external perturbations and poorly formed reference trajectories, by exploiting the demonstrated task's redundancy to regulate the task controller's impedance gains [5]. These studies determine task redundancy from the variance of the movement demonstrations. The task controller is typically some version of the Dynamical Movement Primitive, or DMP, [6] and contains an attractive Proportional-Derivative (PD) term along with a learned forcing term, for example,

$$\xi_{des}(t + \delta t) = K_p \epsilon(t) + K_d \dot{\epsilon}(t) + f. \quad (1)$$

Here $\xi_{des}(t + \delta t)$ is the desired task-space acceleration term, $\epsilon(t)$ and $\dot{\epsilon}(t)$ are the current pose error and its derivative, K_p and K_d , their proportional and derivative gains respectively, and f , the forcing term learned via regression techniques from multiple movement demonstrations [6]. Task variance is measured from the variability of the learned motions and may be adapted based on new demonstrations [7], [8], [9]. An inverse relationship between the task variance and the K_p gain is then formed to regulate the attractor term during the movement. Consequently, when variance is high, the robot is compliant, and when variance is low, the robot is stiff. Variable compliance (a.k.a. gain scheduling), allows the robot to adapt to uncertainties/incompatibilities in its environment. Unfortunately, these conflicts must often be directly integrated into the task controller, making variable compliance, task specific [5]. In addition, the K_p gains may vary by orders of magnitude for a single task [10]. Nevertheless, there is clearly some relationship between the variance of a task and its execution [11].

In this paper, we employ task variance to modulate soft task hierarchies, represented by continuous real valued weights, within a Whole-Body (WB) controller, rendering it more robust to incompatibilities, perturbations and poorly designed reference trajectories. By varying the task weights during execution, the WB controller can temporarily deviate high variance tasks in the presence of incompatibilities on an as-needed basis. This allows the robot to reactively solve a

¹ The authors are with - Sorbonne Universités, UPMC Univ Paris 06, UMR 7222, Institut des Systèmes Intelligents et de Robotique, F-75005, Paris, France - CNRS, UMR 7222, Institut des Systèmes Intelligents et de Robotique, F-75005, Paris, France
e-mail: firstname.lastname@isir.upmc.fr

range of task incompatibilities. We demonstrate how variance can be mapped to task weights for the individual Degrees of Freedom (DoF) of a task and develop a method for computing variance for a task if none is available, as is the case with typical trajectory planners. Finally, we test our variable weighting method in three common incompatibility scenarios on a humanoid robot in simulation.

II. METHODS

In this section, we first give a broad overview of WB task-based hierarchical control. We then propose a method of mapping from a task's variance to its weight in order to modulate its priority over the course of execution. A technique for computing variance for a single task is also developed.

A. Whole-Body Control

WB controllers seek to reactively calculate the joint torques, τ , necessary to minimize a combination of task errors using all of the DoF of the given robot. Task errors can be formulated as the difference between task-space reference commands and their joint-space representations,

$$T_i(\mathbf{q}, \dot{\mathbf{q}}, \xi_i^*, \mathbb{X}) = \left\| \left(J_i(\mathbf{q})\ddot{\mathbf{q}} + \dot{J}_i(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - \xi_i^* \right) \right\|^2. \quad (2)$$

Here $T_i(\mathbf{q}, \dot{\mathbf{q}}, \xi_i^*, \mathbb{X})$ is an acceleration task error, J_i and $\dot{J}_i$, the task Jacobian and its derivative, $[\mathbf{q}, \dot{\mathbf{q}}]$, the joint-space variable states and ξ_i^* the reference task-space acceleration to affect for some frame attached to the robot. The dynamic variable, $\mathbb{X} = [\ddot{\mathbf{q}}^T, \mathbf{w}_e^T, \tau^T]^T$, groups the joint-space accelerations and torques with the external wrenches, $\mathbf{w}_e$. The variable ξ_i^* is commonly provided by a task-level feedforward Proportional-Derivative (PD) controller,

$$\xi_i^*(t + \delta t) = \xi_{des_i}(t + \delta t) + K_p \epsilon_i(t) + K_d \dot{\epsilon}_i(t), \quad (3)$$

where $\xi_{des_i}(t + \delta t)$ is the feedforward frame acceleration term. An optimization problem can then be designed to find the minimum of the weighted sum of n_T task errors, subject to the problem constraints,

$$\begin{aligned} \underset{\mathbb{X}}{\operatorname{argmin}} \quad & \frac{1}{2} \sum_{i=1}^{n_T} w_i T_i + w_0 T_0 \\ \text{subject to:} \quad & G\mathbb{X} \preceq \mathbf{h} \\ & A\mathbb{X} = \mathbf{b}. \end{aligned} \quad (4)$$

The dynamic variable, allows the dynamic equations of motion to be represented as the equality constraint, $A\mathbb{X} = \mathbf{b}$. Inequality constraints such as $G\mathbb{X} \preceq \mathbf{h}$ can account for considerations such as contacts, joint limits, and actuator limits. The importance of each task is governed by its weight w_i , and a regularization task, T_0 , is used to ensure a unique optimization solution with $w_0 \ll w_i$. Varying the task weights, and consequently their priorities, generates joint torque commands which favor the minimization of task errors with higher associated weights¹. Equation (4) can be minimized efficiently using a Linear Quadratic Program. More details on WBC can be found in [12], [2], [3].

¹It is also common to resolve this optimization hierarchically in order of task priority, projecting the lower priority tasks into the null space of the higher priority tasks. [4].

B. Task Formalism

Here we look at Cartesian goal reaching tasks, and without loss of generality, only their translation components are considered.

Each task follows some trajectory, Υ , which passes through one or more waypoints. A trajectory has two components, its path which consists of a series of vectors of spatial coordinates, $\mathbf{r}_i = [x, y, z]$ with $\{i \in \mathbb{N} | 1 \leq i \leq N_r\}$, where N_r is the total number of spatial coordinate vectors, and its temporal evolution, t , which dictates the dynamics of the movement.

Looking at these tasks in a general probabilistic fashion, we can use the position vectors $\mathbf{r}_i$ as the mean, μ_i , of our task trajectory. The variance of the movement at each timestep, $\sigma_\Upsilon^2(t) = [\sigma_{\Upsilon_x}^2(t), \sigma_{\Upsilon_y}^2(t), \sigma_{\Upsilon_z}^2(t)]$, can be obtained through multiple demonstrations² as in [5], [7], [8], [9], or computed from scratch. The concatenation of these position means and variances, respectively yields M_Υ and V_Υ for the given trajectory, Υ .

C. Mapping Variances to Weights

Given a trajectory, Υ , with some variance, V_Υ , we can create a relationship between V_Υ and the task's weight, w_i , at each timestep, t , making the task weights now time and variance dependent within the WB controller. We would like to restrict our variable weight evolution to the $[0.0, 1.0]$ range, therefore all tasks in the WB controller are defined with a baseline weight of 1.0 and we rescale the trajectory variance such that $\{\bar{V}_\Upsilon \in \mathbb{R} | 0 \leq \bar{V}_\Upsilon \leq 1\}$,

$$\bar{V}_\Upsilon = \frac{V_\Upsilon}{\max(V_\Upsilon)}. \quad (5)$$

Equation (5) also ensures that the DoF variances are scaled relative to one another. The variance of each DoF may not be the same, so we map a variance to a weight for each. Therefore, $w_i(\sigma_\Upsilon^2(t))$ becomes the diagonal weight matrix, $W_i(\sigma_\Upsilon^2(t))$, and using a maximum weight factor, β , we can map from variance to weights using this basic approach,

$$W_i(\sigma_\Upsilon^2(t)) = \begin{bmatrix} \frac{1 - \sigma_{\Upsilon_x}^2(t)}{\beta} & 0 & 0 \\ 0 & \frac{1 - \sigma_{\Upsilon_y}^2(t)}{\beta} & 0 \\ 0 & 0 & \frac{1 - \sigma_{\Upsilon_z}^2(t)}{\beta} \end{bmatrix}. \quad (6)$$

Where the variance of the movement is high, $\bar{V}_\Upsilon$ is close to 1 and so the weight/importance of the task diminishes. When the variance is small, $\bar{V}_\Upsilon$ approaches 0 and the importance of the task is at a maximum. The factor β allows us to scale the overall importance of the task relative to the other tasks, while still maintaining variability. For instance, assuming all tasks have a baseline weight of 1, $\beta < 1$ means the variable weight task is less important than the other tasks, while $\beta > 1$ the inverse. This is useful when combining uncritical tasks with safety-critical tasks such as balancing; however, it does not guarantee that the safety-critical task will go unperturbed.

²In these works, the covariances of the forcing term basis functions are used.

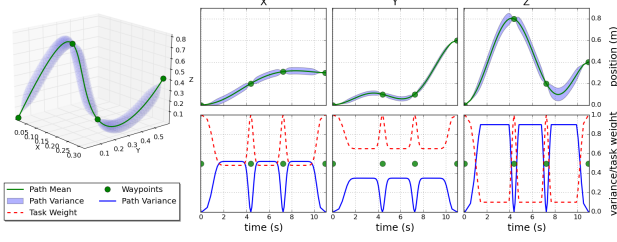


Fig. 1: An example of a 3D task trajectory with variance. This figure shows how variance can be computed given a trajectory, then mapped to the weights of the individual DoF of the task.

In practice, if the variance is too close to 1.0, the weight of the task becomes infinitesimal and the controller no longer executes it. In order to avoid such behavior, the maximum $\bar{V}_\Upsilon$ can be bounded at a value just less than 1.0 (e.g., 0.99 is used in this study).

D. Computing Variance

Historically, task variance has been calculated from multiple demonstrations of the same movement [5], [7]. Unfortunately, demonstration data is not always available, and it is advantageous to be able to compute task variance when we only have one example, as is the case with trajectory generators. Here, we use a covariance function for this purpose.

Covariance functions are commonly used in the field of Gaussian Process Regression (GPR) [13] and allow one to calculate the variance of a new output point in an existing data set by paving the input data space with kernel functions. Here we use Gaussian kernels³:

$$k_i(m) = \sigma_k^2 \exp\left(\frac{-(m - c_i)^2}{2l_k^2}\right). \quad (7)$$

The variable σ_k^2 is the maximum allowable covariance, l_k is the length parameter which influences how much adjacent kernel centers, c_i , influence each other and m is the input value for which we wish to calculate the kernel output. Typically one kernel is centered on each input datum.

Given some new input, m^* , and N_k total kernels, we can calculate the variance of its output as [13],

$$\text{var}(m^*) = K_{**} - K_* K^{-1} K_*^T \quad (8)$$

where,

$$K = \begin{bmatrix} k_1(c_1) & k_2(c_1) & \cdots & k_{N_k}(c_1) \\ k_1(c_2) & k_2(c_2) & \cdots & k_{N_k}(c_2) \\ \vdots & \vdots & \ddots & \vdots \\ k_1(c_{N_\lambda}) & k_2(c_{N_\lambda}) & \cdots & k_{N_k}(c_{N_\lambda}) \end{bmatrix}, \quad (9)$$

$$K_* = [k_1(m^*) \quad k_2(m^*) \quad \cdots \quad k_{N_k}(m^*)] \quad (10)$$

and

$$K_{**} = k_{m^*}(m^*), \quad (11)$$

³Also referred to as the squared exponential in GPR literature.

with $k_{m^*}(m^*)$, a kernel centered and evaluated on the new input, m^* . In this formulation, the kernel centers are points of zero variance, and the variance of the intermediate points is calculated by evaluating (8) between the kernel centers. In terms of goal reaching tasks, variance should be zero at the waypoints meaning that kernel centers should be placed on each one.

Given a single demonstrated trajectory, we can only confidently interpret two waypoints, one at the beginning of the movement and one at the end, based on the assumptions that the trajectory was generated from the EE starting state, and that the final state of the trajectory represents the goal of the movement. If more waypoints are given, such as in the case of programmed trajectories, then they too may be used. The ensemble of waypoints, λ_j , can be indexed by the order in which they are to be attained, $\{j \in \mathbb{N} | 1 \leq j \leq N_\lambda\}$, where N_λ is the total number of waypoints.

We define our kernel centers on the indexes j of the waypoints inferred from the trajectory. We can then create m , our evaluation domain, by resampling the position vectors r_i as r_m such that, $\{m \in \mathbb{R} | 1 \leq m \leq N_\lambda\}$. Now, to calculate the variance of some position r_{m^*} , (8) is evaluated at m^* , the resampled index of r_{m^*} .

For each DoF of the movement, x , y and z we must calculate the kernel parameters, σ_k^2 and l_k . The variance of the position values for each DoF can be used to calculate their individual maximum allowable variances, $\sigma_k^2 = [\sigma_{k_x}^2, \sigma_{k_y}^2, \sigma_{k_z}^2]^T$ using,

$$\sigma_k^2 = \frac{\sum_{i=1}^{N_r} (r_i - \bar{M}_\Upsilon)^2}{N_r - 1}. \quad (12)$$

Again, N_r is the total number of positions vectors, r , and $\bar{M}_\Upsilon$ is the mean of each DoF of the movement. The length parameter, l_k , can be set using,

$$l_k = \frac{N_\lambda}{\alpha_l}, \quad (13)$$

where α_l is some scaling coefficient; here we use $\alpha_l = 10.0$. Figure 1 shows a 3D Cartesian trajectory with 4 waypoints and the variance computed using the aforementioned techniques. Given this variance, we can map to task weights using (6), for each DoF. This is shown by the DoF plots in Fig. 1.

III. EXPERIMENTAL SETUP

To test the efficacy of using variable task weights in a WB controller, three simulated scenarios are presented to highlight some common issues encountered when combining multiple incompatible tasks. In each scenario, a set of tasks is hand-coded for a simulation of the humanoid robot, iCub, which possesses 32 actuated DoF⁴. The XDE physics simulator and environment [14], [15] is used in this study. Successful task combination is characterized as the proximity of the hand task frames to their respective goal locations within a margin of 3.0 cm. This margin is

⁴The real iCub robot has 18 hand DoF and 3 camera DoF that are not modeled in the simulation.

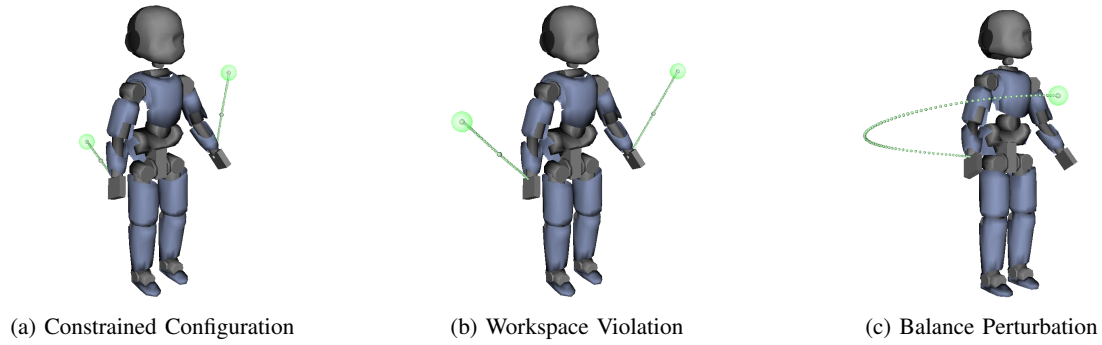


Fig. 2: Three common multi-task incompatibility scenarios. The desired hand task trajectories are indicated by the green markers. Medium size spheres represent waypoints, and large transparent spheres represent the final waypoints or goals.

represented by the large transparent spheres at the end of the hand trajectories in Fig. 2, and has been selected solely to aid in visualization. More precise margins can be applied without loss of generality. We present the execution of these scenarios using both static and variable task weights. The task variances are computed using (8) with a maximum allowable scaled variance of 0.99.

A. Constrained Configuration

In this first scenario, Fig. 2(a), three principle tasks are combined to force the robot into a constrained configuration. Such configurations commonly occur on highly redundant systems when multiple tasks require the same DoF. In humanoid, this often occurs due to solicitation of the torso DoF. The first standing task maintains the center of the robot's waist at a constant height with a static weight of 1.0. The two variable weight tasks are associated with the left and right hands, specifically the center of the base of the palms. These tasks are defined by trajectories passing through waypoints at the beginning, middle and end of the movements. The task objectives are for them to attain the final waypoints, or goal positions, while passing through the other waypoints. The left and right hand tasks last 6.4s and 6.1s respectively, and are executed with $\beta = 1.0$.

B. Workspace Violation

The second scenario, Fig. 2(b), combines the same three primary tasks as in Sec. III-A; however, this time the hand trajectory goal positions are further apart than the maximum workspace of the robot (in this standing configuration). This scenario is designed to represent a typical workspace conflict during picking procedures. The trajectories pass through waypoints at the beginning, middle and end of the movements. The left and right hand tasks last 6.3s and 6.2s respectively, and are executed with $\beta = 1.0$. When one of the hands attains its goal position, that is, within 3.0 cm of the final waypoint, that task is deactivated (i.e. the object has been picked). Task deactivation means that it no longer contributes to the control solution, or equivalently, that its weight is set to 0.0.

C. Balance Perturbation

Here we combine Zero Moment Point (ZMP) balancing [16] with a right hand task (see Fig. 2(c)). The objective of the ZMP balancing task is to maintain the Center of Pressure, or CoP, (x, y) coordinates at $(0, 0)$, its initial position. The right hand follows a sweeping trajectory from the hand's starting waypoint to its end waypoint - no intermediary waypoints are considered. This scenario is meant to replicate activities similar to wiping surfaces. The right hand task lasts 8.6s and is executed with $\beta = 10.0$.

IV. RESULTS

In this section we provide the results of the scenario simulations described in Sec. III. A video presenting these experiments and their results can be found in the attachments of this submission.

A. Constrained Configuration

When the two hand tasks are combined with static weights, we can see in Fig. 3(a) that the left hand task achieves its goal location, contrary to the right hand task. This occurs because individually, the hand tasks require the torso to rotate left and right; therefore, when they are combined this DoF is constrained between the two. The arm DoF attempt to compensate for this reduction in redundancy by moving to their limits, and forcing the robot into a constrained configuration. This is shown in the left arm DoF plots in Fig. 3(c). Consequently, the right hand task is no longer feasible and incurs high task errors at both the middle and goal waypoints due to its combination with the left hand task. This can be observed in the task error plot of Fig. 3(c). The waypoints along the trajectory are indicated by the peaks in the task weight curves in Fig. 3(c).

In Fig. 3(b) the robot has successfully accomplished its tasks through the use of variable weights. By looking at the left arm DoF plots in Fig. 3(c) we can see that the right hand task weight increases approximately 0.25s prior to the left hand weight, forcing the robot to dedicate more DoF to its execution and causing the left arm elbow pitch, shoulder pitch and shoulder roll to deviate. These deviations pull the left arm DoF away from their limit values, freeing these

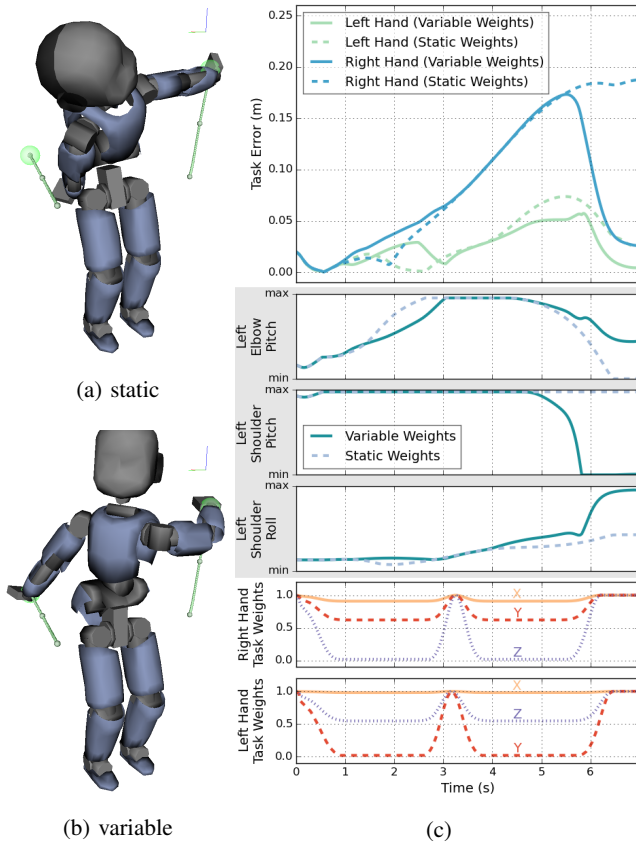


Fig. 3: The constrained configuration scenario. Figures (a) and (b) show the task combination results using static and variable weights respectively. The plots in (c) provide the simultaneous evolution of various task parameters.

articulations for the left hand movement when its weight increases.

B. Workspace Violation

Figure 4(a) shows the static execution of the two hand tasks and although the hands seem to reach their goal positions, close inspection of the distance to goal plot in 4(c) shows that they never attain the 3.0cm error threshold limit. As a result, they rest in a local minimum between their two objectives.

When variable weights are applied to the simultaneous execution of the two hand tasks, the robot achieves its right hand goal first, thereby deactivating the right hand task, and then proceeds to finish the left hand task; this is shown in Fig. 4(b). The instants that the hand tasks are deactivated can be seen in the task error and distance to goal plots, and are indicated by circular markers.

In both the right and left hand movements, the y directional component develops large errors near the goal locations. The errors are roughly equivalent (see Fig. 4(c) static task error plot) and therefore whichever task has the largest w_{y_i} dominates in the WB controller output - the right hand task in this case (see Fig. 4(c) hand task weight

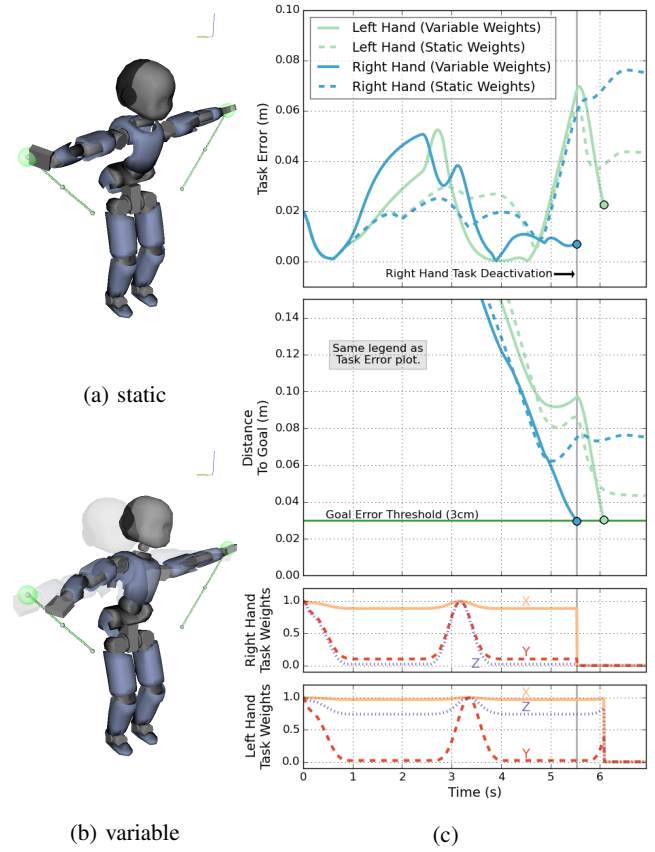


Fig. 4: The workspace violation scenario. See Fig. 3 description for layout details.

plots). Once the right hand task is deactivated, all conflicts are removed and the left hand task is able to recuperate its accumulated error and be deactivated as well.

C. Balance Perturbation

Figures 5(a) and 5(b), show the balance perturbation results. Using static weights for the right hand task results in a loss of balance and ultimately a failure for both tasks; this can be seen in Fig. 5(a). We can confirm this loss of balance by observing that the CoP moves outside of the Polygon of Support, or PoS, in Fig. 5(c). Despite the ZMP balancing being 10x more important than the right hand task, it still fails because the accumulated error at the apex of the sweeping movement generates large enough accelerations in the y direction to perturb the ZMP balancing.

In the variable weight case, we can see in Fig. 5(b) that the robot successfully attains the goal position of the hand task while remaining balanced. The task error plot shows that, the right hand task incurs a large amount of error as in the static case, but because this occurs during a period of high variance, this error only partially perturbs the ZMP balancing. The CoP is deviated somewhat from its goal location in order to compensate for some of the right hand error but it remains safely within the PoS as shown in Fig. 5(c).

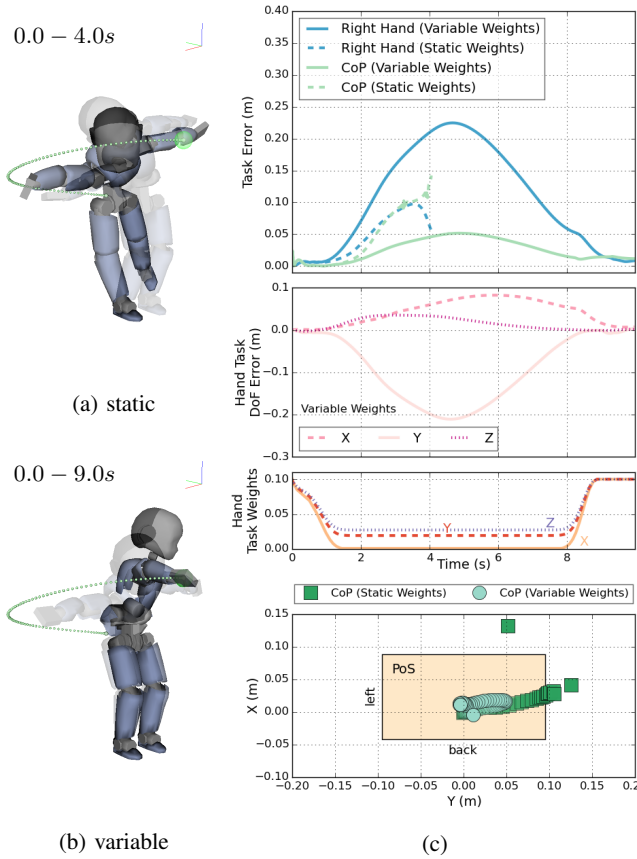


Fig. 5: The balance perturbation scenario. See Fig. 3 description for layout details.

V. CONCLUSION

Regulating task weights based on their variance is a powerful concept, which when coupled with WBC methods, can solve difficult control problems on-line. The use of variable weights diminishes the need for manual tuning of task priorities, and provides WB behaviors which are more robust to incompatibilities, perturbations and poorly designed reference trajectories.

In this paper, we presented a simple technique for utilizing task variance as a means of modulating task weights automatically in a WB controller. These variable weights permit the WB controller to temporarily deviate high variance tasks in the presence of incompatibilities. Through three emblematic scenarios, we showed how variable task weights resolve a broad set of issues encountered in multi-task execution with minimal tuning and in a reactive manner. In addition to the variance to weights mapping, we developed a method of computing variance for a single trajectory demonstration using a covariance function (8). This tool is essential in cases where only one trajectory has been provided for the task, as in trajectory generation.

High task variance allows one to handle conflicts between tasks but provides no guarantee that the tasks will be accomplished. If an incompatibility occurs when all tasks require low variance, or high priority, then our method will not work

and some form of planning must occur. In [17], we show that by optimizing tasks over their entire execution, we can ensure task completion; however, this method is time consuming. In the future, we will investigate how to combine such global optimization methods with variance modulated weighting, to provide a fast and robust task control framework which can assure task realization.

ACKNOWLEDGMENTS

This work was partially supported by the European Commission, within the CoDyCo project (FP7-ICT-2011-9, No.600716) and by the RTE company through the RTE/UPMC chair Robotics Systems for field intervention in constrained environments held by Vincent Padois.

REFERENCES

- [1] M. de Lasa and A. Hertzmann, "Prioritized optimization for task-space control," *IEEE International Conference on Intelligent Robots and Systems*, vol. 3, no. 2, pp. 5755–5762, Oct 2009.
- [2] L. Saab, N. Mansard, F. Keith, J.-Y. Fourquet, and P. Souères, "Generation of dynamic motion for anthropomorphic system under prioritized equality and inequality constraints," in *IEEE International Conference on Robotics and Automation*, Shanghai, China, May 2011.
- [3] J. Salini, V. Padois, and P. Bidaud, "Synthesis of complex humanoid whole-body behavior: a focus on sequencing and tasks transitions," in *IEEE International Conference on Robotics and Automation*, 2011.
- [4] A. Escande, N. Mansard, and P.-B. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," *The International Journal of Robotics Research*, 2014.
- [5] S. Calinon, I. Sardellitti, and D. G. Caldwell, "Learning-based control strategy for safe human-robot interaction exploiting task and robot redundancies," in *IEEE International Conference on Intelligent Robots and Systems*, Oct 2010, pp. 249–254.
- [6] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical movement primitives: learning attractor models for motor behaviors," *Neural computation*, vol. 25, no. 2, pp. 328–73, Feb 2013.
- [7] P. Kormushev, S. Calinon, and D. G. Caldwell, "Approaches for learning human-like motor skills which require variable stiffness during execution," in *IEEE International Conference on Humanoid Robots*, 2010.
- [8] F. Stulp, J. Buchli, E. Theodorou, and S. Schaal, "Reinforcement learning of full-body humanoid motor skills," *IEEE-RAS International Conference on Humanoid Robots*, pp. 405–410, Dec 2010.
- [9] J. Buchli, F. Stulp, E. Theodorou, and S. Schaal, "Learning variable impedance control," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 820–833, April 2011.
- [10] S. Calinon, D. Bruno, and D. G. Caldwell, "A task-parameterized probabilistic model with minimal intervention control," in *IEEE International Conference on Robotics and Automation*, May 2014.
- [11] E. Todorov and M. I. Jordan, "Optimal feedback control as a theory of motor coordination," *Nature neuroscience*, vol. 5, no. 11, pp. 1226–35, Nov 2002.
- [12] O. Kanoun, F. Lamiroux, P.-B. Wieber, F. Kanehiro, E. Yoshida, and J.-P. Laumond, "Prioritizing linear equality and inequality systems: application to local motion planning for redundant robots," in *IEEE International Conference on Robotics and Automation*, 2009.
- [13] C. E. Rasmussen and C. Williams, *Gaussian processes for machine learning*. MIT Press, 2006.
- [14] X. Merhiot, J. L. Garrec, G. Saupin, and C. Andriot, "The xde mechanical kernel: Efficient and robust simulation of multibody dynamics with intermittent nonsmooth contacts," in *Joint International Conference on Multibody System Dynamics*, 2012.
- [15] H. Sovannara. (2013, Dec) Xde-isir wiki. [Online]. Available: <http://pages.isir.upmc.fr/~hak/xdewiki/doku.php?id=start>
- [16] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, "Biped walking pattern generation by using preview control of zero-moment point," in *IEEE International Conference on Robotics and Automation*, vol. 2. IEEE, 2003, pp. 1620–1626.
- [17] R. Lober, V. Padois, and O. Sigaud, "Multiple task optimization using dynamical movement primitives for whole-body reactive control," in *IEEE International Conference on Humanoid Robots*, 2014, pp. 1–6.

Control and design of robots with tasks and constraints in mind

Abstract: The work presented in this dissertation is mostly concerned with the problem of controlling robots. While a large part of the scientific literature in Robotics is dedicated to this problem, there still exists a gap between what has been proposed using advanced control techniques and the majority of existing applications. Indeed, in simulation or in lab conditions, ad hoc environment and situations can be generated in order to simplify the control problem and only address one of its sub-parts. This can be necessary in the preliminary stages of research, when trying to address a challenging problem. However, the risk is to provide solutions which are intrinsically incompatible with real life constraints. These constraints often constitute strong non-linearities which have to be accounted for to maintain the robot and its surrounding environment in proper working conditions. Nevertheless, these limits have often been dealt with as exceptions or secondary objectives. If these approaches make sense in cases where all motions can be pre-planned, they do not apply to situations where motions have to be generated in reaction to the environment. In that sense, there is still a need for advanced control techniques that intrinsically account for real world constraints. These constraints both lie in the actuation and configuration spaces of the robot but also in the tasks spaces, i.e. in spaces where the tasks assigned to the robot can be expressed in a straightforward way by the programmer or by the user through interactive programming. Thus, robotics control paradigms have to provide the means to naturally express and optimally perform concurrent tasks while natively accounting for constraints in a computationally efficient way.

In an attempt to tackle this general problem, the first research direction presented in this dissertation describes some work performed in constraints compatible, multi-tasks robot control. The proposed contribution in that domain leads to the formulation of the robotics control problem as a constrained optimisation one. This formulation provides the benefit of computing locally optimal solutions intrinsically compatible with the constraints. This is particularly suitable for applications in constrained and dynamic environments.

While the first chapter advocates for the formulation of the reactive control problem as an optimisation one, the limits of reactive approaches are quite tangible when dealing with the constraints compatibility problem. Global optimality has to be tackled despite its complexity. This complexity reaches its climax with humanoid robots and the second contribution presented in this dissertation is centred on a review of optimisation-based control approaches for balanced humanoid behaviours.

Control is the central focus of the presented work. Nonetheless, designing the right robot for a given application is also a very interesting and important research topic if one envisions less traditional use of robots with respect to standard manufacturing applications. Thus, the final contribution presented in this document is related to the performance based design and evaluation of robots with collaborative applications in mind.

Keywords: robotics, redundant systems, tasks and constraints, model-based control, whole-body control, collaborative robotics, humanoid robotics
