



HAL
open science

Balance control of robots in compliant contacts

Marwan Hamze

► **To cite this version:**

Marwan Hamze. Balance control of robots in compliant contacts. Automatic. Université Paris-Saclay, 2023. English. NNT : 2023UPAST047 . tel-04196579

HAL Id: tel-04196579

<https://theses.hal.science/tel-04196579>

Submitted on 5 Sep 2023

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.

Balance control of robots in compliant contacts

Contrôle d'équilibre des robots en contacts compliant

Thèse de doctorat de l'université Paris-Saclay

École doctorale n° 580, Sciences et technologies de l'information et de
la communication (STIC)

Spécialité de doctorat : Robotique

Graduate School : Sciences de l'ingénierie et des systèmes.

Référent : Université de Versailles-Saint-Quentin-en-Yvelines

Thèse préparée dans l'unité de recherche **LISV** (Université Paris-Saclay, UVSQ),
sous la direction de **Abdelaziz Benallegue**, professeur, et le co-encadrement
de **Mehdi Benallegue**, docteur.

Thèse soutenue à Vélizy-Villacoublay, le 11 Avril 2023, par

Marwan Hamze

Composition du jury

Membres du jury avec voix délibérative

Gentiane Venture

Professeure, Université de Tokyo

Florent Lamiroux

Directeur de Recherche, LAAS-CNRS

Rochdi Merzouki

Professeur des universités, Polytech Lille CNRS

Andrea Cherubini

Professeur des universités, LIRMM

Présidente

Rapporteur & Examineur

Rapporteur & Examineur

Examineur

Titre : Contrôle d'Équilibre Des Robots en Contacts Compliant

Mots clés : Humanoïdes, Multi-contacts, Compliance, Modélisation, Contrôle en force, LQR, QP

Résumé : Le contrôle d'équilibre des robots reste un problème complexe qui nécessite des recherches supplémentaires, en particulier dans le cas des robots humanoïdes. Ceci est particulièrement vrai lorsque l'interaction robot-environnement est souple, par exemple lorsque les surfaces de contact sont faites de matériaux mous. L'hypothèse classique d'interaction rigide échoue dans ce cas, car le contrôleur construit sur cette hypothèse ne réussit pas à suivre les forces requises pour l'équilibre du robot. Certains travaux dans la littérature tentent de compenser l'effet de compliance, mais cela ne fonctionne que pour des effets de compliance relativement faibles. D'autres se sont appuyés sur la modélisation de l'environnement compliant, mais leurs approches ne fonctionnent que sur des robots spécifiques contrôlés en couple.

Cette thèse traite le contrôle de l'équilibre des robots pendant l'interaction compliant avec l'environnement, et propose une stratégie de contrôle qui est applicable à différents types de robots. Un modèle réduit utilisant le modèle visco-élastique des forces est proposé pour étudier la dynamique imposée par l'interaction compliant sur le robot. Le modèle prend un nombre quelconque de contacts avec l'environnement, ce qui le rend utilisable sur des robots quadrupèdes ainsi que sur des robots humanoïdes avec des configurations de contact multiples. Un processus de linéarisation est utilisé pour linéariser le système non linéaire, et une

représentation de l'espace d'état est utilisée pour représenter la dynamique linéaire. Pour suivre les forces et les moments au niveau des contacts avec une bonne précision, les données des capteurs de force sont utilisées dans un compromis avec la cinématique des contacts. Un programme quadratique linéaire est utilisé pour générer des signaux d'accélération des membres du robot en contact avec l'environnement, nécessaires pour minimiser l'erreur d'état. Les signaux d'accélération sont ensuite incorporés dans un programme quadratique, qui est utilisé pour générer le mouvement du robot en question.

Une première validation a été effectuée en simulation sur un bipède à 26 degrés de liberté contrôlé en couple, dans des situations à deux contacts et à contacts multiples. Les simulations ont été effectuées dans MATLAB SIMULINK et ont montré comment le contrôleur proposé se compare à un contrôleur classique supposant une interaction rigide, et la robustesse du contrôleur a été évaluée. Une deuxième validation a été effectuée sur des robots contrôlés en position, de type HRP4 en simulation à l'aide de l'environnement de simulation Choreonoid, et sur un vrai robot de type HRP2-KAI. Le contrôleur a été formulé en C++ à l'aide du contrôle framework mc_rtc. Le contrôleur peut maintenir l'équilibre du robot lorsque le robot a une compliance intégrée et lorsqu'il se tient debout sur un matelas mou en même temps.

Title : Balance of Robots in Compliant Contacts

Keywords : Humanoids, Multi-contacts, Compliance, Modeling, Force Control, LQR, QP

Abstract : The balance control of robots is still a complex problem that requires additional research, especially in the case of humanoid robots. This is particularly true when the robot-environment interaction is compliant, such as when the contact surfaces are made of soft material. The classical rigid interaction assumption fails in such case, as the controller built on this assumption struggles to produce the required forces for the robot's balance. Some works in the literature try to compensate for the compliance effect, but this works only with relatively small compliance effects. Others have relied on modeling the compliant environment, but their approaches only work on specific torque-controlled robots.

This thesis deals with the balance control of robots during compliant interaction with the environment, and proposes a control strategy that is applicable to different types of robots. A reduced model using the viscous-elastic model of the forces is proposed to study the dynamics imposed by the compliant interaction on the robot. The model takes any number of contacts with the environment, which make it usable on quadruped robots as well as humanoid robots with multiple contact configurations. A linearization process is used to linearize the non-linear system, and a state-space

representation is used to represent the linear dynamics. To follow the desired contact forces and moments, the force sensor data is used in a trade-off with the contacts' kinematics. To minimize the error of the state, a linear quadratic program is used to generate the required command signal for this minimization, consisting of acceleration signals of the robot's limbs in contact with the environment. The acceleration signals are then incorporated into a whole body quadratic program, which is used to generate the motion of the robot in question.

A first validation was performed in simulation on a 26 degrees of freedom torque-controlled biped, in two contacts and in multiple contacts situations. Simulations were run in MATLAB SIMULINK and showed how the proposed controller compares to a classical controller assuming rigid interaction, and the robustness of the controller was gauged. A Second validation was performed on position-controlled robots, of type HRP4 in simulation using the Choreonoid simulation environment, and on a real HRP2-KAI robot. The controller was formulated in C++ using the mc_rtc control framework. The controller can maintain the robot's balance when the robot has a built-in compliance and while standing on a soft mattress at the same time.

Résumé

Le contrôle d'équilibre des robots reste toujours un problème complexe qui nécessite des recherches supplémentaires, en particulier dans le cas des robots humanoïdes. Ceci est particulièrement vrai lorsque l'interaction robot-environnement est souple ou non-rigide, par exemple lorsque les surfaces de contact sont faites de matériaux mous. L'hypothèse classique d'interaction rigide échoue dans ce cas, parce que les contrôleurs développés sur cette hypothèse ne réussissent pas à suivre les forces requises pour l'équilibre du robot. Certains travaux dans la littérature tentent de compenser l'effet de compliance, limités sur des effets de compliance relativement faibles. D'autres se sont appuyés sur la modélisation de l'environnement compliant, mais leurs approches ne fonctionnent que sur des robots spécifiques contrôlés en couple.

Cette thèse traite le contrôle de l'équilibre des robots durant l'interaction compliant avec l'environnement, et propose une stratégie de contrôle qui sera applicable à différents types de robots. Un modèle réduit du robot utilisant le modèle visco-élastique des forces est proposé pour étudier la dynamique imposée par l'interaction compliant sur le robot. Le modèle prend un nombre quelconque de contacts avec l'environnement, ce qui le rend utilisable sur des robots quadrupèdes ainsi que sur des robots humanoïdes avec des configurations de contact multiples. Un processus de linéarisation est utilisé pour linéariser le système non linéaire, et une représentation de l'espace d'état est utilisée pour représenter la dynamique linéaire, simple pour la synthèse de la loi de contrôle. En plus, pour suivre les forces et les moments au niveau des contacts avec une bonne précision, les données des capteurs de forces sont utilisées dans un compromis avec la cinématique des contacts. Un programme quadratique linéaire est utilisé pour générer des signaux d'accélération des membres du robot en contact avec l'environnement, nécessaires pour minimiser l'erreur d'état. Les signaux d'accélération sont ensuite incorporés dans un programme quadratique, qui est utilisé pour générer le mouvement du robot en question, en considérant les différentes contraintes physiques des actionneurs du robot, et de collision.

Une première validation a été effectuée en simulation sur un bipède à 26 degrés de liberté contrôlé en couple, dans des situations à deux contacts et à contacts multiples. Les simulations ont été effectuées dans MATLAB SIMULINK et ont montré comment le contrôleur proposé se compare à un contrôleur classique supposant une interaction rigide, et la robustesse du contrôleur a été évaluée. Une deuxième validation a été effectuée sur des robots humanoïdes contrôlés en position, de type HRP4 en simulation, en utilisant l'environnement de simulation Choreonoid, où la dynamique de l'interaction souple entre le robot et l'environnement peut être simulée. Le contrôleur a été formulé en C++ à l'aide du contrôle framework `mc_rtc`. Les simulations montrent l'efficacité du contrôleur même dans des situations de contacts multiples.

Finalement, des expérimentations ont été effectuées sur un vrai robot humanoïde de type HRP2-KAI avec le contrôleur proposé. Ce robot est construit avec des flexibilités internes (ressorts entre la semelle et la cheville de chaque pied) pour absorber l'impact de l'interaction avec l'environnement, créant une interaction non-rigide qui vise à protéger les capteurs des forces. Le contrôleur a réussi de maintenir l'équilibre du robot lorsque qu'il est soumis à des perturbations successives, malgré cette compliance intégrée. Des validations supplémentaires ont été obtenus, lorsque le robot se tient debout sur un matelas mou qui rend l'interaction beaucoup plus souple. L'importance de retour des capteurs des forces dans le cas des erreurs de modèles assez élevés était bien montré, lorsque le contrôleur ne peut réussir de maintenir l'équilibre du robot qu'avec ce retour, sans lequel le robot tombe vers l'arrière même sans perturbations.

Acknowledgements

As I was writing this document, I could not stop thinking about my entire journey with this thesis during the past few years. Looking back, I encountered a lot of hurdles, some of which were demotivating at times. However, there was nothing that made me give up or lose confidence in myself, which is why I kept pushing myself in order to have fruitful results.

The reason why I always kept going is simple. It is because I sincerely enjoyed this work. It felt good to overcome difficulties and understand why my code does not work, and it felt even better to produce a good result. The numerous meetings and discussions with my supervisors and colleagues were exciting and enjoyable. Looking back at when I started my thesis, I have improved my soft skills quite a lot. And it is for this reason that I will always be grateful to everyone who assisted me or supported me during this thesis.

I want to send my sincerest thanks to Prof. Samer Al Fayad for accepting me in his research project at first, and for accepting my decision to change my research topic.

I want to express my honest and deepest gratitude to my supervisors, Prof. Abdelaziz Benallegue and Dr. Mehdi Benallegue, who were always there to answer any question I have, and were always working hard for my sake so that I could have the best possible results. I will always be grateful to Prof. Benallegue for accepting me to work on this thesis, and for all the aid he was able to give me to continue working on my thesis.

I am in big debt to Dr. Benallegue, as he was the one to propose the novelty of my work, a project that he started thinking about a few years before I started my thesis. He never hesitated to launch a meeting for discussions, despite living in a timezone eight hours ahead of France's, even when it was as late as 23:00 for him! I would like to thank him sincerely for the invitation to the JRL during the last two months of my thesis.

I am also grateful to my colleagues and to the administration in LISV, where I spent the majority of my thesis, and where I have received a lot of support.

I want to send enormous thanks to Prof. Abderrahmane Kheddar, who accepted me to spend some time of my thesis in LIRMM. I had a great time discussing with my colleagues

over there, and I eternally appreciate all the advice and help that I have received from every single one of them.

I would like to thank my colleagues in the JRL lab in Japan, where I tested my work on a real humanoid robot, a time when I enjoyed working on my thesis the most. I would like to give additional thanks to Dr. Rafael Cisneros Limón for constantly helping me in my work, even during the time when I was in France.

Finally, a special thanks to my family and to my close relatives, who were always praying for my success, and supported me all the way through.

Contents

| | |
|--|-----------|
| Introduction | 1 |
| 1 Background and State of the Art | 7 |
| 1.1 Introduction | 7 |
| 1.2 Compliance | 8 |
| 1.3 Tackling the Robot's dynamics | 10 |
| 1.3.1 Centroidal Model | 11 |
| 1.3.2 Linear Inverted Pendulum Mode | 11 |
| 1.3.3 Viscous-Elastic Model of the Contact Forces | 12 |
| 1.4 Balance Criteria | 13 |
| 1.4.1 Contact Stability | 15 |
| 1.4.2 Force Control | 16 |
| 1.5 Multiple Contacts Situations | 19 |
| 1.6 Works Related to Balance | 20 |
| 1.7 A Unified Balance-Force Control for Compliant Interactions | 26 |
| 1.8 Conclusion | 27 |
| 2 Dynamics of Compliant Robot-Environment Interaction | 29 |
| 2.1 Introduction | 29 |
| 2.2 Full Rigid Body Dynamics | 30 |
| 2.3 Dynamics of the Compliant Interaction | 32 |
| 2.3.1 Dynamics of a Simple Monopod | 32 |
| 2.3.2 Dynamics Study in the General Case | 35 |
| 2.4 Linearization of the Non-linear System | 40 |
| 2.5 Trade-off Between Force Feedback and Kinematics | 41 |
| 2.5.1 Definitions | 42 |

| | | |
|----------|--|-----------|
| 2.5.2 | Dynamics of the New State | 43 |
| 2.6 | Conclusion | 43 |
| 3 | Architecture of the Balance-Force Control | 45 |
| 3.1 | Introduction | 45 |
| 3.2 | Optimal Control Law for the Linear Dynamics | 45 |
| 3.2.1 | Linear Quadratic Regulator | 46 |
| 3.2.2 | Testing the Control | 47 |
| 3.3 | Quadratic Program Motion Solver | 49 |
| 3.3.1 | Optimization Problem | 50 |
| 3.3.2 | Objectives and Constraints | 52 |
| 3.4 | Closed Loop Control Architecture | 55 |
| 3.5 | Conclusion | 57 |
| 4 | Validation on a Torque Controlled Biped | 59 |
| 4.1 | Introduction | 59 |
| 4.2 | SIMULINK Simulation Environment | 59 |
| 4.3 | Validation Scenarios and Results | 61 |
| 4.3.1 | Case of 2 Contacts | 61 |
| 4.3.2 | Cases of 4 Contacts | 64 |
| 4.4 | Conclusion | 66 |
| 5 | Validation on Position Controlled Humanoid Robots | 71 |
| 5.1 | Introduction | 71 |
| 5.2 | mc_rtc and Choreonoid | 72 |
| 5.3 | Details Concerning the Implementation of the Controller | 73 |
| 5.3.1 | Estimation | 73 |
| 5.3.2 | Frame Transformations and Additional Tasks | 74 |
| 5.3.3 | LQR Gain Calculation | 76 |
| 5.4 | Results | 76 |
| 5.4.1 | Simulations on HRP4 | 76 |
| 5.4.2 | Experiments on HRP2-KAI | 85 |
| 5.5 | Conclusion | 90 |
| | Conclusion and Future Work | 95 |

A Details of the Linearization Process 99

B Correspondence of the Stiffness and Damping of the contacts between Simulink and the proposed model 105

 B.1 Linear Stiffness and Damping 105

 B.2 Angular Stiffness and Damping 107

 B.3 Additional Calculations 108

C General Guide to the Tuning of the Controller 111

List of Figures

| | | |
|------|---|----|
| 1.1 | Preemptive foot compliance in the HRP-2 Kai robot. [Guadarrama-Olvera et al., 2022] | 9 |
| 1.2 | Example of a soft surface made of foam for motion tests on soft surfaces. [Bosworth et al., 2016] | 10 |
| 1.3 | Illustration of the linear inverted pendulum mode. [Kajita et al., 2005] | 12 |
| 1.4 | Illustration of the viscous-elastic model, via a spring with a stiffness K_p and a damper with a damping constant K_d | 13 |
| 1.5 | Support polygons for the CoM and ZMP regions in a static state (left) and in a dynamic state (right). [Kajita et al., 2005] | 15 |
| 1.6 | Simple diagram showing a simple use of admittance control. | 17 |
| 1.7 | Simple impedance control diagram. | 18 |
| 1.8 | Various manipulation strategies by the humanoid robot HRP-2 while maintaining balance [Murooka et al., 2017]. | 20 |
| 1.9 | RoboCat-1, capable of running while relying on force sensors for force control [Kotaka et al., 2013]. (a) shows the actual robot, (b) shows its CAD model, (c) shows a simulation environment | 21 |
| 1.10 | MIT Cheetah, running at 6 m/s [Seok et al., 2015] | 22 |
| 1.11 | HyQ balancing on rough terrain [Barasuol et al., 2013] and Big Dog descending on rubble [Raibert et al., 2008] | 22 |
| 1.12 | ANYmal walking on uneven terrain, trotting and climbing stairs [Hutter et al., 2016] | 23 |
| 1.13 | HyQ traversing multiple soft terrains of different compliances [Fahmi et al., 2019] | 23 |
| 1.14 | HRP-4 Walking on Pavement (left) [Kajita et al., 2010] and Climbing Stairs (right) [Caron et al., 2021] | 24 |

| | |
|---|----|
| 1.15 Multiple contact motion: by rolling a bobbin (left) [Murooka et al., 2021] and climbing ladders (right) [Kanazawa et al., 2015] | 25 |
| 1.16 Locomotion through mud [Kuindersma et al., 2014] | 26 |
| 1.17 Careful locomotion over a mattress [Mesesan et al., 2019] | 27 |
| 2.1 Illustration of a 1 d.o.f robot making a compliant contact | 33 |
| 2.2 Reduced model of the robot making 3 compliant contacts with a viscous-elastic interaction with the environment | 37 |
| 3.1 CoM position error (x component) | 49 |
| 3.2 Second Contact position error (x component) | 50 |
| 3.3 Contact Force error (z component) | 51 |
| 3.4 Contact Moment error (x component) | 52 |
| 3.5 Illustration of the implementation of the controller. The robot's model is reduced, and the controller is formulated based on the reduced model. Accelerations are formulated as objectives in the motion solver, which generates the robot's motion. | 53 |
| 3.6 Diagram showing the control loop with a torque controlled robot. The yellow block represents the LQR control used on the linearized dynamics of the reduced model. | 56 |
| 3.7 Diagram showing how the controller is implemented with a position controlled robot. | 57 |
| 4.1 Starting postures of scenario (a) (left), (b) (middle) and (c) (right) | 62 |
| 4.2 CoM position error (x component) in scenario (a) | 64 |
| 4.3 Right foot CoP (x component) in scenario (a) | 65 |
| 4.4 RF friction ratio along the x and y axis in scenario (a) | 66 |
| 4.5 CoM position error (x component) for different modeling errors | 67 |
| 4.6 CoM position error (x component) in scenarios (b) and (c) | 68 |
| 4.7 Right foot and left hand CoP (x component) in scenario (b) | 68 |
| 4.8 Right foot and right hand CoP (x component) in scenario (c) | 69 |
| 5.1 mc-rtc with Choreonoid: The left window is for Choreonoid, the top right is for the terminal, the bottom right is for the control panel RViz | 73 |
| 5.2 Control architecture when the controller is implemented in mc_rtc with a position controlled robot. | 75 |

| | |
|--|-----|
| 5.3 Scenarios 1 (left) with rigid contacts and 2 (right) with deformable contacts | 77 |
| 5.4 CoM error in Scenario 1 | 79 |
| 5.5 CoP of the RF in Scenario 1 | 80 |
| 5.6 CoM error in Scenario 2 | 81 |
| 5.7 CoP of the RF in Scenario 2 | 82 |
| 5.8 CoM error in Scenario 2 under successive pushes | 82 |
| 5.9 CoP of the RF in Scenario 2 under successive pushes | 83 |
| 5.10 Friction (in x and y directions) of the RF in Scenario 2 under successive pushes | 84 |
| 5.11 Scenario 3 with 3 deformable contacts | 85 |
| 5.12 CoM error in Scenario 3 | 86 |
| 5.15 HRP2-KAI on rigid floor (left) and on a soft mattress (right) | 86 |
| 5.13 CoP of the RF in Scenario 3 | 87 |
| 5.14 Normal force error at the RH in Scenario 3 | 88 |
| 5.16 CoM error of HRP2-KAI on rigid floor while resisting pushes | 89 |
| 5.17 CoP of the RF of HRP2-KAI on rigid floor while resisting pushes | 90 |
| 5.18 CoM error of HRP2-KAI on a mattress while resisting pushes | 91 |
| 5.19 CoP of the RF of HRP2-KAI on a mattress while resisting pushes | 92 |
| 5.20 CoM error of HRP2-KAI on a mattress with modeling errors while resisting pushes | 92 |
| 5.21 CoP of the RF of HRP2-KAI on a mattress with modeling errors while resisting pushes | 93 |
| B.1 Illustration showing the compliant force model in Matlab Simulink (Left) and the compliance of the proposed model on the contact level (right). A corre- spondence between the parameters is required for the implementation of the controller. | 106 |

List of Tables

| | | |
|-----|--|----|
| 3.1 | Contacts positions, orientations, stiffness and damping as used in the test . . . | 47 |
| 4.1 | Stiffness and damping for each foot in a 2-contact case | 62 |
| 4.2 | Stiffness and damping for each contact in Scenarios (b) and (c) | 65 |
| 4.3 | Max absolute value of the friction ratio at each contact | 67 |
| 5.1 | Stiffness and damping for each foot in a rigid 2-contact case | 78 |
| 5.2 | Real and controller's stiffness/damping for each foot in Scenario 2 | 79 |
| 5.3 | Real and controller's stiffness/damping for each of the 3 contacts in Scenario 3 | 84 |

Introduction

Our human ancestors have existed on this planet for a couple of millions of years, and like other species that have existed on this planet, their purpose in life was centered around their survival. As our species continued to evolve, our ancestors wanted to improve their chances of survival, so they started to use tools made of stones, to hunt their prey, or to start a fire. Then, about twelve thousand years ago, our ancestors started the agricultural revolution, and started to develop the earliest societies. From that point onward, humans have taken a serious approach to developing various tools and technologies, not only to increase their chances of survival, but to make life easier and safer, and to reduce the difficulty of labor.

Unfortunately, our history was not always about the collective well-being and ease of living of the human race; rather many individuals were self-centered, as they cared about their quality of life and not about others. Eventually, they found a way to reduce one's labor while leading a safe life: slavery. In other words, let another human do your work with little to no compensation.

While slavery continued in different parts of the world until the 20th century, it did not stop the intellectual thinking of developing new tools and technologies. On the contrary, it might have actually helped those who oppose slavery to develop alternatives to abolish it, and the idea of "let someone else do your work" might have been an inspiration for the emergence of "automatons", or in other words, self-operating machines. Archytas, a mathematician from the fourth century B.C., postulated a mechanical bird he called "The Pigeon", which was propelled by steam. The Greek philosopher Aristotle speculated in his writings that automata could probably someday bring about human equality by making possible the abolition of slavery:

"There is only one condition in which we can imagine managers not needing subordinates, and masters not needing slaves. This condition would be that each instrument could do its own work, at the word of command or by intelligent anticipation..."

As scientists and philosophers were figuring out how to make machines that could do

someone's work in their stead, some of them succeeded in designing many mechanical systems that were capable of doing simple tasks, most notably Al Jazari, who is sometimes called in our present day "the father of robotics and modern day engineering", mainly because of his book "Book in knowledge of engineering tricks" that he wrote in 1206, where he described 50 mechanical devices, along with instructions on how to construct them. And it wasn't long before the term "robot" first appeared in a play published by the Czech Karel Čapek in 1921. The term "Robot" is actually drawn from an old Church Slavonic word, "robota", for "servitude," or "forced labor". This marked the point when researchers started to refer to the re-programmable industrial machines that are meant to do humans' work as robots.

The first modern-day robots were industrial robots, that came to life in the early 1960s, and were used in car manufacturing. Industrial robots flourished throughout the 20th and 21st centuries in different forms of production, and nowadays, where it is estimated that more than 3 million industrial robots are in use, they are capable of performing a wide array of complex tasks, such as painting, welding, complex product assembly, and even surgery.

However, the problem with industrial robots is that they do not actually move, instead they can only be confined in industrial factories or laboratories, and moving them to other places can be complicated.

Wheeled robots, or robots mounted on wheeled platforms, have been an option to easily move the robot, or let it move on its own to a specific place. While wheels might even provide more speed for whole-body motion compared to artificial feet, their use is limited to roads and even terrains, and they are ineffective for climbing stairs or ladders, and struggle a lot in uneven terrains and narrow places.

This is why other types of robots were developed in parallel, mainly animaloids (robots with the shape of animals) and humanoids (robots with human shape), which fall under the category of floating base systems.

These robots are intended to move freely through the environment, regardless of the shape of the terrain, and accomplish various tasks at the same time. This will enable them to extend their array of tasks to include carrying objects for distances, or even accomplishing tasks that can be dangerous for humans, most notably rescue missions. However, this is where the first challenge emerges: how can these mechatronic systems move in a challenging terrain without losing balance and colliding with themselves?

We as human beings have adapted to our environment, and thanks to our biological features, such as nerves, muscle tissue, and most importantly, self-awareness, we are able to move our limbs and apply the appropriate interaction force in the environment in order to shape our movements safely. Based on these facts, researchers have been working on

replicating our human characteristics and implementing them on these robots. To replicate our nervous system, they ought to use integrated circuits that can provide information such as forces applied into the environment, otherwise known as sensors. To replicate our joints, they designed an actuating system consisting of motors that allow the control of the orientation of body parts via the applied torque. To figure out what kind of force to apply in order to make a stable movement possible, they ought to use control theory.

The objective within this use of control theory is to develop a model or algorithm to drive the robot to a desired state, while ensuring a level of control stability, meaning that the algorithm does not provide an output out of control. During the last few decades, diverse controllers have been developed for quadruped robots' locomotion, ensuring balanced locomotion over various types of terrain, and research is still underway to find the best control strategy to achieve the optimal behavior.

With that being said, controlling robots can still be a complex problem to tackle, especially when dealing with humanoid robots, due to the complexity of their design and limitations in terms of actuation (whether it's the number of actuators that a robot has, or the actual torque that each actuator is capable of generating). Humanoid robots can easily lose their balance and fall to the ground if the necessary criteria for equilibrium is not met, even if the control itself is stable. Many works have tackled the problem of balance and provided different solutions, including relying on the robot's additional limbs, such as hands, to make multiple contacts with the environment, to ensure balance and to generate a safe motion. However these solutions are still limited to very few scenarios and require a number of conditions and assumptions.

One of these assumptions is that usually the interaction between a robot's limb and a surface is assumed to be rigid with little or without any compliance. This assumption cannot always be met, especially when moving outside the testing laboratory to deploy our robots. For example, the terrain can be muddy or snowy after a storm. In this case, the rigid interaction assumption is no longer valid, and the robot risks of being submerged into the ground, making the controllers that were working in rigid enough conditions struggle, and eventually the robot loses balance.

While other works in the literature have considered compliance in the interaction between a robot with the environment, the solutions provided are limited to specific types of robots or do not usually consider multiple contact situations. For example, some controllers produce an additional wrench to compensate for the compliance effect, when this compliance is strong enough to invalidate the rigid contact assumption, which is only possible in torque-controlled robots that use joint torque feedback. Others propose simplified dynamics caused by the

presence of compliance which are not accurate enough and can provide good results only with relatively small compliance effect.

The work that I'm going to present in this thesis takes a unique approach to how to manage the balance of robots under compliant robot-environment interaction. Rather than studying the displacement of the CoM due to the overall compliance, I will study the compliance at the contact level of the interaction. I will address the problem by modeling this interaction and studying the contact forces resulting from such interaction, and see how they relate to the CoM in order to track its reference for the robot's balance. Additionally, I will use a trade-off between the model of the contact forces and the actual measurement of these forces provided by force sensors, to increase the accuracy of force tracking rather than relying on the proposed model on its own. This way, force tracking and balance control are combined in a single feedback loop for the controller in task space.

As for the controller design, an optimal control on the studied dynamics is used, minimizing the state of the robot. The output of this control is going to be used to formulate a task within a motion solver, which takes multiple tasks and physical constraints into consideration for the robot's motion generation. The controller is designed such that it can be used with different types of robots using any number of limbs for the interaction, thanks to the study of the compliant interaction at the contact level.

The controller will be tested on two different bipedal robots to demonstrate the robustness of its stable control under compliant interaction.

The first chapter is going to present the state of the art related to this work. The chapter goes over the different approaches used to treat the balance and locomotion of floating-based robots that assume rigid interactions. It will also present the works that do not use this assumption and treat the problem of environmental compliance, and situate the contribution of this thesis.

The second chapter goes over the dynamics of the robot, and treats the compliant interaction problem. A reduced model of the robot and of the robot-environment interaction will be proposed, and a detailed study of the dynamics of the system is going to be presented. Then, a linearization method to transform the non-linear dynamics into a linear system will be detailed, which can be represented by a state-space formulation.

The third chapter is where the proposed control structure is introduced. An optimal control generates the command signals, which ensure the robot's balance and the tracking of the reference contact forces. Then the controller will be integrated within a whole-body motion solver for the generation of the command for the robot's joints. The chapter details the difference in the control architecture according to how the robot is controlled.

The fourth chapter presents simulation results when testing the proposed controller on a torque-controlled robot, when having compliant interactions with its surrounding. The controller is tested in different conditions, such as when the robot is in multiple contact configurations, and the performance is compared to a classical balance control relying on the rigid interaction assumption.

The fifth chapter presents additional results, this time when applying the controller on a position-controlled robot. A robot is simulated in compliant interactions, and in the simple case when there is an interaction with almost negligible compliance, to showcase the controller's wide usage options. It also presents the results when the controller is successfully applied to a real humanoid robot of type HRP2-KAI.

Finally, this document ends with a conclusion of the presented work, and discusses the future works, which can be developed to expand and improve on the final presented product of this thesis.

Chapter 1

Background and State of the Art

1.1 Introduction

Controlling robots has numerous obstacles, and there are some highly important and necessary concepts that require a careful study in order to overcome these obstacles.

The problem I am treating in this thesis is related to the balance of robots, which can be studied in a standing configuration for humanoids, or in multiple contact configurations for humanoids and quadruped robots. These configurations result in different balance control strategies.

More importantly, a robot uses its limbs to make contacts and interact with the environment, thus this interaction should be studied, by studying the contact forces and how they affect the robot's dynamics. If the interaction has a degree of compliance, such as an interaction with a soft surface, then the compliance's effect on the robot's dynamics should be taken into consideration.

Taking a robot with complex dynamics, proposing a reduced model and developing control strategies to tackle the reduced dynamics might be simpler to implement and could provide results similar to when the full dynamics are considered.

Conditions to maintain a robot's equilibrium should be respected throughout the robot's motion. Therefore these conditions should be well studied and taken into consideration when assigning tasks to the robot.

This is why in this chapter, I will cover in detail the aspects of notion of balance and control of floating base robots. I will start by answering these questions:

- What does the term compliance mean in the context of this thesis?

- How to tackle the complex dynamics of robots?
- What are the conditions to be respected for a robot to maintain its equilibrium?

I will also talk in general about multiple contacts situations for humanoid robots, and how balance control strategies might differ in such situations. Then, I will mention works in the state of the art that deal with these questions, and cover their balance control approaches, before presenting and situating the contribution of this work.

1.2 Compliance

The term compliance used in this thesis refer to two concepts. The first one is mechanical compliance which is usually found inside the rigid robot’s mechanical structure. Examples are the flexible rubber bushes, placed at the ankles of the robots of the Honda robots [Hirai et al., 1998], [Takenaka et al., 2009] and the HRP series [Kanehira et al., 2002], [Kaneko et al., 2019], [Kaneko et al., 2019], to absorb foot impacts [Kajita et al., 2001b], protecting the force-torque sensors at the feet and helping in a more fluid locomotion. Same goes for the WL-12 biped, which is equipped with a special foot mechanism for shock absorption [Yamaguchi et al., 1995], or any specifically designed feet for this purpose [Guadarrama-Olvera et al., 2022]; other examples include intrinsically structural compliance within robots, such as the robot COMAN [Li et al., 2013], cCub [Tsagarakis et al., 2011], Chobino1D [Vanderborght et al., 2011], or Sarcos Primus [Stephens and Atkeson, 2010], which inherently provide protection against impacts or accidental collisions. The other concept is the compliant surfaces, which comes from the nature of the environment that the robot is interacting with by applying forces, such as soft surfaces, like mattresses or muddy terrains, or any surface that undergoes some deformation when a force is applied.

For either categories, this compliance displace the application point of the contact forces, altering the robot’s motion, and can cause balance problems if not taken into consideration. The reason is that the contact forces and moments become unfeasible in the classical approaches that assume rigid interactions as they cannot be instantaneously modified [Flayols et al., 2020]. Additionally, ignoring the compliance generates destabilizing oscillations [Wieber et al., 2016], [Jung-Hoon Kim and Jun-Ho Oh, 2004]. These reasons make compliance interaction a very interesting and important topic to study when it comes to balance control and stability.

It’s important to note that the compliance that I am covering in this thesis is for rigid robots. On the other hand, soft robotics has been emerging as a domain of high interest for

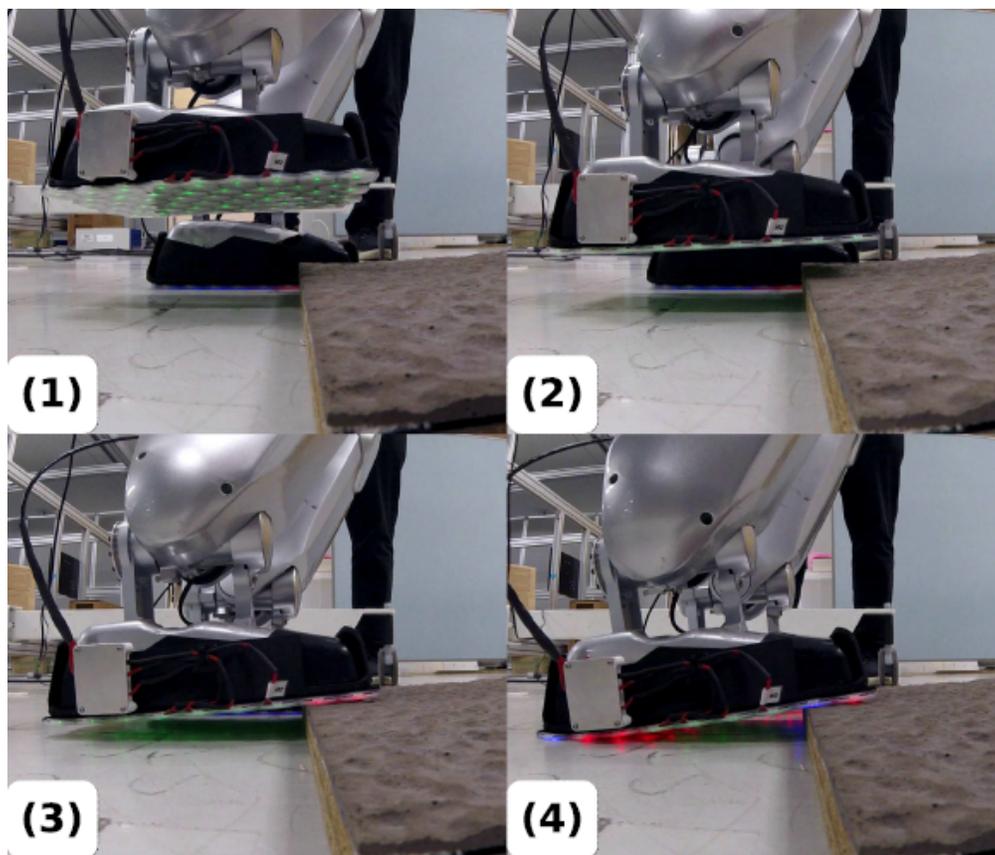


Figure 1.1: Preemptive foot compliance in the HRP-2 Kai robot. [Guadarrama-Olvera et al., 2022]

researchers [Albu-Schaffer et al., 2008], [Trimmer, 2015], [Laschi et al., 2016]. Soft robots have bodies made out of intrinsically soft and/or extensible materials (for example, silicone rubbers) that can deform and absorb much of the energy arising from a collision. These robots have a continuously deformable structure with muscle-like actuation that emulates biological systems and results in a relatively large number of degrees of freedom compared with their rigid bodies counterparts, potentially enabling agile movement in rough terrains [Rus and Tolley, 2015]. However, these robots require different approaches and algorithms to be controlled, and the conventional approaches to robot control assuming rigidity in the linkage structure of the robot itself are a poor fit for controlling these soft bodies [Rus and Tolley, 2015], [Iida and Laschi, 2011], which is why they are not going to be addressed in this thesis when I'm talking about compliance or softness.

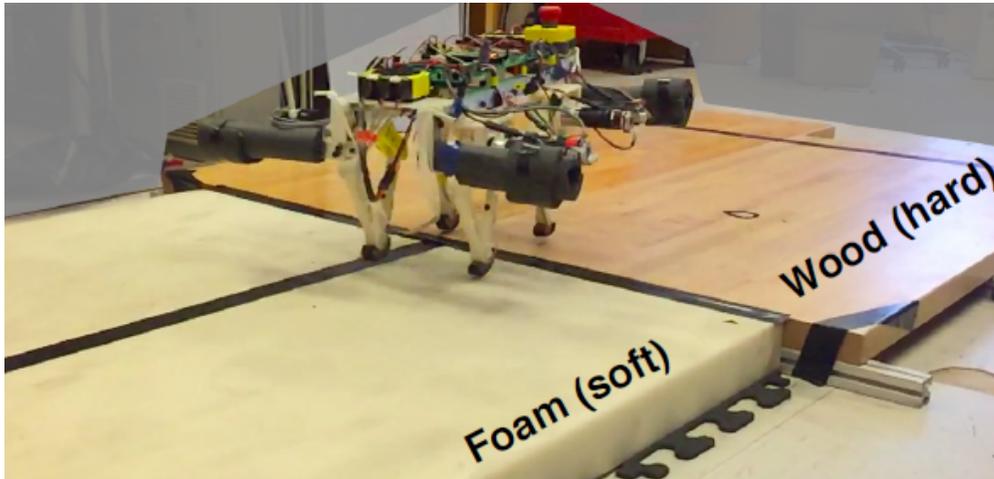


Figure 1.2: Example of a soft surface made of foam for motion tests on soft surfaces. [Bosworth et al., 2016]

1.3 Tackling the Robot's dynamics

An important part of research seeks at using a precise knowledge of the robot in question via the dynamic parameters, such as the mass, inertia, and center of mass, of each link that forms the robot, in order to achieve balance or make complex movements, such as walking or manipulations using the end-effectors. Therefore, it relies heavily on the accuracy of the model data. [Yamaguchi et al., 1999], [Qiang Huang et al., 2000], [Yang et al., 2021].

By contrast, other popular approaches don't rely on this precise knowledge, rather they use simplified models and build the appropriate controllers based on this limited information. These simplified models are low-dimensional approximations that capture the nature of the system's dynamics, and usually remove the non linearities, allowing fast computations due to their linear nature. This way, the control structure is simpler and relatively easy to implement, and can produce very good results in terms of balance, locomotion, and manipulation.

Finally, other works try to find consensus between reduced models and Whole-Body Models or combine them [Ponton et al., 2016], [Herzog et al., 2016], [Budhiraja et al., 2019].

In the following, I will talk about a few simple models that are commonly used in the literature to simplify the study of the robot or the contact forces that the robot exert when interacting with its surrounding.

1.3.1 Centroidal Model

The centroidal dynamics are a reduced-order representation of the full dynamics of the robot that considers the momentum wrench acting at a single point that is called the center of mass (CoM) of the robot.

The CoM of a robot is the adequate location of the robot’s total mass, thus it is the point where its overall linear momentum is naturally defined. It is also the point through which the resultant gravity force acts. These are one of the many reasons why it is considered a vital point in the study of the robot’s dynamics.

The motion of the CoM follows a point-mass trajectory, which can be easily described and studied, contrary to the dynamics of individual member links of the robot, which can be quite complex in most situations. Additionally, the rotational motion of the aggregate chain of links obeys the conservation of angular momentum around the CoM, or in other words, the conservation of the centroidal angular momentum. For many applications, such simplified description is essential in the study of the control design of the system [Orin et al., 2013].

One of the main benefits to this approach is the ability to utilize any controllable limb of the robot’s body to interact with its surroundings, or to undergo locomotion, while always obeying the general full rigid body dynamics.

It should then come as no surprise that almost all reduced models for complex robots like humanoids, and control algorithms rely on the CoM of the robot as an integral component for the study of balance, locomotion and manipulation [Dai et al., 2014], [Shah et al., 2021], [Murooka et al., 2022].

1.3.2 Linear Inverted Pendulum Mode

The linear inverted pendulum is considered to be one of the simplest yet very reliable models available in the literature [Kajita, 1993]. In this model, the totality of the robot’s mass is concentrated at its CoM, the legs are links with zero mass that make contacts with the ground at single rotating joints. Lateral motion is decoupled, and the robot’s motion is constrained by the sagittal plane defined by the vertical axis and the axis of walking direction.

This model has been extended as a 3D linear inverted pendulum to cover 3D motion in [Kajita et al., 2001a] and variable heights in [Caron, 2020], and has been widely used to achieve locomotion for humanoid robots [Caron et al., 2021], [Englsberger et al., 2011], [Kajita et al., 2003], [Kajita et al., 2010], [Kajita et al., 2017], [Kajita et al., 2018], [Parietti and Geyer, 2011], [Wiedebach et al., 2016].

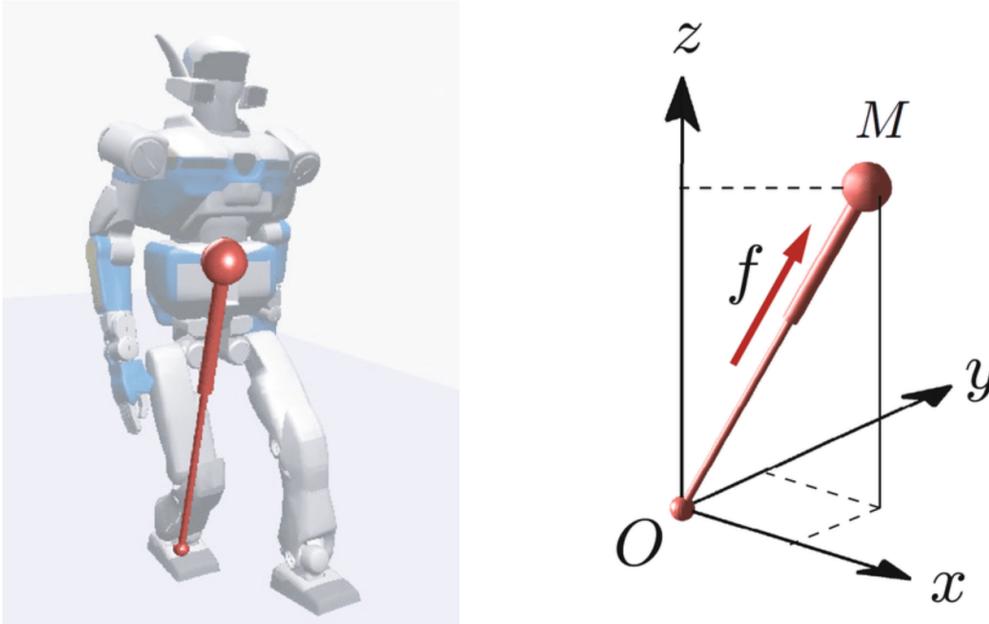


Figure 1.3: Illustration of the linear inverted pendulum mode. [Kajita et al., 2005]

1.3.3 Viscous-Elastic Model of the Contact Forces

A model such as the linear inverted pendulum depends, at their core, on the rigid contact assumption, and balance approaches based on this model will encounter difficulties when dealing with compliance.

In general, any control that always assumes rigid interaction with the environment struggles when this assumption is no longer valid [Kudruss et al., 2015], [Li et al., 2013]. Some controllers like [Polverini et al., 2017] are based on robust control theory and can provide good results, but only with limited compliance and with simple robot models. This is why there has been always the need to model this compliant interaction, and use it to complement the full or reduced dynamics of the robot. Some works use a non-linear modeling approach of the normal contact force [Azad and Featherstone, 2014], [Lankarani and Nikravesh, 1990], however the most popular approach of compliance modeling has been the viscous-elastic model, and it has a variety of slightly different implementations [Liu and Padois, 2015], [Li et al., 2019], [Vasilopoulos et al., 2018], [Romualdi et al., 2021], [Flayols et al., 2020], [Mifsud et al., 2016]. To put it simply, the robot-environment interaction is represented by a spring and a damper, and the contact force is caused by the displacement of the relative motion between the desired and actual contact positions, or by the deformation of the environment. Similarly, the moment produced by the contact's orientation caused by this compliance can

also be modeled using a rotational spring [Benallegue and Lamiraux, 2015], [Winter et al., 1998], although it is considered less accurate than its linear counterpart.

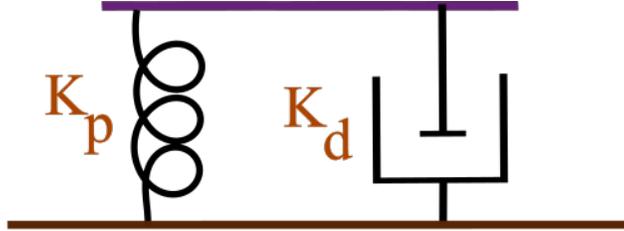


Figure 1.4: Illustration of the viscous-elastic model, via a spring with a stiffness K_p and a damper with a damping constant K_d

1.4 Balance Criteria

A system, whether it's a robot or a living creature, is generally supposed to be in a state of static balance if and only if its CoM projects vertically inside the convex hull of the points at each contact with a specific surface, also known as the support polygon [Wieber et al., 2002]. In other words, balance mainly depends on gravity and contact forces.

However, dynamic balance is not governed by the CoM's projection, instead the point known as the zero moment point (ZMP) should exist inside the support polygon, which is a 2D point representing the center of pressure of the contact forces. The ZMP was first introduced by Vukobratovic and reviewed in his works [Vukobratovic, 1973] [Vukobratović and Borovac, 2004] and [Kajita et al., 2005] goes in detail of its computations and its use in the study of the robot's dynamics. In general, the ZMP is calculated using

$$z_x = \frac{\sum_i (p_{i,x} f_{i,z} - p_{i,z} f_{i,x} - t_{i,y})}{\sum_i f_{i,z}}, \quad (1.1)$$

$$z_y = \frac{\sum_i (p_{i,y} f_{i,z} - p_{i,z} f_{i,y} + t_{i,x})}{\sum_i f_{i,z}}, \quad (1.2)$$

where

- x , y , and z represent the coordinates in the respective Cartesian axis,
- $z \in \mathbb{R}^2$ is the ZMP of the entire body,
- $p_i \in \mathbb{R}^3$ is the position of contact i ,

- $f_i \in \mathbb{R}^3$ is the force at contact i ,
- $t_i \in \mathbb{R}^3$ is the moment at contact i .

The existence of the ZMP inside the support polygon is a necessary condition, however it is not enough, as it does not take into consideration moments around the yaw axis (as seen from equations (1.1) and (1.2)), which can be present during the locomotion of certain robots such as humanoids. When this undesired yaw moment reaches a certain threshold and exceeds the frictional moment, it results in undesired rotations that may cause slippage and deviate the robot from its stable reference trajectory [Cisneros et al., 2014], [Ugurly et al., 2012]. Furthermore, the classical ZMP study does not work with non-coplanar contacts. Which is why an additional study of the contact forces or wrenches is required and used for the robot's balance, especially in the case when the robot has non-coplanar contacts with the environment.

With that being said, it is obvious to conclude that the more contacts a robot make with the environment, the better the overall static balance, as the support polygon will be bigger and wider. This is a main reason why quadruped's balance is considered to be easy to achieve compared to biped's in general [Biswal and Mohanty, 2021] [Li et al., 2011].

This is also an important reason among many why researchers try to exploit additional contacts with their bipedal robot [Samadi et al., 2021], [Samadi et al., 2020], [Samadi, 2021].

In general, exploiting contacts requires respecting conditions allowing contacts with the environment to be maintained amidst possible slippage due to surface friction.

In any case, there is always a need to build a specific control to deal with the position of the CoM or of the ZMP of a bipedal robot, to ensure they stay in their support region. This control should also ensure that the contact forces and moments follow their desired trajectories during a robot's motion and when the robot is perturbed, which can be challenging especially in the presence of compliance.

Force control is not limited in use within balance control only, rather it is also a very important tool for manipulation accuracy, and is used in the control of industrial manipulators and on the end-effector of the floating-base robots performing manipulation tasks to accurately execute such tasks.

In the following, I will talk about the conditions for the maintenance of the contacts and mention various force control algorithms that are used on robots in the literature.

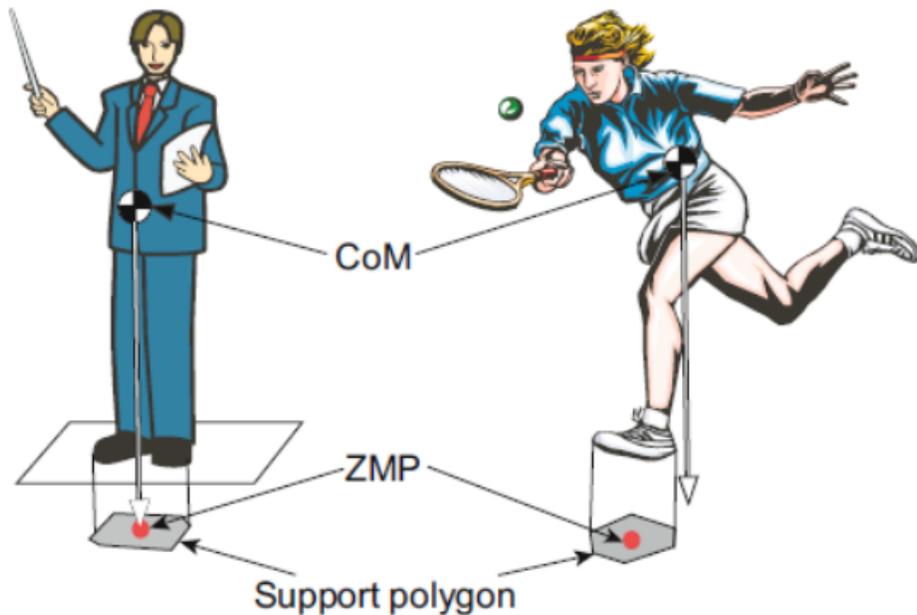


Figure 1.5: Support polygons for the CoM and ZMP regions in a static state (left) and in a dynamic state (right). [Kajita et al., 2005]

1.4.1 Contact Stability

I will start this section by putting an emphasis on the definition of a contact. A contact is the physical touching made by one of the robot's limbs or end effectors with the environment. A contact can be of point type such as the contacts made by pointy legs of quadruped robots, edge type such as the line contacts made by the robot Cassie [Gong et al., 2018], or surface type such as the surface feet of humanoid robots.

Humanoid robots' locomotion is usually relying on the transition between one contact and two contact phases. Multiple contacts configurations for humanoids involves their use of hands or any end-effectors they are equipped with. For the robot to maintain its balance while establishing a contact, this contact should be stable, and research has been done to rely on the stability of the contacts to guarantee the robot's balance, especially when the contacts are non-coplanar [Caron et al., 2015b], [Caron et al., 2015a], [Escande et al., 2013], [Hirai, 1991], [Hirukawa et al., 2006].

Having a stable contact is usually done using constraints. A reliable model used in the study of contacts' stability is the Coulomb friction model, which indicates that the contact will be still without motion if the contact force f lays inside the friction cone. This is

represented by

$$f_n \geq 0 \tag{1.3}$$

$$\|f_t\| \leq \mu f_n, \tag{1.4}$$

Where μ is the friction coefficient, f_n and f_t represent respectively the normal and tangential component of the contact force f at the surface with respect to the frame attached to the contact surface.

Equation (1.4) uses the Euclidean norm $\|\cdot\|$, resulting in friction cones with circular sections. Implementing the friction condition in the study of contact stability and balance is usually done with linearized cones [Del Prete et al., 2016], [Hauser, 2014], represented by

$$f_x \leq \mu f_z \tag{1.5}$$

$$f_y \leq \mu f_z. \tag{1.6}$$

Equation (1.3) represents the unilateral aspect of the contacts, and it implies that the the robot can push through ground, and not pull.

1.4.2 Force Control

The most successful way of controlling the external forces has been to explicitly control the joint torque while relying on the data provided by the force-torque sensors equipped at the robot’s limbs, and on the joint torque feedback, since this allows to virtually monitor the forces applied on the environment at any point.

Force-torque sensors are usually expensive and fragile, and joint torque feedback can be absent in robots due to its complex implementation, which is why other researchers try different approaches to detect collisions or estimate the contact forces without relying on these sensors, such as [De Luca et al., 2006], [Geravand et al., 2013], [Benallegue et al., 2018] [Mifsud et al., 2015].

Other common approaches to force control are admittance and impedance controls, both aiming at shaping the dynamical relation between end effector velocity and applied contact forces [Calanca et al., 2016]. The difference between the two approaches is that the former controls motion after measuring the contact force, while the latter controls the contact force after measuring the motion of the end effector [Keemink et al., 2018].

Admittance control uses closed-loop force control with an open-loop position control, and

is usually implemented in position controlled robots. Force feedback is used to generate the appropriate motion of the end effector as a function of the error of the contact force. It is important to note that in the overall balance control architecture of complex robots such as humanoids, the classical admittance control algorithm is usually used in parallel with a separate CoM control loop, while assuming that the force control loop is fast enough to converge within the time required to maintain balance via the CoM control loop. While this has nevertheless produced stable controllers for balanced dynamic behaviors in different situations, is not practically robust to different dynamic situations, and it requires a careful tuning of the control gains. Instead of relying on this assumption, some works use a first order delay on the dynamics of the ZMP [Murooka et al., 2021].

Overall, Admittance control has been used in many different applications with different types of robots [Lecours et al., 2012], [Seraji, 1994], [Dimeas and Aspragathos, 2016], [Yokoi et al., 2004], [Caron et al., 2021], [Okunev et al., 2012], [Kim et al., 2009], [T., 2002].

The diagram in figure 1.6 shows a simple implementation of an admittance control task within a solver that generates the robots' joints positions and velocities q and \dot{q} . The block $Y(S)$ represents the admittance control, generating a velocity in the Cartesian space v of an end effector using

$$\begin{aligned} v &= Y f^\Delta \\ &= Y (f_{ref} - f_m), \end{aligned} \tag{1.7}$$

where Y represents the admittance gain, f_{ref} is a reference force to track, and f_m is a measured force provided by force sensors. The position controller uses the solver's output with feedback values to produces the joint torques.

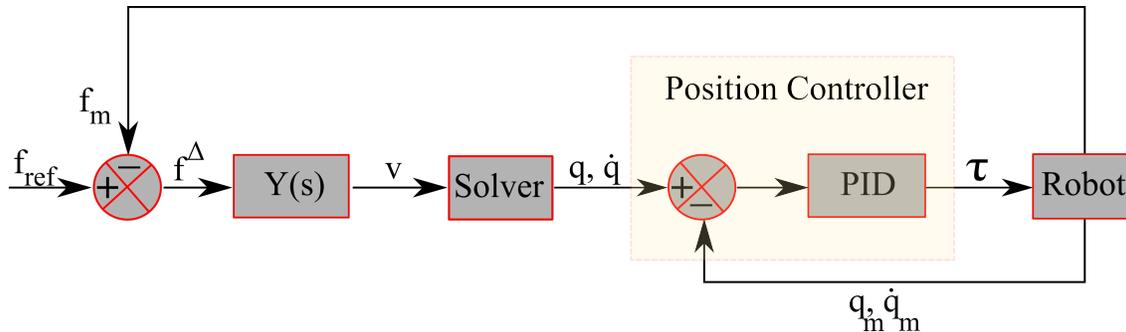


Figure 1.6: Simple diagram showing a simple use of admittance control.

Impedance control uses closed-loop motion control with an open-loop force control, and

is usually implemented in torque controlled robots. When the interaction with the environment is rigid to an extent, the impedance approach ensures steady contact forces which can be regulated through the impedance gains. The basic goal of impedance control in general is to achieve a desired dynamical relationship between external forces and the robot's motion. This type of control also has a wide use in the literature [Hogan, 1987], [Albu-Schaffer and Hirzinger, 2002], [Ott et al., 2004], [Albu-Schaffer et al., 2004], [Dietrich et al., 2011], [Schindlbeck and Haddadin, 2015], [Boaventura et al., 2013], [Herzog et al., 2014].

Figure 1.7 shows a simple diagram with an impedance control approach. The block $Z(S)$ represents the impedance control, generating a force in the Cartesian space f of an end effector using

$$\begin{aligned} f &= Zv^\Delta \\ &= Z(v_{ref} - v_m), \end{aligned} \quad (1.8)$$

where Z represents the impedance gain, v_{ref} is a reference velocity in Cartesian space to track, and v_m is the measured Cartesian space velocity of the end effector (obtained from the robot by using encoders to obtain the joint positions and velocities and jacobians to transform them into Cartesian space). Jacobians are used to transform the force into a reference torque τ_r

$$\tau_r = J^\top f, \quad (1.9)$$

which is used with the torque feedback τ_m in the torque controller to produce the torques for the robot's joints.

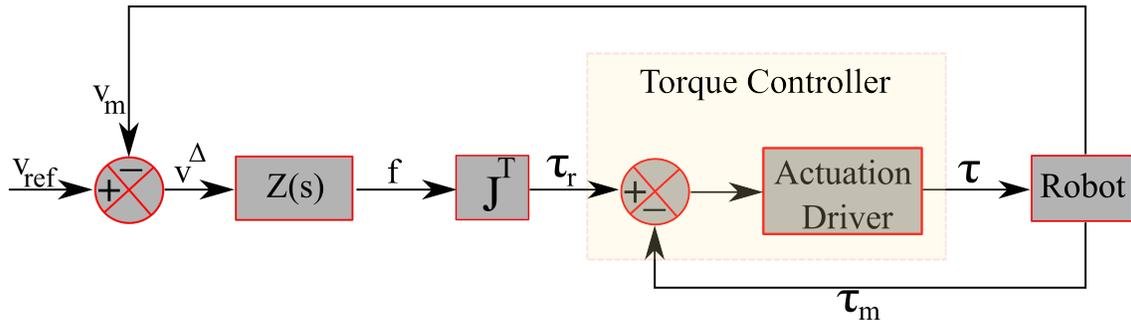


Figure 1.7: Simple impedance control diagram.

The Admittance and impedance control strategies are complementary, they implement opposed solutions in order to ensure force control and thus behave well in different situations. The admittance control is better suited for interactions with compliant environments or

operations in free space and the impedance control is better suited for dynamic interactions with stiff environments [Ramuzat, 2022]. [Sharkawy and Koustoumpardis, 2022] details the differences in performance between the two strategies.

Both strategies can be unified into a hybrid admittance/impedance control, to get the best of both approaches and to avoid the limitations of each approach as much as possible, as done in [Ott et al., 2010].

Other approaches have been proposed to control the normal compliant force, by controlling the movements of the contact points using contact models [Azad et al., 2016].

1.5 Multiple Contacts Situations

The term multiple contacts, or in short multi-contacts, is usually used for humanoid robots making contacts with their hands or end effectors in addition to the contacts made by their feet for standing or locomotion. While a quadruped robot is arguably a robot in a multi-contact situation compared to a default humanoid locomotion, usually all of its limbs are practically identical to each other in terms of design and control, as a result they are not considered in this section.

When it comes to humanoid robots, which are supposedly designed and developed to work along humans, it is only natural that these robots are in need of using their end effectors or hands to replicate some of the tasks that humans do, such as pushing an object, or manipulating it. This requires careful planning for the contacts positions and trajectories, and the state of the art covering the topic of motion planning is quite abundant [Escande et al., 2006], [Murooka et al., 2015], [Murooka et al., 2017], [Murooka et al., 2018], [Harada et al., 2004], [Nozawa et al., 2012], [Takubo et al., 2005], [Kumagai et al., 2020], [Kumagai et al., 2021], [Bouyarmane and Kheddar, 2012], [Bouyarmane et al., 2012], [Bouyarmane et al., 2019].

Furthermore, it's expected for robots to exploit additional contacts with the environment using their limbs and end effectors to improve their balance and movement throughout terrains, especially when the environment presents many obstacles that might hinder the robot's balance when relying on minimal contacts required for locomotion or any particular motion. Additionally, many manipulation tasks require the use of additional contact for improved dexterity, an example would be a humanoid robot with human-like hands using an electric screw driver to insert a screw in a wooden board, in this case the robot should use the screw driver in one of his hand and use his other hand to maintain the surface of the board in the

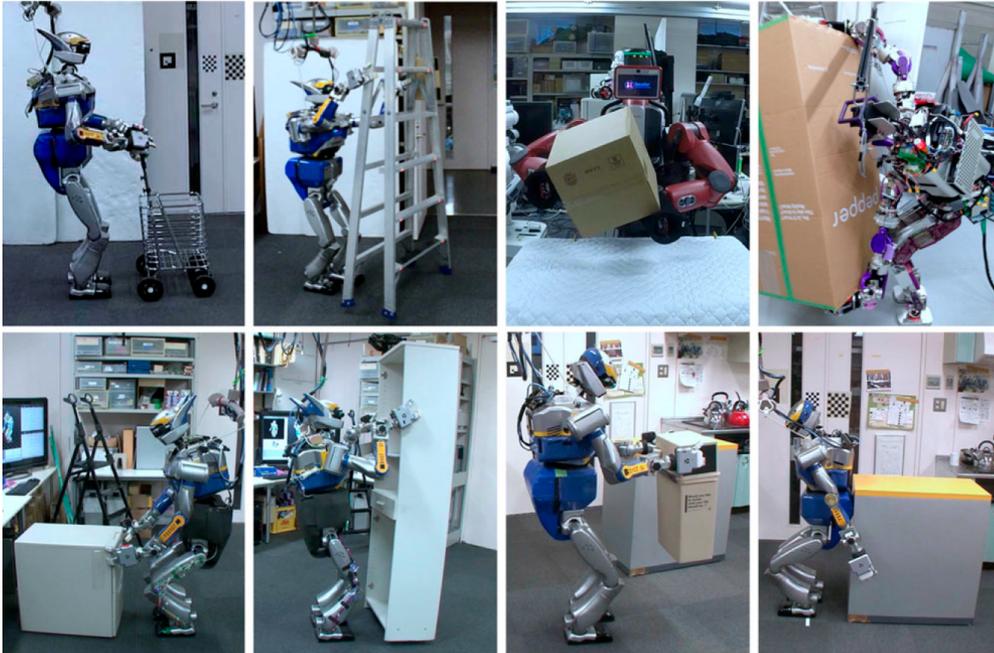


Figure 1.8: Various manipulation strategies by the humanoid robot HRP-2 while maintaining balance [Murooka et al., 2017].

required position to accomplish his task.

Additional contacts offers a bigger support region for the CoM and ZMP, and while this provides a bigger balance zone for robots having the projection of their CoM or the ZMP in the middle of this support region by design, such as quadruped robots, humanoid robots might encounter additional difficulties as a consequence of this increased balance region. For starters, a humanoid robot is usually adding contacts that are non-coplanar with those made by its feet. During dynamic behaviors, it is then complicated to keep the ZMP in its support region, as the ZMP comes with the assumption of contact surfaces being in the same plane. Some researchers consider the contact forces used for manipulation as external forces, and study their effect on the CoM and ZMP in their balance control [Murooka et al., 2021], while some others define a generalized ZMP and seek to keep this upgraded ZMP in the convex support region [Harada et al., 2006], [Caron et al., 2017].

1.6 Works Related to Balance

Now I will put emphasis on multiple works that studied the questions and topics that I mentioned so far in order to command robots to do specific tasks while ensuring that their

balance is well maintained.

As I mentioned earlier, quadruped robot's balance is usually easy to maintain on flat and rigid ground, as their general structure offers a safe large support area for the CoM and ZMP. Preliminary running experiments were done as early as [Raibert et al., 1986], where they applied simple one-legged algorithms, to make their quadruped run with a trotting gait. Challenges occur when the motion is fast, hence why force control is used to follow the desired contact forces without saturating the joint torques, as done in [Kotaka et al., 2013], where force feedback is used to respect the constraints of the CoM and ZMP, or in [Valenzuela and Sangbae Kim, 2012], [Seok et al., 2015], where force planning and impedance control algorithms were implemented. Reduced models such as Spring Loaded Inverted Pendulum are used in [Poulakakis et al., 2003], [Poulakakis et al., 2006], [Poulakakis et al., 2005] to simplify the dynamics, and reliance on self-stabilization was enough to propose a running gait on the quadruped robot Scout II.

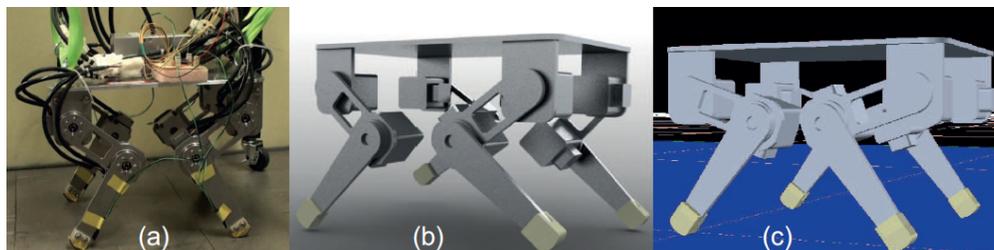


Figure 1.9: RoboCat-1, capable of running while relying on force sensors for force control [Kotaka et al., 2013]. (a) shows the actual robot, (b) shows its CAD model, (c) shows a simulation environment

Additional challenges to quadruped include balance on unevenness/ ruggedness in the terrain or in case of non-coplanar contacts. [Barasuol et al., 2013] used a capture point based model for push recovery on their HyQ robot to balance on a rough terrain, [Raibert et al., 2008] managed locomotion on rubble and snow with little details on the control algorithms used for balance, and [Neunert et al., 2018] used model predictive control for a careful balanced locomotion on rough terrain. Additionally, [Chen et al., 2012] used careful motion planning for a stable stairs climbing. Not to forget ANYmal, the robot presented in [Hutter et al., 2016] and [Hutter et al., 2017], which is capable of dynamic locomotion, walking over obstacles, and climbing stairs thanks to the high mobility of its feet, allowing the movement of a single foot per step, all while relying on impedance control and proportional derivative integrator (PID) control to follow the desired trajectories.

Finally and most importantly, works have been done on quadruped to ensure a safe



Figure 1.10: MIT Cheetah, running at 6 m/s [Seok et al., 2015]



Figure 1.11: HyQ balancing on rough terrain [Barasuol et al., 2013] and Big Dog descending on rubble [Raibert et al., 2008]

and stable locomotion over compliant soft surfaces, such as [Bosworth et al., 2016], where locomotion over hard and soft terrain was possible with a careful tuning of the impedance gains. The work done in [Fahmi et al., 2019] uses the viscous elastic contact model to model contact forces, and estimate the stiffness of the ground online using supervised learning, which is used to produce a desired penetration through the soft ground, while relying on impedance control to follow the desired forces.

With that being said, all of these robots have point contacts, and the tasks that they can perform are limited due to their small size and design, which is not convenient to perform complex tasks like humans do. Reasons why researchers develop humanoid robots.

As for humanoid robots, the locomotion achieved by Honda on their robots caught the interest of the robotics community as early as [Hirai et al., 1998], where balance is achieved by carefully adjusting the position of the ZMP during the motion, and it was improved on

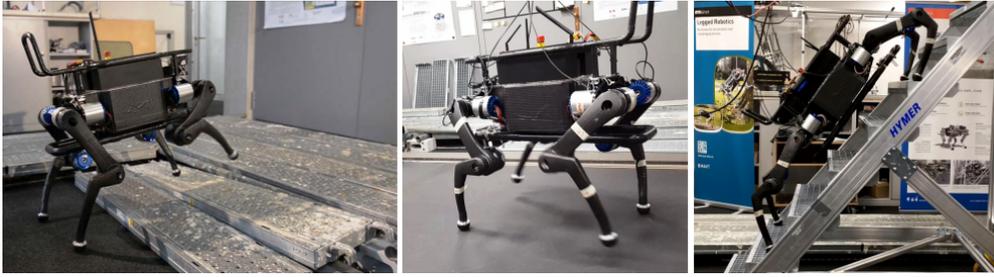


Figure 1.12: ANYmal walking on uneven terrain, trotting and climbing stairs [Hutter et al., 2016]

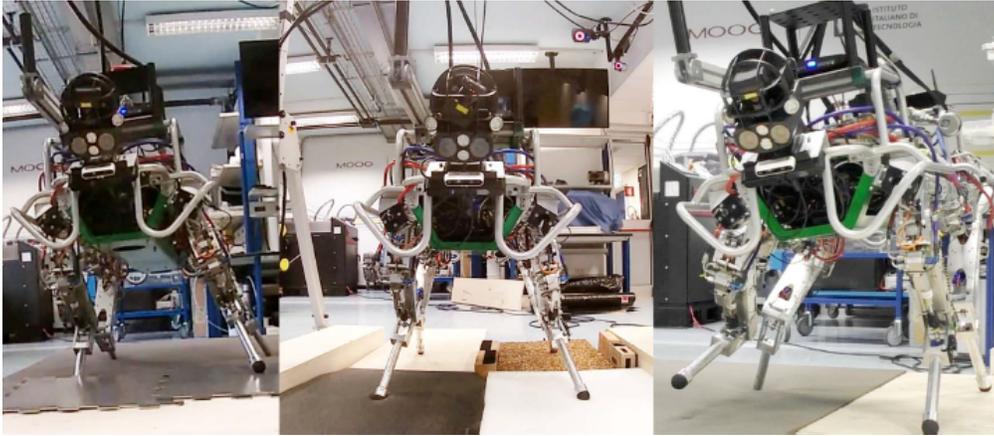


Figure 1.13: HyQ traversing multiple soft terrains of different compliances [Fahmi et al., 2019]

Asimo [Shigemi, 2019], where locomotion on uneven surfaces became possible by using hard pressing on the sole of the foot, manipulating the contact forces and the ZMP position, while relying on force-torque sensors to detect the central point of the actual ground reaction force.

As mentioned earlier, the linear inverted pendulum model has been intensively used for bipedal walking, and used with preview control to generate ZMP trajectories for locomotion [Kajita et al., 2003], the divergent component of motion (DCM) dynamics to calculate desired ZMP [Kajita et al., 2010] and admittance control to guarantee a balanced motion, or to apply wrenches [Caron et al., 2021] to climb stairs. Similar strategies have been used to achieve balance on uneven terrain [Jong Hyeon Park and Eung Seo Kim, 2009], [Morisawa et al., 2011], [Morisawa et al., 2012]. Pattern generation approaches for ZMP compensation for balancing on slightly rough terrain [Nishiwaki and Kagami, 2007] were also used.

Full dynamics with task-space differential inverse kinematics that exploits the entire six-dimensional force measurement information in [Yang et al., 2021] is shown to guarantee the balance of a position controlled robot when in contact with moving and rotating surfaces.



Figure 1.14: HRP-4 Walking on Pavement (left) [Kajita et al., 2010] and Climbing Stairs (right) [Caron et al., 2021]

Linearization of the full dynamics and the use of the linear quadratic regulator (LQR) allowed the hydraulically actuated torque controlled Sarcos to squat in [Mason et al., 2014].

As for humanoids multiple contact balancing, computation of the CoM support region approaches has proven to be a good solution for quasi-static motion [Samadi, 2021], [Roux et al., 2021].

For locomotion and multiple contact scenarios, [Cisneros-Limon et al., 2020] relied on the DCM error dynamics to generate a desired ZMP trajectory, from which desired forces are calculated to be used for admittance control. [Murooka et al., 2021] also relied on the classical DCM dynamics and admittance control, and added CoM and ZMP strategies to compensate for the effect of the manipulation contact forces (which are considered as external forces) on the CoM and ZMP positions. In [Murooka et al., 2022], centroidal dynamics with pre-view control generates centroidal trajectories, which are stabilized by centroidal feedback and admittance control. This architecture allows the simulation of many multiple contact locomotion applications, such as walking with hands on the wall and climbing handrail stairs. Ladder climbing has been done in [Vaillant et al., 2016] and [Kanazawa et al., 2015] with careful motion planning, position estimations and dynamics slow enough that the displacement caused by the built-in compliance of the robots is minimal. Force distribution and CoM trajectory generation based on centroidal dynamics [Morisawa et al., 2018] and DCM dynamics with admittance control [Morisawa et al., 2019] offered balanced locomotion with

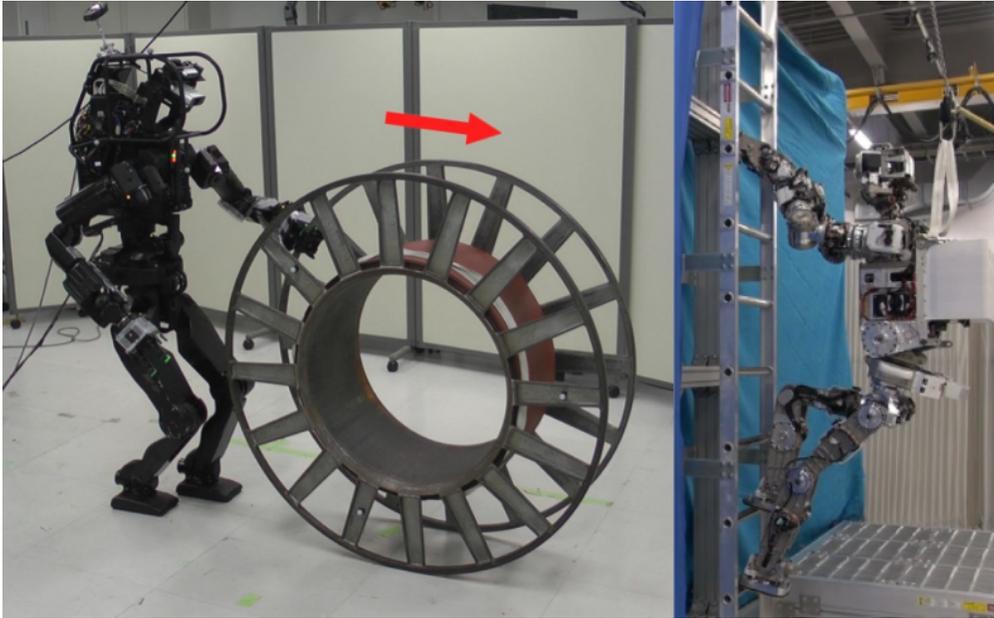


Figure 1.15: Multiple contact motion: by rolling a bobbin (left) [Murooka et al., 2021] and climbing ladders (right) [Kanazawa et al., 2015]

multiple contacts on rough terrain. The combined LQR and QP control has been successfully used with multi-contact motion planning [Posa et al., 2016], [Kuindersma et al., 2016].

To tackle the compliant interaction between the robot and its surroundings in the balance control architecture, [Mifsud et al., 2016] proposed a single viscous-elastic reaction mass pendulum model for the overall compliance effect, while using a kalman filter to estimate the compliance and an LQR to control it. [Flayols et al., 2020] combined the centroidal dynamics with the viscous-elastic model of the forces, and worked on minimizing the snap of the CoM considering under-damped contacts, and tested the control on a 2D model of a bipedal robot with compliant contacts.

As for walking on compliant surfaces, [Mesesan et al., 2019] relied on the DCM dynamics with impedance control, and the calculation of a stabilization wrench in the case when the stance feet moves through the ground, for their torque-controlled robot TORO. [Kuindersma et al., 2014] used an LQR on the linearized ZMP dynamics within a QP formulation and terrain estimation for the feet position planner to simulate walking on a muddy terrain.



Figure 1.16: Locomotion through mud [Kuindersma et al., 2014]

1.7 A Unified Balance-Force Control for Compliant Interactions

Based on the works presented on balance, it is easy to note that there has not been a lot of studies dealing with interaction with soft surfaces. The most notable of these works are [Fahmi et al., 2019] and [Bosworth et al., 2016], however their works were applied on torque-controlled quadruped which can produce the desired torques accurately thanks to the joint torque feedback, as a result these approaches don't work perfectly on position controlled robots. Furthermore, the contacts with the environment are point contact, which are easier to manage compared to contact surfaces.

[Mesesan et al., 2019]'s approach was successful but it can be only applicable on torque-controlled robots. Additionally, it does not have a multiple contact formulation.

There are works that considered treating the compliance within the HRP-2 robots, such as [Murooka et al., 2021], by modeling the effect of the compliance as first order delay on the ZMP and used admittance control in their balance control, which is not accurate enough since the simplest physics based model has at least a second order dynamics.

This is why I propose a balance control that take into account the compliance at the contact level, with a multiple contact formulation, that can be implemented on both position and torque controlled robots. I will use a reduced model based on the viscous-elastic interaction, linearize the non-linear dynamics and use an LQR solver to generated accelerations for the robot's limbs, which will be used within a QP solver for the joint positions. Relying on the forces from the viscous-elastic model makes the control prone to inaccuracies due to modeling errors, which is why I will use the force-moment feedback from force sensors and establish a trade-off between the positions/rotations of the contacts and the contact



Figure 1.17: Careful locomotion over a mattress [Mesesan et al., 2019]

force/moment within the linear dynamics before computing the LQR solution. The control will be validated on a torque-controlled biped and on a position-controlled humanoid in two contacts and multiple contact situations with compliant surfaces.

1.8 Conclusion

This chapter laid out the background related to robots' balance, the study of their body dynamics and the dynamics of the compliant interaction with the environment. It focused on the theoretical approaches of balance for robots, and presented many works that deal with this topic in the literature, before positioning a novel contribution.

Studying the dynamics of the robots is an essential step for balance control. The study of the reduced dynamics of a reduced model in place or in parallel with the full dynamics provide effective solutions at lower computation costs. Robots have sometimes a built-in compliance, having the role of reducing the power of impacts and making the robot's motion more fluid. However, this compliance can alter the desired position and orientation of the robot's limb. Additionally, compliance in the environment itself, such as soft surfaces, displaces the point

of application of the contact forces applied by the robot. In general, compliance might cause the robot to struggle with balance or lose it entirely, which is why there's a need to take the study of this compliance in the control synthesis.

The works presented in the literature usually use rigid contacts assumptions in their controller architecture, and while this can be proven effective for interactions with very low compliance, these controllers struggle when the compliance is high enough.

The chapter covered the works that deal with the compliance in their balance control, and explicitly mentioned their limitations, and situated the contribution of the work of this thesis, which aims to formulate a controller that deals with the compliance at the contact level in the robot-environment interaction, and can be used with a wide variety of robots: whether the robot is a humanoid or a quadruped, whether the robot is position or torque controlled, whether the humanoid is multiple-contact configuration or not.

The next chapter goes in detail over the theoretical structure of the contribution, by describing a reduced model of the robot-environment compliant interaction, and studying the reduced dynamics.

Chapter 2

Dynamics of Compliant Robot-Environment Interaction

2.1 Introduction

The previous chapter gave a general outline to the balance control in robotics, and the contact force control schemes were highlighted for their importance. Most importantly, the state of art tackling the compliant interaction between the robot and its surrounding was presented. When the environment itself is compliant, there are only a handful of works that propose a balance control for their robots. However, these algorithms can only be applied to specific robots or in limited applications.

This chapter takes a novel approach on the study of the compliant robot-environment interaction, and presents a system for a balance control that can be applied to a wide selection of robots.

The chapter proposes a reduced model for the compliant robot-environment interaction, studies its non-linear dynamics, and deals with the linearization of these dynamics. The study is treating dynamic behaviors around static references, meaning that the robot is considered in contact with the environment and around its desired reference, and the study of the linear dynamics is done around this reference.

I will start by describing the full dynamics of a rigid robot and propose a reduced model using a viscous-elastic interaction to represent the robot-environment compliant interaction. I will then describe how the contact forces of this interaction affect the center of mass of the robot using Newton-Euler equations. Then by choosing a state-space representation, I will linearize the dynamics of the system around the desired point of reference. Finally, I use the

measurement of the contact forces obtained from sensors data to make a trade-off between the kinematics and force feedback.

The chapter is divided as follows: section 2.2 lays out the full body dynamics, section 2.3 models the compliant interaction and studies the reduced dynamics, section 2.4 covers the linearization of the non-linear dynamics, section 2.5 covers the force-kinematic trade-off, finally section 2.6 concludes the chapter.

2.2 Full Rigid Body Dynamics

I will start by considering a multi-body robot with $n + 6$ degrees of freedom (dof), having a configuration described as

$$\Psi = (p_B, R, q), \quad (2.1)$$

where $p_B \in \mathbb{R}^3$ and $R \in SO(3)$ represent the position and orientation matrix of the non-actuated floating base, and $q \in \mathbb{R}^n$, the joint angles vector.

For many applications, it is usually sufficient to model the robot as a rigid multi-body system, in other words, as interconnections of perfectly rigid bodies by ideal joints without mechanical play [Duindam and Stramigioli, 2007]. I define $\alpha \in \mathbb{R}^{n+6}$ as the robot's configuration velocity vector given by

$$\alpha = \left(\dot{p}_B^\top \quad \omega^\top \quad \dot{q}^\top \right)^\top, \quad (2.2)$$

where \dot{p}_B and $\omega \in \mathbb{R}^3$ are the linear and angular velocities of the base. The angular velocities are such that the derivative with respect to time of the rotation matrix is given by $\dot{R} = S(\omega)R$, with $S(\omega)$ being the skew-symmetric matrix operator allowing to perform cross-product

$$S(\omega) = \begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix}, \quad (2.3)$$

with ω_x , ω_y , and ω_z representing the Cartesian coordinates of ω .

Given the inertial properties of the robot's rigid bodies, the dynamics of the robot can be described using the Lagrangian equation [Murray et al., 2017] of the form

$$H(\Psi) \dot{\alpha} + C(\Psi, \alpha) \alpha + G(\Psi) - F = \tau, \quad (2.4)$$

where:

- $H(\Psi) \in \mathbb{R}^{(n+6) \times (n+6)}$ is the robot's inertia matrix,
- $C(\Psi, \alpha) \in \mathbb{R}^{(n+6) \times (n+6)}$ is a matrix accounting for the Coriolis and centrifugal effects,
- $G(\Psi) \in \mathbb{R}^{n+6}$ is the vector of gravitational effects,
- $\dot{\alpha}$ is the robot's configuration acceleration vector (time derivative of α),
- $\tau \in \mathbb{R}^{n+6}$ is the input torque vector corresponding to both the under-actuated and actuated dof,
- $F \in \mathbb{R}^{n+6}$ is the vector of external forces acting through the contacts with the environment, which is calculated using

$$F = \begin{pmatrix} J_{c,1}^\top & \cdots & J_{c,n_c}^\top \end{pmatrix} \begin{pmatrix} f_{c,l,1} \\ t_{c,l,1} \\ \vdots \\ f_{c,l,n_c} \\ t_{c,l,n_c} \end{pmatrix}, \quad (2.5)$$

where $J_{c,i} \in \mathbb{R}^{6 \times (n+6)}$ is the Jacobian matrix of the contact i , and $f_{c,l,i}$, $t_{c,l,i}$ are respectively the force and torque applied at contact i ($i = 1, 2, \dots, n_c$; n_c is the number of contacts), expressed in the local frame of the contact body.

The classical use of equation (2.5) is to consider that the robot-environment interaction is rigid. In this case, these forces depend mostly on the torque τ in a way that any force that is feasible (within unilaterality, friction, and torque limit constraints) can be generated instantly. However, it is not accurate to assume perfect rigidity especially when the interaction surface is compliant, and can modify the position or orientation of the limb of the robot used to apply a certain contact force, which is why I cannot rely on this assumption as the forces are not feasible instantly. The contact forces at a given instant depend on many factors other than joint torque, such as deformation in the contact surface.

In particular, if the forces depend only on the local deformation caused by the interaction with the contact body, I can write the forces as a function of the robot state $F(\Psi, \alpha)$. In this case, I can control the forces indirectly by modifying this deformation through the state of the robot. This requires to explicitly consider the coupled dynamics between the kinematics and the contact forces to ensure the convergence to a given reference.

Here, the reader might hint at a contradiction in what I am presenting, which is that I criticize the rigid contact assumption and say that the forces depend on many factors on one hand, but on the other hand I assume a particular case where the forces depend only on the deformation caused by this interaction. One thing that should be noted is that no assumption is perfectly accurate, and each assumption will struggle under different conditions. Assuming rigidity in the robot-environment interaction has resulted in many successful balance control for robots in static and dynamic cases, and lots of them were mentioned in the state of art. However as the topic of interest of the work in this thesis is centered around a compliant interaction, especially compliant surfaces, where this assumption struggles, I opt to use a different assumption that is suitable for the topic in hand, and more accurate in representing the real contact forces in general.

2.3 Dynamics of the Compliant Interaction

In order to study the coupled dynamics of the contact forces and the kinematics, I will propose a simple model for the robot-environment interaction and study the balance of the robot based on the reduced dynamics of this model.

To simply parameterize the contact wrenches and formulate how they affect the position of the CoM of the robot, the dynamics are reduced to consider the robot as a single rigid body, having a single mass and a single inertia tensor. The robot is capable of making contacts with its surroundings using mass-less limbs, with a negligible momentum around the CoM.

To model a compliant contact, I will rely on the viscous-elastic approximation, allowing to emulate a linear passive interaction. Contacts are defined as the intersection of the robot's limbs applying forces on the environment, with each contact having their own local frame (that I am going to simply call contact frame) with a yaw axis that is perpendicular to the environment and centered in the middle of the contact surface in case of surface contacts.

In the following, I will consider a simple robot with a compliant contact, and use a simple model of the contact force, in order to validate the choice of the command vector in task space, and show how the actual command of the robot (joint actuation) affects the dynamics of the robot's CoM.

2.3.1 Dynamics of a Simple Monopod

Consider a robot of mass m with a single prismatic joint making a compliant contact with the environment, as illustrated in figure [2.1](#). The joint has no mass, hence it's acceleration

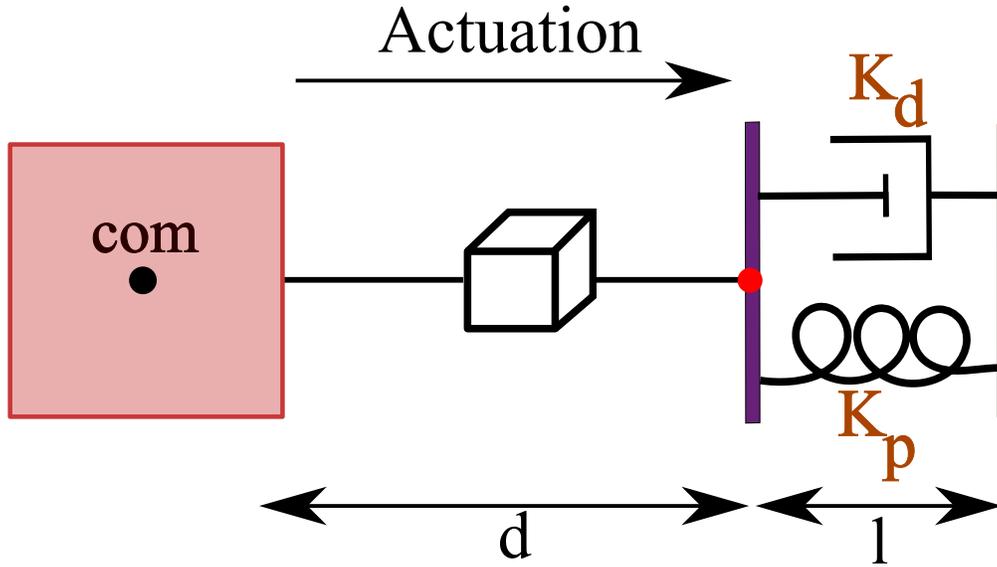


Figure 2.1: Illustration of a 1 d.o.f robot making a compliant contact

controlled:

$$u = \ddot{d}. \quad (2.6)$$

The compliant contact is simply modeled as a spring with a damper, resulting in a contact force

$$f = -K_p(l - l_0) - K_d\dot{l}, \quad (2.7)$$

where l_0 is the length of the spring when uncompressed with a force equals zero, K_p and K_d represent the stiffness and damping of the spring respectively.

The position of the CoM is com . The world frame being on the contact surface, the position of the CoM in this frame is

$$com = d + l. \quad (2.8)$$

To represent the dynamics using a state-space representation, I choose the state x such that

$$x = \begin{pmatrix} com \\ \dot{com} \\ l - l_0 \\ \dot{l} \end{pmatrix}. \quad (2.9)$$

Newton's Law gives

$$\begin{aligned} c\ddot{om} &= \frac{f}{m} \\ &= -\frac{K_p}{m}(l - l_0) - \frac{K_d}{m}\dot{l}. \end{aligned} \quad (2.10)$$

Deriving equation (2.8) twice gives

$$c\ddot{om} = \ddot{d} + \ddot{l}. \quad (2.11)$$

Using (2.10) in (2.11), I can write

$$\ddot{l} = -\ddot{d} - \frac{K_p}{m}(l - l_0) - \frac{K_d}{m}\dot{l}. \quad (2.12)$$

Finally, using (2.10) and (2.12), the dynamics of the state can be written as

$$\begin{pmatrix} c\dot{om} \\ c\ddot{om} \\ \dot{l} \\ \ddot{l} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{K_p}{m} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{K_p}{m} & -\frac{K_d}{m} \end{pmatrix} \begin{pmatrix} com \\ c\dot{om} \\ (l - l_0) \\ \dot{l} \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ -1 \end{pmatrix} \ddot{d}. \quad (2.13)$$

Looking at the dynamics in (2.13), the joint command \ddot{d} only affects the acceleration at the compliance level through \ddot{l} , and is absent from $c\ddot{om}$, which means that the contact force is not directly controlled by the joint command. Furthermore, $c\ddot{om}$ depends on the deformation of the compliance through the stiffness of the compliance and on its velocity through the damping component, which is usually hard to estimate. Having the joint command appear in the expression of the CoM is possible by deriving (2.10) and using (2.12), which gives

$$\begin{aligned} c\ddot{om} &= -\frac{K_p}{m}\dot{l} - \frac{K_d}{m}\ddot{l} \\ &= \frac{K_d}{m} \left(u + \frac{K_p}{m}(l - l_0) + \frac{K_d}{m}\dot{l} \right) - \frac{K_p}{m}\dot{l}. \end{aligned} \quad (2.14)$$

Equation (2.14) shows that the joint command influences the jerk of the CoM, making the latter a candidate for a command variable in the task space. However, in the case of a multiple contact configuration, it is better to benefit from the redundancy of the contacts

and control each contact separately, by controlling the derivative of the contact forces

$$\dot{f} = -K_p \dot{l} - K_d \ddot{l}, \quad (2.15)$$

which is done by controlling the acceleration at the contact level, which is in this case, \ddot{l} .

2.3.2 Dynamics Study in the General Case

Back to the general case introduced in the beginning of this section. For a concise explanation, I will list the assumptions and suppositions taken in the full model.

Assumptions for the Full Model Proposition

- The model is represented in 3D space.
- The entire robot is made of a single floating mass, and has a single inertia tensor.
- The robot uses mass-less limbs to interact with the environment. There are no constraints on the number of contacts.
- Deformations due to compliance are represented by linear and angular stiffness and dampers along each axis of 3D space.
- Contacts with the environment can be of type surface, edge or point contacts.
- The model represents the robot in contact with the environment, around its static desired reference. The dynamics study deals with dynamic behaviors around this static reference.
- Contact friction is not explicitly studied, instead the tracking of the desired forces guarantee the non-slippage of each contact.

An illustration of the model is depicted in figure [2.2](#). The blue frames represent the contact frames, and the red frame represents the world frame. Finally, the green frame represents the floating mass' CoM frame, which is the frame centered at the CoM of the floating mass and having the same orientation as the world base.

Formulation of the Dynamics

The model of the contact forces and moments is represented by

$$f_{c,l,i} = K_{f,p,i} (p_{c,i} - p_{c,i,r}) + K_{f,d,i} \dot{p}_{c,i} \quad (2.16)$$

$$t_{c,l,i} = K_{t,p,i} \Theta (R_{c,i,r} R_{c,i}^\top) + K_{t,d,i} \omega_{c,i} \quad (2.17)$$

where:

- $p_{c,i} \in \mathbb{R}^3$ is the position of contact i ,
- $\dot{p}_{c,i} \in \mathbb{R}^3$ is the linear velocity of contact i ,
- $R_{c,i} \in SO(3)$ is the orientation matrix of the contact i ,
- $\omega_{c,i} \in \mathbb{R}^3$ is the angular velocities of contact i ,
- $p_{c,i,r} \in \mathbb{R}^3$ is the rest position of the contact i (i.e, when forces are zero),
- $R_{c,i,r} \in SO(3)$ is the rest orientation matrix of the contact i (i.e, when moments are zero),
- $K_{f,p,i}$ is the linear stiffness at contact i ,
- $K_{f,d,i}$ is the linear damping at contact i ,
- $K_{t,p,i}$ is the angular stiffness at contact i ,
- $K_{t,d,i}$ is the angular damping at contact i .

The difference in orientation represented by $R_{c,i} R_{c,i,r}^\top$ is considered so small that the approximation $\sin\theta \approx \theta$ is used, thus I define the function $\Theta(\cdot)$ as the axis-sine of angle representation, approximated by

$$\Omega(R) \simeq \frac{1}{2} \text{Vec}(R - R^\top), \quad (2.18)$$

where $\text{Vec}(\cdot)$ is the inverse operator of the skew symmetric:

$$\text{Vec}(S(\omega)) = \omega \quad (2.19)$$

The values of stiffness and damping can be set independently from a contact to another, allowing for implementations on surfaces of different proprieties at the same time. Note that in the case of a point contact i , $K_{t,p,i}$ and $K_{t,d,i}$ are set to zero, and in the case of edge contact, these matrices are set to have semi definite values.

I want to note that this formulation of the contact forces and moments can still be used to represent a relatively rigid interaction, simply by giving high values for the stiffness and damping components. So while the contribution is to design a controller that ensures the robot's balance during a compliant interaction, the controller will also be capable of dealing with relatively rigid robot-environment interactions.

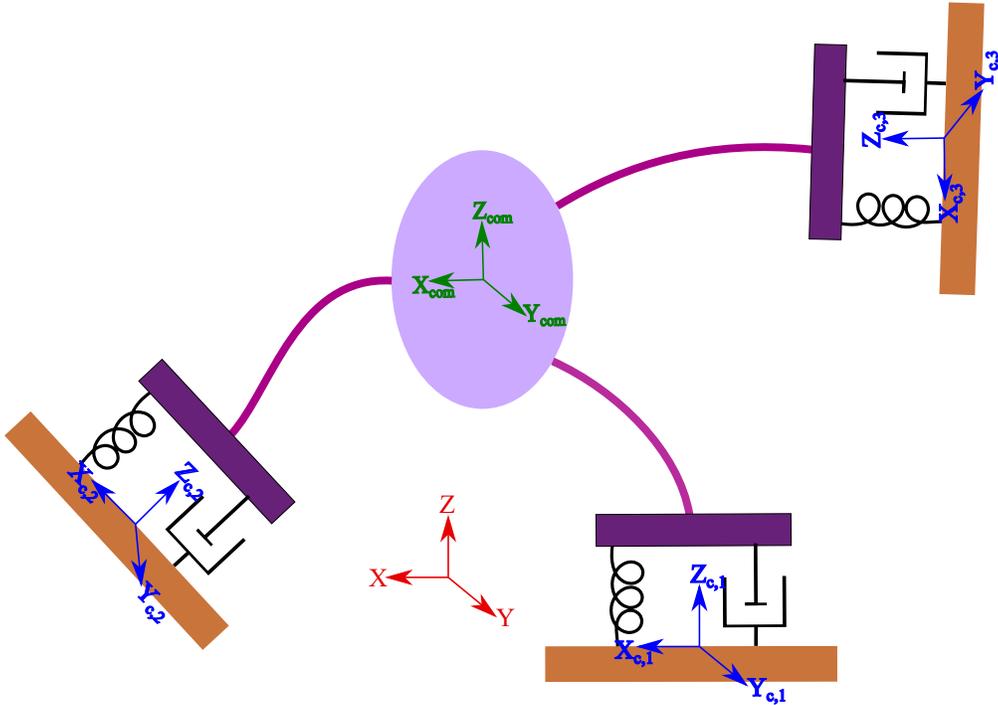


Figure 2.2: Reduced model of the robot making 3 compliant contacts with a viscous-elastic interaction with the environment

As mentioned above, the robot in this reduced model is a single mass rigid body with mass-less legs, which gives a constant inertia tensor $I \in \mathbb{R}^{3 \times 3}$ in the base frame, expressed as RIR^T in the world frame (R here is the orientation matrix of the robot's base). Thus, the angular momentum of the robot is $RIR^T\omega$.

Based on the analysis done in section [2.3.1](#), the command in task space consists of the

accelerations at the contact level, hence why I define the control vector as

$$u \triangleq \left(\ddot{p}_{c,1,com}^\top \quad \dot{\omega}_{c,1,com}^\top \quad \cdots \quad \ddot{p}_{c,n_c,com}^\top \quad \dot{\omega}_{c,n_c,com}^\top \right)^\top \quad (2.20)$$

where $\ddot{p}_{c,i,com} \in \mathbb{R}^3$ and $\dot{\omega}_{c,i,com} \in \mathbb{R}^3$ are respectively the linear and angular accelerations of contact i , written in the CoM frame.

Next, I'm going to describe the dynamics of the model using Newton and Euler's equations, while expressing the contacts positions, orientations, linear and angular velocities in the CoM frame.

This notation has an advantage over the classical world frame expression, which is that the reference trajectories are easier to plan. The feasibility of a desired motion can be studied by studying the kinematics of the robot's limbs arriving at the robot's base, which is more evident compared to the planning of the trajectories with a reference to an anchor frame in the environment, with little information on the feasibility of the robot's motion. A practical example which showcases this advantage is a free fall case. The robot cannot follow a reference trajectory of its limbs with respect to an anchor frame, as it is constantly moving away from the desired reference, however the robot can follow trajectories for its limbs when planned with respect to its own body. Furthermore, this representation replicates the human behavior more accurately, since human move their limbs with respect to their own body, not with respect to an anchor point in the environment.

Using Euler's second law, the relation between angular momentum and the total external moment, is expressed by

$$\sum_{i=1}^{n_c} (S(Rp_{c,i,com}) f_{c,i} + t_{c,i}) = S(\omega) RIR^\top \omega + RIR^\top \dot{\omega} \quad (2.21)$$

where:

- $p_{c,i,com} \in \mathbb{R}^3$ is the position of the contact i in the CoM frame,
- $f_{c,i}$ and $t_{c,i}$ are respectively the force and moment at contact i written in the world frame, which can be obtained from equations (2.16) and (2.17) using

$$f_{c,i} = R_{c,i} f_{c,l,i} \quad (2.22)$$

$$t_{c,i} = R_{c,i} t_{c,l,i} \quad (2.23)$$

Using Newton's second law, and equation (2.21), the linear and angular accelerations of

the CoM and the floating base of the robot can be expressed as

$$c\ddot{om} = \frac{1}{m} \sum_{i=1}^{n_c} f_{c,i} + g \quad (2.24)$$

$$\begin{aligned} \dot{\omega} = & RI^{-1}R^\top \sum_{i=1}^{n_c} (S(Rp_{c,i,com}) f_{c,i} + t_{c,i}) \\ & - RI^{-1}R^\top S(\omega) RIR^\top \omega \end{aligned} \quad (2.25)$$

where m is the mass of the robot, $c\ddot{om}$ is the CoM's acceleration, $g = \begin{bmatrix} 0 & 0 & -g_0 \end{bmatrix}^\top$ and g_0 is the gravity constant.

With these second order dynamics, I use the state-space representation in order to transform them into a first order representation. I define here a state vector focusing on the contacts with the environment. The state variables are the positions, orientations, and velocities of these contacts, in addition to those of the CoM. The contact positions and orientations are going to be defined with respect to the CoM frame.

The state vector is thus defined as

$$x \triangleq \left(x_{com} \quad x_1 \quad \cdots \quad x_n \right)^\top \quad (2.26)$$

with $x_{com} = (com^\top \quad \Omega^\top \quad c\dot{om}^\top \quad \omega^\top)^\top$ and $x_i = (p_{c,i,com}^\top \quad \Omega_{c,i,b}^\top \quad \dot{p}_{c,i,com}^\top \quad \omega_{c,i,b}^\top)^\top$, where:

- $p_{com} \in \mathbb{R}^3$ is the position of the CoM in the world frame,
- Ω is the orientation of the floating base in the world frame,
- $\dot{p}_{com} \in \mathbb{R}^3$ is the velocity of the CoM in the world frame,
- $\omega \in \mathbb{R}^3$ is the angular velocity of the floating base in the world frame,
- $p_{c,i,com} \in \mathbb{R}^3$ is the position of the contact i in the CoM frame,
- $\dot{p}_{c,i,com} \in \mathbb{R}^3$ is the position of the contact i in the CoM frame,
- $\Omega_{c,i,com} \in \mathbb{R}^3$ is the linear velocity of the contact i in the CoM frame,
- $\omega_{c,i,com} \in \mathbb{R}^3$ is the position of the contact i in the CoM frame,

It should be noted that the orientations (Ω and $\Omega_{c,i,com}$) can be written by using any representation of the orientation, such as the quaternion, the axis-angle, Euler angles, etc. Each

Ω has an associated rotation matrix $R \in SO(3)$. Another thing to note is that the dynamics development done above is made using the rotation matrices and not a specific orientation vector. Which means that with whichever orientation notation chosen, the equations are going to be the same.

Using equations (2.20), (2.22), (2.23), (2.24), and (2.25), the non-linear model of the robot can be finally written as:

$$\dot{x} = f(x, u). \quad (2.27)$$

It should be noted that the non-linearity of the dynamics is only present in the state variables representing the single mass rigid body, while the variables representing the contacts have linear dynamics as function of the state and command vectors, since the commanded accelerations are immediate time-derivatives of the velocities in the state vector.

2.4 Linearization of the Non-linear System

The dynamics given by equation (2.27) can be used to derive different types of control. I propose to balance the robot around a fixed, stationary desired equilibrium state x^* , where $(.)^*$ is an index referring to the reference of $(.)$. But even in that case, the dynamics are non-linear. Non-linear systems are complex and hard to solve due to the the high dependency of the system variables on each others. Therefore, to simplify the control synthesis, the local dynamics are approximated by linearization around this desired state. To simplify the notation, I represent the state error x^Δ between the actual state and the desired one by using an operator noted Δ . For the positions and velocities in the state vector, Δ represents the Euclidean difference:

$$(v)^\Delta = (v) - (v)^*. \quad (2.28)$$

As for the orientations, it is the axis-sine of angle representation of the relative orientation between the orientation matrices:

$$\Omega^\Delta = \Theta(RR^{*\top}). \quad (2.29)$$

The linearization of the dynamics is detailed in appendix [A](#).

It is important to note that this linearization is very different from the one commonly performed with a model like the inverted pendulum. In my case, no assumptions are made either on the kinematics of the CoM, on the position nor on the orientation of the contacts. So, this linearization is not less precise in the case of multiple non-coplanar contacts with

different stiffness and damping, including point and edge contacts.

Having linearized the reduced model given in equation (2.27), I define the matrices A and B (also given in appendix A) such that

$$\dot{x}^\Delta = Ax^\Delta + Bu^\Delta. \quad (2.30)$$

With what I have presented so far, these dynamics are well suited for a state feedback control law. However, it would be purely based on kinematics and cannot track reference forces except through the viscous-elastic model. In the case where the stiffness and damping of the environment can be accurately estimated, a state feedback control law can be enough for minimizing the error of the state and for maintaining the robot's balance. On the other hand, the error tracking will worsen when this estimation has errors.

To counter the effect of these estimation and modeling errors, I will use the information from the measured contact forces and moments using force sensors and include this information along with the kinematics of the model in a new state vector, which is going to be covered in the following section.

2.5 Trade-off Between Force Feedback and Kinematics

The classical admittance control, which are used in series with the CoM balance control as done in [Caron et al., 2021], assumes that the dynamics of the force control are fast enough to converge within the time required to maintain balance. However, not only both dynamics remain always coupled, the admittance control becomes significantly slower with more compliant and uncertain environments. For this reason, there is a need for a controller able to take both dynamics into account in a single loop. On the other hand, since the contact forces and moments are dependent of the state variables as seen in 2.22 and 2.23, adding these forces to the state vector will cause a redundancy in the matrices of the linear system, making them singular. A solution would then be to include them in a different vector and then append them to the state vector, the result would be a new state vector that has the kinematics and force information.

However, appending the entire information of the contact forces from force feedback will cause a redundancy in the information about the forces, since the kinematics also have these forces based from the viscous-elastic model. Thus there is a need to establish a trade-off between the conflicting kinematics and force control, which is common to have.

For these reasons, I will define the following vectors and matrix:

- The vector z , having the force values from force sensors.
- The weight matrix W , responsible for the trade-off between the kinematics and force.
- The new state vector y , where the contact forces and moments are combined with the positions and orientations of the corresponding contact bodies using the trade-off.

2.5.1 Definitions

I define the vector z as

$$z \triangleq \begin{pmatrix} 0_{1 \times 12} & z_1 & \cdots & z_{n_c} \end{pmatrix}^\top, \quad (2.31)$$

where $z_i = \begin{pmatrix} f_{s,i}^\top & t_{s,i}^\top & 0_{1 \times 3} & 0_{1 \times 3} \end{pmatrix}^\top$. The vectors $f_{s,i}$ and $t_{s,i} \in \mathbb{R}^3$ are respectively the force and moment at contact i , scaled to the positions and orientations by dividing by the stiffness of the contact, and written in the base frame to make their appendage to the positions and orientations possible. The actual forces and moments are measured with force sensors at the contacts.

As for the weight matrix W used for the trade-off, one way of defining it is as

$$W \triangleq \text{diag}(0_{1 \times 12}, w_1, \cdots, w_{n_c}) \quad (2.32)$$

where $\text{diag}(\cdot)$ is an operator that gives a square matrix, having on its diagonal the values given in between the parenthesis and zeros elsewhere, and

$$w_i \triangleq \begin{pmatrix} w_{fi} & w_{ti} & 0_{1 \times 3} & 0_{1 \times 3} \end{pmatrix} \quad (2.33)$$

where $w_{fi} \in \mathbb{R}^3$ and $w_{ti} \in \mathbb{R}^3$ are vectors having values between 0 and 1 that respectively multiply $f_{s,i}$ and $t_{s,i}$ in the vector z .

Finally, I define the error of the new state vector y^Δ :

$$y^\Delta \triangleq (\mathbb{I} - W)x^\Delta + Wz^\Delta. \quad (2.34)$$

where \mathbb{I} is an identity matrix. Note that by setting the values in W to zero, I will go back to the case of the old state vector x , which does not include the force feedback information.

2.5.2 Dynamics of the New State

Considering that the forces and moments in z can be written as functions of the variables in x using equations (2.16) and (2.17), I can write, after linearization, the vector z^Δ as a matrix M (given in the appendix A) multiplying the vector x^Δ :

$$z^\Delta = Mx^\Delta. \quad (2.35)$$

Using equation (2.35) in (2.34), I can write y^Δ as

$$y^\Delta = Nx^\Delta \quad (2.36)$$

with $N = \mathbb{I} - W + WM$ and W is chosen so that matrix N is non-singular.

I can now write the dynamics of y^Δ as

$$\dot{y}^\Delta \simeq A_y y^\Delta + B_y u^\Delta \quad (2.37)$$

with $A_y = NAN^{-1}$ and $B_y = NB$.

Note that in the case of absence of force sensors for force feedback, this formulation can still be valid by setting W to zero. This will take the linear dynamics back to equation (2.30), where the contact forces are only tracked via the viscous-elastic model.

2.6 Conclusion

In this chapter, I proposed a model for the compliant robot-environment interaction. The model considers the robot as a single rigid mass with mass-less limbs used to make contacts with the environment via a viscous-elastic interaction. Then, I studied the non-linear dynamics of the proposed model and went through a linearization process to obtain a linear system. Then, I proposed a force-kinematics trade-off for an accurate tracking of the contact forces.

In summary, this chapter serves as a description of the proposed reduced model and a careful study of the dynamics imposed by it, alongside the chosen linearization approach, and a method of incorporating the contact forces from force feedback into the kinematics of the system.

Having a linear system, the next step would be to formulate a control law that generates the command which minimizes the error of the state vector. Having chosen the command as

accelerations, the controller in this case must generate the accelerations of the contact limbs, minimizing the errors of the CoM and other state variables as part of its balancing control.

Furthermore, it is important to note that controlling robots is done by sending commands to the joint articulations, and it is usually done by using a motion solver. The proposed controller should be used to send objectives for the solver, which will generate the necessary commands for the joint articulations to maintain the robot's balance.

The next chapter will focus on the control aspect of this thesis, by proposing a controller for the linear dynamics at first, then by integrating the proposed controller within a motion solver.

Chapter 3

Architecture of the Balance-Force Control

3.1 Introduction

In the previous chapter, I proposed a reduced model for the robot-environment interaction, and studied the dynamics of the system while taking the kinematics of the robot and force feedback into the same loop via a trade-off, while choosing the accelerations of the robot's limbs in contact with the environment as command for the control.

This chapter will start by introducing an optimal control law for the linearized system, which will generate the command required to ensure the balance of the robot and the tracking of the contact forces. A test of this controller is shown in a simple simulation of the non linear system, to emphasize on the importance of the trade-off between the force feedback and the kinematics of the system in force tracking. Then I will introduce a motion solver to be used to generate the command to the robot's articulations, while detailing the difference in the control architecture depending on how the robot is controlled.

This chapter is divided as the following: section [3.2](#) introduces the control law and tests it, section [3.3](#) describes the motion solver used and the formulation of the objectives and constraints for the validation presented later on, finally section [3.5](#) concludes the chapter.

3.2 Optimal Control Law for the Linear Dynamics

I will divide this section into 2 parts. The first part will describe the control law, while the second part will test the controller in a simple simulation.

3.2.1 Linear Quadratic Regulator

With the linear dynamics presented in equation [2.37](#), I seek to choose an optimal control that operates the system at minimum cost, which is why I opt to choose a linear quadratic regulator (LQR). The LQR method is a powerful technique for designing controllers for complex systems that have strict and precise performance requirements, and it has been commonly used in control laws for the balance of robots [\[Kuindersma et al., 2014\]](#), [\[Kuindersma et al., 2016\]](#), [\[Mason et al., 2016\]](#), [\[Mason et al., 2014\]](#). It seeks to find the optimal controller that minimizes a given cost function L_y such that

$$L_y = \int_{t_0}^{\infty} (y^{\Delta\top} Q_y y^\Delta + u^{\Delta\top} P u^\Delta) dt, \quad (3.1)$$

where Q_y and P are respectively the symmetric positive semi-definite weight matrices for the state and control vectors that parameterize the cost function. A larger P penalizes the control, so that the accelerations in the command vector will be smaller in norm relative to a case where P has smaller values. On the other hand, a larger Q_y contributes in a quicker convergence of the error of the state to zero. Note that tuning Q_y corresponds to minimizing the error of $y^{\Delta\top}$, meaning that it minimizes the hybrid position-force of the contact instead of the contact position. However, I wanted to visualize the tuning differently, by tuning the contact forces and moments in the matrix W and by tuning the contacts' positions and orientations in the LQR. Which is why I resulted to the following strategy.

Earlier on, I presented the linear dynamics without including the force feedback in equation [2.30](#) and mentioned that force tracking could struggle since it relies on the accuracy of the model of the forces. Here, I can use the LQR formulation to minimize a cost L_x such that

$$L_x = \int_{t_0}^{\infty} (x^{\Delta\top} Q x^\Delta + u^{\Delta\top} P u^\Delta) dt. \quad (3.2)$$

Here the matrix Q tunes the positions and orientations of the contacts without the corresponding contact forces and moments, and the matrix P tunes the same command vector u^Δ . Using equation [2.36](#) in [3.1](#), and by supposing that the costs in [3.2](#) and [3.1](#) are equal, I can write Q_y as function of Q :

$$Q_y = N^{-\top} Q N^{-1} \quad (3.3)$$

This way, I can tune Q and use equation [3.3](#) to use Q_y in the gain calculation.

The problem then boils down to solving a Riccati Equation, which provides us with the optimal gain matrix K such that $u^\Delta = -Ky^\Delta$ induces the minimum cost L_y .

| Contact | First | Second | Third |
|---------------------|---|---|---|
| $p_{c,i,com}$ | $[0 \ -0.1 \ -0.9]^\top$ | $[0 \ 0.1 \ -0.7]^\top$ | $[0.2 \ 0 \ -0.4]^\top$ |
| $\Omega_{c,i,com}$ | $[0 \ 0 \ 0]^\top$ | $[30^\circ \ 0 \ 0]^\top$ | $[0 \ 0 \ 0]^\top$ |
| $K_{f,p} (N/m)$ | $4 \times 10^4 \mathbb{I}_{3 \times 3}$ | $3.5 \times 10^4 \mathbb{I}_{3 \times 3}$ | $3 \times 10^4 \mathbb{I}_{3 \times 3}$ |
| $K_{f,d} (N.s/m)$ | $1500 \mathbb{I}_{3 \times 3}$ | $1200 \mathbb{I}_{3 \times 3}$ | $1000 \mathbb{I}_{3 \times 3}$ |
| $K_{t,p} (N/rad)$ | $800 \mathbb{I}_{3 \times 3}$ | $750 \mathbb{I}_{3 \times 3}$ | $700 \mathbb{I}_{3 \times 3}$ |
| $K_{t,d} (N.s/rad)$ | $50 \mathbb{I}_{3 \times 3}$ | $45 \mathbb{I}_{3 \times 3}$ | $40 \mathbb{I}_{3 \times 3}$ |

Table 3.1: Contacts positions, orientations, stiffness and damping as used in the test

3.2.2 Testing the Control

I will test the LQR controller in a simple simulation of the non-linear model, to showcase the difference between the inclusion and absence of force feedback in the control.

The simulation is done using Matlab SIMULINK, where the LQR gain can be calculated simply using functions built in Matlab, using the continuous form of the linear matrices. I relied on the non-linear model in equation [2.27](#) to simulate the proposed model, i.e. a floating mass with mass-less limbs, making contacts with the environment whose contact forces and moments are modeled according to the viscous-elastic model (basically I am simulating the illustration given in figure [2.2](#)). The simulations were run using a variable step solver.

In this test, the robot is making 3 contacts. I chose the rigid floating base to have a mass $m = 101Kg$, with an inertia tensor

$$I = \begin{pmatrix} 270 & 0 & 0 \\ 0 & 212 & 0 \\ 0 & 0 & 70 \end{pmatrix}$$

I chose the contacts positions so that I have 3 contact surfaces that are non-coplanar. Each contact is compliant with different stiffness and damping values.

In general, estimating the real stiffness in a real testing environment is not perfect, and leads to modeling errors in these values. Which is why the controller should be expected to behave correctly and achieve balance despite these errors. Which is why in this test, I set their values in the controller 10% lower than their values in the non-linear system.

The reference positions of the contacts and their orientations in the CoM frame, in addition to the stiffness and damping values given in the non-linear model and in the controller are given in table [3.1](#). The orientation of contacts is using the axis-angle notation.

The CoM is pushed from its reference position by 10 cm along the x axis at the start of the simulation, the controller works on converging this error to zero.

The interest of this test is to showcase the difference in force tracking when considering the force feedback in the control, which is why I am going to show the error of the contact force and moment at a contact position, in addition to the error of the CoM of the simulated model (to verify balance), and the error of a contact position. I showcase a component per variable, in order to make the images clearer, especially when the results and analysis are the same.

Two simulations were run, one with $W = 0$ (meaning no force feedback used) and the other while choosing

- $w_{fi} = \begin{bmatrix} 0.75 & 0.75 & 0.75 \end{bmatrix}$,
- $w_{ti} = \begin{bmatrix} 0.75 & 0.75 & 0.75 \end{bmatrix}$.

As for the LQR weight matrices, I set them identical in both simulations. The priority is set for the CoM position and contacts positions in the Q matrix, and their respective linear velocities is also tuned to damp possible oscillations. In other words, I set them as

- $Q = \text{diag}(Q_{com} \quad Q_c \quad Q_c \quad Q_c)$,
- $Q_{com} = (10^4\mathbb{I}_3 \quad \mathbb{I}_3 \quad 300\mathbb{I}_3 \quad \mathbb{I}_3)$,
- $Q_c = (10^4\mathbb{I}_3 \quad \mathbb{I}_3 \quad 300\mathbb{I}_3 \quad \mathbb{I}_3)$,
- $P = \mathbb{I}_{18}$.

Figures [3.1](#) and [3.2](#) show the errors of the CoM position and of the position of the second contact written in the CoM frame, respectively. The controller succeeds in the convergence of these errors to zero without oscillations, whether the force feedback is used or not. It is important to note that the few millimeters that appear as a steady state error for the CoM error curves are caused by the modeling errors in the controller setup.

Figures [3.4](#) and [3.3](#) respectively show of the contact force and moment errors at the second contact. Here a clear difference can be noticed between the 2 cases. When the force feedback is used, the forces nearly converge towards zero, while steady state errors of $10.53N$ for the contact force and of $0.73N.m$ for the contact moment appear otherwise, showing that in this case the reference tracking of the forces and moments is not as reliable.

In this simple case, the controller could only follow the desired forces when the force feedback was used, however the controller could always follow the desired CoM position, due to the simplicity of the simulated system. When simulating a more complex robot, the bad tracking of the desired forces could lead to unexpected behaviors and to the loss of balance.

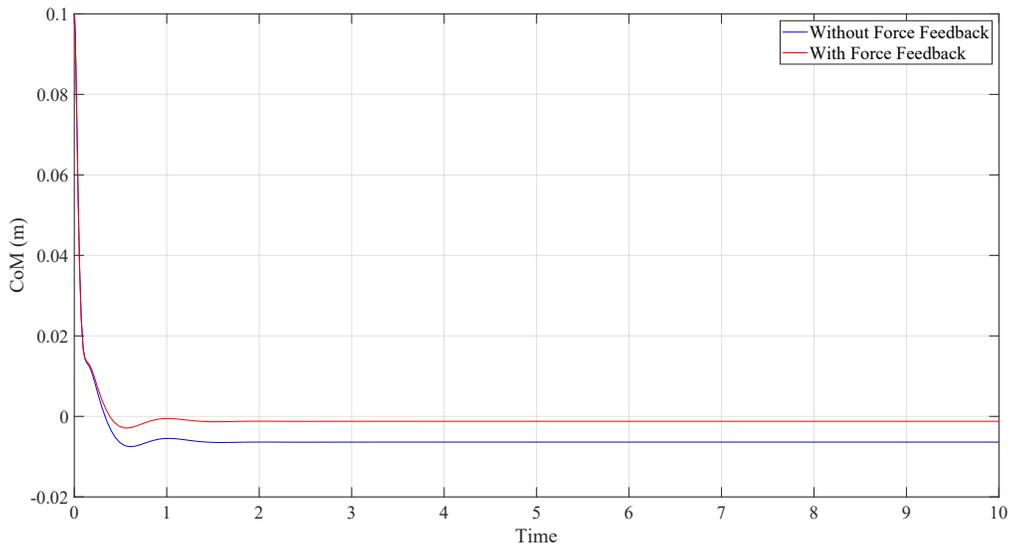


Figure 3.1: CoM position error (x component)

3.3 Quadratic Program Motion Solver

Commanding robots to move is done by sending commands to the articulations of the robot, which is done using motion solvers. Motion solvers receive the command in task space and generate the reference trajectory for the robot's joints.

The proposed controller is using $6 \times n_c$ variables of control in its command vector u . Many robots, especially humanoid ones, are equipped with more degrees of freedom to be able to perform many concurrent tasks.

The tasks, or objectives, to be performed by robots, have a wide variety, and can be divided into point to point tasks (such as moving a limb into a specific position in space) and areas of satisfaction tasks (such as keeping the end effector inside a specific region in space). Furthermore, any task performed by the robot is bound by the performance and limitations of the actuators the robot is equipped with, in other words, any task is usually constrained by certain conditions.

A wide spread expression for representing the objectives to be performed by the robot is the task-function approach, which is also used to deduce from this expression the control to

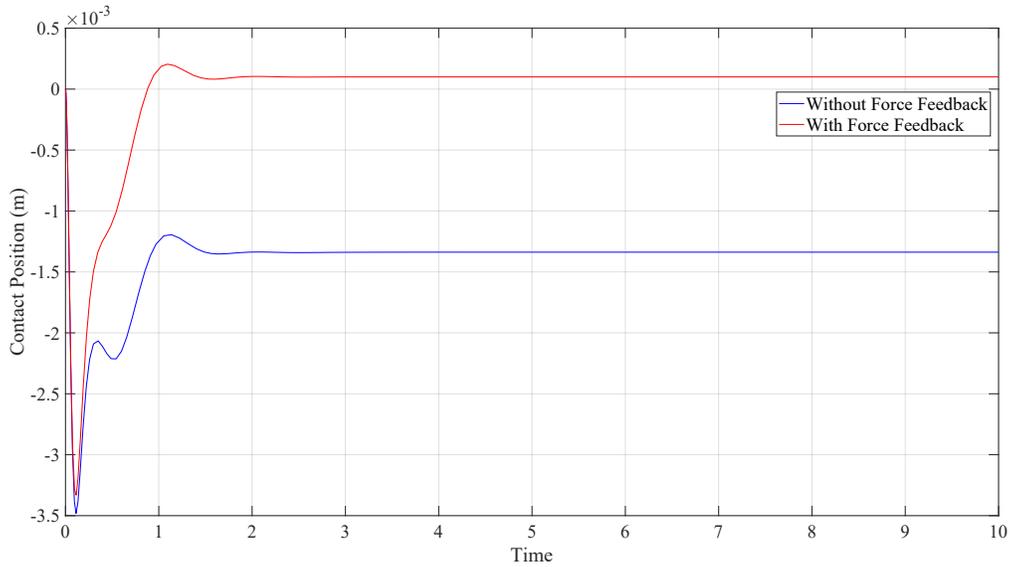


Figure 3.2: Second Contact position error (x component)

be applied at the joint level [Escande et al., 2014]. The objectives are written in the task space, and transformed to the joint spaces using the appropriate jacobian matrices.

To deal with the redundancy and with multiple objectives at the same time, it's common to use a QP solver. This optimization problem minimizes the tracking error of different weighted objectives, while abiding by several defined constraints.

The proposed control law introduced earlier in this chapter generates accelerations for the robot's limbs in contacts with the environment. These accelerations are going to be used to formulate objectives for the limbs in contact with the environment in the QP, which is going to serve as a motion solver for the robot to execute the necessary motion to maintain its balance while following the desired forces. Figure 3.5 shows a simple illustration of the implementation of the controller with the QP motion solver without presenting the full control architecture. The reduced model is deduced from the robot (and its compliant interaction), the controller is designed based on the reduced model, and the QP motion solver takes the controller's command to generate the robot's motion.

3.3.1 Optimization Problem

The output of the QP, or the decision variable, can be chosen depending on the type of robots to be controlled, however it is convenient to have the acceleration \dot{a} as the decision variable, to take into account dynamical constraints, such as the joint torque's range. Additionally, this will make the formulation of the accelerations tasks for the contact limbs simpler, and

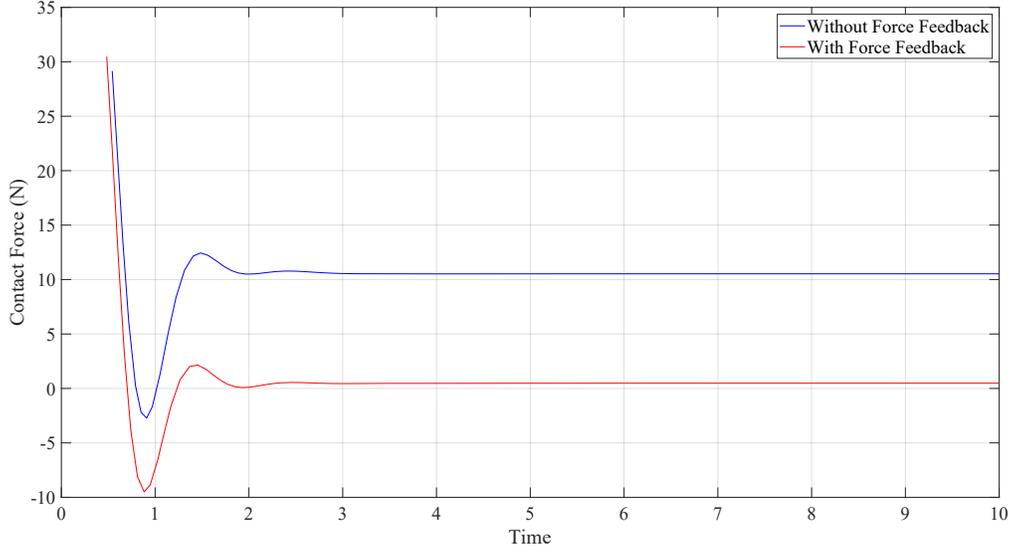


Figure 3.3: Contact Force error (z component)

the proposed controller will naturally fit into this motion solver, since it directly provides the desired Cartesian accelerations of the contact limbs, and needs only simple Jacobians.

The QP calculates the optimal reference acceleration $\dot{\alpha}_r$, by solving

$$\begin{aligned} \dot{\alpha}_r &= \underset{\xi}{\operatorname{argmin}} \|W_{task} (A_{ob}\xi - b_{ob})\|^2, \\ s.t. \quad &A_{eq}\xi = b_{eq}, \quad A\xi \leq b, \quad l_b \leq \xi \leq u_b, \end{aligned} \quad (3.4)$$

where:

- W_{task} is a positive diagonal matrix made up of diagonal weighting matrices for each objective,
- the matrix A_{ob} and the vector b_{ob} contain the corresponding objectives,
- the matrix A_{eq} and the vector b_{eq} contain the equality constraints,
- the matrix A and the vector b contain the inequality constraints,
- the vectors l_b , and u_b are respectively the lower and upper bounds for the bounded constraints.

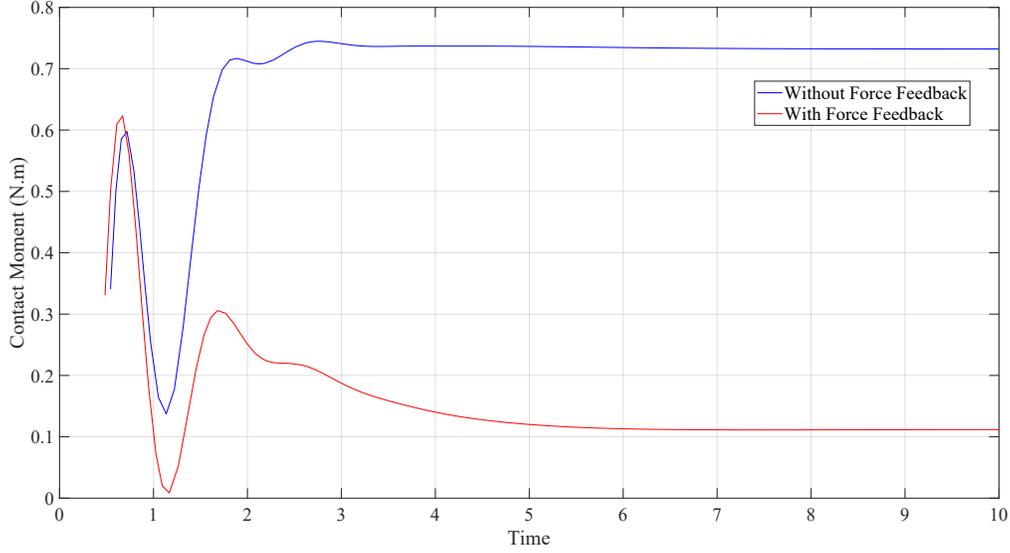


Figure 3.4: Contact Moment error (x component)

3.3.2 Objectives and Constraints

I will now introduce a set of objectives and constraints used in the optimization problem in equation (3.4) for the balance-force controller.

The objectives to be used during the validation of the controller are as the following:

1. **Posture objective:** an acceleration objective, tracked with a proportional derivative (PD) of scalar gains k_p and k_v , is used to track the posture of the robot. The objective is calculated using

$$A_{ob} = \begin{bmatrix} 0_{n \times 6} & \mathbb{I}_n \end{bmatrix} \quad (3.5)$$

$$b_{ob} = k_p (q_d - q) + k_v (\dot{q}_d - \dot{q}) + \ddot{q}_d. \quad (3.6)$$

This objective will have a relatively low weight, since it does not serve in the balance control.

2. **Contacts objectives:** Acceleration objectives are defined for the contact limbs, labeled as hands and feet objectives, calculated using

$$A_{ob} = J_{c,i,com}(q) \quad (3.7)$$

$$b_{ob} = \left(\ddot{p}_{c,i,com}^\top \quad \dot{\omega}_{c,i,b}^\top \right)^\top - \dot{J}_{c,i,com}(q, \alpha)\alpha, \quad (3.8)$$

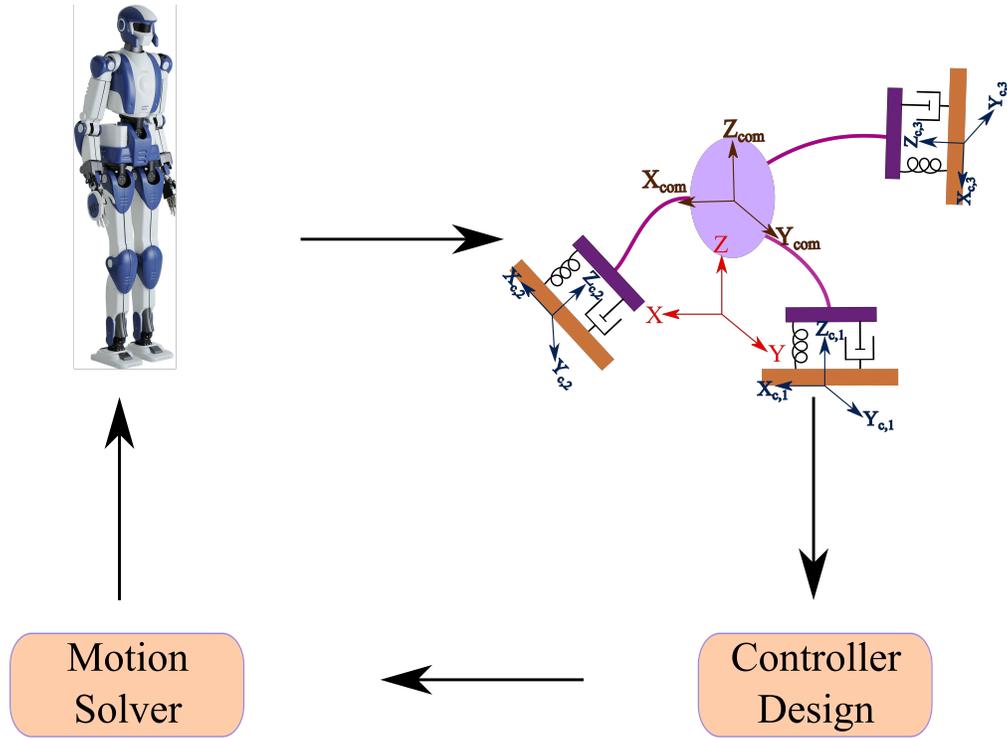


Figure 3.5: Illustration of the implementation of the controller. The robot’s model is reduced, and the controller is formulated based on the reduced model. Accelerations are formulated as objectives in the motion solver, which generates the robot’s motion.

where $\left(\ddot{p}_{c,i,com}^\top \quad \dot{\omega}_{c,i,b}^\top \right)^\top$ are the accelerations of the contacts given by the proposed controller, and $J_{c,i,com}$ is the Jacobian linking the velocities of the contact i written in the CoM frame with the joint velocities, and $\dot{J}_{c,i,com}$ being its time derivative. These objectives will take care of the balance and force control, which is why they are given the highest weight in the solver.

As for the constraints, two physical constraints are considered, however I divided the second constraint into 2 based on how they are formulated:

1. **Joint limits constraints:** the range and speed limit of the joints can be specified using bounded constraints while specifying the lower and upper bounds for each joint.

This is done similarly as in [\[Vaillant et al., 2016\]](#), by choosing

$$l_b = \begin{bmatrix} -1e6\mathbb{I}_{1 \times 6} & l_{bj}^\top \end{bmatrix}^\top \quad (3.9)$$

$$u_b = \begin{bmatrix} 1e6\mathbb{I}_{1 \times 6} & u_{bj}^\top \end{bmatrix}^\top \quad (3.10)$$

$$l_{bj}dt = \max \left(\dot{q}_{min}, \zeta \frac{(q - q_{min}) - q_s}{q_i - q_s} \right) - \dot{q} \quad (3.11)$$

$$u_{bj}dt = \min \left(\dot{q}_{max}, \zeta \frac{(q_{max} - q) - q_s}{q_i - q_s} \right) - \dot{q} \quad (3.12)$$

$$q_i = 0.1 (q_{max} - q_{min}) \quad (3.13)$$

$$q_s = 0.01 (q_{max} - q_{min}) \quad (3.14)$$

where ζ is a velocity damper vector, and q_s represents a security range.

2. **Under-actuation:** this constraint ensures the generation of a feasible motion for the floating base of the robot. it is formulated as an equality constraint, where the index B is used to refer to the first 6 rows of the matrices corresponding to the under-actuation. This constraint is written with

$$A_{eq} = M_B \quad (3.15)$$

$$b_{eq} = -C_B \alpha - G_B + F_B \quad (3.16)$$

3. **Torque constraint:** this constraint ensures that the joint torques are within the limitations of the actuators. 2 inequality constraints are used (one for the minimum and another one for the maximum), and the index j is used to refer to the rows of the matrices corresponding to the actuation part (which immediately follows those of the under-actuation). This constraint can be specified by choosing

$$A = M_j \quad (3.17)$$

$$b = \tau_{max} - C_j \alpha - G_j + F_j \quad (3.18)$$

for the maximum torque condition and

$$A = -M_j \quad (3.19)$$

$$b = -\tau_{min} + C_j \alpha + G_j + F_j \quad (3.20)$$

for the minimum torque condition.

3.4 Closed Loop Control Architecture

So far I have identified the optimization control law for the linear dynamics based on the reduced model of the compliant robot-environment interaction, and I have introduced the solver used to generate the command for the robot's joints based on the different objectives and constraints.

Now I will discuss how the LQR control, the QP solver, and the robot are connected in a closed loop control, depending on how the robot is controlled.

Indeed, the LQR control minimizes the error of the chosen state vector, which is taken as feedback. The way this feedback is taken differs when the robot is torque controlled or position controlled.

Torque Controlled Robot

In this case, the feedback can be entirely taken from the robot itself, by applying kinematics on the joint's position and velocities provided by the robot's encoders. Figure 3.6 gives an overall view of the control structure when applied to a torque controlled robot. The LQR control generates the accelerations for the limbs of the robot in contact with the environment, whose objectives, alongside the posture objective and the above mentioned constraints, are minimized using the QP. The latter generates the reference accelerations used in the joint torque calculation using 2.4. A passivity based term $\delta\tau$ is added to the joint torques, calculated using

$$\delta\tau = (C + \lambda H) (\alpha_r - \alpha), \quad (3.21)$$

where α_r is the reference velocity vector, obtained by integrating $\dot{\alpha}_r$, and λ is a constant.

Position Controlled Robot

In this case, taking the feedback entirely from the robot might put the stability of the closed loop in risk. As the QP solver generates the joint accelerations for the required motion, torque drivers are required to generate the joint torques, using joint positions and velocities errors, which use the integration of the QP's output of joint accelerations. Torque drivers are usually unknown to the user, and the torque calculation method differ depending on the

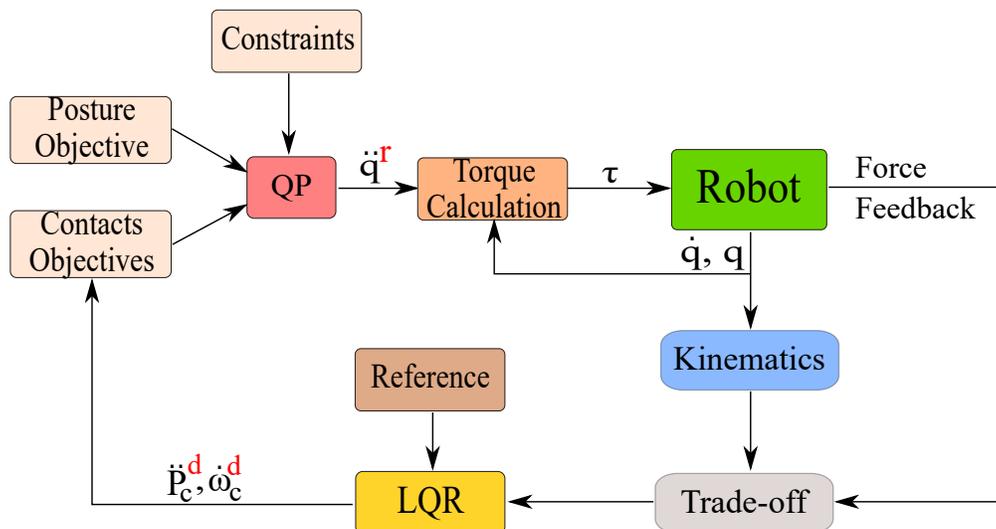


Figure 3.6: Diagram showing the control loop with a torque controlled robot. The yellow block represents the LQR control used on the linearized dynamics of the reduced model.

robot's motors. One way of writing the formula of the joint torque is

$$\tau = K_1 (q - q^r) + K_2 (\dot{q} - \dot{q}^r), \quad (3.22)$$

where q^r is the joint reference, calculated by integrating the joint accelerations at the output of the QP, and K_1 and K_2 are gain matrices. These gains are proprieties of the robot's motors and can be tuned in order to achieve the desired performance.

Unfortunately, the torque in equation (3.22) is not perfectly achievable, instead the actual torque produced by the motors have an additional term τ_{fr} :

$$\tau_{actual} = \tau + \tau_{fr}. \quad (3.23)$$

This term comes from non-modeled parameters such as the friction of the motors, making it impossible to produce a perfect torque as desired. This will add uncertainty about the stability of the control loop and will render it unstable in many situations, and no tuning of the motor gains can completely eliminate this effect.

For these reasons, I will divide the feedback taken from the robot into two groups:

1. The first one is the positions, orientations, linear and angular velocities of the robots limbs in contact with the environment. This data is going to be taken directly by transforming the joint space from the QP output to the cartesian space, this way equations (3.22) and (3.23) are avoided.

- The second group is the CoM position and velocity, the orientation of the robot's base and its angular velocity, which will be taken from the robot using estimations, as these variables don't need require a position-torque transformation. Additionally, the contact forces and moments are taken from the force sensors equipped at the robot's hands and feet.

It is important to mention that in this case, only the joints limits constraint is used from the constraints described above, due to the impossibility of accurately producing the desired torque, making the under-actuation and torque limits constraints difficult to abide by.

Figure 3.7 shows the control loop with a position controlled robot. Notice how the feedback for the LQR controller is divided between kinematics from the QP and from the Robot. The torque calculation could be done using (3.22) and (3.23).

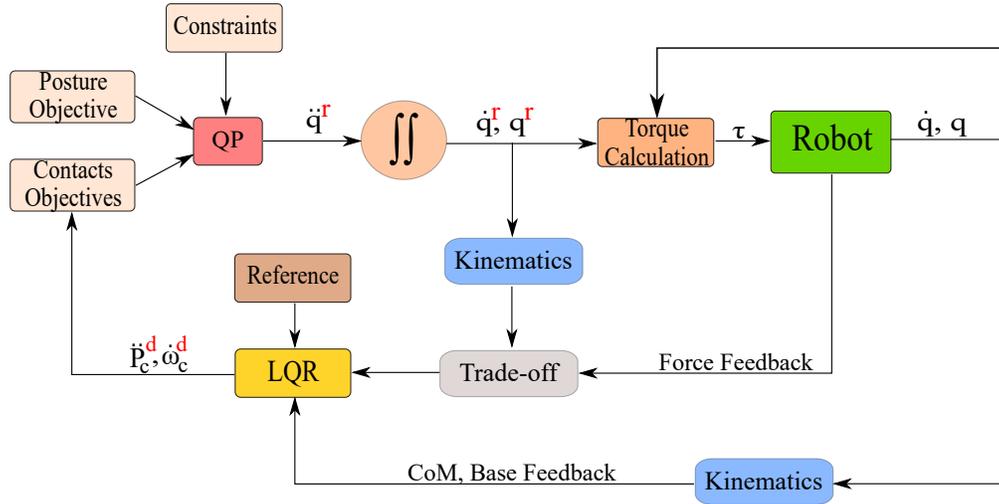


Figure 3.7: Diagram showing how the controller is implemented with a position controlled robot.

3.5 Conclusion

In this chapter, an LQR controller was proposed as an optimal control for the linearized dynamics, in order to generate the contact limbs accelerations which minimize the error of the forces and of the kinematics for balance. It was then tested in a simple simulation of the non-linear model. In the simulation, the difference between the inclusion and absence of force feedback in the control loop is highlighted. While the controller can ensure balance in

both cases, the inclusion of force feedback helps in the reference force tracking, leading to the convergence of the force error to zero amidst modeling errors in the controller.

Then, a motion solver based on a QP formulation was introduced, along with the objectives and constraints to be used in the control architecture.

Finally, the control structure was presented depending on how the robot is controlled. In the case of a torque controlled robot, the kinematics feedback can be taken from the robot without a risk on the stability of the closed loop. In the case of a position controlled robot on the other hand, the feedback for the positions, orientations and velocities of the contacts should be taken by applying kinematics on the joint positions and velocities from the QP output, in order to avoid the risk of instability of the closed loop.

The next chapter will showcase the first part of the validation of the controller, by applying it in simulation to a torque controlled robot.

Chapter 4

Validation on a Torque Controlled Biped

4.1 Introduction

In the previous chapter, the control architecture for the balance-force control was presented for torque and position controlled robots, and the integration of the LQR optimal control within a QP motion solver was detailed. In order to validate the theoretical study, the controller is in need to be applied to a robot, and it is expected that the controller is able to maintain the robot's balance and minimize the error of the state and contact forces amidst external perturbations and modeling errors.

In this chapter, I will validate the proposed controller in simulation by implementing it on a 26 d.o.f torque controlled biped. The robot is going to be put in multiple configurations while receiving perturbations, and the controller is expected to maintain the balance of the robot while minimizing the error of the contact forces and of the state vector.

This chapter is divided as follows. Section [4.2](#) describes the simulation environment in SIMULINK, section [4.3](#) describes the experiments done along with the results. Finally section [4.4](#) concludes the chapter.

4.2 SIMULINK Simulation Environment

The control is tested in simulation on the biped robot described in [Cisneros et al., 2018b](#) using Matlab SIMULINK and the Simscape Multibody library. The robot is torque controlled, has 2 arms with 7 d.o.f each, 2 legs with 6 d.o.f each, and an under-actuated main body, accumulating to 26 d.o.f in total. With a height of $1.8m$, it has a total mass of 77 kg, distributed on every link. The body, hands and feet are illustrated as boxes while the

other links have a cylindrical shape with a radius of $0.001m$. The inertia tensors are then calculated for each part according to its shape. Additionally, the joints were implemented as driven by geared servo-motors considering an additional rotor inertia. The input to these servos is the joint torque, scaled by the corresponding gear ratio.

The Simscape Multibody Contact Forces Library is used to generate the non-rigid contact models. Each corner of the base of a foot or a hand has a virtual sphere attached to it, with its own linear stiffness and damping parameters, generating a force

$$f_j = K_{f,p,j} (p_j - p_j^*) + K_{f,d,j} \dot{p}_j \quad (4.1)$$

$$p_j = p_{c,i} + R_{c,i} p_{j,l}, \quad (4.2)$$

where p_j is the position of the contact sphere in the world frame ($p_{j,l}$ refers to the position of the contact sphere in the contact frame), and $K_{f,p,j}$ and $K_{f,d,j}$ are the stiffness and damping constants for the forces for a sphere j .

The total force at each contact is then

$$f_{c,i} = \sum_{j=1}^4 f_j \quad (4.3)$$

The moment created by each sphere around the contact frame is calculated using

$$\begin{aligned} t_j &= f_j \times (p_{c,i} - p_j) \\ &= - f_j \times R_{c,i} p_{j,l} \\ &= S(R_{c,i} p_{j,l}) f_j. \end{aligned} \quad (4.4)$$

Thus, the total moment created by the four spheres of each contact is

$$t_{c,i} = \sum_{j=1}^4 t_j \quad (4.5)$$

According to the model introduced in the previous chapter, the contact force and moment are produced via deformation using linear and angular stiffness and damping, with each one of these parameters being represented by a matrix representing the total stiffness and damping at each contact. Since these parameters are used in the linear matrices describing the linear dynamics and in the calculation of the control gain, I need to provide the controller with the equivalent of these parameters in the model used in the contact forces library. In other

words, I need to find $K_{f,p,i}$, $K_{f,d,i}$, $K_{t,p,i}$ and $K_{t,d,i}$ from equations [2.22](#) and [2.23](#) in equations [4.3](#) and [4.5](#). This is done by making an equivalence between how the forces are formulated in the proposed model and the ones generated by the contact forces library in SIMULINK, and by deriving the equations with respect to the position and velocities of the contact limbs. The calculation is detailed in appendix [B](#).

The maximum static friction of the contact can also be set in the contacts sphere configurations, and it is set to 0.4.

The simulations were run while allowing the simulator to automatically choose the max and min time step size using a variable step solver. This setting provides a simulation with high precision that is able to break down highly dynamical effects and is computationally very costly [\[Li et al., 2021\]](#).

4.3 Validation Scenarios and Results

The tests are done on scenarios with different starting postures, as illustrated in Figure [4.1](#). The simulation is started with the robot in contact with the environment, slightly above the equilibrium state, to avoid introducing too much initial energy. Then, the robot is perturbed to assess control performance. The robot is expected to maintain its balance and to keep the contacts with the environment, which is why I will focus on the analysis of the CoM position of the robot, and on the CoP of each contact (in the contact frame). I will also highlight the friction at each contact to ensure that no slippage is occurring.

Furthermore, to challenge the controller, white noise is added to the measured forces, and modeling error is introduced by providing reduced stiffness and damping values to the controller compared to the actual values that I chose for the environment. The stiffness is reduced by 20% and the damping by 10%.

To obtain the best performance, the controller needs to be tuned. Tuning consists of adjusting the gains of the matrices Q , P , and W , and configuring the stiffness and damping of each contact. For a general guide on how to tune the controller, check appendix [C](#).

4.3.1 Case of 2 Contacts

The robot, in this case, is in a half-sitting configuration while making contacts with its feet on the ground, as shown in Figure [4.1](#) (a). The stiffness and damping constant parameters at the contacts are given in Table [4.1](#) for the Right Foot (RF) and Left Foot (LF). The table

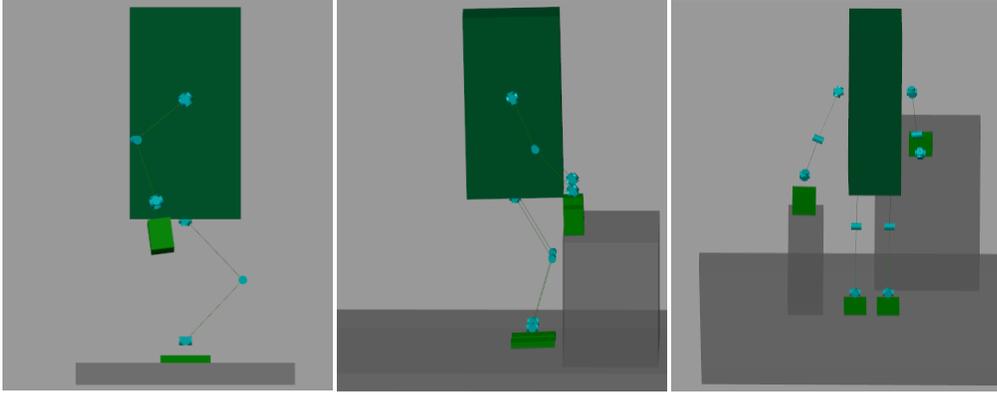


Figure 4.1: Starting postures of scenario (a) (left), (b) (middle) and (c) (right)

| Contact | RF | LF |
|---------------------|-----------------|-----------------|
| $K_{f,p}$ (N/m) | 3×10^4 | 3×10^4 |
| $K_{f,d}$ (N.s/m) | 1000 | 1000 |
| $K_{t,p}$ (N/rad) | 400 | 400 |
| $K_{t,d}$ (N.s/rad) | 15 | 15 |

Table 4.1: Stiffness and damping for each foot in a 2-contact case

shows the real values at the robot's feet, the controller is provided with reduced values as mentioned above).

I set the force-kinematics trade-off matrix W such that

- $w_{fi} = \begin{bmatrix} 0 & 0 & 0.5 \end{bmatrix}$
- $w_{ti} = \begin{bmatrix} 0.5 & 0.5 & 0 \end{bmatrix}$,

to emphasize on the tracking of the normal force and of the horizontal moments.

I choose the LQR control matrices such that:

- $Q = \text{diag}(Q_{com} \quad Q_c \quad Q_c)$,
- $Q_{com} = (1 \times 10^4 \mathbb{I}_3 \quad \mathbb{I}_3 \quad 3 \times 10^4 \mathbb{I}_3 \quad \mathbb{I}_3)$,
- $Q_c = (3 \times 10^4 \mathbb{I}_3 \quad \mathbb{I}_3 \quad 3 \times 10^4 \mathbb{I}_3 \quad \mathbb{I}_3)$,
- $P = \text{diag}(\mathbb{I}_6)$,

giving importance to the positions of the CoM and the contact positions, and adding a damping effect to the position tracking via the tuning of the linear velocities.

The robot is hit at its base with a perturbation of $250N$ at $t = 0.5s$ for $0.1s$ along the x-axis, and the controller is expected to bring it back to its equilibrium position.

To fully demonstrate the advantages of my controller over the classical balance control with a rigid interaction assumption, I compare my controller in this case to the stabilizer of [Murooka et al., 2021], where the effect of the compliance built in the robot’s feet is modeled as a first order lag system on the ZMP. This stabilizer relies on the DCM dynamics to calculate a desired wrench, which is used to create a CoM balance task and admittance control for the feet. It replaces my controller in Figure 3.6. Feet objectives are formulated with PD gains while using the stabilizer’s velocities as references.

Additionally, the QP in this case solves for the contact forces alongside the reference accelerations, and has friction cone constraints, just like in [Cisneros et al., 2018a]. I will refer to this stabilizer as the feet damping controller in the comparison below.

My proposed controller manages to keep the robot’s balance, while the feet damping controller struggles until the robot falls down. This is clear when looking at the CoM position in 4.2 and the CoP of the right foot in Figure 4.3. The feet damping controller struggles to follow the proper forces in the non-rigid environment, which explains the large fluctuations in the foot CoP and the inability of the controller to maintain the robot’s balance after the perturbation. On the other hand, my proposed controller manages to minimize the error of the CoM and to keep the CoP of each foot within its support zone (each foot has a surface of $x : 0.23m \times y : 0.13m$). The friction ratio at the right foot is shown in figure 4.4. The maximum value does not exceed 0.176, and no slippage occurs. Note that the friction curves oscillate slightly before convergence to zero, due to the presence of slight oscillations in the horizontal contact forces before they also converge to zero in this scenario. For a better tracking of the horizontal contact forces, they should be tuned in w_{fi} .

To gauge the robustness to modeling errors of my controller, the same scenario is repeated with the same tuning while reducing the linear and angular stiffness of the contacts by 50% and then again by 70% in the controller. Despite the presence of stronger oscillations of the CoM and a bigger steady-state error as seen in Figure 4.5, the controller is still able to maintain the robot’s balance. Reducing the stiffness in the controller by more than 70% might cause the robot to fall due to a CoM error that places the CoM outside of the support region. A finer tuning is perhaps needed to make the controller handle such modeling errors.

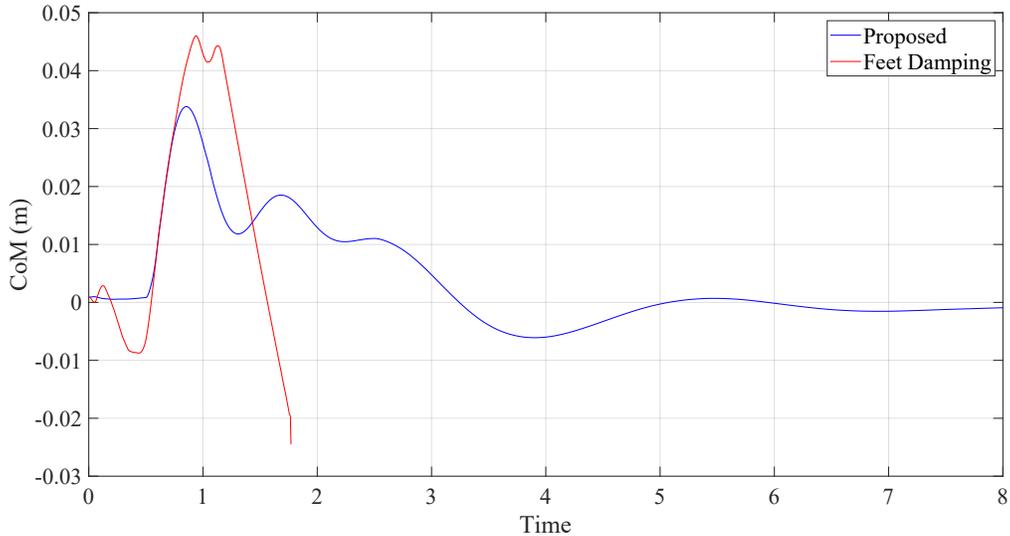


Figure 4.2: CoM position error (x component) in scenario (a)

4.3.2 Cases of 4 Contacts

Here, I test two different scenarios while the robot is making four contacts with the environment, as illustrated in Figure 4.1: the robot is putting its hands on a table-like surface (scenario (b)), and the robot is making non-coplanar contacts with its hands (scenario (c)). The stiffness and damping constants at the contacts for the Right Foot (RF), Left Foot (LF), Right Hand (RH) and Left Hand (LH) are given in Table 4.2 under scenarios (b) and (c). While I did test the feet damping controller in these multi-contact scenarios, I will not be sharing the plots of the signals from those tests. This is because the robot was already struggling to follow the contact forces calculated by the QP even before any perturbation due to the non-rigidity of the contacts, and because the fluctuation in the CoP of each contact was very strong. Therefore, the contacts broke quickly and the robot fell. While this stabilizer can take into consideration external forces at additional contact levels, it doesn't aim to calculate a stabilizing wrench using these additional contacts for the sake of minimizing the CoM error to maintain balance, rather the forces at the hands level are taken as external perturbations whose effects on the CoM are compensated. This is another reason to why it is already expected to not behave correctly, as it is not suited for such experiment.

With the proposed control, the robot is hit with a perturbation of $200N$ at $t = 0.5s$ and then again with $250N$ at $t = 2.5s$ for $0.1s$. In the case of scenario (b), a bias of 5° to the orientation of the base is added to the robot at the start of the simulation to test the robustness of the control against additional orientation errors. In this case, I set the force-kinematics trade-off matrix W such that

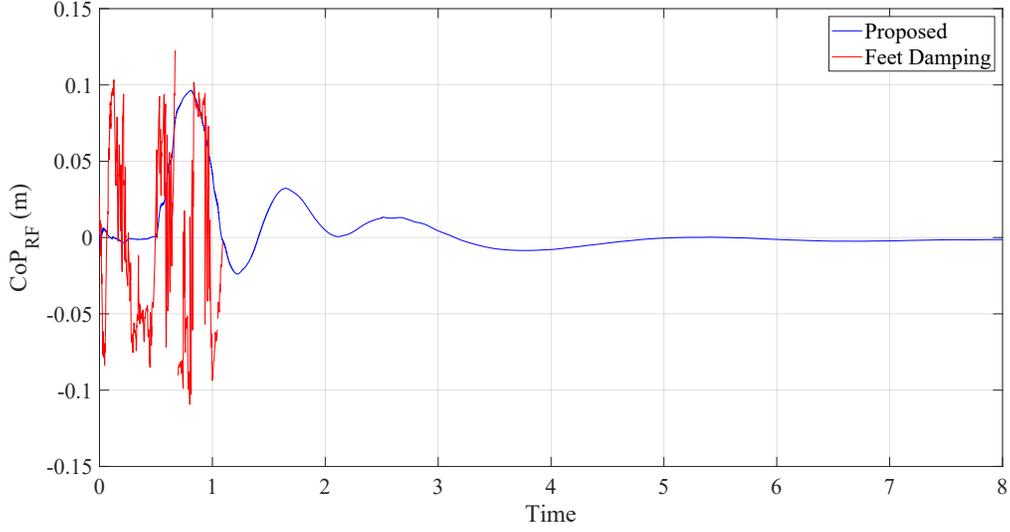


Figure 4.3: Right foot CoP (x component) in scenario (a)

| Contact | RF | LF | RH | LH |
|---------------------|-----------------|------------------|-----------------|--------|
| $K_{f,p}$ (N/m) | 3×10^4 | 25×10^3 | 2×10^4 | 10^4 |
| $K_{f,d}$ (N.s/m) | 4000 | 3000 | 4000 | 4000 |
| $K_{t,p}$ (N/rad) | 400 | 350 | 180 | 90 |
| $K_{t,d}$ (N.s/rad) | 60 | 60 | 30 | 30 |

Table 4.2: Stiffness and damping for each contact in Scenarios (b) and (c)

- $w_{fi} = \begin{bmatrix} 0 & 0 & 0.5 \end{bmatrix}$
- $w_{ti} = \begin{bmatrix} 0.5 & 0.5 & 0 \end{bmatrix}$,

and the control matrices of the LQR such that

- $Q = \text{diag}(Q_{com} \quad Q_c \quad Q_c \quad Q_c \quad Q_c)$,
- $Q_{com} = (10^4 \mathbb{I}_3 \quad 100 \mathbb{I}_3 \quad 3 \times 10^4 \mathbb{I}_3 \quad 100 \mathbb{I}_3)$,
- $Q_c = (3 \times 10^4 \mathbb{I}_3 \quad 100 \mathbb{I}_3 \quad 3 \times 10^4 \mathbb{I}_3 \quad 100 \mathbb{I}_3)$,
- $P = \text{diag}(\mathbb{I}_6)$.

By setting the angular positions and velocities gains at 100, the controller was able to minimize the CoM error while regaining the default orientation of the base. I noticed that giving too much weight to the orientations results in a slightly worse behavior, as the robot does not seem to minimize the contacts position as efficiently.

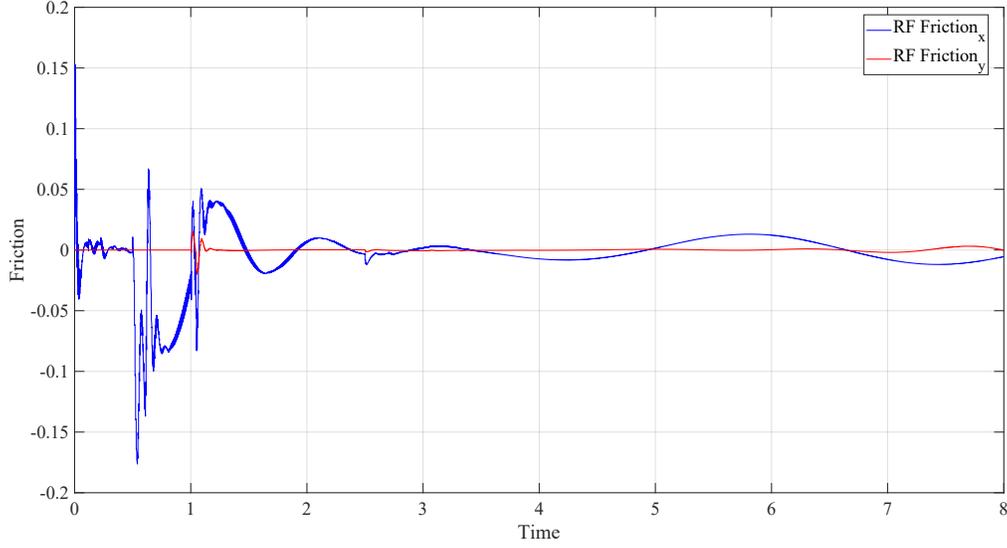


Figure 4.4: RF friction ratio along the x and y axis in scenario (a)

As for scenario (c), I chose:

- $w_{fi} = \begin{bmatrix} 0 & 0 & 0.5 \end{bmatrix}$ (for $i = 1, 2, 4$),
- $w_{f3} = \begin{bmatrix} 0.5 & 0 & 0 \end{bmatrix}$,
- $w_{ti} = \begin{bmatrix} 0.5 & 0.5 & 0 \end{bmatrix}$ (for $i = 1, 2, 4$),
- $w_{t3} = \begin{bmatrix} 0 & 0.5 & 0.5 \end{bmatrix}$,
- $Q_{com} = (10^5 \mathbb{I}_3 \quad \mathbb{I}_3 \quad 10^4 \mathbb{I}_3 \quad \mathbb{I}_3)$
- $Q_c = (10^4 \mathbb{I}_3 \quad \mathbb{I}_3 \quad 1000 \mathbb{I}_3 \quad \mathbb{I}_3)$,
- $P = diag(\mathbb{I}_6)$.

Figures 4.6, 4.7 and 4.8 show that the proposed controller is able to maintain the balance of the robot, minimizing the CoM error and maintaining the CoP in its support zone (each hand has a dimension of $x : 0.105m \times y : 0.135m$). No slippage of contacts occurred, and the maximum value of the friction ratio at each contact in each scenario is given in Table 4.3

4.4 Conclusion

In this chapter, the controller was tested on a biped in a two contacts and multi-contacts with the environment in different scenarios. The contacts being compliant, the classical admittance

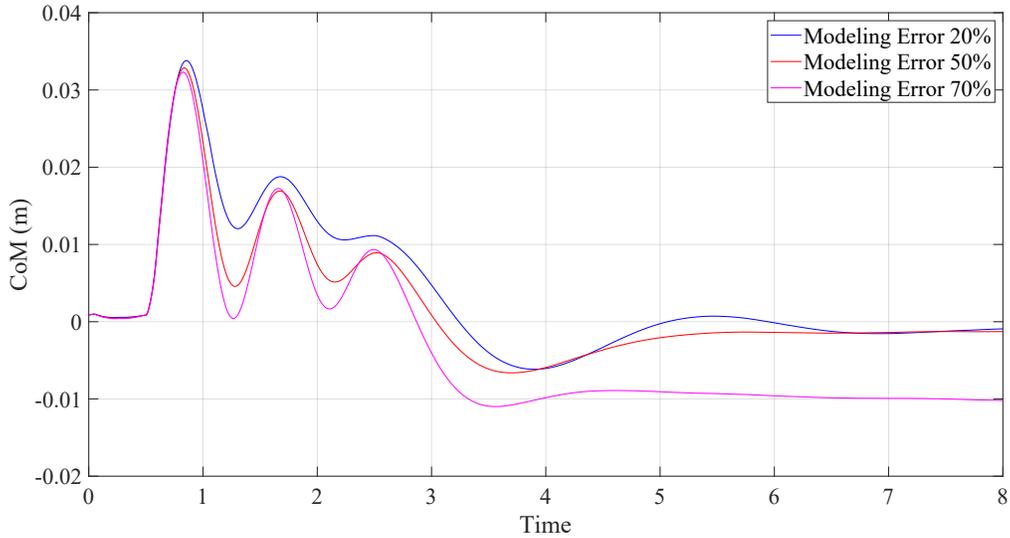


Figure 4.5: CoM position error (x component) for different modeling errors

| Contact | RF | LF | RH | LH |
|------------|-------|-------|-------|-------|
| Scenario b | 0.31 | 0.354 | 0.372 | 0.355 |
| Scenario c | 0.278 | 0.193 | 0.361 | 0.359 |

Table 4.3: Max absolute value of the friction ratio at each contact

control relying on the DCM dynamics and a calculation of the contact wrenches for CoM and admittance control fails to maintain the balance of the robot in face of perturbations, as the robot struggles to calculate the required wrench for its balance in the non-rigid environment. On the other hand, the proposed controller succeeds in balancing the robot and minimizing the error of the forces at each contact thanks to the force feedback and kinematics trade-off.

That being said, this validation has many assumptions that aren't met when applying the control on a real robot. First, the robot's model is simple and doesn't correspond to a model of a real robot. Second, no state estimation of the robot's under-actuated base is being considered, instead the values obtained from the feedback are assumed to be perfectly measured and without any delay. Third, an integration from Matlab to a real robot is complicated, hence the need of the validation of the controller using a software that can be easily applied to a real robot.

For these reasons, the next chapter is going to present a validation of the control on a model of a real position control robot, before moving on to the validations on a real robot. The next chapter is going to cover the framework used, the simulation environment and tests, in addition to the experimental results on a real robot before concluding.

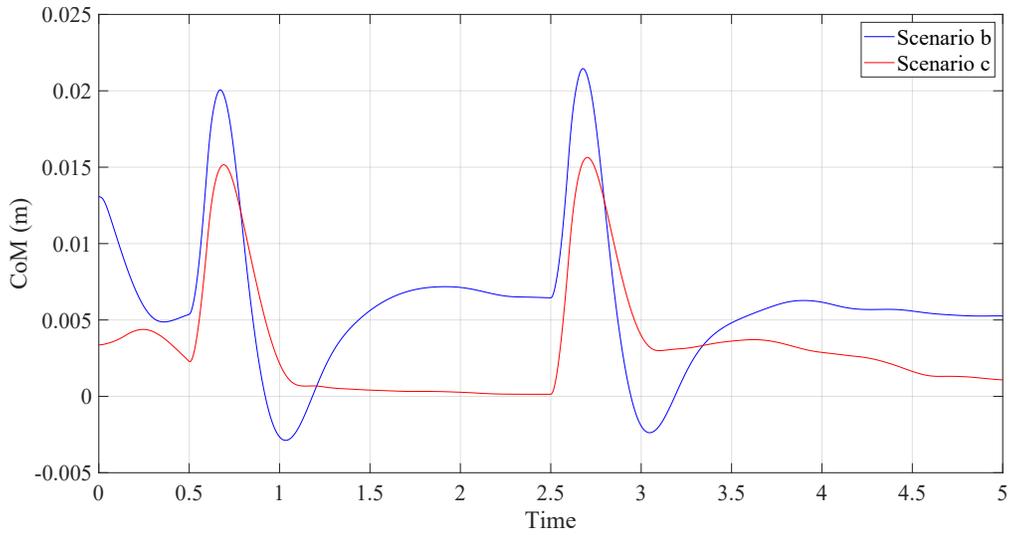


Figure 4.6: CoM position error (x component) in scenarios (b) and (c)

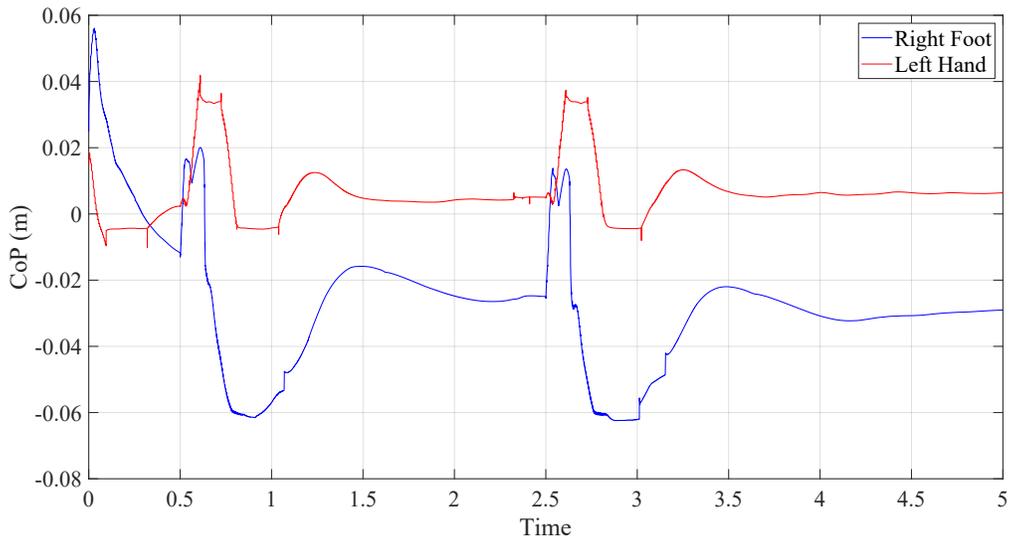


Figure 4.7: Right foot and left hand CoP (x component) in scenario (b)

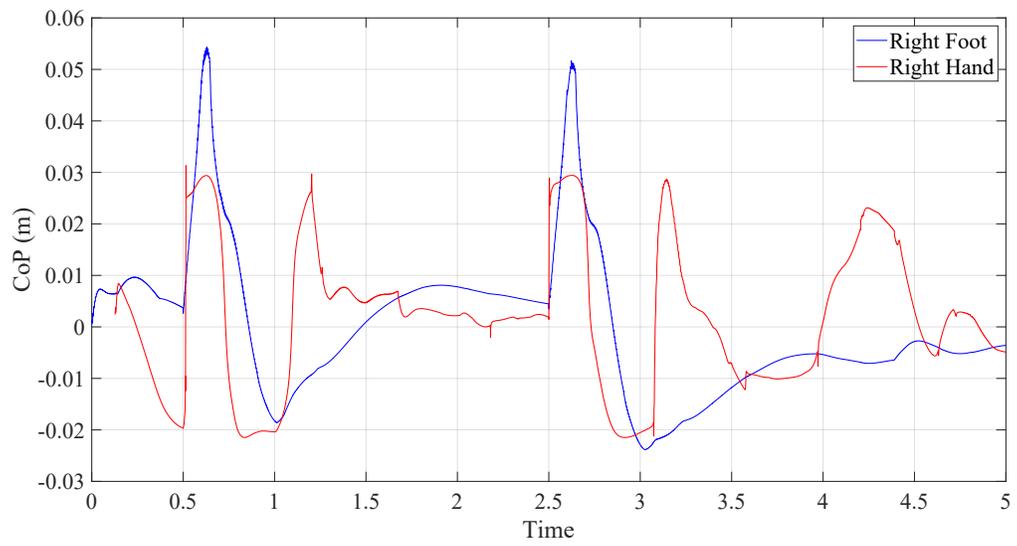


Figure 4.8: Right foot and right hand CoP (x component) in scenario (c)

Chapter 5

Validation on Position Controlled Humanoid Robots

5.1 Introduction

In the previous chapter, I presented the implementation of the controller on a torque controlled 26 d.o.f biped in Matlab and Simulink. I presented in the conclusion of the previous chapter several points of improvement to be made for a better validation on simulations and implementation on the real humanoid robot. This chapter is going to take those points into consideration, while testing the controller on a position controlled robot.

The controller is going to be implemented in C++, using the `mc_rtc` framework, first on a model of a position controlled robot of type HRP4. The simulation is incorporated in choreonoid, where dynamics behavior can be simulated. State estimation is used to estimate the robot's floating base's position and orientation, and encoders provide the positions and velocities of the articulation joints.

Tests will be done in a default configuration with 2 legs contacts, first with an almost rigid interaction and second with compliant surfaces, then a validation on a multi-compliant contacts will be demonstrated.

Finally, the controller is going to be tested on a real humanoid robot of type HRP2-KAI. The robot will be in the default half-sitting position with 2 contacts, first on a rigid ground, where the spring bush in the robot's feet are going to apply the compliant contacts, and then on a mattress for an increased compliant effect.

This chapter is divided as follows. Section [5.2](#) describes the framework used and the simulation environment, while section [5.3](#) adds some details about calculations and imple-

mentation of the controller in `mc_rtc`. Section 5.4 presents the results, finally section 5.5 concludes the chapter.

5.2 `mc_rtc` and Choreonoid

The controller is now written in the C++ programming language using the control framework `mc_rtc`. `mc_rtc` is a QP based control architecture for real-time robotic applications that allows the build of complex behaviors using hierarchical state machines. It provides a friendly way to implement these behaviours, such as the use of finite state machines (FSM) to implement robotic scenarios using simple transitions, along with graphical tools providing user input in addition to the monitoring and logging of data. This framework has been used to successfully integrate robots in a wide variety of applications, such as aircraft automation as shown in [Kheddar et al., 2019], or in physically interacting with humans [Bolotnikova et al., 2020a]. It is build at the top of libraries such as `SpaceVecAlg`, which provides a concise vector notation to write positions, velocities and accelerations of a rigid body with respect to a non-inertial frame, `RBDyn` (Rigid Body Dynamics) which provides a set of classes and functions to model the dynamics of rigid body systems, and `Tasks`, a library for real time control of robots and kinematic trees using constrained optimization. `SpaceVecAlg` and `RBDyn` are based on Roy Featherstone Rigid Body Dynamics Algorithms book [Featherstone, 2008].

Once the controller is written, it's necessary to have a simulation environment to test the behavior of the robot when such controller is implemented. `mc_rtc` is compatible with a number of interfaces and simulators, such as the Robot Operating System (ROS), `MuJoCo` [Singh et al., 2022], `NAOqi` [Bolotnikova et al., 2020b] and `Choreonoid` [Nakaoka, 2012].

`Choreonoid` is an open source integrated robotics environment with a graphical user interface (GUI), that was developed to handle humanoid robots' choreography [Nakaoka, 2012], teleoperation [Nakaoka et al., 2014], and various other robotics operations. I opt to use `choreonoid` for its dynamic physics engine, to test the controller in a setting close to a real testing environment, which is very useful to have before testing on actual real robots.

Figure 5.1 shows how a controller is run using `mc_rtc` and `Choreonoid`. The `Choreonoid` environment shows the behavior of a controller on the robot. The `RViz` panel is used as an input/output GUI interface, where the user can add or remove tasks from the solver, switch between different controllers, with many other options. Additionally, a virtual robot built from the output of the QP can be observed. Finally, the logs information and debug messages

are displayed in the terminal window. This is the setting that I use to implement and test my controller.

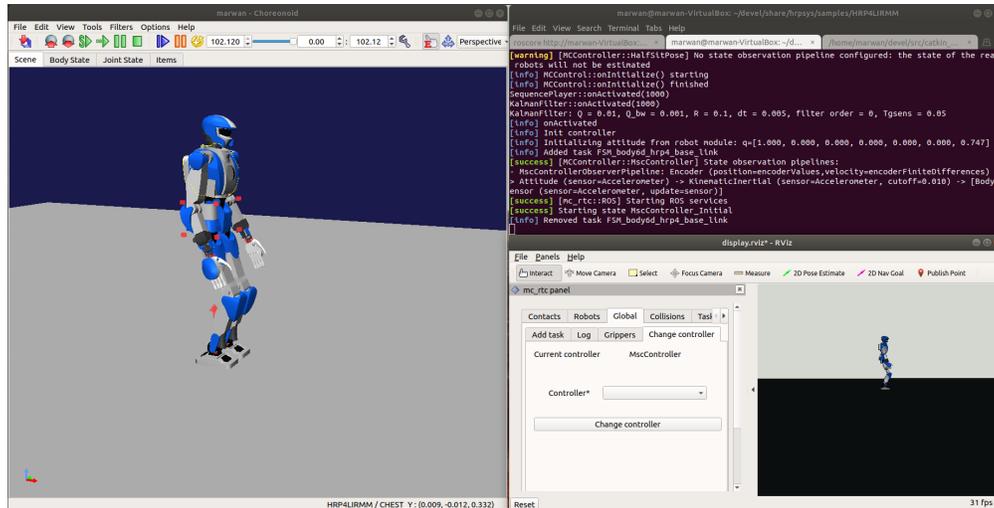


Figure 5.1: mc-rtc with Choreonoid: The left window is for Choreonoid, the top right is for the terminal, the bottom right is for the control panel RViz

5.3 Details Concerning the Implementation of the Controller

In this section, I will go over some details regarding the implementation of the controller using mc_rtc. This includes estimation, frame transformations, and the calculation of the LQR gain.

5.3.1 Estimation

Since the robot to be controlled is position controlled, the control architecture and feedback are as described in [3.4](#).

It is important to note that the feedback is taken after estimation of the robot's state. An estimation of the robot's base position and orientation is used. This estimation is called the [anchor point's solution](#), and is based on measurements obtained from another estimation of the orientation by an IMU sensor. Encoders provide the joint's positions and velocities, and forward kinematics are used to determine the position of the robot's limbs in a certain reference frame.

5.3.2 Frame Transformations and Additional Tasks

As the `mc_rtc` framework can only formulate tasks written in the world frame, additional frame transformations are required for transforming the accelerations of the contact limbs generated by the proposed controller from the CoM/base frames to the world frame. Similarly, the position, orientations and velocities of the contact limbs need to be transformed from the world frame to the CoM/Base frame of the robot before being used by the controller.

- Transforming the position and orientation of a contact from the world frame to the CoM/Base frames is done using

$$p_c^{com} = R_b^\top (p_c - com), \quad (5.1)$$

$$R_c^b = R_b^\top R_c. \quad (5.2)$$

- Transforming the linear and angular velocities of a contact from the world frame to the CoM/Base frames is done using

$$\dot{p}_c^{com} = R_b^\top (\dot{p}_c - \dot{com} - S(\omega_b)(p_c - com)), \quad (5.3)$$

$$\omega_c^b = R_b^\top (\omega_c - \omega_b). \quad (5.4)$$

- Transforming the linear and angular accelerations of a contact from the CoM/Base frames to the world frame is done using

$$\begin{aligned} \ddot{p}_c = & R_b \ddot{p}_c^{com} - S^2(\omega_b)(p_c - com) + S(\dot{\omega}_b)(p_c - com) \\ & + 2S(\omega_b)(\dot{p}_c - \dot{com}) + \ddot{com}, \end{aligned} \quad (5.5)$$

$$\dot{\omega}_c = R_b \dot{\omega}_c^b + S(\omega_b)(\omega_c - \omega_b) + \dot{\omega}_b. \quad (5.6)$$

Looking at equations (5.5) and (5.6), transforming the accelerations to the world frame relies on the acceleration of the CoM and the angular acceleration of the floating base. Additionally in this case, the objectives handled by the QP are the posture task and acceleration tasks for the contact limbs in the world frame, which can cause the robot's base linear and angular accelerations at the output of the QP to diverge. Which is why there's a need to send additional CoM task and an angular acceleration task for the robot's base to the QP,

and use these accelerations in the frame transformations in (5.5) and (5.6). This is done using

$$\ddot{com} = -k_p (com - com^*) - k_d (\dot{com} - \dot{com}^*) + \ddot{com}^*, \quad (5.7)$$

$$\dot{\omega}_b = -k_p \Theta (R_b R_b^{*\top}) - k_d (\omega_b - \omega_b^*) + \dot{\omega}_b^*, \quad (5.8)$$

where k_p and k_d represent proportional and derivative terms to be tuned to obtain the best behavior. Note that these 2 tasks are given a low weight compared to the accelerations of the contact limbs, as they do not serve in the balancing control.

It is important to note that if the accelerations for the contacts were written in the robot's base frame, then the motion of the base would be constrained by the motion of the contact limbs, and the diagram shown in figure 3.7 would be valid to use. I chose the above described approach due to its simple integration within mc_rtc.

The diagram of the control architecture in this case requires a few additions, and it is as illustrated and detailed in figure 5.2.

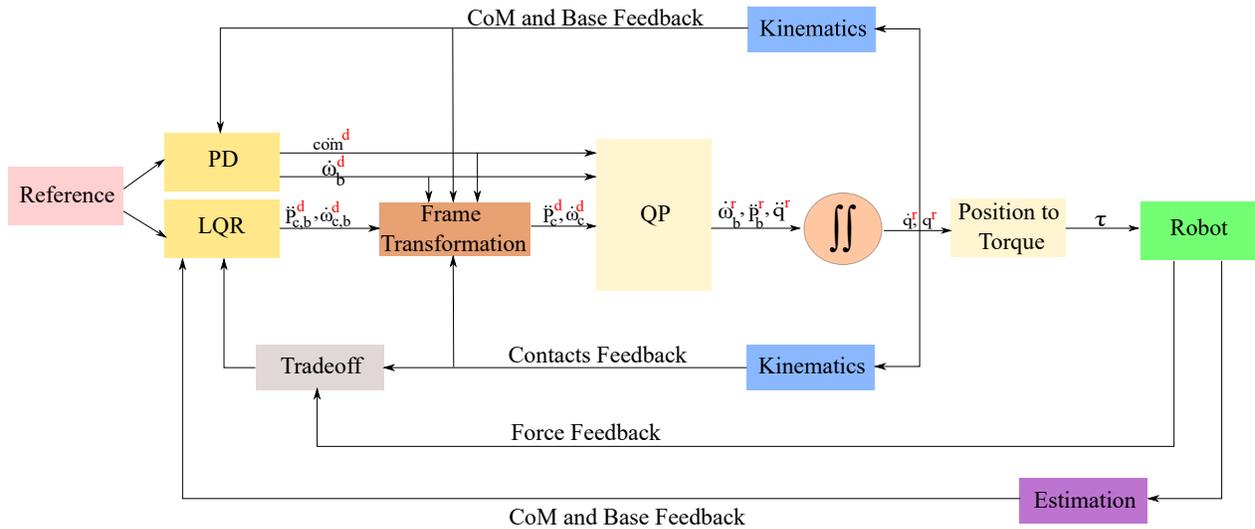


Figure 5.2: Control architecture when the controller is implemented in mc_rtc with a position controlled robot.

5.3.3 LQR Gain Calculation

I implemented an LQR gain calculation for a discrete-time linear system. Which is why I use the discrete form of the linearized matrices:

$$A_{y,d} = \mathbb{I} + A_y dt, \quad (5.9)$$

$$B_{y,d} = B_y dt. \quad (5.10)$$

The notation $(.)^d$ is used to refer to the discrete form, and dt is the discretization step.

U is the unique positive definite solution to the following discrete time algebraic Riccati equation (5.11):

$$U = A_{y,d}^\top U A_{y,d} - A_{y,d}^\top U B_{y,d} (P + B_{y,d}^\top U B_{y,d})^{-1} B_{y,d}^\top U A_{y,d} + Q_y. \quad (5.11)$$

For the first iteration, I set $U = Q_y$, and then U is calculated at every iteration using its old value from the previous iteration, until the difference between the 2 results in a matrix whose maximum number is inferior to a constant ϵ I define, in other words when

$$\max |U - U_{old}| \leq \epsilon. \quad (5.12)$$

Finally, the gain K_y is calculated using

$$K_y = (P + B_{y,d}^\top U B_{y,d})^{-1} B_{y,d}^\top U A_{y,d}. \quad (5.13)$$

5.4 Results

5.4.1 Simulations on HRP4

I used the 34 degrees of freedom HRP4 robot in the choreonoid environment to test my controller. To simulate compliant contacts, I used an already programmed deformable surface on each contact. These surfaces have their own stiffness and damping parameters, and move or rotate similar to a soft surface according to the proposed viscous-elastic model, meaning that I can simply set their $K_{f,p}$, $K_{f,d}$, $K_{t,p}$ and $K_{t,d}$ values.

I present 3 different scenarios for testing: the first two scenarios have the robot in 2 contacts mode, the difference is that the first scenario does not use the deformable surfaces to simulate compliant contacts like the second scenario, instead it tests the controller on

rigid ground. The reason why I chose to showcase the performance with a relatively rigid interaction is because I want to show that the controller can still be used in such situations, where the user is not interested in deploying robots in terrains where compliance is visibly present. Finally, the third scenario is a multi-contact scenario on deformable surfaces.

In each case, I will use the push feature of the robot in the choreonoid environment, the controller is expected to shift the robot back to its equilibrium while maintaining all the contacts and minimizing the error of the state and forces.

In all three scenarios, I set the k_p and k_d terms in equations [5.7](#) and [5.8](#) such that

- $k_p = 100\mathbb{I}_3$,
- $k_d = 30\mathbb{I}_3$,

which was enough to prevent divergence of the robot's base position at the output of the QP.

The tuning of the controller is done similarly to the previous chapter, which is described in the appendix [C](#).

Two Contacts Configuration

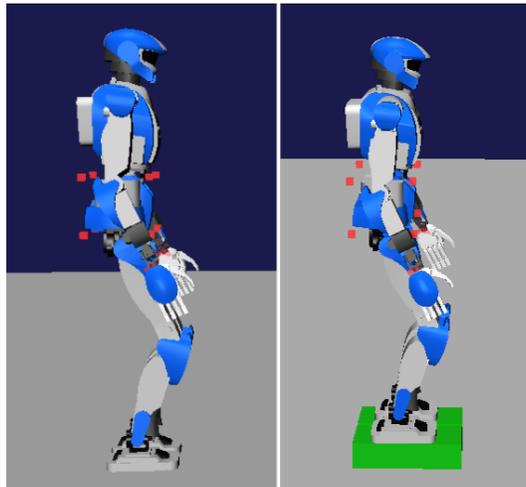


Figure 5.3: Scenarios 1 (left) with rigid contacts and 2 (right) with deformable contacts

Figure [5.3](#) shows scenarios 1 and 2. The simulation starts from the position illustrated in this figure, the state and force references are taken from the robot once all oscillations due to the initial energy stops.

| Contact | RF | LF |
|---------------------|--------|--------|
| $K_{f,p}$ (N/m) | 10^5 | 10^5 |
| $K_{f,d}$ (N.s/m) | 400 | 400 |
| $K_{t,p}$ (N/rad) | 5000 | 5000 |
| $K_{t,d}$ (N.s/rad) | 30 | 30 |

Table 5.1: Stiffness and damping for each foot in a rigid 2-contact case

Starting with scenario 1, this is a simple case to showcase the reliability of the controller even when tested with relatively rigid parameters. Indeed, using the viscous-elastic formulation for the forces when the contacts are relatively rigid might bring difficulties in balance amidst modeling errors and noise, as seen in [Flayols et al., 2020]. For the controller to be compatible with a relatively rigid environment, the linear and angular stiffness of the viscous-elastic model given in the controller’s configuration should be set to high values so that the contacts barely move through the environment, hence why I set them as shown in table 5.1. Note that increasing the stiffness beyond the values in table 5.1 can slow down or complicate numerical computation of the LQR gain.

After carefully tuning the LQR control matrices and the force-Kinematics trade-off matrix W , I got the best performance by choosing

- $w_{fi} = \begin{bmatrix} 0.95 & 0.95 & 0.95 \end{bmatrix}$,
- $w_{ti} = \begin{bmatrix} 0.95 & 0.95 & 0.95 \end{bmatrix}$,
- $Q = \text{diag}(Q_{com} \quad Q_c \quad Q_c)$,
- $Q_{com} = (3 \times 10^{10}\mathbb{I}_3 \quad 10^9\mathbb{I}_3 \quad 3 \times 10^8\mathbb{I}_3 \quad 3 \times 10^6\mathbb{I}_3)$,
- $Q_c = (3 \times 10^{11}\mathbb{I}_3 \quad 3 \times 10^7\mathbb{I}_3 \quad 10^9\mathbb{I}_3 \quad 3 \times 10^8\mathbb{I}_3)$,
- $P = \mathbb{I}_{12}$.

Note that it’s necessary to rely on the entire 6-dimension force feedback vector for each contact for better force tracking. Giving full weight to the force feedback, i.e. setting w_{fi} and w_{ti} as identity worsens the contact’s position and orientations tracking, and the robot always loses its balance.

With these gains and as I applied an external push to the robot as perturbation, the controller is able to maintain the balance of the robot while minimizing the error of the

| | Real | Controller |
|---------------------|-----------------|-------------------|
| $K_{f,p}$ (N/m) | 2×10^4 | 2.5×10^4 |
| $K_{f,d}$ (N.s/m) | 500 | 300 |
| $K_{t,p}$ (N/rad) | 1500 | 500 |
| $K_{t,d}$ (N.s/rad) | 7 | 1 |

Table 5.2: Real and controller’s stiffness/damping for each foot in Scenario 2

forces. This is can be seen by looking at the CoM error curve and the right foot CoP curve in figures 5.4 and 5.4. As seen from the CoP curve, the perturbation pushed the CoP to the edge of its support zone along the x axis, yet the CoM error didn’t go over a few millimeters and the convergence to zero was very quick.

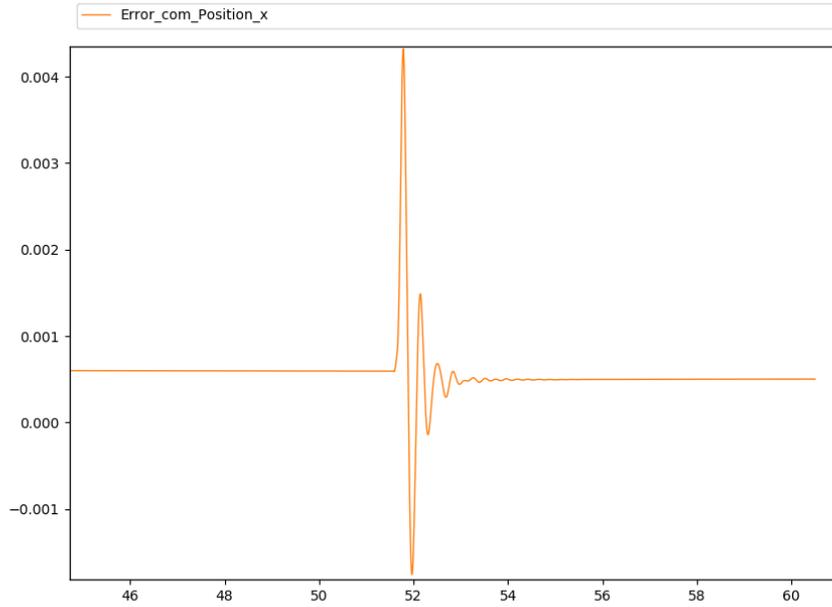


Figure 5.4: CoM error in Scenario 1

Now to scenario 2, where each foot makes contact with a deformable surface. Similarly as I did in the Matlab simulations, I set the stiffness and damping for the deformable surfaces as the real environment compliance values, and I choose different values for these same parameters in my controller as modeling errors to test the robustness of the controller, as shown in table 5.2.

After another tuning of the LQR control matrices, I got the best performance by choosing:

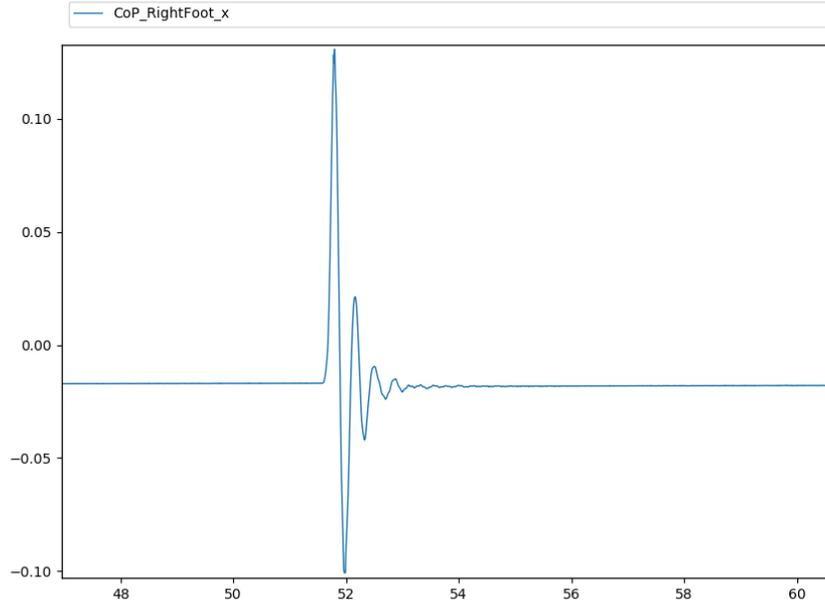


Figure 5.5: CoP of the RF in Scenario 1

- $w_{fi} = \begin{bmatrix} 0.8 & 0.8 & 0.8 \end{bmatrix}$,
- $w_{ti} = \begin{bmatrix} 0.8 & 0.8 & 0.8 \end{bmatrix}$,
- $Q = \text{diag}(Q_{com} \quad Q_c \quad Q_c)$,
- $Q_{com} = (10^6 \mathbb{I}_3 \quad 3 \times 10^4 \mathbb{I}_3 \quad 300 \mathbb{I}_3 \quad 3000 \mathbb{I}_3)$,
- $Q_c = (3 \times 10^6 \mathbb{I}_3 \quad 10^5 \mathbb{I}_3 \quad 10^5 \mathbb{I}_3 \quad 10^4 \mathbb{I}_3)$,
- $P = \mathbb{I}_{12}$.

In this case I found the tuning to be a bit easier to obtain a very good performance concerning balance and minimizing the state space and force errors in spite of external pushes. Figure 5.7 shows the CoP curve of the right foot following a push of the robot that moved this CoP to the edge of its support zone, and figure 5.6 shows the CoM error convergence during the experiment.

Figures 5.8 and 5.9 shows the CoM error and the CoP of the right foot for both x and y components in another test for the same scenario with the same gains. Here the robot was pushed three times in succession, twice along the x axis in different directions, followed by a lateral push. Even when the CoP was pushed repeatedly and to the limit of its support

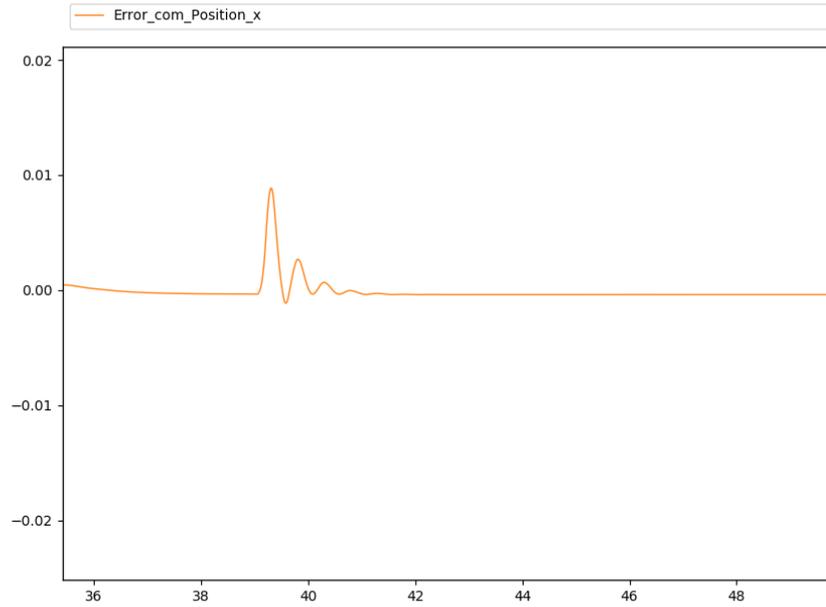


Figure 5.6: CoM error in Scenario 2

region as seen from Figure 5.9, the maximum CoM error does not exceed 1.2 centimeters and the error converges to zero without much oscillations. Figure 5.10, also shows the friction components of the right foot during this experiment, where the maximum absolute value barely exceeds 0.2, showing the importance of taking the entire 6-dimension force feedback vector in the force tuning, as the contact forces follow their references accurately.

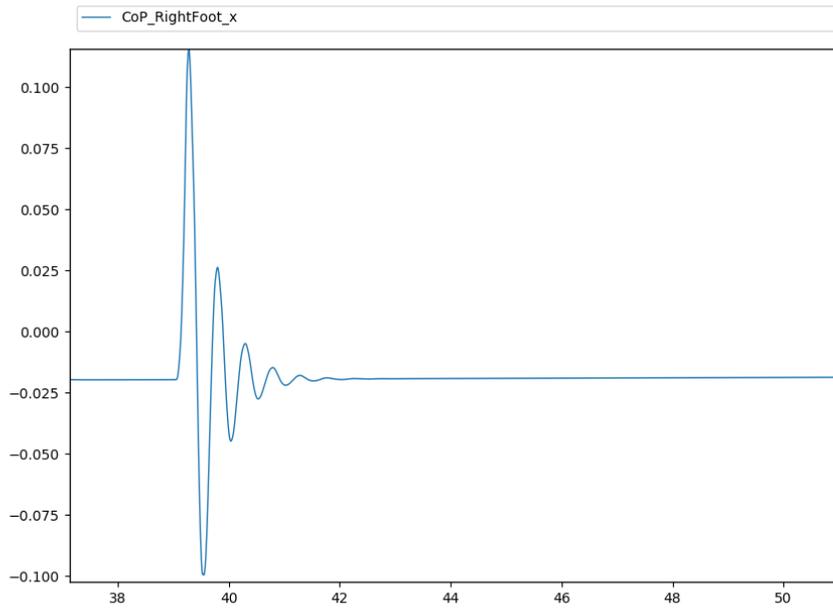


Figure 5.7: CoP of the RF in Scenario 2

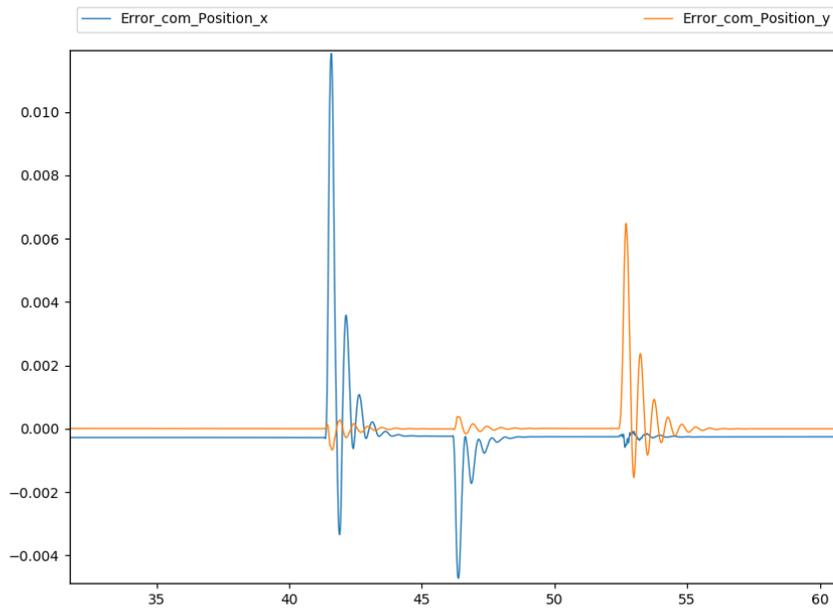


Figure 5.8: CoM error in Scenario 2 under successive pushes

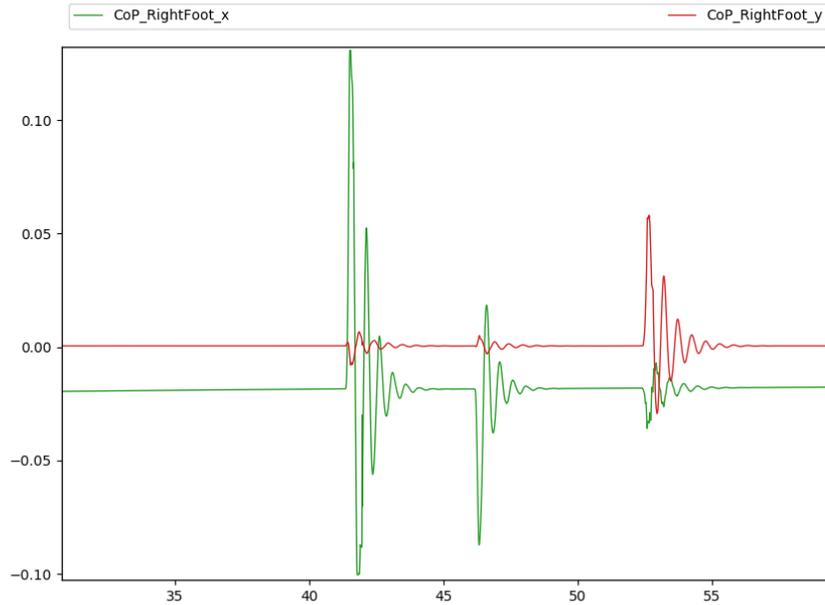


Figure 5.9: CoP of the RF in Scenario 2 under successive pushes

Multi-contacts Configuration

Figure 5.11 shows the three-contacts scenario 3. The robot starts from a half-sitting position, and with the use of the FSM states feature in `mc_rtc`, I use a position task to move the right hand of the robot to the deformable table-like surface, while moving the robot's by 5cm to the front along the x axis. Once the robot reaches the desired reference state and maintains its position in equilibrium as seen in figure 5.11, the controller is manually started, i.e. it calculates the appropriate gain around the desired reference, along with the appropriate accelerations to the right and left feet and to the right hand in order to maintain the robot's balance. The acceleration tasks are manually introduced into the QP solver.

To test the controller, I push the robot backwards. What I am expecting from the controller is to maintain the three contacts, and minimize the state and forces error. As always, the deformable surfaces have stiffness and damping values that differ from the controller's to test the robustness against modeling errors, which is what I show in table 5.3

After another tuning of the gain matrices, I got the best performance by choosing:

- $w_{fi} = \begin{bmatrix} 0.95 & 0.95 & 0.95 \end{bmatrix}$,
- $w_{ti} = \begin{bmatrix} 0.95 & 0.95 & 0.95 \end{bmatrix}$,

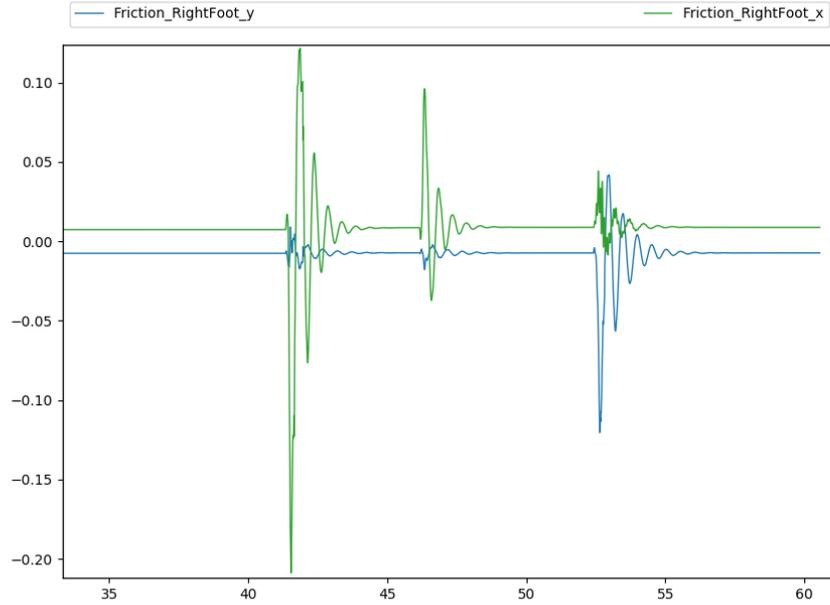


Figure 5.10: Friction (in x and y directions) of the RF in Scenario 2 under successive pushes

| | Real | Controller |
|---------------------|-----------------|-------------------|
| $K_{f,p}$ (N/m) | 2×10^4 | 1.5×10^4 |
| $K_{f,d}$ (N.s/m) | 500 | 400 |
| $K_{t,p}$ (N/rad) | 1500 | 1000 |
| $K_{t,d}$ (N.s/rad) | 7 | 5 |

Table 5.3: Real and controller's stiffness/damping for each of the 3 contacts in Scenario 3

- $Q = \text{diag}(Q_{com} \quad Q_F \quad Q_F \quad Q_H)$,
- $Q_{com} = (10^{11}\mathbb{I}_3 \quad 3 \times 10^{10}\mathbb{I}_3 \quad 3 \times 10^{10}\mathbb{I}_3 \quad 10^{10}\mathbb{I}_3)$,
- $Q_F = (10^{11}\mathbb{I}_3 \quad 10^{12}\mathbb{I}_3 \quad 2 \times 10^{11}\mathbb{I}_3 \quad 10^{12}\mathbb{I}_3)$,
- $Q_H = (3 \times 10^{12}\mathbb{I}_3 \quad 3 \times 10^{12}\mathbb{I}_3 \quad 10^{13}\mathbb{I}_3 \quad 3 \times 10^{12}\mathbb{I}_3)$,
- $P = \text{diag}(\mathbb{I}_{18})$.

Figure 5.13 shows the CoP curve at the right foot. Looking at this curve, the CoP was pushed to the limit of the foot's support region, yet the controller is able to quickly minimize the amplitude of the error, however, small oscillations persist for a few seconds. This is caused by a small rebound of the robot when it was exerting an additional force via its right

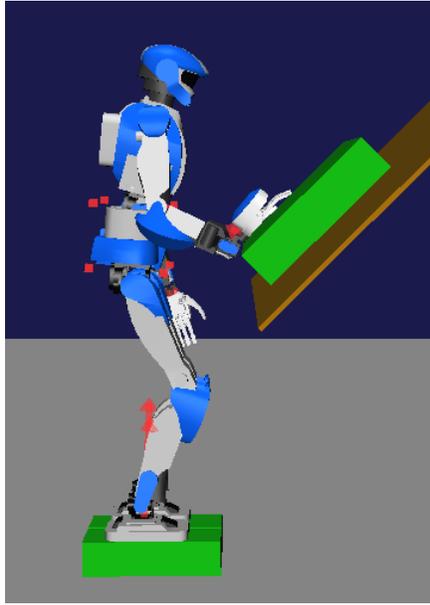


Figure 5.11: Scenario 3 with 3 deformable contacts

hand in order to maintain the hand's position as seen in figure 5.14. Following the push of the robot, and as the weight given in the matrix Q to the right hand's position is more important than other contacts', the controller forced the robot to apply additional force to the right hand in order to keep the contact in place, while minimizing the error of this force at the same time. The CoM error curve is also shown in figure 5.12. The error converges to zero despite the presence of modeling errors, after negligible oscillations due to rebound, with a maximum error of less than $2cm$.

5.4.2 Experiments on HRP2-KAI

HRP2-KAI is a position controlled robot having a spring bush between the sole and the ankle of each foot. Due to this design, the interaction with the floor of the lab is compliant, and the robot actually struggles to stand still when slightly pushed. For the experiments, I will test the control when the robot is on rigid floor first, to deal with the compliance at the feet level, then I will put the robot on a soft mattress and test the control to see if it can deal with an increased compliant effect, due to the compliance by design and by surfaces of interaction at the same time. The experimental setup is shown in figure 5.15.

To test the controller, the robot is pushed repeatedly, and the controller is expected to maintain the robot balance by minimizing the error of the CoM and follow the desired contact forces, just like in simulation.

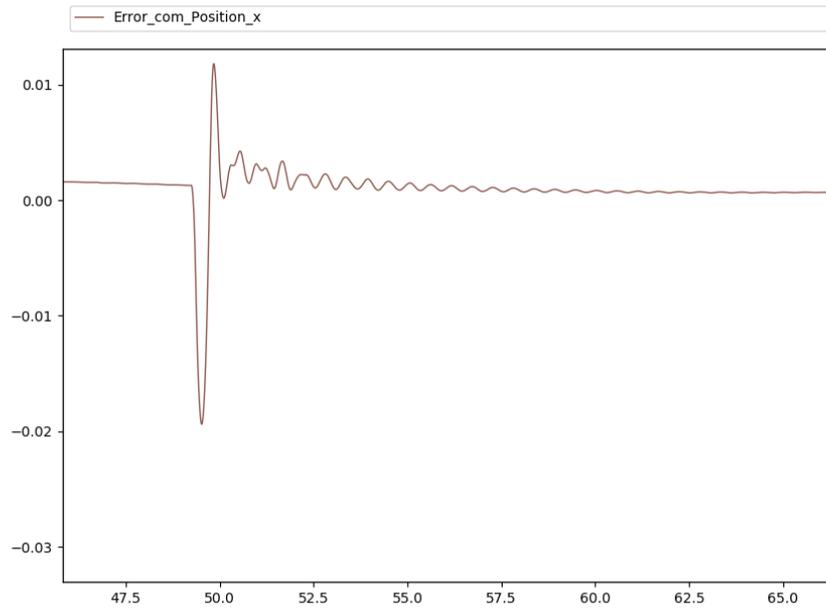


Figure 5.12: CoM error in Scenario 3

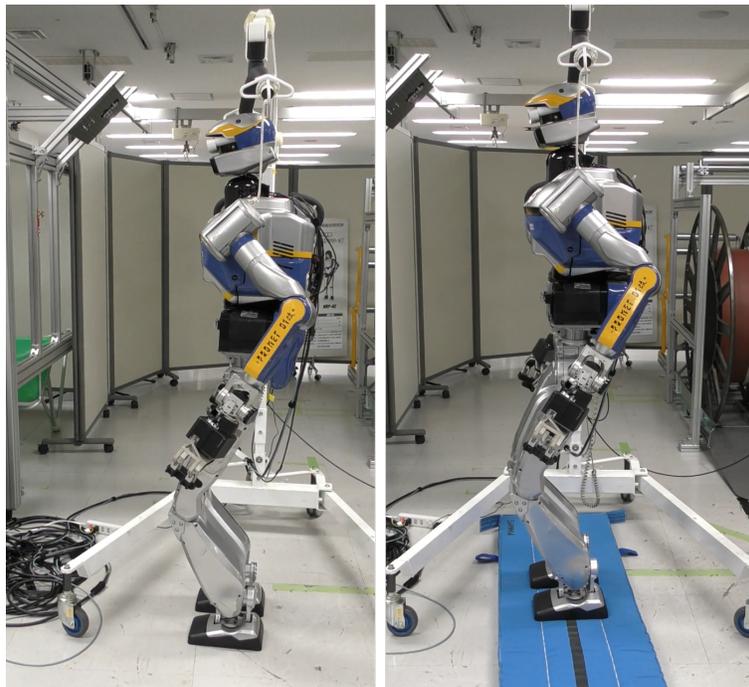


Figure 5.15: HRP2-KAI on rigid floor (left) and on a soft mattress (right)

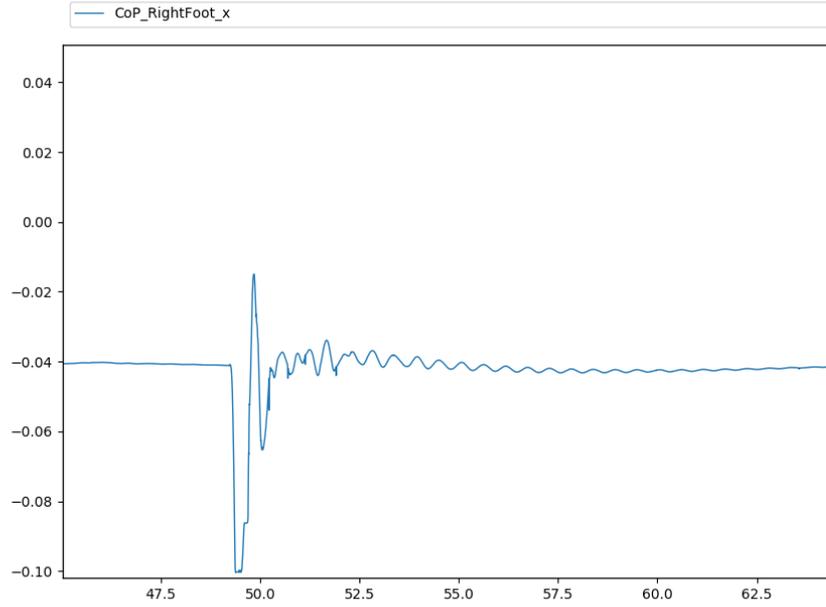


Figure 5.13: CoP of the RF in Scenario 3

HRP2-KAI on Rigid Floor

In this case, the compliant interaction with the environment is due solely to the spring bushes of the feet. Just like in simulations, the controller needs tuning to obtain the best performance. In this case, I tuned the compliance parameters along with the other matrices in order to have an accurate estimation of the stiffness imposed by the spring bushes. As for the damping, due to the difficulty of estimating the damping of the contacts, I set their matrices as identity, in other words, I am considering a model with negligible damping effects. For these reasons, a good performance is obtained by choosing

- $K_{f,p,i} = \text{diag}(4 \times 10^4, 4 \times 10^4, 10^4)$,
- $K_{f,d,i} = \text{diag}(1, 1, 1)$,
- $K_{t,p,i} = \text{diag}(2000, 2000, 2000)$,
- $K_{t,d,i} = \text{diag}(1, 1, 1)$,
- $Q = \text{diag}(Q_{com} \quad Q_c \quad Q_c)$,
- $Q_{com} = \left(\begin{array}{cccc} \text{diag}(10^4, 1000, 1) & \text{diag}(1000, 1000, 1) & \text{diag}(100, 100, 1) & \text{diag}(300, 300, 1) \end{array} \right)$,

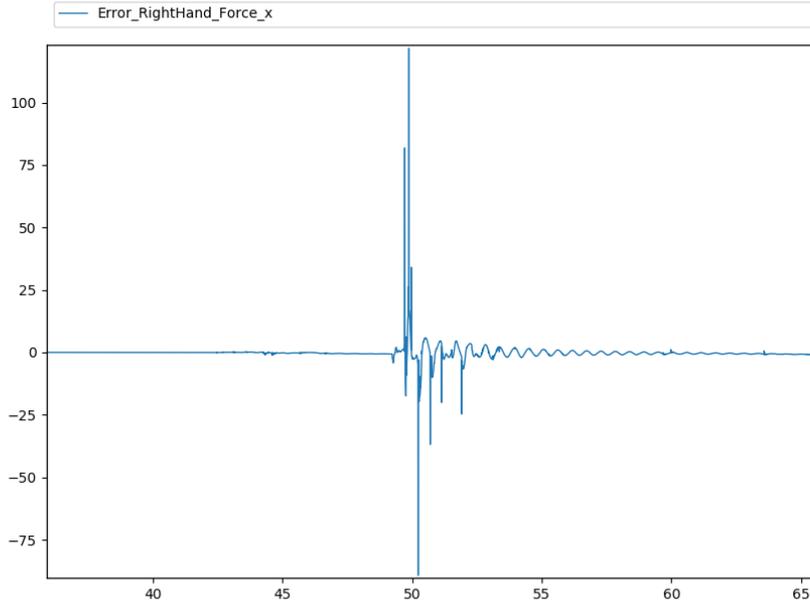


Figure 5.14: Normal force error at the RH in Scenario 3

- $Q_c = (3 \times 10^4 \mathbb{I}_3 \quad 10^4 \mathbb{I}_3 \quad 3000 \mathbb{I}_3 \quad 300 \mathbb{I}_3)$,
- $P = \mathbb{I}_{12}$,
- $W = 0_{36 \times 36}$,

Note that due to the tuning of the stiffness of the contacts, a good performance is achieved without relying on the force feedback information (since I set W to have zero values).

Figures [5.16](#) and [5.17](#) respectively shows the error of the CoM and of the CoP of the right foot. The controller manages to minimize the error of the CoM and follow the desired forces in spite of consecutive pushes, without any use of force feedback information, meaning that the controller can also be used when force sensors are absent from the robot, on the condition that the compliance parameters are well tuned or estimated.

HRP2-KAI on Soft Mattress

Starting from the controller's configuration of the previous test, the robot started falling on its back due to the rotational effect caused by the soft mattress. Additionally, the robot's feet started moving up and down due to the linear compliance along the z-axis. Due to these reasons, I lowered the linear stiffness along the z-axis and the rotational stiffness. I also

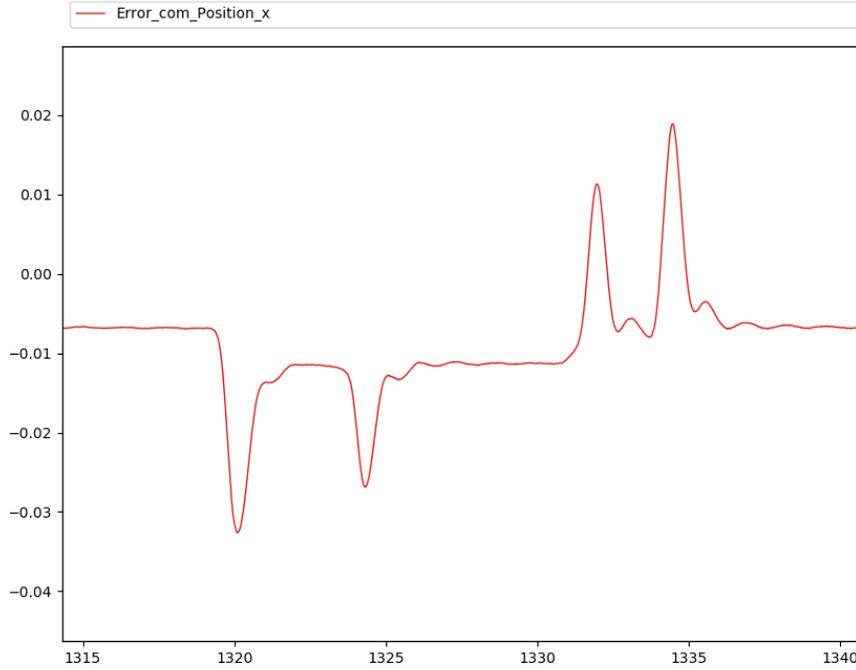


Figure 5.16: CoM error of HRP2-KAI on rigid floor while resisting pushes

added more tracking to the CoM error along the x-axis in Q , in other words, the following changes were made to the previous configuration:

- $K_{f,p,i} = \text{diag}(4 \times 10^4, 4 \times 10^4, 3000)$,
- $K_{t,p,i} = \text{diag}(600, 600, 600)$,
- $Q_{com} = \left(\begin{array}{cccc} \text{diag}(10^6, 1000, 1) & \text{diag}(1000, 1000, 1) & \text{diag}(100, 100, 1) & \text{diag}(300, 300, 1) \end{array} \right)$.

The trajectories of the error of the CoM and of the CoP of the right foot are shown in figures 5.18 and 5.19. The robot can endure the consecutive perturbations even without force feedback (since W is set to zeros again) and maintain its balance. A steady state error of 1cm is visible for the CoM error, due to a non-ideal tuning of the compliance. Additionally, oscillations could have been tuned down by raising the tuning for minimizing the CoM velocity in Q . Nonetheless, the controller did well to maintain the robot's balance with the increased compliance effect. It is important to note that the robot falls on its back even without pushes when the controller is not enabled due to the strong compliance.

The same scenario is repeated with modeling errors, by increasing $K_{t,p,i}$ by 100%: $K_{t,p,i} = \text{diag}(1200, 1200, 1200)$.



Figure 5.17: CoP of the RF of HRP2-KAI on rigid floor while resisting pushes

With these errors, the controller struggles without force feedback, hence the tuning is slightly changed by including force feedback, done by setting $w_{ti} = \begin{bmatrix} 0 & 0.3 & 0 \end{bmatrix}$.

Figures 5.20 and 5.21 show the CoM error and the CoP of the right foot when. The controller manages to maintain the robot’s balance amidst the strong compliance and modeling errors. The steady state of the CoM error is less than $3mm$, thanks to the inclusion of the force feedback.

5.5 Conclusion

In this chapter, the controller’s performance was tested on a model of the position controlled robot HRP4 using the control framework `mc_rtc` and the `choreonoid` simulation environment. It also presented the results on a real HRP2-KAI robot.

The chapter explained the details in the integration of the controller in `mc_rtc`, including the frame transformations and the LQR gain computation.

The controller’s performance is validated as it maintains the balance of the robot while on rigid floor, deformable floor and in a multi-contact situation, while maintaining the contacts and minimizing the error signals of the contact forces in simulation.

The validation was also done on HRP2-KAI, a robot with a built-in compliance between

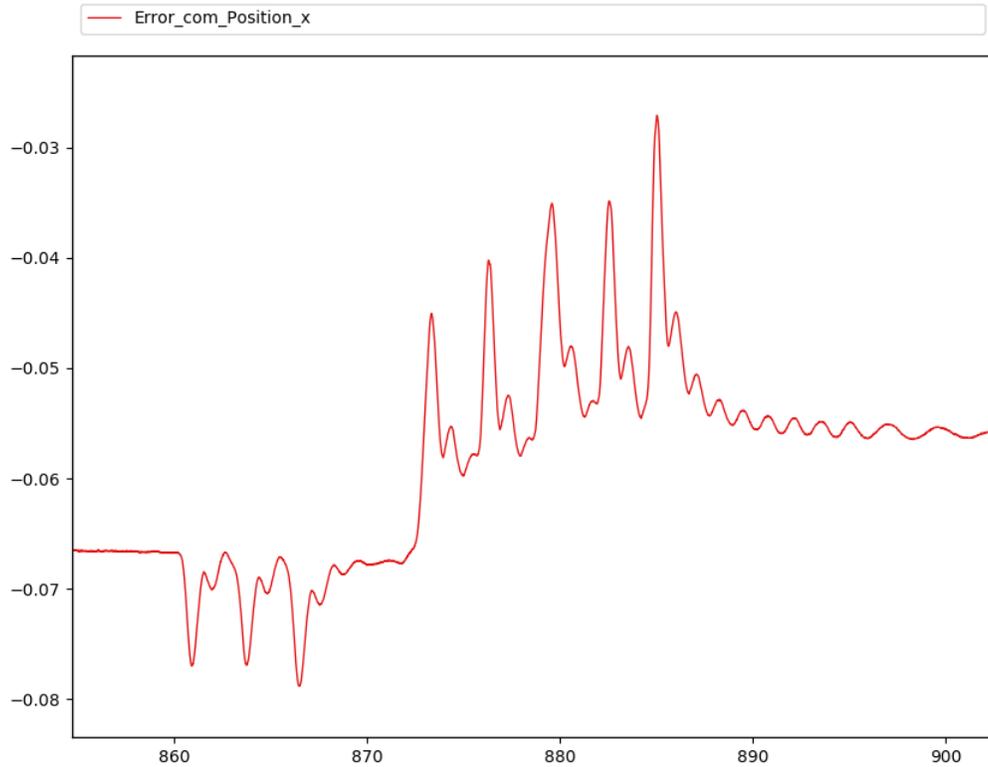


Figure 5.18: CoM error of HRP2-KAI on a mattress while resisting pushes

the sole and the spring of each foot. The controller was able to deal with the built-in compliance of the robot when on rigid ground and on a soft mattress, and maintained the robot’s balance when pushed multiple times, without relying on force feedback, thanks to an approximate estimation of the compliance parameters via tuning. Force feedback was necessary when modeling errors were considered to balance the robot and follow the desired contact forces.

More experiments could be done with different contact configurations, especially on the real robot, or when the controller is tested while the robot is in quasi-static motions. However, this requires slow motions that do not put the stability of the linearized system in risk. Instead, evolving the controller so that it can take references around a time variant trajectory rather than a stationary reference is the better option.

This topic is going to be discussed in the future works after concluding the work in the conclusion.

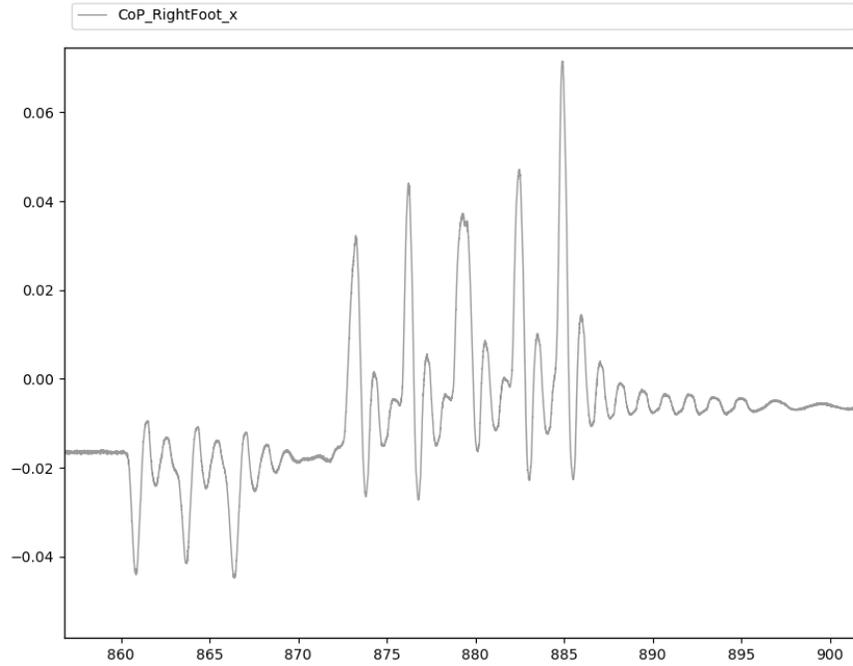


Figure 5.19: CoP of the RF of HRP2-KAI on a mattress while resisting pushes

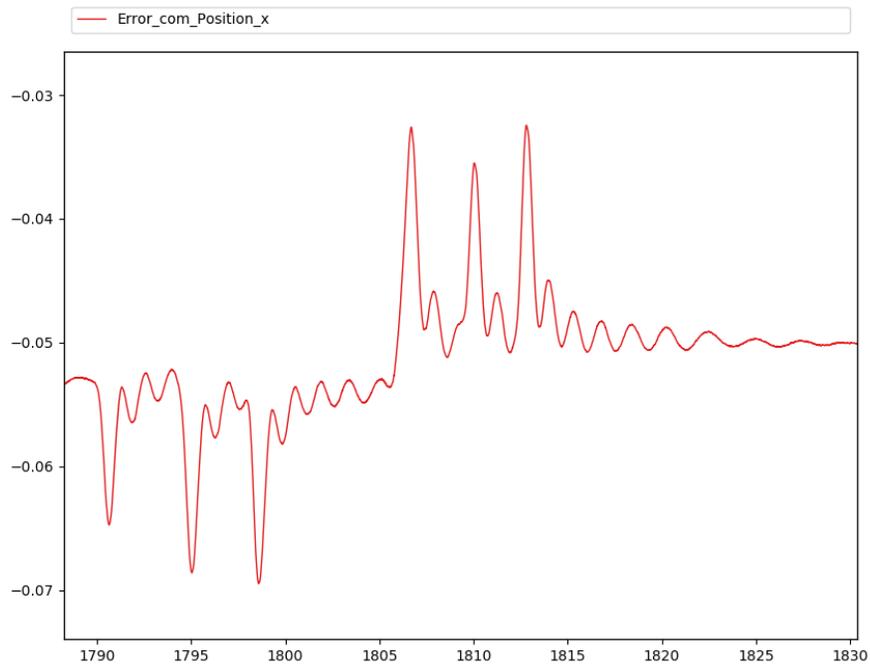


Figure 5.20: CoM error of HRP2-KAI on a mattress with modeling errors while resisting pushes

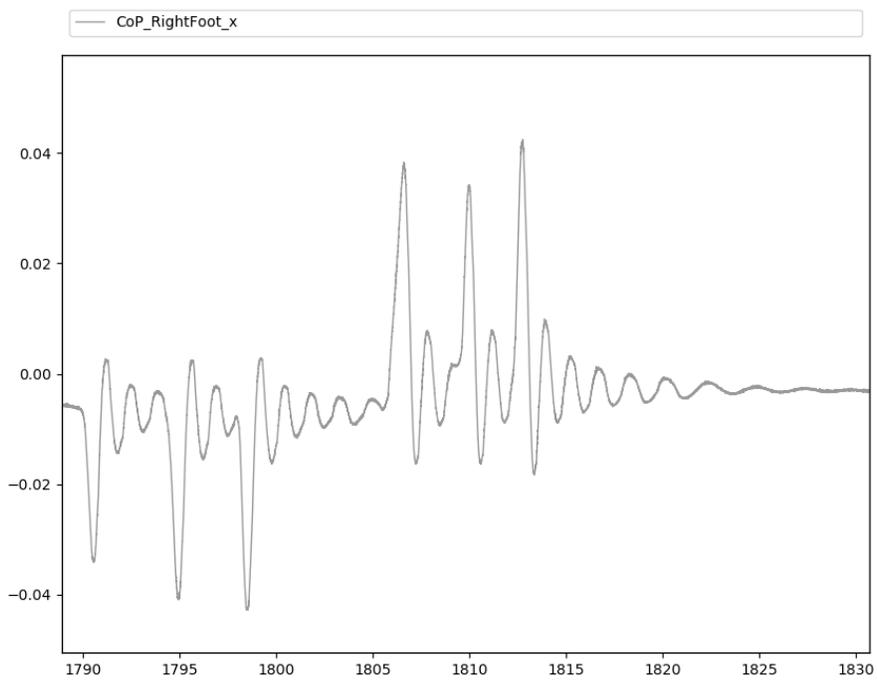


Figure 5.21: CoP of the RF of HRP2-KAI on a mattress with modeling errors while resisting pushes

Conclusion and Future Work

Conclusion

For a robot to accomplish a certain task, whether it's a humanoid robot using tools, or a legged robot traversing a terrain via locomotion, balance should be guaranteed for the robot to succeed in his task. However, due to inaccuracies of the modeling of the robot's dynamics and the limitations of actuation, robots usually struggle to follow the desired trajectories for the contact forces to be applied by their end effectors or legs, risking of displacing their CoM and ZMP outside of their support regions, and lose balance as a consequence. This is particularly true when unexpected perturbations hit the robot, or most importantly, when the robot-environment interaction is compliant. If ignored, compliance alters the reference position of the robot's limbs, and present a challenge for the robot to produce the desired contact forces and moments.

The works in the literature usually rely on the assumption that the robot-environment interaction is rigid, and build the appropriate controllers under this assumption to ensure the balance of the robot. While this strategy works when the interaction is not compliant enough, these controllers struggle otherwise. A few works consider the compliance effect and design controllers that compensate for this effect, or model the compliance and study the dynamics imposed by it for the controller's design, however the solutions are either applied to quadrupeds, or to torque controlled humanoids using only their feet as contacts with the environment.

I started the work on this thesis by proposing a way to take the compliant robot-environment interaction into the balance controller design. Modeling the wrench of the interaction using the viscous-elastic model, I incorporated the contact forces and moments in a reduced model, where the robot is assumed to be a single rigid mass having mass-less limbs to interact with its surroundings, and without any assumptions on the lateral motion or the height of the CoM. Then I studied the non-linear dynamics of the system while writing

the positions and orientations of the contacts in the robot’s CoM frame. This model in itself can be used to build different controllers depending on the approach taken to deal with the non-linear dynamics of the system.

My solution was to transform the non-linear system into a linear one using a linearization process. In order to track the desired contact forces and moments accurately, I used a trade-off between the force and moments obtained from force sensors and the positions and orientations of the contacts. Finally, I relied on the LQR gain to generate the commanded accelerations of the contacts which are used to maintain the robot’s balance.

I incorporated the controller within a whole body QP solver and validated the controller on 2 robots in simulation. The first validation was performed on a 26 d.o.f torque controlled biped in MATLAB SIMULINK. The proposed controller showed superior results compared to the classical control with rigid contacts assumption, as the latter struggled to track the desired forces in spite of the compliance in the contact surface, while the former ensured the biped’s balance under external perturbations. The proposed controller succeeded in multiple scenarios, including multiple-contacts configurations.

The second validation was performed on position-controlled robots, first on HRP4 in simulation and then on a real HRP2-KAI robot. The controller was written using the `mc_rtc` framework and simulated using the choreonoid simulator. The controller managed to maintain the balance of the real robot with a strong compliance effect, due to the built-in compliance in HRP2-KAI and the soft mattress at the same time, without the need of use of force sensors when the compliance parameters were tuned along the gain matrices of the LQR. Force feedback information was needed for the controller to maintain the robot’s balance when modeling errors were considered.

Future Work

The controller in its current state can be applied to a variety of robots in different situations and applications. Performing additional tests and on different robots should be considered.

What follows are improvements on a theoretical structure of the controller itself. First, a lacking feature in the current architecture is contact constraints. Indeed, there are no constraints on the friction cones at the contacts that bind the forces to respect the friction on the contact surfaces to not cause slippage. In certain situations, especially in dynamic locomotion, this might cause the controller to fail in its balance control when the robot is in a complicated configuration, even when following the desired forces correctly.

Second, the controller should be applied to dynamic behaviors and movements. The linearization should consider a trajectory reference rather than a stationary one, and a pattern generator should be used to generate the trajectory references for the required tasks.

An additional step could include the use of an online estimator for the stiffness and damping of the compliant environment according to the proposed model. While the controller is robust to modeling errors, the performance is better when the controller has more accurate knowledge of these parameters, and can be done without relying on force feedback information.

Controlling robots in a compliant environment is an ongoing research topic. The work described in this thesis proposes a solution for a number of situations, and lays out the plan for improvements to be done in order to make the controller more universal in use.

Appendix A

Details of the Linearization Process

The operator Δ represents the Euclidean difference for positions and velocities in the state vector, for instance $v^\Delta = v - v^*$. As for the orientations, it is the axis-sine of angle representation of the relative orientation between the orientation matrices, for instance $\Omega^\Delta = \Theta(RR^{*\top})$.

The formula used in the linearization process is

$$(XY + Z + S(w))^\Delta \simeq X^\Delta Y^* + X^* Y^\Delta + Z^\Delta + S(w^\Delta) \quad (\text{A.1})$$

where X , Y and Z all $\in \mathbb{R}^{3 \times 3}$, and $w \in \mathbb{R}^3$. $(.)^\Delta$ is the error of $(.)$, $(.)^*$ is its reference value. $S(w)$ is the skew-symmetric operator of vector w . As we can notice, the operator Δ functions similarly to the derivative operator. As for the term $\Theta(R_{c,i,r}R_{c,i}^\top) = \Omega(R_{c,i,r}R_{c,i,b}^\top R^\top)$, the linearization was done in the limits where the estimation $\sin\theta \approx \theta$ can be used, as the following:

$$\begin{aligned} \Omega(R_M)^\Delta &\simeq \frac{1}{2} \text{Vec}(R_M - R_M^\top)^\Delta \\ &= C_b(\Omega^\Delta + R^* \Omega_{c,i,b}^\Delta), \end{aligned} \quad (\text{A.2})$$

with:

$$C_b = \frac{1}{2} \sum_{i=1}^3 S(e_i) R_M^* S(e_i) \quad (\text{A.3})$$

$$R_M = R_{c,i,r} R_{c,i,b}^\top R^\top \quad (\text{A.4})$$

$$e_1 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \quad (\text{A.5})$$

$$e_2 = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \quad (\text{A.6})$$

$$e_3 = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}. \quad (\text{A.7})$$

While assuming that the reference of the angular acceleration of the floating base is zero: $\dot{\omega}^* = 0$, applying the operator Δ on equations [2.24](#) and [2.25](#) gives

$$c\ddot{om}^\Delta = \frac{1}{m} \sum_{i=1}^{n_c} f_{c,i}^\Delta \quad (\text{A.8})$$

and

$$\begin{aligned} \dot{\omega}^\Delta = & R_I \left(\sum_i S(f_{c,i}^*) S(R^* p_{c,i,com}^*) + S(\omega^*) S(R_I^{-1} \omega^*) + S(\omega^*) R_I^{-1} S(\omega^*) \right) \Omega^\Delta \\ & - R_I \left(\sum_i S(f_{c,i}^*) R^* p_{c,i,com}^\Delta + \sum_i S(R^* p_{c,i,com}^*) f_{c,i}^\Delta + \sum_i t_{c,i}^\Delta \right) \omega^\Delta \\ & - R_I \left((S(R_I^{-1} \omega^*) - S(\omega^*) R_I^{-1}) \right) \omega^\Delta \end{aligned} \quad (\text{A.9})$$

with

$$\begin{aligned} R_I = & R^* I^{-1} R^{*\top} \\ f_{c,i}^\Delta = & -K_{f,p,i} com^\Delta - K_{f,d,i} c\ddot{om}^\Delta + K_{f,d,i} S(R^* p_{c,i,com}^*) \omega^\Delta \\ & + [K_{f,p,i} S(R^* p_{c,i,com}^*) + K_{f,d,i} (S(R^* \dot{p}_{c,i,com}^*) + S(\omega^*) S(R^* p_{c,i,com}^*))] \Omega^\Delta \\ & - [K_{f,d,i} S(\omega^*) R^* + K_{f,p,i} R^*] p_{c,i,com}^\Delta - K_{f,d,i} R^* \dot{p}_{c,i,com}^\Delta \end{aligned} \quad (\text{A.10})$$

$$\begin{aligned} t_{c,i}^\Delta = & [K_{t,p,i} C_{b,i} + K_{t,d,i} S(R^* \omega_{c,i,b}^*)] \Omega^\Delta \\ & - K_{t,d,i} \omega^\Delta + K_{t,p,i} C_{b,i} R^* \Omega_{c,i,b}^\Delta - K_{t,d,i} R^* \omega_{c,i,b}^\Delta \end{aligned} \quad (\text{A.11})$$

Using this linearization, I can write the linear dynamics as in equation [2.30](#), with ma-

trices A and B given by

$$A = \begin{pmatrix} F_0 & F_1 & F_2 & \dots & F_n \\ 0 & D_1 & 0 & \dots & 0 \\ 0 & 0 & D_2 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & D_n \end{pmatrix}, \quad (\text{A.12})$$

$$B = \begin{pmatrix} 0 & 0 & \dots & 0 \\ G_1 & 0 & 0 & 0 \\ 0 & G_2 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & G_n \end{pmatrix}, \quad (\text{A.13})$$

$$F_0 = \begin{pmatrix} 0 & 0 & \mathbb{I} & 0 \\ 0 & 0 & 0 & \mathbb{I} \\ A_{31} & A_{32} & A_{33} & A_{34} \\ A_{41} & A_{42} & A_{43} & A_{44} \end{pmatrix}, \quad (\text{A.14})$$

$$F_i = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ F_{i31} & 0 & -\frac{1}{m}K_{f,d,i}R^* & 0 \\ F_{i41} & F_{i42} & F_{i43} & F_{i44} \end{pmatrix}, \quad (\text{A.15})$$

$$D_i = \begin{pmatrix} 0 & 0 & \mathbb{I} & 0 \\ 0 & 0 & 0 & \mathbb{I} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad (\text{A.16})$$

$$G_i = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ \mathbb{I} & 0 \\ 0 & \mathbb{I} \end{pmatrix}. \quad (\text{A.17})$$

with

$$\begin{aligned}
A_{31} &= -\frac{1}{m} \sum_{i=1}^n K_{f,p,i} \\
A_{32} &= \frac{1}{m} \sum_{i=1}^n (A_{321} + A_{322}) \\
A_{33} &= -\frac{1}{m} \sum_{i=1}^n K_{f,d,i} \\
A_{34} &= \frac{1}{m} \sum_{i=1}^n (K_{f,d,i} S(R^* \mathbf{p}_{c,i,com}^*)) \\
A_{321} &= K_{f,p,i} S(R^* \mathbf{p}_{c,i,com}^*) \\
A_{322} &= K_{f,d,i} (S(R^* \dot{\mathbf{p}}_{c,i,com}^*) + S(\boldsymbol{\omega}^*) S(R^* \mathbf{p}_{c,i,com}^*)) \\
A_{41} &= -R_I \sum_{i=1}^n S(R^* \mathbf{p}_{c,i,com}^*) K_{f,p,i} \\
A_{42} &= R_I S(\boldsymbol{\omega}^*) R_I^{-1} S(\boldsymbol{\omega}^*) \\
&\quad + R_I \sum_{i=1}^n (A_{421} + A_{422} + A_{423}) \\
A_{43} &= -R_I \sum_{i=1}^n S(R^* \mathbf{p}_{c,i,com}^*) K_{f,d,i} \\
A_{44} &= R_I \left(S(R_I^{-1} \boldsymbol{\omega}^*) - S(\boldsymbol{\omega}^*) R_I^{-1} - \sum_{i=1}^n A_{441} \right) \\
A_{421} &= S(\mathbf{f}_{c,i}^*) S(R^* \mathbf{p}_{c,i,com}^*) \\
A_{422} &= S(R^* \mathbf{p}_{c,i,com}^*) K_{f,p,i} S(R^* \mathbf{p}_{c,i,com}^*) \\
&\quad + S(R^* \mathbf{p}_{c,i,com}^*) K_{f,d,i} (S(R^* \dot{\mathbf{p}}_{c,i,com}^*) + S(\boldsymbol{\omega}^*) S(R^* \mathbf{p}_{c,i,com}^*)) \\
A_{423} &= K_{t,p,i} C_{b,i} + K_{t,d,i} S(R^* \boldsymbol{\omega}_{c,i,b}^*) \\
A_{441} &= K_{t,d,i} + S(R^* \mathbf{p}_{c,i,com}^*) K_{f,d,i} S(R^* \mathbf{p}_{c,i,com}^*) \\
F_{i31} &= -\frac{1}{m} (K_{f,p,i} + K_{f,d,i} S(\boldsymbol{\omega}^*)) R^* \\
F_{i41} &= R_I (-S(\mathbf{f}_{c,i}^*) - S(R^* \mathbf{p}_{c,i,com}^*) K_{dp}) R^* \\
K_{dp} &= K_{f,d,i} S(\boldsymbol{\omega}^*) + K_{f,p,i} \\
F_{i42} &= R_I K_{t,p,i} C_{b,i} R \\
F_{i43} &= -R_I S(R^* \mathbf{p}_{c,i,com}^*) K_{f,d,i} R^* \\
F_{i44} &= -R_I K_{t,d,i} R^*
\end{aligned}$$

The matrix M in equation 2.35 can be deduced from equations (A.10) and (A.11), and is written as

$$M = \begin{pmatrix} 0 & 0 & \dots & \dots & 0 \\ T_1 & V_1 & 0 & \dots & 0 \\ T_2 & 0 & V_2 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ T_n & 0 & \dots & 0 & V_n \end{pmatrix}, \quad (\text{A.18})$$

$$T_i = \begin{pmatrix} R^{*\top} & T_{i12} & T_{i13} & -T_{i12}S(R^*\mathbf{p}_{c,i}^*) \\ 0 & T_{i22} & 0 & R^{*\top}K_{t,p,i}^{-1}K_{t,d,i} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad (\text{A.19})$$

$$V_i = \begin{pmatrix} V_{i11} & 0 & V_{i13} & 0 \\ 0 & -R^{*\top}C_{b,i}R & 0 & V_{i24} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad (\text{A.20})$$

with

$$\begin{aligned} T_{i12} &= -R^{*\top}S(R^*\mathbf{p}_{c,i}^*) - T_{i13}(S(R^*\dot{\mathbf{p}}_{c,i}^*) + S(\boldsymbol{\omega}^*)S(R^*\mathbf{p}_{c,i}^*)) \\ T_{i13} &= R^{*\top}K_{f,p,i}^{-1}K_{f,d,i} \\ T_{i22} &= -R^{*\top}(C_{b,i} + K_{t,p,i}^{-1}K_{t,d,i}S(R^*\boldsymbol{\omega}_{c,i}^*)) \\ V_{i11} &= R^{*\top}(\mathbb{I} + K_{f,p,i}^{-1}K_{f,d,i}S(\boldsymbol{\omega}^*))R^* \\ V_{i13} &= R^{*\top}K_{f,p,i}^{-1}K_{f,d,i}R^* \\ V_{i24} &= R^{*\top}K_{t,p,i}^{-1}K_{t,d,i}R^* \end{aligned}$$

Appendix B

Correspondence of the Stiffness and Damping of the contacts between Simulink and the proposed model

Here I go in detail on how the linear and angular stiffness and damping of the proposed model are deduced from the forces and moments generated by the contact spheres model used in Simulink.

Figure [B.1](#) shows an illustration of the procedure. On the left is the Matlab model of the compliant forces. Each sphere placed at the vertex of the contact surface has a stiffness $K_{f,p,j}$ and a damping $K_{f,d,j}$, generating a force f_j . The moment t_j is deduced by multiplying the f_j by the distance to the contact frame, situated at the center of the contact surface. On the right is the proposed model. The compliance at the contact level is concentrated on a single location on the contact surface, with a linear stiffness $K_{f,p,i}$ and a linear damping $K_{f,d,i}$, generating a force $f_{c,i}$. Additionally, the compliance has an angular stiffness $K_{t,p,i}$ and an angular damping $K_{t,d,i}$, generating a moment $t_{c,i}$.

B.1 Linear Stiffness and Damping

First of all, I want to note that in this appendix, any variable $.^*$ (such as $R_{c,i}^*$, etc...) marks the point around which the derivation takes place in the development below. The formulation is general and can be used around a reference point (which is what I am considering in this thesis) or other, depending on the usage.

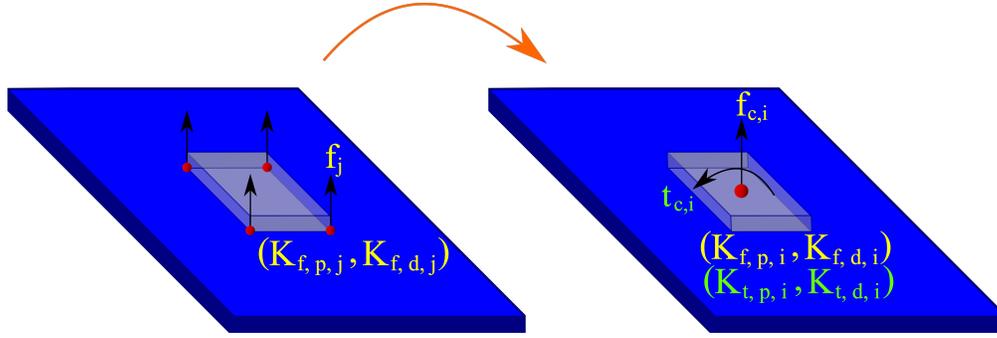


Figure B.1: Illustration showing the compliant force model in Matlab Simulink (Left) and the compliance of the proposed model on the contact level (right). A correspondence between the parameters is required for the implementation of the controller.

Using equation (4.2) in (4.1) gives

$$\begin{aligned} f_j &= K_{f,p,j} (p_{c,i} + R_{c,i} p_{j,l} - p_j^*) + K_{f,d,j} \frac{d}{dt} (p_{c,i} + R_{c,i} p_{j,l}) \\ &= K_{f,p,j} (p_{c,i} + R_{c,i} p_{j,l} - p_j^*) + K_{f,d,j} (\dot{p}_{c,i} + S(\omega_{c,i}) R_{c,i} p_{j,l}). \end{aligned} \quad (\text{B.1})$$

Then, using (B.1) in equation (4.3) allows me to write

$$f_{c,i} = \sum_{j=1}^4 (K_{f,p,j} (p_{c,i} + R_{c,i} p_{j,l} - p_j^*) + K_{f,d,j} (\dot{p}_{c,i} + S(\omega_{c,i}) R_{c,i} p_{j,l})). \quad (\text{B.2})$$

$f_{c,i}$ in equation (B.2) is considered equivalent to the contact force of the proposed model, i.e. $f_{c,i}$ in equation (2.16).

By Deriving equation (2.16) with respect to $p_{c,i}$:

$$\frac{\partial f_{c,i}}{\partial p_{c,i}} = K_{f,p,i}, \quad (\text{B.3})$$

and deriving equation (B.2) with respect to $p_{c,i}$:

$$\frac{\partial f_{c,i}}{\partial p_{c,i}} = \sum_{j=1}^4 K_{f,p,j}, \quad (\text{B.4})$$

I can conclude from equations (B.3) and (B.4) that

$$K_{f,p,i} = \sum_{j=1}^4 K_{f,p,j}. \quad (\text{B.5})$$

Similarly, by deriving (2.16) and (B.2) by $\dot{p}_{c,i}$ this time, I get

$$\begin{aligned} K_{f,d,i} &= \frac{\partial f_{c,i}}{\partial \dot{p}_{c,i}} \\ &= \sum_{j=1}^4 K_{f,d,j}. \end{aligned} \quad (\text{B.6})$$

B.2 Angular Stiffness and Damping

Using equation (B.2) in equation (4.5), I get:

$$t_{c,i} = \sum_{j=1}^4 (S(R_{c,i}p_{j,l}) (K_{f,p,j} (p_{c,i} + R_{c,i}p_{j,l} - \mathbf{p}_j^*) + K_{f,d,j} (\dot{p}_{c,i} + S(\omega_{c,i}) R_{c,i}p_{j,l}))). \quad (\text{B.7})$$

Similarly to the force, $t_{c,i}$ in equation (B.7) is considered equivalent to the contact moment of the model, i.e. $t_{c,i}$ in equation (2.17). To find the stiffness and damping constants, I will write the error dynamics of the moment $t_{c,i}$ in equation (4.5) (using the cross product form):

$$t_{c,i}^\Delta = \sum_{j=1}^4 t_j^\Delta, \quad (\text{B.8})$$

with

$$\begin{aligned} t_j^\Delta &= - \left(f_j^\Delta \times R_{c,i}^* p_{j,l}^* + f_j^* \times (R_{c,i} p_{j,l})^\Delta \right) \\ &= R_{c,i}^* p_{j,l}^* \times f_j^\Delta - f_j^* \times (R_{c,i} p_{j,l})^\Delta \\ &= S(R_{c,i}^* p_{j,l}^*) f_j^\Delta - S(f_j^*) (R_{c,i} p_{j,l})^\Delta. \end{aligned} \quad (\text{B.9})$$

For the calculation of $(R_{c,i} p_{j,l})^\Delta$, f_j^Δ , and then t_j^Δ , see (B.3) below.

Finally, the corresponding angular stiffness can be determined:

$$\begin{aligned}
K_{t,p,i} &= \left. \frac{\partial t_{c,i}}{\partial \Omega_{c,i}} \right|_{x=x^*} \\
&= \sum_{j=1}^4 (S(f_j^*) S(R_{c,i}^* p_{j,l}^*) - S(R_{c,i}^* p_{j,l}^*) (K_{f,p,j} S(R_{c,i}^* p_{j,l}^*) + K_{f,d,j} S(\omega_{c,i}^*) S(R_{c,i}^* p_{j,l}^*))) \\
&= \sum_{j=1}^4 (S(f_j^*) S(R_{c,i}^* p_{j,l}^*) - S(R_{c,i}^* p_{j,l}^*) (K_{f,p,j} + K_{f,d,j} S(\omega_{c,i}^*)) S(R_{c,i}^* p_{j,l}^*)) \\
&= \sum_{j=1}^4 S(f_j^*) S(R_{c,i}^* p_{j,l}^*) - \sum_{j=1}^4 S(R_{c,i}^* p_{j,l}^*) (K_{f,p,j} + K_{f,d,j} S(\omega_{c,i}^*)) S(R_{c,i}^* p_{j,l}^*). \quad (\text{B.10})
\end{aligned}$$

and damping:

$$\begin{aligned}
K_{t,d,i} &= \left. \frac{\partial t_{c,i}}{\partial \omega_{c,i}} \right|_{x=x^*} \\
&= \sum_{j=1}^4 S(-R_{c,i}^* p_{j,l}^*) K_{f,d,j} S(R_{c,i}^* p_{j,l}^*) \quad (\text{B.11})
\end{aligned}$$

Notice that $K_{t,p,i}$ in equation [B.10](#) depends on the reference forces of each sphere, however in my case where the contact frame lies at the center of the contact sphere, and each sphere at a given contact surface has the same parameters, the following term becomes zero:

$$\sum_{j=1}^4 S(f_j^*) S(R_{c,i}^* p_{j,l}^*) = 0.$$

Thus, $K_{t,p,i}$ becomes independent of f_j^* .

B.3 Additional Calculations

Here are some additional calculations to be used in equation [B.8](#). I put them in a separate section for clarity.

The Δ approximation of $R_{c,i} p_{j,l}$:

$$\begin{aligned}
(R_{c,i} p_{j,l})^\Delta &\simeq R_{c,i}^* p_{j,l}^\Delta + S(\Omega_{c,i}^\Delta) R_{c,i}^* p_{j,l}^* \\
&= R_{c,i}^* p_{j,l}^\Delta - S(R_{c,i}^* p_{j,l}^*) \Omega_{c,i}^\Delta. \quad (\text{B.12})
\end{aligned}$$

The Δ approximation of f_j :

$$\begin{aligned}
f_j^\Delta &= (K_{f,p,j} (p_{c,i} + R_{c,i} p_{j,l} - \mathbf{p}_j^*) + K_{f,d,j} (\dot{p}_{c,i} + S(\omega_{c,i}) R_{c,i} p_{j,l}))^\Delta \\
&= K_{f,p,j} (p_{c,i}^\Delta + (R_{c,i} p_{j,l})^\Delta) + K_{f,d,j} (\dot{p}_{c,i}^\Delta + S(\omega_{c,i}^\Delta) R_{c,i}^* p_{j,l}^* + S(\omega_{c,i}^*) (R_{c,i} p_{j,l})^\Delta) \\
&= K_{f,p,j} (p_{c,i}^\Delta + R_{c,i}^* p_{j,l}^\Delta - S(R_{c,i}^* p_{j,l}^*) \Omega_{c,i}^\Delta) \\
&\quad + K_{f,d,j} (\dot{p}_{c,i}^\Delta - S(R_{c,i}^* p_{j,l}^*) \omega_{c,i}^\Delta + S(\omega_{c,i}^*) (R_{c,i}^* p_{j,l}^\Delta - S(R_{c,i}^* p_{j,l}^*) \Omega_{c,i}^\Delta)) \\
&= K_{f,p,j} p_{c,i}^\Delta + (K_{f,p,j} R_{c,i}^* + K_{f,d,j} S(\omega_{c,i}^*) R_{c,i}^*) p_{j,l}^\Delta + K_{f,d,j} \dot{p}_{c,i}^\Delta \\
&\quad - K_{f,d,j} S(R_{c,i}^* p_{j,l}^*) \omega_{c,i}^\Delta - (K_{f,p,j} S(R_{c,i}^* p_{j,l}^*) + K_{f,d,j} S(\omega_{c,i}^*) S(R_{c,i}^* p_{j,l}^*)) \Omega_{c,i}^\Delta. \tag{B.13}
\end{aligned}$$

Therefore:

$$\begin{aligned}
t_j^\Delta &= S(R_{c,i}^* p_{j,l}^*) f_j^\Delta - S(f_j^*) (R_{c,i} p_{j,l})^\Delta \\
&= S(R_{c,i}^* p_{j,l}^*) [K_{f,p,j} p_{c,i}^\Delta + (K_{f,p,j} R_{c,i}^* + K_{f,d,j} S(\omega_{c,i}^*) R_{c,i}^*) p_{j,l}^\Delta + K_{f,d,j} \dot{p}_{c,i}^\Delta] \\
&\quad - S(R_{c,i}^* p_{j,l}^*) [K_{f,d,j} S(R_{c,i}^* p_{j,l}^*) \omega_{c,i}^\Delta - (K_{f,p,j} S(R_{c,i}^* p_{j,l}^*) + K_{f,d,j} S(\omega_{c,i}^*) S(R_{c,i}^* p_{j,l}^*)) \Omega_{c,i}^\Delta] \\
&\quad - S(f_j^*) [R_{c,i}^* p_{j,l}^\Delta - S(R_{c,i}^* p_{j,l}^*) \Omega_{c,i}^\Delta]. \tag{B.14}
\end{aligned}$$

The term $K_{t,p,i} = \left. \frac{\partial t_{c,i}}{\partial \Omega_{c,i}} \right|_{x=x^*}$ is the term multiplying $\Omega_{c,i}^\Delta$ and the term $K_{t,d,i} = \left. \frac{\partial t_{c,i}}{\partial \omega_c} \right|_{x=x^*}$ is the term multiplying $\omega_{c,i}^\Delta$ in equation (B.10) (preceded by a $\sum_{j=1}^4$)

Appendix C

General Guide to the Tuning of the Controller

The tuning of the controller is done by setting the compliance parameters (i.e., the stiffness and damping of each contact: $K_{f,p,i}$, $K_{f,d,i}$, $K_{t,p,i}$ and $K_{t,d,i}$) and by adjusting the gains of the matrices Q , P , and W .

It is very important to start testing the controller in simulation first before moving to tests on real robots, where the compliance parameters can be manually set in the simulation environment and in the controller.

It is recommended to run the first simulations without any modeling errors on the compliance parameters, so that a good performance can be achieved without the need of tuning the matrix W , in other words, without including the force feedback in the feedback for the controller. This is done by choosing W to be a null matrix.

I will write the matrix Q as

$$\begin{aligned} Q &= \text{diag} \left(Q_{com} \quad Q_{c,1} \quad \cdots \quad Q_{c,n} \right), \\ Q_{com} &= \begin{pmatrix} q_{com} & q_R & q_{\dot{com}} & q_\omega \end{pmatrix}, \\ Q_{c_i} &= \begin{pmatrix} q_{c,i} & q_{R_{c,i}} & q_{\dot{c},i} & q_{\omega_{c,i}} \end{pmatrix}. \end{aligned} \tag{C.1}$$

$q_{com} \in SO(3)$, $q_R \in SO(3)$, $q_{\dot{com}} \in SO(3)$, and $q_\omega \in SO(3)$ represent the weights for minimizing the error of the CoM, orientation of the floating base, CoM velocity and the base's angular velocity respectively. Similarly, $q_{c,i} \in SO(3)$, $q_{R_{c,i}} \in SO(3)$, $q_{\dot{c},i} \in SO(3)$, and $q_{\omega_{c,i}} \in SO(3)$ represent the weights for minimizing the error of the contact i 's position,

orientation, linear velocity, and angular velocity respectively.

The position and orientation associated terms serve for the better tracking of the CoM and contact positions and orientations, while the velocity associated terms are important to reduce oscillations that could be present in the curves of the CoM or a contact's position and orientation.

For the first test, it is advisable to start from high weights for $q_{c,i}$ and $q_{R_{c,i}}$, and to have at least $q_{\dot{c},i} = 2\sqrt{q_{c,i}}$ and $q_{\omega_{c,i}} = 2\sqrt{q_{R_{c,i}}}$. As for q_{com} , q_R , $q_{\dot{com}}$, and q_ω , it is better to have them at very low values (identity matrix or even null matrix). This way, the controller is asking the robot to maintain its contacts without balance control (i.e. without minimizing the error of the CoM), which is equivalent to sending simple position and orientation tasks for the contacts. At this stage, if the robot is slightly pushed, it should oscillate for a few seconds. Then, slowly increase $q_{\dot{com}}$ to reduce the oscillations of the CoM. Eventually, oscillations should disappear if the robot is perturbed, with a very slow convergence of the CoM toward its desired reference. Similarly, an increase of q_ω might be required to eliminate the oscillations of the floating base. Once oscillations no longer appear, start increasing q_{com} slowly to improve the minimization of the error of the CoM, and if needed, start increasing q_R to minimize the error of the orientation of the base, until the best performance is achieved. Note that by increasing q_{com} for example, oscillations might re-appear, thus $q_{\dot{com}}$ should be increased again to eliminate them. It is up to the user to choose the final desirable performance.

If the accelerations generated by the LQR have high values, they can be reduced by tuning the matrix P .

Once balance control is up to the desired performance, modeling errors can be introduced on the compliance parameters in the controller's configuration. The performance will worsen, and can be improved by tuning the matrix W responsible for tuning the force feedback in the kinematic-force trade-off. If the robot is struggling to maintain the contacts position (e.g. the feet are seemingly going up and down) or follow the desired contact forces, slowly increase w_{f_i} to include the contact force. If the robot is struggling with the orientation of the contacts and is falling on its back due to these orientations, slowly increase w_{t_i} to include the contact moment. Note that the values in W should be between 0 and 1, and it is not advisable to have ones for the three components of a contact force or moment, because the controller will lose the contact position or rotation information, and the robot will fall.

Once the behavior of the controller is understood in simulation, and a good performance is achieved while considering modeling errors, experiments on a real robot can be done, while restarting the tuning with the same strategy, as the gains will differ from those set in simula-

tion. One problem emerges which is that the compliance parameters are not known in such case. In this case, these parameters should be tuned alongside the matrix Q until a good performance is achieved without force feedback, meaning that the compliance parameters chosen could be considered as the "real" values. It is important to note that it might be enough to set the damping terms ($K_{f,d,i}$ and $K_{t,d,i}$) to identity matrices, i.e. considering a compliance with negligible damping effect and avoid tuning or estimating them, as demonstrated in chapter 5, and tune only $K_{f,p,i}$ and $K_{t,p,i}$.

Furthermore, one can choose not to tune any compliance parameter, rather have their values fixed, and then tune W alongside Q to compensate for modeling errors. After all, tuning these parameters is done similarly to the tuning of W : if the robot is swinging and falling on its back for example, this means that the angular stiffness has errors and should be adjusted; if the robot's feet are moving up and down or right and left, this means that the linear stiffness has errors and should be adjusted. It is important to mention that increased compliance means lower values for $K_{f,p,i}$ and $K_{t,p,i}$.

It is also advisable to tune all variables along a single axis at a time when testing on real robot, since the performance can be quite different from simulations, for increased safety.

Bibliography

- A. Albu-Schaffer and G. Hirzinger. Cartesian impedance control techniques for torque controlled light-weight robots. In *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, volume 1, pages 657–663. IEEE, 2002. ISBN 978-0-7803-7272-6. doi: 10.1109/ROBOT.2002.1013433. URL <http://ieeexplore.ieee.org/document/1013433/>.
- A. Albu-Schaffer, C. Ott, and G. Hirzinger. A passivity based cartesian impedance controller for flexible joint robots - part II: full state feedback, impedance design and experiments. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, pages 2666–2672 Vol.3. IEEE, 2004. ISBN 978-0-7803-8232-9. doi: 10.1109/ROBOT.2004.1307463. URL <http://ieeexplore.ieee.org/document/1307463/>.
- Alin Albu-Schaffer, Oliver Eiberger, Markus Grebenstein, Sami Haddadin, Christian Ott, Thomas Wimbock, Sebastian Wolf, and Gerd Hirzinger. Soft robotics. *IEEE Robotics & Automation Magazine*, 15(3):20–30, 2008. ISSN 1070-9932. doi: 10.1109/MRA.2008.927979. URL <http://ieeexplore.ieee.org/document/4624580/>.
- Morteza Azad and Roy Featherstone. A new nonlinear model of contact normal force. *IEEE Transactions on Robotics*, 30(3):736–739, 2014. ISSN 1552-3098, 1941-0468. doi: 10.1109/TRO.2013.2293833. URL <http://ieeexplore.ieee.org/document/6689296/>.
- Morteza Azad, Valerio Ortenzi, Hsiu-Chin Lin, Elmar Rueckert, and Michael Mistry. Model estimation and control of compliant contact normal force. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 442–447. IEEE, 2016. ISBN 978-1-5090-4718-5. doi: 10.1109/HUMANOIDS.2016.7803313. URL <http://ieeexplore.ieee.org/document/7803313/>.
- Victor Barasuol, Jonas Buchli, Claudio Semini, Marco Frigerio, Edson R. De Pieri, and Darwin G. Caldwell. A reactive controller framework for quadrupedal locomotion on

- challenging terrain. In *2013 IEEE International Conference on Robotics and Automation*, pages 2554–2561. IEEE, 2013. ISBN 978-1-4673-5643-5 978-1-4673-5641-1. doi: 10.1109/ICRA.2013.6630926. URL <http://ieeexplore.ieee.org/document/6630926/>.
- Mehdi Benallegue and Florent Lamiroux. Estimation and stabilization of humanoid flexibility deformation using only inertial measurement units and contact information. *International Journal of Humanoid Robotics (IJHR)*, 12(3):1550025, 2015. ISSN 0219-8436, 1793-6942. doi: 10.1142/S0219843615500255. URL <https://www.worldscientific.com/doi/abs/10.1142/S0219843615500255>.
- Mehdi Benallegue, Pierre Gergondet, Herve Audrerr, Alexis Mifsud, Mitsuharu Morisawa, Florent Lamiroux, Abderrahmane Kheddar, and Fumio Kanehiro. Model-based external force/moment estimation for humanoid robots with no torque measurement. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3122–3129. IEEE, 2018. ISBN 978-1-5386-3081-5. doi: 10.1109/ICRA.2018.8460809. URL <https://ieeexplore.ieee.org/document/8460809/>.
- Priyaranjan Biswal and Prases K. Mohanty. Development of quadruped walking robots: A review. *Ain Shams Engineering Journal*, 12(2):2017–2031, 2021. ISSN 20904479. doi: 10.1016/j.asej.2020.11.005. URL <https://linkinghub.elsevier.com/retrieve/pii/S2090447920302501>.
- Thiago Boaventura, Gustavo A. Medrano-Cerda, Claudio Semini, Jonas Buchli, and Darwin G. Caldwell. Stability and performance of the compliance controller of the quadruped robot HyQ. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1458–1464. IEEE, 2013. ISBN 978-1-4673-6358-7 978-1-4673-6357-0. doi: 10.1109/IROS.2013.6696541. URL <http://ieeexplore.ieee.org/document/6696541/>.
- Anastasia Bolotnikova, Sebastien Courtois, and Abderrahmane Kheddar. Autonomous initiation of human physical assistance by a humanoid. In *2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, pages 857–862. IEEE, 2020a. ISBN 978-1-72816-075-7. doi: 10.1109/RO-MAN47096.2020.9223519. URL <https://ieeexplore.ieee.org/document/9223519/>.
- Anastasia Bolotnikova, Pierre Gergondet, Arnaud Tanguy, Sébastien Courtois, and Abderrahmane Kheddar. Task-space control interface for SoftBank humanoid robots and its human-robot interaction applications, 2020b. URL <http://arxiv.org/abs/2010.04573>.

Will Bosworth, Jonas Whitney, Sangbae Kim, and Neville Hogan. Robot locomotion on hard and soft ground: Measuring stability and ground properties in-situ. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3582–3589. IEEE, 2016. ISBN 978-1-4673-8026-3. doi: 10.1109/ICRA.2016.7487541. URL <http://ieeexplore.ieee.org/document/7487541/>.

Karim Bouyarmane and Abderrahmane Kheddar. Humanoid robot locomotion and manipulation step planning. *Advanced Robotics*, 26(10):1099–1126, 2012. ISSN 0169-1864, 1568-5535. doi: 10.1080/01691864.2012.686345. URL <https://www.tandfonline.com/doi/full/10.1080/01691864.2012.686345>.

Karim Bouyarmane, Joris Vaillant, Francois Keith, and Abderrahmane Kheddar. Exploring humanoid robots locomotion capabilities in virtual disaster response scenarios. In *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)*, pages 337–342. IEEE, 2012. ISBN 978-1-4673-1369-8. doi: 10.1109/HUMANOIDS.2012.6651541. URL <http://ieeexplore.ieee.org/document/6651541/>.

Karim Bouyarmane, Stéphane Caron, Adrien Escande, and Abderrahmane Kheddar. Multi-contact motion planning and control. In Ambarish Goswami and Prahlad Vadakkepat, editors, *Humanoid Robotics: A Reference*, pages 1763–1804. Springer Netherlands, 2019. ISBN 978-94-007-6045-5 978-94-007-6046-2. doi: 10.1007/978-94-007-6046-2_32. URL http://link.springer.com/10.1007/978-94-007-6046-2_32.

Rohan Budhiraja, Justin Carpentier, and Nicolas Mansard. Dynamics consensus between centroidal and whole-body models for locomotion of legged robots. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 6727–6733. IEEE, 2019. ISBN 978-1-5386-6027-0. doi: 10.1109/ICRA.2019.8793878. URL <https://ieeexplore.ieee.org/document/8793878/>.

Andrea Calanca, Riccardo Muradore, and Paolo Fiorini. A review of algorithms for compliant control of stiff and fixed-compliance robots. *IEEE/ASME Transactions on Mechatronics*, 21(2):613–624, 2016. ISSN 1083-4435, 1941-014X. doi: 10.1109/TMECH.2015.2465849. URL <http://ieeexplore.ieee.org/document/7182329/>.

Stephane Caron, Quang Cuong Pham, and Yoshihiko Nakamura. Leveraging cone double description for multi-contact stability of humanoids with applications to statics and

- dynamics. In *Robotics: Science and Systems XI*. Robotics: Science and Systems Foundation, 2015a. ISBN 978-0-9923747-1-6. doi: 10.15607/RSS.2015.XI.028. URL <http://www.roboticsproceedings.org/rss11/p28.pdf>.
- Stéphane Caron, Quang-Cuong Pham, and Yoshihiko Nakamura. ZMP Support Areas for Multicontact Mobility Under Frictional Constraints. *IEEE Transactions on Robotics*, 33(1):67–80, February 2017. doi: 10.1109/TRO.2016.2623338. URL <https://hal.science/hal-02108589>.
- Stéphane Caron. Biped stabilization by linear feedback of the variable-height inverted pendulum model, 2020. URL <http://arxiv.org/abs/1909.07732>.
- Stéphane Caron, Quang-Cuong Pham, and Yoshihiko Nakamura. Stability of surface contacts for humanoid robots: Closed-form formulae of the contact wrench cone for rectangular support areas, 2015b. URL <http://arxiv.org/abs/1501.04719>.
- Stéphane Caron, Abderrahmane Kheddar, and Olivier Tempier. Stair climbing stabilization of the HRP-4 humanoid robot using whole-body admittance control, 2021. URL <http://arxiv.org/abs/1809.07073>.
- Shen-Chiang Chen, Chih-Chung Ko, Cheng-Hsin Li, and Pei-Chun Lin. Stair climbing in a quadruped robot. *International Journal of Automation and Smart Technology*, 2(1):11–20, 2012. ISSN 2223-9766. doi: 10.5875/ausmt.v2i1.77. URL <http://www.ausmt.org/index.php/AUSMT/article/view/77>.
- Rafael Cisneros, Kazuhito Yokoi, and Eiichi Yoshida. Yaw moment compensation by using full body motion. In *2014 IEEE International Conference on Mechatronics and Automation*, pages 119–125, 2014. doi: 10.1109/ICMA.2014.6885682.
- Rafael Cisneros, Mehdi Benallegue, Abdelaziz Benallegue, Mitsuharu Morisawa, Herve Audren, Pierre Gergondet, Adrien Escande, Abderrahmane Kheddar, and Fumio Kanehiro. Robust humanoid control using a QP solver with integral gains. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7472–7479. IEEE, 2018a. ISBN 978-1-5386-8094-0. doi: 10.1109/IROS.2018.8593417. URL <https://ieeexplore.ieee.org/document/8593417/>.
- Rafael Cisneros, Mehdi Benallegue, Mitsuharu Morisawa, Eiichi Yoshida, Kazuhito Yokoi, and Fumio Kanehiro. Partial yaw moment compensation using an optimization-based

- multi-objective motion solver. In *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*, pages 1017–1024. IEEE, 2018b. ISBN 978-1-5386-7283-9. doi: 10.1109/HUMANOIDS.2018.8625076. URL <https://ieeexplore.ieee.org/document/8625076/>.
- Rafael Cisneros-Limon, Mitsuharu Morisawa, Mehdi Benallegue, Adrien Escande, and Fumio Kanehiro. An inverse dynamics-based multi-contact locomotion control framework without joint torque feedback. *Advanced Robotics*, 34(21):1398–1419, 2020. ISSN 0169-1864, 1568-5535. doi: 10.1080/01691864.2020.1842140. URL <https://www.tandfonline.com/doi/full/10.1080/01691864.2020.1842140>.
- Hongkai Dai, Andres Valenzuela, and Russ Tedrake. Whole-body motion planning with centroidal dynamics and full kinematics. In *2014 IEEE-RAS International Conference on Humanoid Robots*, pages 295–302. IEEE, 2014. ISBN 978-1-4799-7174-9. doi: 10.1109/HUMANOIDS.2014.7041375. URL <http://ieeexplore.ieee.org/document/7041375/>.
- Alessandro De Luca, Alin Albu-Schaffer, Sami Haddadin, and Gerd Hirzinger. Collision detection and safe reaction with the DLR-III lightweight manipulator arm. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1623–1630. IEEE, 2006. ISBN 978-1-4244-0258-8. doi: 10.1109/IROS.2006.282053. URL <https://ieeexplore.ieee.org/document/4058607/>.
- Andrea Del Prete, Steve Tonneau, and Nicolas Mansard. Fast algorithms to test robust static equilibrium for legged robots. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1601–1607. IEEE, 2016. ISBN 978-1-4673-8026-3. doi: 10.1109/ICRA.2016.7487299. URL <http://ieeexplore.ieee.org/document/7487299/>.
- A. Dietrich, T. Wimbock, and Alin Albu-Schaffer. Dynamic whole-body mobile manipulation with a torque controlled humanoid robot via impedance control laws. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3199–3206. IEEE, 2011. ISBN 978-1-61284-456-5 978-1-61284-454-1 978-1-61284-455-8. doi: 10.1109/IROS.2011.6094445. URL <http://ieeexplore.ieee.org/document/6094445/>.
- Fotios Dimeas and Nikos Aspragathos. Online stability in human-robot cooperation with admittance control. *IEEE Transactions on Haptics*, 9(2):267–278, 2016. ISSN 1939-1412. doi: 10.1109/TOH.2016.2518670. URL <http://ieeexplore.ieee.org/document/7384497/>.

- Vincent Duindam and Stefano Stramigioli. Lagrangian dynamics of open multibody systems with generalized holonomic and nonholonomic joints. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3342–3347. IEEE, 2007. ISBN 978-1-4244-0911-2 978-1-4244-0912-9. doi: 10.1109/IROS.2007.4399066. URL <http://ieeexplore.ieee.org/document/4399066/>.
- Johannes Engelsberger, C. Ott, M. A. Roa, Alin Albu-Schaffer, and G. Hirzinger. Bipedal walking control based on capture point dynamics. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4420–4427. IEEE, 2011. ISBN 978-1-61284-456-5 978-1-61284-454-1 978-1-61284-455-8. doi: 10.1109/IROS.2011.6094435. URL <http://ieeexplore.ieee.org/document/6094435/>.
- Adrien Escande, Abderrahmane Kheddar, Jrl Cnrs, Sylvain Miossec, and Jrl Aist. Planning support contact-points for humanoids robots and experiments on HRP-2. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, page 6, 2006.
- Adrien Escande, Abderrahmane Kheddar, and Sylvain Miossec. Planning contact points for humanoid robots. *Robotics and Autonomous Systems*, 61(5):428–442, 2013. ISSN 09218890. doi: 10.1016/j.robot.2013.01.008. URL <https://linkinghub.elsevier.com/retrieve/pii/S0921889013000213>.
- Adrien Escande, Nicolas Mansard, and Pierre-Brice Wieber. Hierarchical quadratic programming: Fast online humanoid-robot motion generation. *The International Journal of Robotics Research (IJRR)*, 33(7):1006–1028, 2014. ISSN 0278-3649, 1741-3176. doi: 10.1177/0278364914521306. URL <http://journals.sagepub.com/doi/10.1177/0278364914521306>.
- Shamel Fahmi, Michele Focchi, Andreea Radulescu, Geoff Fink, Victor Barasuol, and Claudio Semini. STANCE: Locomotion adaptation over soft terrain, 2019. URL <http://arxiv.org/abs/1904.12306>.
- Roy Featherstone. *Rigid Body Dynamics Algorithms*. Springer New York, NY, 1 edition, 2008. ISBN 978-0-387-74314-1. URL <https://doi.org/10.1007/978-1-4899-7560-7>.
- Thomas Flayols, Andrea Del Prete, Majid Khadiv, Nicolas Mansard, and Ludovic Righetti. Reactive balance control for legged robots under visco-elastic contacts. *Applied Sciences*, 11(1):353, 2020. ISSN 2076-3417. doi: 10.3390/app11010353. URL <https://www.mdpi.com/2076-3417/11/1/353>.

Milad Geravand, Fabrizio Flacco, and Alessandro De Luca. Human-robot physical interaction and collaboration using an industrial robot with a closed control architecture. In *2013 IEEE International Conference on Robotics and Automation*, pages 4000–4007. IEEE, 2013. ISBN 978-1-4673-5643-5 978-1-4673-5641-1. doi: 10.1109/ICRA.2013.6631141. URL <http://ieeexplore.ieee.org/document/6631141/>.

Yukai Gong, Ross Hartley, Xingye Da, Ayonga Hereid, Omar Harib, Jiunn-Kai Huang, and Jessy Grizzle. Feedback control of a cassie bipedal robot: Walking, standing, and riding a segway, 2018. URL <http://arxiv.org/abs/1809.07279>.

J. Rogelio Guadarrama-Olvera, Shuuji Kajita, and Gordon Cheng. Preemptive foot compliance to lower impact during biped robot walking over unknown terrain. *IEEE Robotics and Automation Letters*, 7(3):8006–8011, 2022. ISSN 2377-3766, 2377-3774. doi: 10.1109/LRA.2022.3187253. URL <https://ieeexplore.ieee.org/document/9810342/>.

K. Harada, H. Hirukawa, F. Kanchiro, K. Fujiwara, K. Kaneko, S. Kajita, and M. Nakamura. Dynamical balance of a humanoid robot grasping an environment. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, volume 2, pages 1167–1173. IEEE, 2004. ISBN 978-0-7803-8463-7. doi: 10.1109/IROS.2004.1389554. URL <http://ieeexplore.ieee.org/document/1389554/>.

K. Harada, S. Kajita, K. Kaneko, and H. Hirukawa. Dynamics and balance of a humanoid robot during manipulation tasks. *IEEE Transactions on Robotics*, 22(3):568–575, 2006. ISSN 1552-3098. doi: 10.1109/TRO.2006.870649. URL <http://ieeexplore.ieee.org/document/1638349/>.

Kris Hauser. Fast interpolation and time-optimization with contact. *The International Journal of Robotics Research (IJRR)*, 33(9):1231–1250, 2014. ISSN 0278-3649, 1741-3176. doi: 10.1177/0278364914527855. URL <http://journals.sagepub.com/doi/10.1177/0278364914527855>.

Alexander Herzog, Ludovic Righetti, Felix Grimminger, Peter Pastor, and Stefan Schaal. Balancing experiments on a torque-controlled humanoid with hierarchical inverse dynamics. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 981–988. IEEE, 2014. ISBN 978-1-4799-6934-0 978-1-4799-6931-9. doi: 10.1109/IROS.2014.6942678. URL <http://ieeexplore.ieee.org/document/6942678/>.

Alexander Herzog, Stefan Schaal, and Ludovic Righetti. Structured contact force optimization for kino-dynamic motion generation. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2703–2710. IEEE, 2016. ISBN 978-1-5090-3762-9. doi: 10.1109/IROS.2016.7759420. URL <http://ieeexplore.ieee.org/document/7759420/>.

K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka. The development of honda humanoid robot. In *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*, volume 2, pages 1321–1326. IEEE, 1998. ISBN 978-0-7803-4300-9. doi: 10.1109/ROBOT.1998.677288. URL <http://ieeexplore.ieee.org/document/677288/>.

Shinichi Hirai Hirai. *Analysis and Planning of Manipulation Using the Theory of Polyhedral Convex Cones*. Theses, Kyoto University, 1991.

H. Hirukawa, S. Hattori, K. Harada, S. Kajita, K. Kaneko, F. Kanehiro, K. Fujiwara, and M. Morisawa. A universal stability criterion of the foot contact of legged robots - adios ZMP. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pages 1976–1983. IEEE, 2006. ISBN 978-0-7803-9505-3. doi: 10.1109/ROBOT.2006.1641995. URL <http://ieeexplore.ieee.org/document/1641995/>.

N. Hogan. Stable execution of contact tasks using impedance control. In *Proceedings. 1987 IEEE International Conference on Robotics and Automation*, volume 4, pages 1047–1054. Institute of Electrical and Electronics Engineers, 1987. doi: 10.1109/ROBOT.1987.1087854. URL <http://ieeexplore.ieee.org/document/1087854/>.

M. Hutter, C. Gehring, A. Lauber, F. Gunther, C. D. Bellicoso, V. Tsounis, P. Fankhauser, R. Diethelm, S. Bachmann, M. Bloesch, H. Kolvenbach, M. Bjelonic, L. Isler, and K. Meyer. ANYmal - toward legged robots for harsh environments. *Advanced Robotics*, 31(17):918–931, 2017. ISSN 0169-1864, 1568-5535. doi: 10.1080/01691864.2017.1378591. URL <https://www.tandfonline.com/doi/full/10.1080/01691864.2017.1378591>.

Marco Hutter, Christian Gehring, Dominic Jud, Andreas Lauber, C. Dario Bellicoso, Vassilios Tsounis, Jemin Hwangbo, Karen Bodie, Peter Fankhauser, Michael Bloesch, Remo Diethelm, Samuel Bachmann, Amir Melzer, and Mark Hoepflinger. ANYmal - a highly mobile and dynamic quadrupedal robot. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 38–44. IEEE, 2016. ISBN 978-1-5090-3762-9. doi: 10.1109/IROS.2016.7758092. URL <http://ieeexplore.ieee.org/document/7758092/>.

- Fumiya Iida and Cecilia Laschi. Soft robotics: Challenges and perspectives. *Procedia Computer Science*, 7:99–102, 2011. ISSN 18770509. doi: 10.1016/j.procs.2011.12.030. URL <https://linkinghub.elsevier.com/retrieve/pii/S1877050911006958>.
- Jong Hyeon Park and Eung Seo Kim. Foot and body control of biped robots to walk on irregularly protruded uneven surfaces. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(1):289–297, 2009. ISSN 1083-4419. doi: 10.1109/TSMCB.2008.2003451. URL <http://ieeexplore.ieee.org/document/4695980/>.
- Jung-Hoon Kim and Jun-Ho Oh. Walking control of the humanoid platform KHR-1 based on torque feedback control. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, pages 623–628 Vol.1. IEEE, 2004. ISBN 978-0-7803-8232-9. doi: 10.1109/ROBOT.2004.1307218. URL <http://ieeexplore.ieee.org/document/1307218/>.
- S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa. The 3d linear inverted pendulum mode: a simple modeling for a biped walking pattern generation. In *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No.01CH37180)*, volume 1, pages 239–246. IEEE, 2001a. ISBN 978-0-7803-6612-1. doi: 10.1109/IROS.2001.973365. URL <http://ieeexplore.ieee.org/document/973365/>.
- S. Kajita, K. Yokoi, M. Saigo, and K. Tanie. Balancing a humanoid robot using backdrive concerned torque control and direct angular momentum feedback. In *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*, volume 4, pages 3376–3382. IEEE, 2001b. ISBN 978-0-7803-6576-6. doi: 10.1109/ROBOT.2001.933139. URL <http://ieeexplore.ieee.org/document/933139/>.
- 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 *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, pages 1620–1626. IEEE, 2003. ISBN 978-0-7803-7736-3. doi: 10.1109/ROBOT.2003.1241826. URL <http://ieeexplore.ieee.org/document/1241826/>.
- S Kajita, M Morisawa, K Miura, S Nakaoka, K Harada, K Kaneko, F Kanehiro, and K Yokoi. Biped walking stabilization based on linear inverted pendulum tracking. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4489–4496. IEEE, 2010.

- ISBN 978-1-4244-6674-0. doi: 10.1109/IROS.2010.5651082. URL <http://ieeexplore.ieee.org/document/5651082/>.
- Shuuji Kajita. Study of dynamic walk control of a biped robot on rugged terrain - derivation and application of the linear inverted pendulum mode. *Journal of Robotics and Mechatronics*, 5(6):516–523, 1993.
- Shuuji Kajita, Hirohisa Hirukawa, Kensuke Harada, and Kazuhito Yokoi. *Introduction to Humanoid Robotics*. Springer Berlin, Heidelberg, 1 edition, 2005. ISBN 978-3-642-54535-1. URL <https://doi.org/10.1007/978-3-642-54536-8>.
- Shuuji Kajita, Mehdi Benallegue, Rafael Cisneros, Takeshi Sakaguchi, Shin'ichiro Nakaoka, Mitsuharu Morisawa, Kenji Kaneko, and Fumio Kanehiro. Biped walking pattern generation based on spatially quantized dynamics. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, pages 599–605. IEEE, 2017. ISBN 978-1-5386-4678-6. doi: 10.1109/HUMANOIDS.2017.8246933. URL <http://ieeexplore.ieee.org/document/8246933/>.
- Shuuji Kajita, Mehdi Benallegue, Rafael Cisneros, Takeshi Sakaguchi, Shin'ichiro Nakaoka, Mitsuharu Morisawa, Hiroshi Kaminaga, Iori Kumagai, Kenji Kaneko, and Fumio Kanehiro. Biped gait control based on spatially quantized dynamics. In *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*, pages 75–81. IEEE, 2018. ISBN 978-1-5386-7283-9. doi: 10.1109/HUMANOIDS.2018.8624942. URL <https://ieeexplore.ieee.org/document/8624942/>.
- Masao Kanazawa, Shunichi Nozawa, Yohei Kakiuchi, Yoshiki Kanemoto, Mitsuhide Kuroda, Kei Okada, Masayuki Inaba, and Takahide Yoshiike. Robust vertical ladder climbing and transitioning between ladder and catwalk for humanoid robots. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2202–2209. IEEE, 2015. ISBN 978-1-4799-9994-1. doi: 10.1109/IROS.2015.7353672. URL <http://ieeexplore.ieee.org/document/7353672/>.
- N. Kanehira, T.U. Kawasaki, S. Ohta, T. Ismumi, T. Kawada, F. Kanehiro, S. Kajita, and K. Kaneko. Design and experiments of advanced leg module (HRP-2l) for humanoid robot (HRP-2) development. In *IEEE/RSJ International Conference on Intelligent Robots and System*, volume 3, pages 2455–2460. IEEE, 2002. ISBN 978-0-7803-7398-3. doi: 10.1109/IRDS.2002.1041636. URL <http://ieeexplore.ieee.org/document/1041636/>.

Kenji Kaneko, Hiroshi Kaminaga, Takeshi Sakaguchi, Shuuji Kajita, Mitsuharu Morisawa, Iori Kumagai, and Fumio Kanehiro. Humanoid robot HRP-5p: An electrically actuated humanoid robot with high-power and wide-range joints. *IEEE Robotics and Automation Letters*, 4(2):1431–1438, 2019. ISSN 2377-3766, 2377-3774. doi: 10.1109/LRA.2019.2896465. URL <https://ieeexplore.ieee.org/document/8630006/>.

Arvid QL Keemink, Herman van der Kooij, and Arno HA Stienen. Admittance control for physical human–robot interaction. *The International Journal of Robotics Research (IJRR)*, 37(11):1421–1444, 2018. ISSN 0278-3649, 1741-3176. doi: 10.1177/0278364918768950. URL <http://journals.sagepub.com/doi/10.1177/0278364918768950>.

Abderrahmane Kheddar, Maximo A. Roa, Pierre-Brice Wieber, Francois Chaumette, Fabien Spindler, Giuseppe Oriolo, Leonardo Lanari, Adrien Escande, Kevin Chappellet, Fumio Kanehiro, Patrice Rabate, Stephane Caron, Pierre Gergondet, Andrew Comport, Arnaud Tanguy, Christian Ott, Bernd Henze, George Mesesan, and Johannes Engelsberger. Humanoid robots in aircraft manufacturing: The airbus use cases. *IEEE Robotics & Automation Magazine*, 26(4):30–45, 2019. ISSN 1070-9932, 1558-223X. doi: 10.1109/MRA.2019.2943395. URL <https://ieeexplore.ieee.org/document/8889461/>.

Jung-Yup Kim, Ill-Woo Park, and Jun-Ho Oh. Realization of dynamic stair climbing for biped humanoid robot using force/torque sensors. *Journal of Intelligent and Robotic Systems*, 56(4):389–423, 2009. ISSN 0921-0296, 1573-0409. doi: 10.1007/s10846-009-9324-z. URL <http://link.springer.com/10.1007/s10846-009-9324-z>.

K. Kotaka, B. Ugurlu, M. Kawanishi, and T. Narikiyo. Prototype development and real-time trot-running implementation of a quadruped robot: RoboCat-1. In *2013 IEEE International Conference on Mechatronics (ICM)*, pages 604–609. IEEE, 2013. ISBN 978-1-4673-1388-9 978-1-4673-1386-5 978-1-4673-1387-2. doi: 10.1109/ICMECH.2013.6519111. URL <http://ieeexplore.ieee.org/document/6519111/>.

M. Kudruss, M. Naveau, O. Stasse, N. Mansard, C. Kirches, P. Soueres, and K. Mombaur. Optimal control for whole-body motion generation using center-of-mass dynamics for predefined multi-contact configurations. In *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pages 684–689. IEEE, 2015. ISBN 978-1-4799-6885-5. doi: 10.1109/HUMANOIDS.2015.7363428. URL <http://ieeexplore.ieee.org/document/7363428/>.

Scott Kuindersma, Frank Permenter, and Russ Tedrake. An efficiently solvable quadratic program for stabilizing dynamic locomotion. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2589–2594. IEEE, 2014. ISBN 978-1-4799-3685-4. doi: 10.1109/ICRA.2014.6907230. URL <http://ieeexplore.ieee.org/document/6907230/>.

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*, 40(3):429–455, 2016. ISSN 0929-5593, 1573-7527. doi: 10.1007/s10514-015-9479-3. URL <http://link.springer.com/10.1007/s10514-015-9479-3>.

Iori Kumagai, Mitsuharu Morisawa, Shizuko Hattori, Mehdi Benallegue, and Fumio Kanehiro. Multi-contact locomotion planning for humanoid robot based on sustainable contact graph with local contact modification. *IEEE Robotics and Automation Letters*, 5(4):6379–6387, 2020. ISSN 2377-3766, 2377-3774. doi: 10.1109/LRA.2020.3013843. URL <https://ieeexplore.ieee.org/document/9158392/>.

Iori Kumagai, Masaki Murooka, Mitsuharu Morisawa, and Fumio Kanehiro. Multi-contact locomotion planning with bilateral contact forces considering kinematics and statics during contact transition. *IEEE Robotics and Automation Letters*, 6(4):6654–6661, 2021. ISSN 2377-3766, 2377-3774. doi: 10.1109/LRA.2021.3095517. URL <https://ieeexplore.ieee.org/document/9478184/>.

H. M. Lankarani and P. E. Nikravesh. A contact force model with hysteresis damping for impact analysis of multibody systems. *Journal of Mechanical Design*, 112(3):369–376, 1990. ISSN 1050-0472, 1528-9001. doi: 10.1115/1.2912617. URL <https://asmedigitalcollection.asme.org/mechanicaldesign/article/112/3/369/429554/A-Contact-Force-Model-With-Hysteresis-Damping-for>.

Cecilia Laschi, Barbara Mazzolai, and Matteo Cianchetti. Soft robotics: Technologies and systems pushing the boundaries of robot abilities. *Science Robotics*, 1(1):eaah3690, 2016. ISSN 2470-9476. doi: 10.1126/scirobotics.aah3690. URL <https://www.science.org/doi/10.1126/scirobotics.aah3690>.

Alexandre Lecours, Boris Mayer-St-Onge, and Clement Gosselin. Variable admittance control of a four-degree-of-freedom intelligent assist device. In *2012 IEEE International Conference*

- on *Robotics and Automation*, pages 3903–3908. IEEE, 2012. ISBN 978-1-4673-1405-3 978-1-4673-1403-9 978-1-4673-1578-4 978-1-4673-1404-6. doi: 10.1109/ICRA.2012.6224586. URL <http://ieeexplore.ieee.org/document/6224586/>.
- Qingqing Li, Zhangguo Yu, Xuechao Chen, Qinqin Zhou, Weimin Zhang, Libo Meng, and Qiang Huang. Contact force/torque control based on viscoelastic model for stable bipedal walking on indefinite uneven terrain. *IEEE Transactions on Automation Science and Engineering*, 16(4):1627–1639, 2019. ISSN 1545-5955, 1558-3783. doi: 10.1109/TASE.2019.2903564. URL <https://ieeexplore.ieee.org/document/8672785/>.
- Yibin Li, Bin Li, Jiahong Ruan, and Xuewen Rong. Research of mammal bionic quadruped robots: a review. In *2011 IEEE 5th International Conference on Robotics, Automation and Mechatronics (RAM)*, page 6, 2011.
- Zhibin Li, Nikos G. Tsagarakis, and Darwin G. Caldwell. Walking pattern generation for a humanoid robot with compliant joints. *Autonomous Robots*, 35(1):1–14, 2013. ISSN 0929-5593, 1573-7527. doi: 10.1007/s10514-013-9330-7. URL <http://link.springer.com/10.1007/s10514-013-9330-7>.
- Zhongyu Li, Xuxin Cheng, Xue Bin Peng, Pieter Abbeel, Sergey Levine, Glen Berseth, and Koushil Sreenath. Reinforcement learning for robust parameterized locomotion control of bipedal robots. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2811–2817. IEEE, 2021. ISBN 978-1-72819-077-8. doi: 10.1109/ICRA48506.2021.9560769. URL <https://ieeexplore.ieee.org/document/9560769/>.
- Mingxing Liu and Vincent Padois. Reactive whole-body control for humanoid balancing on non-rigid unilateral contacts. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3981–3987. IEEE, 2015. ISBN 978-1-4799-9994-1. doi: 10.1109/IROS.2015.7353938. URL <http://ieeexplore.ieee.org/document/7353938/>.
- Sean Mason, Ludovic Righetti, and Stefan Schaal. Full dynamics LQR control of a humanoid robot: An experimental study on balancing and squatting. In *2014 IEEE-RAS International Conference on Humanoid Robots*, pages 374–379. IEEE, 2014. ISBN 978-1-4799-7174-9. doi: 10.1109/HUMANOIDS.2014.7041387. URL <http://ieeexplore.ieee.org/document/7041387/>.
- Sean Mason, Nicholas Rotella, Stefan Schaal, and Ludovic Righetti. Balancing and walking using full dynamics LQR control with contact constraints. In *2016 IEEE-RAS*

16th International Conference on Humanoid Robots (Humanoids), pages 63–68. IEEE, 2016. ISBN 978-1-5090-4718-5. doi: 10.1109/HUMANOIDS.2016.7803255. URL <http://ieeexplore.ieee.org/document/7803255/>.

George Mesesan, Johannes Engelsberger, Gianluca Garofalo, Christian Ott, and Alin Albu-Schaffer. Dynamic walking on compliant and uneven terrain using DCM and passivity-based whole-body control. In *2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids)*, pages 25–32. IEEE, 2019. ISBN 978-1-5386-7630-1. doi: 10.1109/Humanoids43949.2019.9035053. URL <https://ieeexplore.ieee.org/document/9035053/>.

Alexis Mifsud, Mehdi Benallegue, and Florent Lamiroux. Estimation of contact forces and floating base kinematics of a humanoid robot using only inertial measurement units. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3374–3379. IEEE, 2015. ISBN 978-1-4799-9994-1. doi: 10.1109/IROS.2015.7353847. URL <http://ieeexplore.ieee.org/document/7353847/>.

Alexis Mifsud, Mehdi Benallegue, and Florent Lamiroux. Stabilization of a compliant humanoid robot using only inertial measurement units with a viscoelastic reaction mass pendulum model. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5405–5410. IEEE, 2016. ISBN 978-1-5090-3762-9. doi: 10.1109/IROS.2016.7759795. URL <http://ieeexplore.ieee.org/document/7759795/>.

Mitsuharu Morisawa, Fumio Kanehiro, Kenji Kaneko, Shuuji Kajita, and Kazuhiro Yokoi. Reactive biped walking control for a collision of a swinging foot on uneven terrain. In *2011 11th IEEE-RAS International Conference on Humanoid Robots*, pages 768–773. IEEE, 2011. ISBN 978-1-61284-868-6 978-1-61284-866-2 978-1-61284-867-9. doi: 10.1109/Humanoids.2011.6100885. URL <http://ieeexplore.ieee.org/document/6100885/>.

Mitsuharu Morisawa, Shuuji Kajita, Fumio Kanehiro, Kenji Kaneko, Kanako Miura, and Kazuhiro Yokoi. Balance control based on capture point error compensation for biped walking on uneven terrain. In *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)*, pages 734–740. IEEE, 2012. ISBN 978-1-4673-1369-8. doi: 10.1109/HUMANOIDS.2012.6651601. URL <http://ieeexplore.ieee.org/document/6651601/>.

Mitsuharu Morisawa, Rafael Cisneros, Mehdi Benallegue, Iori Kumagai, Adrien Escande, and Fumio Kanehiro. Online 3d CoM trajectory generation for multi-contact locomotion.

- tion synchronizing contact. In *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*, pages 1–8. IEEE, 2018. ISBN 978-1-5386-7283-9. doi: 10.1109/HUMANOIDS.2018.8625014. URL <https://ieeexplore.ieee.org/document/8625014/>.
- Mitsuharu Morisawa, Mehdi Benallegue, Rafael Cisneros, Iori Kumagai, Adrien Escande, Kenji Kaneko, and Fumio Kanehiro. Multi-contact stabilization of a humanoid robot for realizing dynamic contact transitions on non-coplanar surfaces. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2252–2258. IEEE, 2019. ISBN 978-1-72814-004-9. doi: 10.1109/IROS40897.2019.8968059. URL <https://ieeexplore.ieee.org/document/8968059/>.
- Masaki Murooka, Shunichi Nozawa, Yohei Kakiuchi, Kei Okada, and Masayuki Inaba. Whole-body pushing manipulation with contact posture planning of large and heavy object for humanoid robot. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5682–5689. IEEE, 2015. ISBN 978-1-4799-6923-4. doi: 10.1109/ICRA.2015.7139995. URL <http://ieeexplore.ieee.org/document/7139995/>.
- Masaki Murooka, Ryohei Ueda, Shunichi Nozawa, Yohei Kakiuchi, Kei Okada, and Masayuki Inaba. Global planning of whole-body manipulation by humanoid robot based on transition graph of object motion and contact switching. *Advanced Robotics*, 31(6):322–340, 2017. ISSN 0169-1864, 1568-5535. doi: 10.1080/01691864.2016.1266965. URL <https://www.tandfonline.com/doi/full/10.1080/01691864.2016.1266965>.
- Masaki Murooka, Shunichi Nozawa, Masahiro Bando, Iori Yanokura, Kei Okada, and Masayuki Inaba. Simultaneous planning and estimation based on physics reasoning in robot manipulation. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3137–3144. IEEE, 2018. ISBN 978-1-5386-3081-5. doi: 10.1109/ICRA.2018.8463156. URL <https://ieeexplore.ieee.org/document/8463156/>.
- Masaki Murooka, Kevin Chappellet, Arnaud Tanguy, Mehdi Benallegue, Iori Kumagai, Mitsuharu Morisawa, Fumio Kanehiro, and Abderrahmane Kheddar. Humanoid locomanipulations pattern generation and stabilization control. *IEEE Robotics and Automation Letters*, 6(3):5597–5604, 2021. ISSN 2377-3766, 2377-3774. doi: 10.1109/LRA.2021.3077858. URL <https://ieeexplore.ieee.org/document/9424385/>.
- Masaki Murooka, Mitsuharu Morisawa, and Fumio Kanehiro. Centroidal trajectory generation and stabilization based on preview control for humanoid multi-contact motion. *IEEE*

Robotics and Automation Letters, 7(3):8225–8232, 2022. ISSN 2377-3766, 2377-3774. doi: 10.1109/LRA.2022.3186515. URL <https://ieeexplore.ieee.org/document/9807369/>.

Richard M. Murray, Zexiang Li, and S. Shankar Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1 edition, 2017. ISBN 978-1-315-13637-0. doi: 10.1201/9781315136370. URL <https://www.taylorfrancis.com/books/9781351469791>.

Shin’ichiro Nakaoka. Choreonoid: Extensible virtual robot environment built on an integrated GUI framework. In *2012 IEEE/SICE International Symposium on System Integration (SII)*, pages 79–85. IEEE, 2012. ISBN 978-1-4673-1497-8 978-1-4673-1496-1 978-1-4673-1495-4. doi: 10.1109/SII.2012.6427350. URL <http://ieeexplore.ieee.org/document/6427350/>.

Shin’ichiro Nakaoka, Mitsuharu Morisawa, Kenji Kaneko, Shuuji Kajita, Fumio Kanehiro, and Eiichi Yoshida. Indirect-type teleoperation interface for humanoid robot based on choreonoid framework. In *2014 IEEE/SICE International Symposium on System Integration*, page 2, 2014.

Michael Neunert, Markus Stauble, Markus Gifftthaler, Carmine D. Bellicoso, Jan Carius, Christian Gehring, Marco Hutter, and Jonas Buchli. Whole-body nonlinear model predictive control through contacts for quadrupeds. *IEEE Robotics and Automation Letters*, 3(3):1458–1465, 2018. ISSN 2377-3766, 2377-3774. doi: 10.1109/LRA.2018.2800124. URL <https://ieeexplore.ieee.org/document/8276298/>.

Koichi Nishiwaki and Satoshi Kagami. Walking control on uneven terrain with short cycle pattern generation. In *2007 7th IEEE-RAS International Conference on Humanoid Robots*, pages 447–453. IEEE, 2007. ISBN 978-1-4244-1861-9. doi: 10.1109/ICHR.2007.4813908. URL <http://ieeexplore.ieee.org/document/4813908/>.

Shunichi Nozawa, Yohei Kakiuchi, Kei Okada, and Masayuki Inaba. Controlling the planar motion of a heavy object by pushing with a humanoid robot using dual-arm force control. In *2012 IEEE International Conference on Robotics and Automation*, pages 1428–1435. IEEE, 2012. ISBN 978-1-4673-1405-3 978-1-4673-1403-9 978-1-4673-1578-4 978-1-4673-1404-6. doi: 10.1109/ICRA.2012.6224884. URL <http://ieeexplore.ieee.org/document/6224884/>.

Vladislav Okunev, Thomas Nierhoff, and Sandra Hirche. Human-preference-based control design: Adaptive robot admittance control for physical human-robot interaction. In *2012*

- IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication*, pages 443–448. IEEE, 2012. ISBN 978-1-4673-4606-1 978-1-4673-4604-7 978-1-4673-4605-4. doi: 10.1109/ROMAN.2012.6343792. URL <http://ieeexplore.ieee.org/document/6343792/>.
- David E. Orin, Ambarish Goswami, and Sung-Hee Lee. Centroidal dynamics of a humanoid robot. *Autonomous Robots*, 35(2):161–176, 2013. ISSN 0929-5593, 1573-7527. doi: 10.1007/s10514-013-9341-4. URL <http://link.springer.com/10.1007/s10514-013-9341-4>.
- C. Ott, A. Albu-Schaffer, A. Kugi, S. Stamigioli, and G. Hirzinger. A passivity based cartesian impedance controller for flexible joint robots - part i: torque feedback and gravity compensation. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, pages 2659–2665 Vol.3. IEEE, 2004. ISBN 978-0-7803-8232-9. doi: 10.1109/ROBOT.2004.1307462. URL <http://ieeexplore.ieee.org/document/1307462/>.
- Christian Ott, Ranjan Mukherjee, and Yoshihiko Nakamura. Unified impedance and admittance control. In *2010 IEEE International Conference on Robotics and Automation*, pages 554–561. IEEE, 2010. ISBN 978-1-4244-5038-1. doi: 10.1109/ROBOT.2010.5509861. URL <http://ieeexplore.ieee.org/document/5509861/>.
- Federico Parretti and Hartmut Geyer. Reactive balance control in walking based on a bipedal linear inverted pendulum model. In *2011 IEEE International Conference on Robotics and Automation*, pages 5442–5447. IEEE, 2011. ISBN 978-1-61284-386-5. doi: 10.1109/ICRA.2011.5980225. URL <http://ieeexplore.ieee.org/document/5980225/>.
- Matteo Parigi Polverini, Davide Nicolis, Andrea Maria Zanchettin, and Paolo Rocco. Robust set invariance for implicit robot force control in presence of contact model uncertainty. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6393–6399. IEEE, 2017. ISBN 978-1-5386-2682-5. doi: 10.1109/IROS.2017.8206544. URL <http://ieeexplore.ieee.org/document/8206544/>.
- Brahayam Ponton, Alexander Herzog, Stefan Schaal, and Ludovic Righetti. A convex model of humanoid momentum dynamics for multi-contact motion generation. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 842–849. IEEE, 2016. ISBN 978-1-5090-4718-5. doi: 10.1109/HUMANOIDS.2016.7803371. URL <http://ieeexplore.ieee.org/document/7803371/>.

Michael Posa, Scott Kuindersma, and Russ Tedrake. Optimization and stabilization of trajectories for constrained dynamical systems. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1366–1373. IEEE, 2016. ISBN 978-1-4673-8026-3. doi: 10.1109/ICRA.2016.7487270. URL <http://ieeexplore.ieee.org/document/7487270/>.

I. Poulakakis, E. Papadopoulos, and M. Buehler. On the stable passive dynamics of quadrupedal running. In *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, volume 1, pages 1368–1373. IEEE, 2003. ISBN 978-0-7803-7736-3. doi: 10.1109/ROBOT.2003.1241782. URL <http://ieeexplore.ieee.org/document/1241782/>.

Ioannis Poulakakis, James Andrew Smith, and Martin Buehler. Modeling and experiments of untethered quadrupedal running with a bounding gait: The scout II robot. *The International Journal of Robotics Research (IJRR)*, 24(4):239–256, 2005. ISSN 0278-3649, 1741-3176. doi: 10.1177/0278364904050917. URL <http://journals.sagepub.com/doi/10.1177/0278364904050917>.

Ioannis Poulakakis, Evangelos Papadopoulos, and Martin Buehler. On the stability of the passive dynamics of quadrupedal running with a bounding gait. *The International Journal of Robotics Research (IJRR)*, 25(7):669–687, 2006. ISSN 0278-3649, 1741-3176. doi: 10.1177/0278364906066768. URL <http://journals.sagepub.com/doi/10.1177/0278364906066768>.

Qiang Huang, K. Kaneko, K. Yokoi, S. Kajita, T. Kotoku, N. Koyachi, H. Arai, N. Imamura, K. Komoriya, and K. Tanie. Balance control of a piped robot combining off-line pattern with real-time modification. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, volume 4, pages 3346–3352. IEEE, 2000. ISBN 978-0-7803-5886-7. doi: 10.1109/ROBOT.2000.845227. URL <http://ieeexplore.ieee.org/document/845227/>.

M. Raibert, M. Chepponis, and H. Brown. Running on four legs as though they were one. *IEEE Journal on Robotics and Automation*, 2(2):70–82, 1986. ISSN 0882-4967. doi: 10.1109/JRA.1986.1087044. URL <http://ieeexplore.ieee.org/document/1087044/>.

Marc Raibert, Kevin Blankespoor, Gabriel Nelson, and Rob Playter. BigDog, the rough-terrain quadruped robot. *IFAC Proceedings Volumes*, 41(2):10822–10825, 2008.

- ISSN 14746670. doi: 10.3182/20080706-5-KR-1001.01833. URL <https://linkinghub.elsevier.com/retrieve/pii/S1474667016407020>.
- Noelie Ramuzat. *Robotics Force/Torque Control for Manufacturing Operations*. Theses, INSA de Toulouse, February 2022. URL <https://theses.hal.science/tel-03675191>.
- Giulio Romualdi, Stefano Daffarra, and Daniele Pucci. Modeling of visco-elastic environments for humanoid robot motion control. *IEEE Robotics and Automation Letters*, 6(3):4289–4296, 2021. ISSN 2377-3766, 2377-3774. doi: 10.1109/LRA.2021.3067589. URL <https://ieeexplore.ieee.org/document/9382068/>.
- Julien Roux, Saeid Samadi, Eisoku Kuroiwa, Takahide Yoshiike, and Abderrahmane Kheddar. Control of humanoid in multiple fixed and moving unilateral contacts. In *2021 20th International Conference on Advanced Robotics (ICAR)*, pages 793–799. IEEE, 2021. ISBN 978-1-66543-684-7. doi: 10.1109/ICAR53236.2021.9659359. URL <https://ieeexplore.ieee.org/document/9659359/>.
- Daniela Rus and Michael T. Tolley. Design, fabrication and control of soft robots. *Nature*, 521(7553):467–475, 2015. ISSN 0028-0836, 1476-4687. doi: 10.1038/nature14543. URL <http://www.nature.com/articles/nature14543>.
- Saeid Samadi. *Humanoid robots balance in multi-contact modes settings*. Theses, Université Montpellier, December 2021. URL <https://theses.hal.science/tel-03629770>.
- Saeid Samadi, Stephane Caron, Arnaud Tanguy, and Abderrahmane Kheddar. Balance of humanoid robots in a mix of fixed and sliding multi-contact scenarios. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6590–6596. IEEE, 2020. ISBN 978-1-72817-395-5. doi: 10.1109/ICRA40945.2020.9197253. URL <https://ieeexplore.ieee.org/document/9197253/>.
- Saeid Samadi, Julien Roux, Arnaud Tanguy, Stephane Caron, and Abderrahmane Kheddar. Humanoid control under interchangeable fixed and sliding unilateral contacts. *IEEE Robotics and Automation Letters*, 6(2):4032–4039, 2021. ISSN 2377-3766, 2377-3774. doi: 10.1109/LRA.2021.3066965. URL <https://ieeexplore.ieee.org/document/9380971/>.
- Christopher Schindlbeck and Sami Haddadin. Unified passivity-based cartesian force/impedance control for rigid and flexible joint robots via task-energy tanks. In

- 2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 440–447. IEEE, 2015. ISBN 978-1-4799-6923-4. doi: 10.1109/ICRA.2015.7139036. URL <http://ieeexplore.ieee.org/document/7139036/>.
- Sangok Seok, Albert Wang, Meng Yee Chuah, Dong Jin Hyun, Jongwoo Lee, David M. Otten, Jeffrey H. Lang, and Sangbae Kim. Design principles for energy-efficient legged locomotion and implementation on the MIT cheetah robot. *IEEE/ASME Transactions on Mechatronics*, 20(3):1117–1129, 2015. ISSN 1083-4435, 1941-014X. doi: 10.1109/TMECH.2014.2339013. URL <https://ieeexplore.ieee.org/document/6880316/>.
- H. Seraji. Adaptive admittance control: an approach to explicit force control in compliant motion. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 2705–2712. IEEE Comput. Soc. Press, 1994. ISBN 978-0-8186-5330-8. doi: 10.1109/ROBOT.1994.350927. URL <http://ieeexplore.ieee.org/document/350927/>.
- Paarth Shah, Avadesh Meduri, Wolfgang Merkt, Majid Khadiv, Ioannis Havoutis, and Ludovic Righetti. Rapid convex optimization of centroidal dynamics using block coordinate descent, 2021. URL <http://arxiv.org/abs/2108.01797>.
- Abdel-Nasser Sharkawy and Panagiotis N. Koustoumpardis. Human–robot interaction: A review and analysis on variable admittance control, safety, and perspectives. *Machines*, 10(7):591, 2022. ISSN 2075-1702. doi: 10.3390/machines10070591. URL <https://www.mdpi.com/2075-1702/10/7/591>.
- Satoshi Shigemi. ASIMO and humanoid robot research at honda. In Ambarish Goswami and Prahlad Vadakkepat, editors, *Humanoid Robotics: A Reference*, pages 55–90. Springer Netherlands, 2019. ISBN 978-94-007-6045-5 978-94-007-6046-2. doi: 10.1007/978-94-007-6046-2_9. URL http://link.springer.com/10.1007/978-94-007-6046-2_9.
- Rohan P. Singh, Pierre Gergondet, and Fumio Kanehiro. mc-mujoco: Simulating articulated robots with FSM controllers in MuJoCo, 2022. URL <http://arxiv.org/abs/2209.00274>.
- Benjamin J. Stephens and Christopher G. Atkeson. Push recovery by stepping for humanoid robots with force controlled joints. In *2010 10th IEEE-RAS International Conference on Humanoid Robots*, pages 52–59. IEEE, 2010. ISBN 978-1-4244-8688-5. doi: 10.1109/ICHR.2010.5686288. URL <http://ieeexplore.ieee.org/document/5686288/>.

- Sugihara T. Whole-body cooperative balancing of humanoid robot using COG jacobian. In *International Conference on Intelligent Robots and Systems*, page 6, 2002.
- Toru Takenaka, Takashi Matsumoto, Takahide Yoshiike, Tadaaki Hasegawa, Shinya Shirokura, Hiroyuki Kaneko, and Atsuo Orita. Real time motion generation and control for biped robot -4th report: Integrated balance control-. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1601–1608. IEEE, 2009. ISBN 978-1-4244-3803-7. doi: 10.1109/IROS.2009.5354522. URL <http://ieeexplore.ieee.org/document/5354522/>.
- T. Takubo, K. Inoue, and T. Arai. Pushing an object considering the hand reflect forces by humanoid robot in dynamic walking. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 1706–1711. IEEE, 2005. ISBN 978-0-7803-8914-4. doi: 10.1109/ROBOT.2005.1570359. URL <http://ieeexplore.ieee.org/document/1570359/>.
- Barry Trimmer. Humanoids and the emergence of soft robotics. *Soft Robotics*, 2(4):129–130, 2015. ISSN 2169-5172, 2169-5180. doi: 10.1089/soro.2015.29005.bat. URL <https://www.liebertpub.com/doi/10.1089/soro.2015.29005.bat>.
- Nikos G. Tsagarakis, Zhibin Li, Jody Saglia, and Darwin G. Caldwell. The design of the lower body of the compliant humanoid robot ‘cCub’. In *2011 IEEE International Conference on Robotics and Automation*, pages 2035–2040. IEEE, 2011. ISBN 978-1-61284-386-5. doi: 10.1109/ICRA.2011.5980130. URL <http://ieeexplore.ieee.org/document/5980130/>.
- Barkan Ugurlu, Jody A. Saglia, Nikos G. Tsagarakis, and Darwin G. Caldwell. YAW MOMENT COMPENSATION FOR BIPEDAL ROBOTS VIA INTRINSIC ANGULAR MOMENTUM CONSTRAINT. In *International Journal of Humanoid Robotics*, volume 09, page 1250033, 2012. doi: 10.1142/S0219843612500338. URL <https://www.worldscientific.com/doi/abs/10.1142/S0219843612500338>.
- Joris Vaillant, Abderrahmane Kheddar, Hervé Audren, François Keith, Stanislas Brossette, Adrien Escande, Karim Bouyarmane, Kenji Kaneko, Mitsuharu Morisawa, Pierre Gergondet, Eiichi Yoshida, Suuji Kajita, and Fumio Kanehiro. Multi-contact vertical ladder climbing with an HRP-2 humanoid. *Autonomous Robots*, 40(3):561–580, 2016. ISSN 0929-5593, 1573-7527. doi: 10.1007/s10514-016-9546-4. URL <http://link.springer.com/10.1007/s10514-016-9546-4>.

Andres K. Valenzuela and Sangbae Kim. Optimally scaled hip-force planning: A control approach for quadrupedal running. In *2012 IEEE International Conference on Robotics and Automation*, pages 1901–1907. IEEE, 2012. ISBN 978-1-4673-1405-3 978-1-4673-1403-9 978-1-4673-1578-4 978-1-4673-1404-6. doi: 10.1109/ICRA.2012.6225251. URL <http://ieeexplore.ieee.org/document/6225251/>.

Bram Vanderborght, Nikos G. Tsagarakis, Ronald Van Ham, Ivar Thorson, and Darwin G. Caldwell. MACCEPA 2.0: compliant actuator used for energy efficient hopping robot chobino1d. *Autonomous Robots*, 31(1):55–65, 2011. ISSN 0929-5593, 1573-7527. doi: 10.1007/s10514-011-9230-7. URL <http://link.springer.com/10.1007/s10514-011-9230-7>.

Vasileios Vasilopoulos, Iosif S. Paraskevas, and Evangelos G. Papadopoulos. Monopod hopping on compliant terrains. *Robotics and Autonomous Systems*, 102:13–26, 2018. ISSN 09218890. doi: 10.1016/j.robot.2018.01.004. URL <https://linkinghub.elsevier.com/retrieve/pii/S0921889017300659>.

Miomir Vukobratovic. How to control artificial anthropomorphic systems. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-3(5):497–507, 1973. ISSN 0018-9472, 2168-2909. doi: 10.1109/TSMC.1973.4309277. URL <http://ieeexplore.ieee.org/document/4309277/>.

Miomir Vukobratović and Branislav Borovac. ZERO-MOMENT POINT — THIRTY FIVE YEARS OF ITS LIFE. *International Journal of Humanoid Robotics (IJHR)*, 01(1):157–173, 2004. ISSN 0219-8436, 1793-6942. doi: 10.1142/S0219843604000083. URL <https://www.worldscientific.com/doi/abs/10.1142/S0219843604000083>.

Pierre-Brice Wieber, INRIA Rhone-Alpes, and St-Ismier Cedex. On the stability of walking systems. In *Proceedings of the International Workshop on Humanoid and Human Friendly Robotics*, page 7, 2002.

Pierre-Brice Wieber, Russ Tedrake, and Scott Kuindersma. Modeling and control of legged robots. In Bruno Siciliano and Oussama Khatib, editors, *Springer Handbook of Robotics*, pages 1203–1234. Springer International Publishing, 2016. ISBN 978-3-319-32550-7 978-3-319-32552-1. doi: 10.1007/978-3-319-32552-1_48. URL https://link.springer.com/10.1007/978-3-319-32552-1_48. Series Title: Springer Handbooks.

Georg Wiedebach, Sylvain Bertrand, Tingfan Wu, Luca Fiorio, Stephen McCrory, Robert Griffin, Francesco Nori, and Jerry Pratt. Walking on partial footholds including line contacts with the humanoid robot atlas. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 1312–1319. IEEE, 2016. ISBN 978-1-5090-4718-5. doi: 10.1109/HUMANOIDS.2016.7803439. URL <http://ieeexplore.ieee.org/document/7803439/>.

David A. Winter, Aftab E. Patla, Francois Prince, Milad Ishac, and Krystyna Gielo-Perczak. Stiffness control of balance in quiet standing. *Journal of Neurophysiology (JNP)*, 80(3): 1211–1221, 1998. ISSN 0022-3077, 1522-1598. doi: 10.1152/jn.1998.80.3.1211. URL <https://www.physiology.org/doi/10.1152/jn.1998.80.3.1211>.

J. Yamaguchi, A. Takanishi, and I. Kato. Experimental development of a foot mechanism with shock absorbing material for acquisition of landing surface position information and stabilization of dynamic biped walking. In *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, volume 3, pages 2892–2899. IEEE, 1995. ISBN 978-0-7803-1965-3. doi: 10.1109/ROBOT.1995.525694. URL <http://ieeexplore.ieee.org/document/525694/>.

J. Yamaguchi, E. Soga, S. Inoue, and A. Takanishi. Development of a bipedal humanoid robot-control method of whole body cooperative dynamic biped walking. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*, volume 1, pages 368–374. IEEE, 1999. ISBN 978-0-7803-5180-6. doi: 10.1109/ROBOT.1999.770006. URL <http://ieeexplore.ieee.org/document/770006/>.

Shunpeng Yang, Hua Chen, Zhen Fu, and Wei Zhang. Force-feedback based whole-body stabilizer for position-controlled humanoid robots. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7432–7439. IEEE, 2021. ISBN 978-1-66541-714-3. doi: 10.1109/IROS51168.2021.9636634. URL <https://ieeexplore.ieee.org/document/9636634/>.

Kazuhito Yokoi, Fumio Kanehiro, Kenji Kaneko, Shuuji Kajita, Kiyoshi Fujiwara, and Hirohisa Hirukawa. Experimental study of humanoid robot HRP-1s. *The International Journal of Robotics Research (IJRR)*, 23(4):351–362, 2004. ISSN 0278-3649, 1741-3176. doi: 10.1177/0278364904042194. URL <http://journals.sagepub.com/doi/10.1177/0278364904042194>.