THÈSE POUR OBTENIR LE GRADE DE DOCTEUR
DE L'UNIVERSITE DE MONTPELLIER

En Systèmes Automatiques et Microélectroniques

École doctorale : Information, Structures, Systèmes

Unité de recherche UMR5506

Contributions to balance and design of humanoid robots

Présentée par Julien Roux

Le 2 Décembre 2024

Sous la direction de Pr. Abderrahmane Kheddar

Devant le jury composé de

Damien Chablat, Professeur, Directeur de Recherche CNRS, LS2N, Nantes	Rapporteur
Féthi Ben Ouezdou, Professeur, Directeur du campus Arts et Métiers, Aix en Provence	Rapporteur
Olivier Stasse, Professeur, Directeur de Recherche CNRS, LAAS, Toulouse	Examinateur
Jean-Pierre Gazeau, Ing. de Recherche CNRS, HDR, Institut PPRIME, Université de Poitiers	Examinateur
Claire Dune, Maître de Conférence, Laboratoire COSMER, Université de Toulon	Examinateur
Abderrahmane Kheddar, Professeur, Directeur de Recherche, CNRS	Directeur de thèse
Hiroshi Kaminaga, Senior Researcher, AIST, Tsukuba	Invité

UNIVERSITÉ DE MONTPELLIER

Acknowledgements

Five years is a long time. I could not have done it without the feedback, input, and support of many people.

First, I thank Pr. Abderrahmane Kheddar, my thesis supervisor, for the many opportunities he has given me. Thanks to him, I was able to work with several world-renowned humanoid platforms, and I am deeply grateful for it. Joining the IDH team at LIRMM, where I met many wonderful people, was a pleasure. All his advice and comments on preparing high-quality scientific papers and his impulse to explore the capabilities of a robotic platform helped me develop many valuable skills.

Then, I would like to thank the jury members for their careful reading of my work and their kind remarks and feedback. I especially want to thank Damien Chablat and Hiroshi Kaminaga, who gave me feedback on my thesis progress earlier.

On a technical level, from software implementation to hardware development and integration, I received plenty of help from many engineers and research engineers. I would like to thank Arnaud Tanguy and Pierre Gergondet for all their help and advice on implementing all the controllers presented in this thesis. Also, I thank Olivier Tempier from LIRMM for his help fixing HRP-4 and for his advice when modifying the robot's hardware. Finally, I want to acknowledge the work of Jean-Baptiste Izard from Alted, who designed the end-effector changer presented in the later part of this thesis.

Many thanks to the coworkers and friends from the IDH team. First, Saeid Samadi, my "co-bureau" for the first two years, for all the nice discussions we had, and all the actuators we burnt. Niels Dehio and Yuquan Wang for all the discussion and insight they gave me and all the nice hiking sessions. Thanks to all the other Ph.D. students of the IDH team and from LIRMM, with whom I shared many wonderful moments and memories at the lab or outside. In particular, I want to thank Johan Medrano, Camille Mutschler, Celia Saghour, Antonin Dallard, Hugo Lefèvre, and Carole Fournier.

During the last two years of my thesis, I have been giving courses on mechanics and other topics at IUT de Nîmes. It was an interesting and enriching experience, and I would like to thank the teaching team for their welcome. In particular, I would like to thank Olivier Company and Simon Le Floch for all the exchanges, encouragement, and advice on teaching and their feedback on parts of my thesis they gave me.

Finally, thank you to my family, who were always asking for news on the progress of my thesis and offered their support many times.

Abstract

Humanoid robots have been the subject of numerous studies for several decades, and their large-scale deployment in industrial contexts has never been closer. In particular, their human-like form makes them perfect candidates for performing low-value tasks in environments that are difficult to access for other mobile platforms. Thus, we see them appearing in factories where contacts with their upper body limbs, often underutilized, are required and have significant consequences on balance. In this thesis, we address the issues of humanoid robotics related to multi-contact balance and the use of the robot's upper limbs. First, the multi-contact balance is studied through a centroidal model that allows for quick calculations. For this, a balance region in which the center of mass must be located is used. Its shape depends on the position of the robot's contacts as well as the tolerated accelerations of the center of mass. To use it online, a method that improves its calculation is proposed. It is then integrated into the control of the humanoid robot HRP-4. Then, several multi-contact balance approaches are compared to the previous method. In particular, an approximation of the equilibrium region of the center of mass by its Chebyshev center and the associated ball is studied.

In the second phase, the capabilities of humanoid robots are enhanced by integrating tool-changers. The robot will then be able to autonomously change its effectors, whether they are hands, feet, or even the head. A study of commercially available tool-changer models is being conducted. Two models are integrated into the humanoid platform HRP-4. We then establish the specifications for a tool-changer that is better suited to the constraints of a humanoid robot. Finally, based on these constraints, a new design for an end-effector changer is proposed. This design has variations aimed at every end-effector of the robots, including hands, feet, and head. It takes advantage of the dexterity of the humanoid robot to lock and unlock the effectors without needing to add extra actuators. The result is demonstrated again on the HRP-4 robot.

Keywords

Humanoid Robotics, Dynamic Balance, Humanoid Robot Design

RÉSUMÉ DE LA THÈSE

Les robots humanoïdes sont l'objet de nombreuses études depuis plusieurs décennies et leur déploiement à grande échelle dans des contextes industriels n'a jamais été aussi proche. En particulier, leur forme similaire à celle de l'humain en fait des candidats parfaits pour la réalisation de tâches à faible valeur ajoutée dans des environements difficiles d'accès pour d'autres plateformes mobiles. Ainsi, on les voit apparaitre dans les usines où les contacts avec les membres supérieurs, souvent non utilisés, sont requis et ont des conséquences importantes sur l'équilibre. Dans cette thèse, nous abordons les problématiques de la robotique humanoïde liées à l'équilibre multi-contacts et à l'utilisation des membres supérieurs du robot.

Tout d'abord l'équilibre multi-contacts est étudié au travers d'un modèle centroidal permettant des calculs rapides. Pour cela, une région d'équilibre dans laquelle doit se trouver le centre de masse est utilisée. Sa forme dépend de la position des contacts du robot ainsi que des accélérations du centre de masse tolérées. Afin de pouvoir l'utiliser en ligne, une méthode améliorant son calcul est proposée. Elle est alors intégrée au contrôle du robot humanoïde HRP-4.
Ensuite, plusieurs approches d'équilibre en multi-contacts sont comparées à la méthode précédente. En particulier, une approximation de la région d'équilibre du centre de masse par son centre de Chebyshev et la boule associée est étudiée.

Dans un second temps, les capacités des robots humanoïdes sont étendues en leur intégrant des échangeurs d'outils. Le robot pourra alors changer ses effecteurs de manière autonome, que ce soit les mains, les pieds ou encore la tête. Une étude des modèles d'échangeurs disponibles dans le commerce est réalisée. Deux modèles sont intégrés à la plateforme humanoïdes HRP-4. Nous établissons alors le cahier des charges d'un échangeur d'outils plus adapté aux contraintes d'un robot humanoïde.
Enfin, à partir de ces contraintes, une nouvelle conception d'échangeur d'outil est proposée. Elle profite de la dextérité du robot humanoïde pour verrouiller et déverrouiller les effecteurs sans nécessiter l'ajout d'actionneur supplémentaires. Le résultat est démontré de nouveau sur le robot HRP-4.

Contents

List of Figures **I**

List of Tables **VII**

Introduction **1**

1 Background and Used works **5**

1.1 Multi-Contact balance 5

1.1.1 General balance of humanoid 6

1.1.2 Multi-Contact balance 7

1.1.3 Center of Mass Region 8

1.2 Background on tool-changing technology 10

1.3 Other related litterature 14

1.3.1 Whole Body QP 14

1.3.2 Vision 17

1.3.3 Parameters estimation 17

1.3.4 Contact estimation 18

1.4 Conclusion 18

2 CoM Balance Region for Scenarios with Multiple Non Coplanar Fixed and Moving Contacts **21**

2.1 Centroidal Model and CoM Balance Region 22

2.2 Iterative Projection Algorithm 25

2.2.1 Linearization of the constraints 26

2.2.2 Support Point and Support Functions 27
2.2.3 Updating the inner and outer approximations 28
2.2.4 Performance Comparison 28
2.3 Integration in the Whole-Body Control 30
2.3.1 Whole Body Balance Framework 30
2.3.2 Formulation of a CoM Admittance Task 31
2.4 Experiments 32
2.4.1 Regulation of contact force 32
2.4.2 Box pushing in multi-contact 33
2.4.3 Evolution of the balance region 35
2.5 Conclusion 36

3 Optimal CoM and Wrench Distribution for Scenarios with Multiple Non Coplanar Sliding Contacts 39

3.1 Formulation of the Chebichev QP 40
3.1.1 Contact formulation 40
3.1.2 Centroidal QP 43
3.1.3 Chebichev center 44
3.2 Integration in the whole body QP 45
3.2.1 Whole body QP integration 46
3.2.2 Resulting experiments 47
3.3 Discussion 48
3.3.1 Interpretation of the Chebyshev radius 49
3.3.2 Comparison whith CoM balance region 50
3.3.3 Target CoM versus CoM region 53
3.4 Conclusion 54

4 End-Effector Changers for Humanoid Robots: Case Study and Limitation Definitions 55

4.1 Terminology 56
4.2 Choosing initial tool-changers 57
4.2.1 Manual Tool Changer 58
4.2.2 Automatic Tool Changer 59
4.3 Hardware integration 61
4.3.1 On Robot 61
4.3.2 Kosmek hardware interfaces 62
4.3.3 Electronic integration of the Kosmek TC 64
4.4 Behavior integration on HRP-4 65
4.4.1 Vision integration 66
4.4.2 Contact control 69
4.4.3 Model update 70
4.5 Limitations and design constraints 72
4.6 Conclusion 74

5 New Design of an End-Effector Changer for Humanoids 75
5.1 Design of the end-effector changer 75
5.1.1 Mechanism and operation phases 76
5.1.2 EEC Variations 78
5.1.3 Stress evaluation 79
5.2 Evaluation and experiments on a humanoid HRP-4 81
5.2.1 Mechatronics integration and mounting 81
5.2.2 Arms' end-effectors changing 83
5.2.3 Feet changing 84
5.2.4 Head changing 90
5.3 Discussion 90
5.4 Conclusion 93

List of Figures

1.1 ©2011 IEEE Tool-changer from Gyimothy and Toth (2011) used in the ACROBOTER project. Left and midle CAD of the TC; Right: prototype seen from below . . . 11

1.2 ©2018 IEEE Tool-changer from Berenstein et al. (2018) (a) CAD of tool side; (b) CAD of robot side with Fetch arm adapter; (c) CAD of tool station; (d) TC prototype mounted on Fetch robotic platform with simple tool; (e) Tool station with simple tool stored in. . . . 11

1.3 ©2019 IEEE Tool-changer from Pettinger et al. (2019). Left: wheeled robotic plateform with robotic arm used on which the TC will be installed; Right: 3 steps to store a tool in the tool station. . . . 12

1.4 ©2022 IEEE Tool-changer from Li et al. (2022). The tool station is mounted in the Freight robotic platform (left) and the tool-changer is used on the Fetch robotic plateform (right) . . . 13

1.5 ©2011 IEEE - Humanoid Robotic Platform 4 (HRP-4) as presented in Kaneko et al. (2011) . . . 15

1.6 Schematics of the WBQP . . . 16

2.1 HRP-4 pushing a 12.5 kg box. It produced a force of up to 30 N horizontally with its left hand while controlling the remaining fixed contacts during the pushing phase. . . . 22

2.2 Schematic of the centroidal model of the robot . . . 22

2.3 The balance CoM region for a robot in a contact configuration similar to Fig. 2.1. The CoM acceleration is bounded to ± 0.4 m.s^{-2} in x-axis, ± 0.3 m.s^{-2} in y-axis and -9.81 ± 0.3 m.s^{-2} in z-axis. The black arrows are the contact used for the computation, each one of them is a vertex of a contact surface. The region is truncated at $0 \leq z \leq 2$ m . . . 24

2.4 Illustration of the projection algorithm for the static case with initial state and the following two iteration (One per line). In dotted black lines is $\mathcal{Y}$, in red is $\mathcal{Y}_{\text{inner}}$ and in green is $\mathcal{Y}_{\text{outer}}$. . . . 25

2.5 Illustration of the measures used to choose the next cutting direction. The regions corresponds to the cuts of $\mathcal{Y}_{\text{outer}}$ with the different faces H_i of $\mathcal{Y}_{\text{inner}}$. . 27

2.6 Impact of linearization; Top: evolution of the volume of the approximation w.r.t. the number of iteration of the projection algorithm for different number of sides for the linearization of the friction cone. Bottom: Evolution of the computation time w.r.t. the number of friction sides for the linearization of the friction cone 29

2.7 Tasks and constraints are formulated as function of the generalized acceleration coordinates $\ddot{\mathbf{q}}$ and contact force λ. The targets of the tasks are given by a FSM or the WBQP's output from previous iteration and are compared to the estimated state of the robot. Using the estimated contact position the balance region is computed and used as constraints. 30

2.8 Right-hand force regulation. HRP-4 right hand is on a 70 cm high horizontal table around 40 cm in front of the standing robot and off centered by 30 cm to the right. The left hand is not in contact. 33

2.9 Box pushing in multi-contact (Fig. 2.1): the right hand is in the same position as the first experiment (Fig. 2.8); the left hand is in contact with a box on the table. When the contact is established, the hand is 85 cm high, 40 cm in front of the robot and 30 cm to its left. The box is 12.5 kg, and during motion, it moves 15 cm forward. 34

2.10 The same experiment as Fig. 2.9 with a box mass of 7.5 kg: the target force on the left hand is not reached (since the box can be moved with a lower force); thus, the CoM keeps moving forward until the target is decreased to remove the contact. 35

2.11 Evolution of the balance region when adding the right-hand contact with the table and increasing the normal force target. For the computation of the balance region, the contact normal force for each contact is bounded. Here, only the bounds on the right-hand contact are changing. From left to right: $[0.0, 0.0]$ [N](no right-hand contact), $[5e-4, 1e-2]$ [N], $[7.0, 20.5]$ [N], $[15.7, 47.2]$ [N], $[22.0, 66.5]$ [N], $[25.0, 75.0]$ [N](transition finished) 36

3.1 Representation of the friction inequalities for a rectangular planar surface according to Caron et al. (2015b). Left the surface contact is modeled with punctual contact at its corner. Right: the contact friction boundaries are expressed as the admissible wrenches expressed at the origin of the surface. . . 40

3.2 Integration of the Chebyshev QP to the WBQP 46

3.3 Wiping and shuffling demonstrations 48

3.4 Hands normal forces during the co-wiping demonstration (As the hand's contact surfaces are non-coplanar, the normal are considered in the local frames) 49

3.5 Hands and shuffled foot normal forces during the shuffling demonstration (As the hands' contact surfaces are non-coplanar, the normal are considered in the local frames) 50

3.6 CoM trajectory projected in the horizontal plane during the co-wiping (top) and shuffling (bottom) experiments 51

3.7 Plot of different CoM balance regions during the co-wiping motion of HRP-4 (Fig. 3.3a). Region A is the static balance region from Chap. 2. Region B is the static balance region using the fixed contact equations presented in Sec. 3.1.1. Region C (blue) is the static balance with the wrenches constrained to the Chebyshev sphere. The red circle is the Chebyshev CoM sphere with a radius defined in Eq. 3.53. 52

4.1 Words used to describe the steps of the tool-change. The top and bottom arrows represent the full sequence. The word *free* stands for not in contact with the tool station as this contact restrains end-effector motions. 57

4.2 From left to right: (a) *Schunk* SHS manual change system with a payload of up to 11 kg and up to 6 pneumatic feed-throughs depending on the variant of the model; (b) *Kosmek* SXR manual robotic hand changer with a payload of up to 10 kg; (c) *Stäubli* SR series with a lifting force of up to 1000N and up to 10 pneumatic feed-throughs. 58

4.3 *OnRobot* TC with a rated payload of 30 kg 59

4.4 Set of automatic tool-changers with similar payload and connectivity options. All of them are actuated by air. From left to right: Stäubli MPS series, Shcunk SWS series, Destaco RHC series, and ATI QC series. 59

4.5 Set of automatic tool-changers with similar payload and connectivity options. From left to right: Triple A wingman, smartshift ATC, Kosmek. 60

4.6 Kosmek tool-changer, from left to right: robot and tool side engaged together but not locked, tool side, robot side. The black parts on each side contain springloaded electric connectors to power tools on the tool side. These parts can be removed if the electric connections are not required to reduce the space occupancy. 61

4.7 Top line, left to right: hand gripper attached to the wrist, head without camera, Azure Kinect with tool side. Bottom line, left to right: foot with tool side, foot attached to ankle 62

4.8 Wrist integration of the Kosmek tool-changer 63

4.9 design of the foot parts for the kosmek TC 64

4.10 Kosmek TC electrical connection to make it work 65

4.11 Experiment with Kosmek tool-changer 66

4.12 Visual servoing using the whycon library 67

4.13 Robot's point of view using the Kinect Azure camera with the models of the parts track by the M3T library superimposed. 68

4.14 Spiraling hand trajectory to adjust the TC position when engaging it. 70

4.15 Trajectory and contact force at TC level when engaging the TC. 71

4.16 design of the foot parts for the kosmek TC 72

5.1 Subassemblies of the proposed design. On the bell, the small sphere is used to lock/unlock the EEC. The surfaces are color-coded so to highlight that surfaces of the same color on both the robot and end-effector/tool sides will match once the ECC is locked. Surfaces E, F, G, and H are each a set of 3 similar surfaces spread around the ECC. 76

5.2 Different steps of EEC engagement. Sub-figure 5.2a is a slice in a plane containing the z-axis. Sub-figures 5.2b, 5.2c, and 5.2d are slices normal to the z-axis at the bottom of the pin looking from the end-effector/tool side toward the robot (effector) side. In each sub-figure the parts of the TC are color-coded in the same way. 77

5.3 Foot end-effector changer 79

5.4 CAD of the head end-effector changer with USB-C pass-through. From left to right: tool side, robot side and engagement step. On the robot side, the USB-C cable can rotate with respect to the EEC as during the last rotation of the engagement, the cable is already plugged-in. 79

5.5 FEA analysis results: in both case the robot side is considered fixed (black ground symbol) while the load is applied on the tool side (orange arrows). . 80

5.6 Prototype of the end-effector changer. From left to right: robot side, end-effector/tool side, mounted and locked tool-changer (both robot and tool side). 82

5.7 Prototype EEC mounted on humanoid HRP-4 (from left to right: feet, hand, head) 82

5.8 Arm end-effector use to support the robot with the tool side of the EEC . . . 83

5.9 Robot locking the EEC after releasing the gripper from the tool station 84

5.10 Balance region of the robot standing on two feet (left) and standing one foot and using its arms (right). The green rectangles correspond to the feet contacts and the red and black volume is the CoM balance region accounting for CoM acceleration up to 0.5 m.s^{-2} in all horizontal directions. 85

5.11 Top view of the (right) foot docking/storage station (dark gray aluminium profile and the 3D printed green part). The light grey rectangles are the feet of the robot. The one on the right is stored at the foot station while the left one is still attached to the robot (black part is the ankle link of the robot and purple part is the locking bell of the EEC). 86

5.12 Tracking of the forces and torque from the ground of the robot at the sole of the right foot during the motions showed in Figure 5.11. Form top to bottom there is (i) the normal vertical force applied by the foot on the floor (along the Z axis) (ii) The horizontal forces applied by the foot, the X-axis being forward and Y-axis from right to left when looking from the robot point of view, (iii) the torque around the Z-axis. The legend hides the part of the graphs were the foot is in the air without tool and moving from one tool to the other and thus all measures and targets are at 0. The letters correspond to Figure 5.11 and are positionned horizontally at the corresponding time 87

5.13 Foot change from the grey one to the white one 89

5.14 Locking and unlocking of the foot EEC 89

5.15 Modification of the locking mechanism to have larger variation of feet. . . . 90

5.16 Head station 91

5.17 Head change on HRP-4 using the hand EEC 92

List of Tables

2.1 Timings in ms 29

4.1 Existing TC data (extracted from data-sheets). Weights account for both robot and tool side. Required locking precision is the allowed positioning error when locking the TC. 73

5.1 Result from FEA on hand/head EEC. 80

5.2 Result from FEA for foot EEC. 81

INTRODUCTION

Robots and automatons have always been imagined by the human mind as having a human morphology with two legs, two arms, and one head. Many literature examples from the early twentieth century such as Asimov's series picture robots with a humanoid shape. It is also the case for later pop culture references such as George Lucas's *Star Wars* or Masamune Shirow's *Ghost in the Shell*. With *Honda*'s ASIMO in the early 2000s, the many humanoid solutions presented at DARPA's robotic challenge from 2012 to 2015, and the fast development of humanoids over the last decade, humanoid robots are now a tangible and realistic technology.

Encouraged by the amazing achievements of *Boston Dynamics* with the humanoid Atlas and the reliability of *Honda*'s ASIMO, new humanoid robots have been disclosed by big companies like *Tesla* or *Kawasaki Robotics*. Others have emerged from dedicated new companies. For example, *Agility Robotics*, under a joint initiative with *Ford* and *Amazon*, is working toward commercial humanoid robots for package delivery applications. Recently *Figure* has signed a contract with the car manufacturer *BMW*.

Agility Robotics' *Digit* and *Figure*'s *O2* are both shown in an industrial context, yet neither is used in challenging scenarios. Currently, they execute simple carrying tasks. Few demonstrations show the use of upper limbs to apply important forces on the environment, for example, to manipulate a large object or push a heavy door. In that case, a human would use its weight to apply the force and such behavior is not trivial to reproduce in the context of humanoid robotics. Several demonstrators from research teams presented these behaviors Kheddar et al. (2019); Kumagai et al. (2019); Chappellet et al. (2023).

Furthermore, grippers and end-effectors used in industry and research demonstrators usually fall into two categories. Some cases use complex grippers for tasks requiring dexterity such *Figure*'s *O2* or in Kheddar et al. (2019). Others consider simpler and more robust end-effectors with one degree of freedom or less to handle heavy objects such as *Agility Robotics*' *Digit* or in Chappellet et al. (2023). In Kumagai et al. (2019), the robot demonstrated both the carrying of a heavy object and the use of a tool to install plaster. The same gripper allowed for both tasks as the tool's shape allowed for a good quality of grasping. In all these cases, the end-effectors are fixed to the robot and remain unchanged during all the showcases.

Following these applications of humanoid robots in industrial contexts, this thesis aims at improving the use of humanoid robot's upper limbs. First, two contributions are made to improve the balance in multi-contact scenarios so that the robot can better handle its weight and apply it to tasks requiring moving and sliding end-effectors. Second, we propose the use of end-effector changers for humanoids as a way to offer a compromise between robust but imprecise end effectors for heavy-duty tasks and dexterous but more fragile hands or

tools. To the best of our capabilities, we implemented all these contributions on the HRP-4 platform and demonstrated their benefits through various scenarios.

Recent developments promoted by the major actors of the humanoid robot companies seem to rely heavily on computer learning for motion generations. Although it is proving to be a proper solution to mimic human behavior, learning-based solutions are computationally demanding to develop and at the moment, extracting guarantees required for safety and industrial regulations is a challenge.

First, we recall the major works related to the contributions of this thesis in Chap. 1. An overview of the different methods used to balance humanoid robots for walking and multi-contact behaviors is done. Then, reviewing existing works that implement tool changers in robotics, helps us outline some benefits and challenges of their implementation. Finally, we detail and introduce several technologies that make humanoid robots possible.

In the first part of this thesis, multi-balance is studied by considering scenarios where the robot has to balance using sliding and moving contacts. The whole body dynamics of a humanoid robot with tens of joints makes it complex to consider balance in general. For this purpose, many simplified models to represent the general behavior of the robot have been used. For example, the Linear Inverted Pendulum represents the humanoid by an inverted pendulum. It is the most notable approach when discussing dynamic walking motion generation. When considering multi-contact balance, we use centroidal models that consider the motion of the Centre of Mass (CoM) according to the different contact conditions imposed by the chosen model. Then, the control of the robot has to account for these CoM positions and generate joint motions to execute them. The first two chapters aim at finding valid and robust CoM positions.

In Chap. 2, we propose to use an approach relying on the computation of a CoM balance region. Such a region is used to constrain the motion of the CoM while the robot moves. The first contribution aims at improving the computation speed so that it can used online and account for moving contact in the balance of the robot. The contributions of this chapter were published at the ICAR conference in 2021 in an article entitled *Control of humanoid in multiple fixed and moving unilateral contacts* with the co-authorship of Saeid Samadi, Eisoku Kuroiwa, Takahide Yoshiike, and Abderrahmane Kheddar.

In Chap. 3, the objective is for the robot to keep its balance in the case of sliding contacts. This is considered as in some environments, the robot may not have enough room to break a contact and thus sliding it will let the robot move. Also, tasks such as wiping a whiteboard need the robot to be able to slide one of its contacts while applying pressure on it. The results have been published as a second author in the journal Robotics and Automation Letters (RA-L) and presented at the ICRA conference in 2021 under the title *Humanoid control under interchangeable fixed and sliding unilateral contacts* with co-authorship of Saeid Samadi (first author), Arnaud Tanguy, Stéphane Caron, Abderrahmane Kheddar.

In the second part of this thesis, we discuss the design challenge that consists of solving a shortcoming observed in the deployment attempts we mentioned previously: the need for different grippers, feet, heads, and even sometimes limb lengths for the same application domains. On the one hand, tool-changing systems have a long history in industrial automation. They range from simple mechanisms to change drill bits or pipe wrench heads

to the more sophisticated ones in industrial robotic settings. On the other hand, almost all existing humanoid robots come with unchangeable mechanisms to attach the gripper and feet, and head, often designed once for all. It is well known that bipedal mobility poses high constraints on the end-effectors in terms of weight, payload, and connectivity making it difficult to design sophisticated systems that do it all. Even if general purpose humanoid robots (GPR) is an interesting perspective in terms of both technological and fundamental aspects, we could also think of humanoids that can change autonomously and quickly appropriate end-effectors such as entire grippers, feet, heads... or other tools to match specific task as we human do with different tools that are all extensions of our own 'end-effectors' (e.g., drill, dedicated shoes, night-vision, etc.).

Chap. 4 presents the integration of two commercial tool-changers to the robot platform HRP-4 from the review and choice of existing solutions to all hardware, firmware, and software changes required for their integration. These initial tests showed the inadequacy of the existing solution for humanoid robots. We outlined a set of design constraints to apply end-effector changing to humanoids.

In Chap. 5, we propose a novel design for end-effector changers better suited for humanoid robots. It is lightweight. The locking mechanism can be actuated externally by interacting with the environment, an operator, or another of the robot's limbs. Finally, the proposed design comes in three versions to accommodate the requirements of different end-effectors. The results of this part are published in the journal RA-L in a paper entitled *End-Effectors Changer Design for Humanoids* co-authored by Jean-Baptiste Izard, Arnaud Tanguy, Hiroshi Kaminaga, Fumio Kanehiro, and Abderrahmane Kheddar.

CHAPTER 1

BACKGROUND AND USED WORKS

Although humanoid robots have recently been booming in popularity, they have been developed since the end of the twentieth century Brooks et al. (1998); Sakagami et al. (2002). Thus, the current developments and successes heavily rely on the build-ups of the whole community. This chapter situates the thesis relative to the previous works of the robotic community. We will review a wide set of topics related to robotics and focus on humanoid robots. It will be divided into three principal axes.

First, in section 1.1, the topic of balance is developed. We will introduce the main models and approaches to study and control the interaction between the humanoid robot and its environment. Further contributions on these topics are later introduced in Chap. 2 and 3, two new and different approaches to balance in multi-contact scenarios are introduced.

Then, in section 1.2, contributions on tool-changers will be discussed. An overall tour of all existing solutions presented in the literature with comments on their benefits and drawbacks will be given. More examples of tool-changers are introduced in Chap. 4: those models are not presented in peer-reviewed papers but are commercially available from manufacturers. Although they are not described in published articles, reviewing them provides useful insights on tool changing.

Finally, in section 1.3, the rest of the references concern works that are useful for humanoids as they provided stepping stones to demonstrate the benefits of the different contributions of this thesis on the robot platform HRP-4. They concern topics such as whole body control, vision, or localization. As each of these topics would require chapters to be fully covered, many of these references are review papers that give an overview of the recent research on these topics. Some more precise papers will detail the works that were directly concerned or used in our contributions.

1.1 Multi-Contact balance

In many papers tackling the topic of balance, the word *stability* does not always correspond to the definition given in the field of control theory, where stability qualifies an equilibrium and the capability of a system to converge to an equilibrium. It is meant in the sense of balance. That is preventing the robot from falling. Other similar notions, such as

viability Sherikov et al. (2015) and capturability Pratt et al. (2006), are used instead to qualify the quality of the balance.

1.1.1 General balance of humanoid

Generating walking patterns allowing humanoids to walk dynamically has been fundamental when considering legged locomotion. In Kajita et al. (2001, 2003), the author introduced an implementation of the Zero Momentum Point (ZMP). It lets the robot track dynamic gait motions. It is considered a cornerstone in humanoid legged locomotion. The ZMP dynamics are computed from the Linear Inverted Pendulum (LIP) model as follows:

$$\ddot{\mathbf{c}} = \omega^2(\mathbf{c} - \mathbf{z}) \tag{1.1}$$

where $\mathbf{c}$ is Center of Mass (CoM) position, $\ddot{\mathbf{c}}$ its acceleration and $\mathbf{z}$ the ZMP position on the ground. From this equation, the CoM diverges from the ZMP, and controlling the ZMP imposes the CoM behavior. However, some major hypotheses are constraining the model:

- variation of the angular momentum at CoM is neglected;
- the CoM must have a constant height for the model to be linear;
- the robot is assumed to walk on flat ground.

Many research works tried to overcome these hypotheses and improve humanoid locomotion. In Popovic et al. (2005), the Foot Rotation Indicator (FRI) and Centroidal Moment Pivot (CMP) are used to account for foot rotation and angular momentum at CoM. In Pratt et al. (2006), the inverted pendulum model is extended with a flywheel body to compute the capture region. Then, when the robot has to recover from a push, if there is no intersection between the support area and the capture region, stepping into the capture region will let the robot regain its balance. The support area is computed as the convex hull of the feet. In Takenaka et al. (2009), the capture point approach is applied to a 3-mass model to generate a walking gait trajectory. In Koolen et al. (2016), they demonstrate the use of CoM height variation to control the balance of an inverted pendulum. In Caron et al. (2019), feedback on the Divergent Component of Motion (DCM) is used in the control of the robot to climb stairs. In Englsberger et al. (2015), the DCM's definition is further extended to be applied to 3D cases. They propose a corresponding trajectory generator that allows for CoM height variations while walking. This approach allows for running and jumping in Mesesan et al. (2023).

Then, the inverted pendulum model has been extended to account for perturbations and manipulated objects with the upper body limbs.

In Morisawa et al. (2018), the LIP model is formulated using force distribution ratios instead of the ZMP. It allows the authors to consider multi-contact locomotion using hands and feet and variable CoM height. Furthermore, the CoM trajectory is generated using a linear time-variant system instead of Model Predictive Control (MPC), allowing to account for an uneven terrain where contact may happen earlier or later than expected.

In Agravante et al. (2019), the author implemented collaborative carrying by considering the interaction between the carried object and the robot as an external force that disturbs

the ZMP motion. The model assumes the robot walks on a flat horizontal ground with no variation of the angular momentum. This model is then used in an MPC outputing the CoM and swing foot trajectories for the robot to follow.

Murooka et al. (2021) derived the inverted pendulum model as in Kajita et al. (2003) but accounting for external wrenches applied on the robot. They also use the DCM feedback as in Caron et al. (2019). The resulting controller lets the robot balance in complex scenarios and dynamically apply important wrenches to its environment as long as the applied external wrenches are known. The method to derive the IPM while accounting for external wrenches can also be applied to many other balance schemes.

The papers presented up to now deal mostly with walking models based on the linear inverted pendulum model. They focus on legged locomotion more than multi-contact behaviors. They provide dynamic motions and behaviors. However, they lack the application to multiple non-horizontal, and non co-planar contacts. The next section introduces works about multi-contact balancing and applying the desired wrench to the environment while considering robustness to external disturbances.

1.1.2 Multi-Contact balance

The works presented in this section tackle the issues related to non-coplanar and non-horizontal contact. Most of them do not rely on the linear inverted pendulum model.

Often, the balance is considered kept when the contact is kept fixed, without sliding. The concept of contact stability is discussed in Caron et al. (2015b). The authors computed the condition to maintain a rectangular surface contact under friction conditions. They introduced the concept of upper and lower bound on the yaw torque that has not been used before.

In Collette et al. (2007) and Collette et al. (2008), the authors proposed proposed a two-level controller. In the lower level Collette et al. (2007), minimized joint torque is computed by solving a Quadratic Programming (QP) optimization problem while tracking target CoM, unilateral contact forces, and grasp wrenches under kinematic and dynamic constraints. The targets for CoM, contact forces, and grasp wrenches are obtained from a higher level QP presented in Collette et al. (2008). In that work, the authors account for disturbance at the CoM level and introduce the concept of stability margin to assess the robustness of contact and CoM position. They compute the maximum admissible disturbance for a set of directions and contraversively for a given set of disturbances a CoM target and contact wrench distribution.

In Sherikov et al. (2015), a task hierarchy is presented to distribute forces in non-coplanar multi-contact scenarios. In particular, they consider the case where one contact is fragile and may not be used other than to keep balance in an emergency case. The method is based on capturability and uses MPC to enforce the static balance of the robot at the end of the sliding preview window.

Generating contact sequences that go beyond walking gaits is a widely studied topic. Yet, having the trajectories between the different contact stances and having the robot follow them is also very complex. In Lengagne et al. (2010), an early solution to this is proposed by formulation of a semi-infinite problem.

In Righetti et al. (2013), the authors derive a controller that generates the joint torques for floating base robots that minimize cost functions. These cost functions describe some desired tasks such as minimizing tangential contact forces to improve the robot's behavior on uneven terrains.

In Nozawa et al. (2017), the authors presented a method to formulate and identify the interaction between an object that the robot is manipulating and its environment. This allows to formulation of the balance of the robot while accounting for the forces required to manipulate the object. However, even though they use ZMP, the motions are quasi-static.

In Li and Nguyen (2023), the authors used MPC to compute the desired contact wrench and account at the same time for switching contact and external perturbations. They show that when carrying an object it is better to consider it as a new contact than as part of the robot thus modifying the formulation of the CoM. Indeed this avoids non-linearities in the formulation of the CoM and thus of the following MPC that would then give poor results. The MPC uses as input the simplified rigid body dynamics and known sequence of contacts and computes the contact wrenches to keep balance. The computed wrenches are then fed into a whole-body controller that computes torque input for a torque-controlled robot.

For torque-controlled robots, many controllers use passivity to generate contact wrenches that compensate for gravity and result in compliant behavior of the robot. In Hyon et al. (2007), the method is introduced to interact with humans while tolerating external wrenches. First, Ozawa and Ishizaki (2011) only considered walking for which they formulated a control law on the joints and a floating base that would ensure balance of the robot. Then as the floating base is not actuated, a power conservative transform is introduced based on the virtual work principle to have the legs generate the missing input on the floating base. They proved the stability of their approach using the Lyapunov approach. Simulated results showed that they can then balance on an uneven terrain. Henze et al. (2016); Ozawa and Kamioka (2022) extended the passivity-based approaches to multi-contact scenarios.

In Abi-Farraj et al. (2019), the authors formulated the problem as a wrench distribution problem and used passivity to distribute the contact wrench and compute a target CoM position. The main limitation of the paper is that the support end-effector used for balance and action end-effector that realize a given task are separated when considering balance. They also demonstrated the capability to slide objects on the ground while the CoM is outside of the support region.

The passivity-based approaches are mostly reserved for torque-controlled robots. For position control, such a method would require a precise identification of the dynamic parameters of the robot so that the joint's torques generated correspond to those computed by the controller.

1.1.3 Center of Mass Region

A balance region is defined such that the robot's CoM should stay inside during its motion to avoid falling. The balance region computation is based on centroidal dynamics derived from the Newton-Euler equations with the friction cones inequalities constraints. Instead of choosing a CoM target, a region where the CoM can stay is computed.

A CoM region is similar in concept to the support polygon constraining the ZMP under the foot sole of the robot. However, the support polygon and ZMP are mostly defined on flat surfaces. To extend this to multi-non-coplanar contact scenarios, the following works use the concept of region to the CoM.

In Bretl and Lall (2008), the authors introduced the concept of a support region. The CoM must be above it for a static equilibrium to exist. This region can be computed for any number of non-coplanar supports. They propose an iterative algorithm to compute an approximation of the region and prove its convergence within a bounded number of iterations. They also provide an algorithm to test if a CoM is inside the support region.

After devising a method to control the balance and the internal forces in multi-contact scenarios, the authors in Sentis et al. (2010), proposed to build a similar balance region by sampling points and testing if they are inside. The method proposed in Nozawa et al. (2016) builds the CoM balance region using a Linear Programming (LP) optimization problem as in Bretl and Lall (2008). However, the direction to find vertices on the boundary of the balance region is fixed. In Carpentier et al. (2017), the set of feasible CoM positions is approximated using a probability distribution. It is then used as a reduced model to constrain the generation of whole-body motions. In Orsolino et al. (2020), the CoM balance region is extended to account for joint actuation limits.

In Audren and Kheddar (2018), the static balance region introduced in Bretl and Lall (2008) is extended to consider the robust cases. A set CoM acceleration is considered. The new CoM region computed contains the CoM position for which all the CoM accelerations from a given are realizable with the current contact set. The result is a 3D polyhedron. The demonstration for convergence given in Bretl and Lall (2008) is still standing.

On a similar topic, one can consider similar regions for contact wrenches and CoM acceleration. In Caron et al. (2015a), they use the Gravito Inertial wrench cone to consider contact stability. In Caron and Kheddar (2016), a CoM acceleration region is computed for a given CoM position (i.e. the "opposite" of the CoM region).

However, such computed "regions" are expensive to compute. Thus, they are either used offline for planning CoM trajectories Audren and Kheddar (2017); Orsolino et al. (2020); Focchi et al. (2017) or pre-computed offline and then used online to test the equilibium.

When switching between contact stances, by removing or adding a contact, there is a large discontinuity in the shape of the balance region. In Audren et al. (2016), the author applied a morphing technique to smoothly transform the balance region from one stance to the next. Furthermore, they are not useful when any contact location undergoes motion tasks, e.g. pushing, sliding, etc. In Samadi et al. (2020), the CoM support area is modified to account for sliding contacts. The balance region is computed analytically in the limited scenario of two flat contacts and one sliding contact. In Samadi et al. (2021), we extended this approach in the general case by approximating the 3D CoM region with the inscribed Chebichev ball as detailed in Chap. 3. It resulted in a fast approximation of the balance region. However, it lacks the added benefits of the balance region, for which there is no need to set a direct CoM task. Indeed, such a task will interfere with other tasks to ensure balance. Thus, in Roux et al. (2021), we introduced a fast method to compute the balance region introduced in Audren and Kheddar (2018). It lets us update the balance region online under moving contacts.

1.2 Background on tool-changing technology

As mentioned in Lundberg (2011), Tool-Changers (TC) have been used in the manufacturing industries since before the mid-1980s. They greatly reduce the manufacturing time of small-batch products. Most of the time, when discussing industrial machining applications, the TCs do not require human intervention and are called Automatic Tool Changers (ATC). Given they have existed for a long time, many works discuss their benefits and disadvantages. This section described works about tool-changers in the literature related to robotics. Most of these works mention commercial tool-changers. In this thesis, two commercial tool-changers were integrated into our robotic platform as presented in Chap. 4. Details on other commercial TCs are also given in that chapter.

In Rogelio and Baldovino (2014), the authors developed an ATC for a 3-axis CNC router machine. They consider the tool-to-tool time to evaluate the benefits of the ATC. Adding an ATC to their CNC router allowed them to reduce by 35% the machining time. In the meantime, to test reliability, the accuracy of the CNC is evaluated after performing several tool changes and it displays a loss of precision (≈ 0.01 mm) superior to what would happen without the ATC. Finally, user safety is mentioned as a benefit of ATC as they reduce the need for operator interventions during the making of a part.

The generalization of ATC led to them being a major source of failure in machining centers. In Tian et al. (2017), the authors evaluated the life expectancy of a spring in a tool holder by measuring the pull force to remove the tool from the tool holder. They modeled the system and using a tool to measure the pulling force, the parameters were identified. The objective is to prevent the tool from falling from the tool station.

Following the democratization of robotic arms, TCs were integrated into their operations further increasing their capabilities and versatility. For instance, in Ryuh et al. (2006), the polishing bits on a robotic arm are changed using TC to improve the productivity of a die polishing station. The solution is inspired by a machining center's ATC as only the bit of the polisher is stored and the tool older is a rotating one. They also developed a method to adapt the trajectory of the previously used tool for the current one to computation save time.

In Clévy et al. (2005, 2008), they implemented the concept of TC into a micromanipulation cell. The TC allows to change the tip of a piezo-electric gripper which manipulates objects at a scale between 20 and 500 μm. The tools are fixed to the gripper using a thermal glue which is solid at room temperature but is melted at 65°C. The glue is melted using resistors that heat up when the current passes through. When the tools are not mounted on the gripper, they are glued to the tool holder.

Similarly in Chen et al. (2014), the authors had a robotic arm change its screwdriver tip in a human-robot collaboration setup. There was no new design for tool changing as a common socket wrench is used as a tool. However, the bits are stored in a tool holder with a notch that the robot can use to remove the tip. The torque-controlled robot, once showed where to start the tool change motion, uses force control and the interaction between the screwdriver and the tool holder to change the tip. A similar setup could be used by a human with one arm.

The TC showed in Fig. 1.1 was introduced in Gyimothy and Toth (2011). It is a small electrically actuated TC. It has electricity and data pass through. After the design had been

detailed, the authors thoroughly evaluated its characteristics with a set of experiments (some were destructive). It provides tolerance to positioning errors of up to ± 4 mm in position and $\pm 1^o$ of rotation in the axis normal to the interface plane. Its main drawback is the complexity of its design and the use of an internal DC motor to lock it as it would require some integration to the robot's hardware.

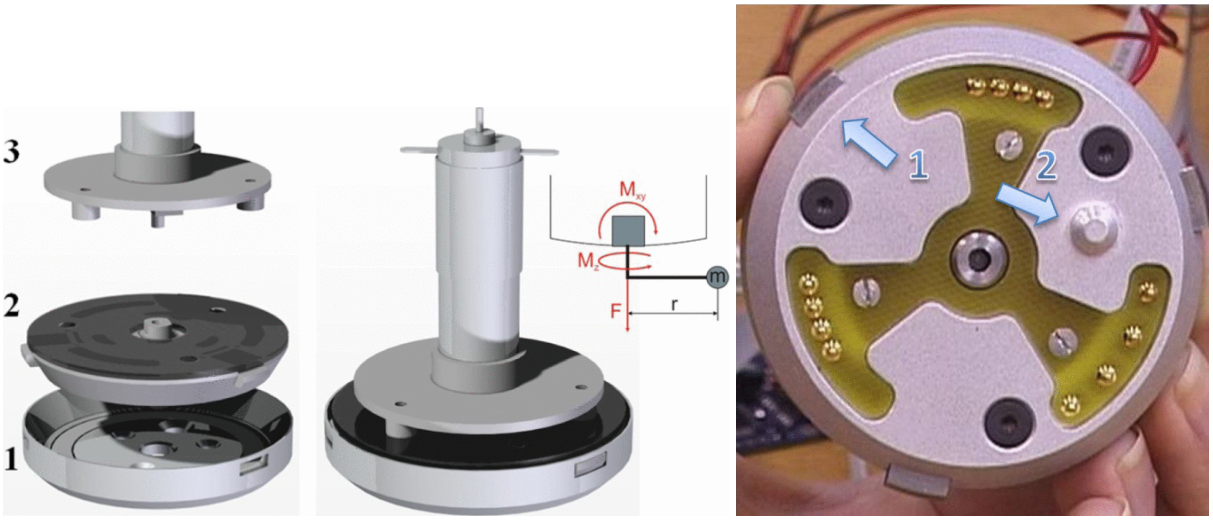


FIGURE 1.1 – *©2011 IEEE Tool-changer from Gyimothy and Toth (2011) used in the ACROBOTER project. Left and midle CAD of the TC; Right: prototype seen from below*

In opposition, Berenstein et al. (2018) introduces a simple passive design for robotic arms with no actuators (see Fig. 1.2). The system relies on springs and a clever locking mechanism that triggers when inserting the tool back in the tool station. The design focuses on increasing its tolerance to positioning errors, as illustrated by a set of tool change experiments where the robot successfully changed the tool 92 times out of a hundred trials. They mention that most of the failures were due to positioning errors. However, in the presented version, there are no electric pass-throughs.

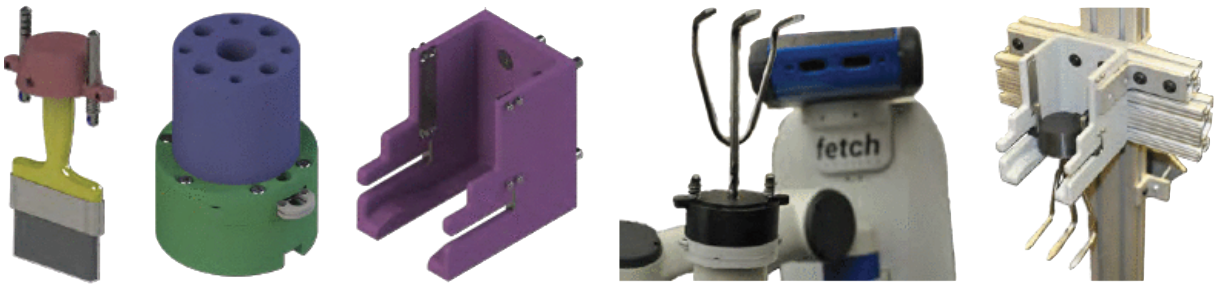


FIGURE 1.2 – *©2018 IEEE Tool-changer from Berenstein et al. (2018) (a) CAD of tool side; (b) CAD of robot side with Fetch arm adapter; (c) CAD of tool station; (d) TC prototype mounted on Fetch robotic platform with simple tool; (e) Tool station with simple tool stored in.*

Similarly, in Pettinger et al. (2019), the authors designed and integrated a resilient TC for a robotic arm mounted on a mobile platform that inspects corrosive and radioactive tunnels (Fig. 1.3). Their application imposed special requirements on the design of the tool-changer and of the tool station. During the inspection mission, the robot needs a camera at its end-effector which is integrated in the TC. The collected samples have to be stored in sealed

chambers that should be easily retrieved. To fit the robot arm, the TC should be lightweight and robust to handle large contact forces. Finally, the TC should be cheap because it will be discarded with the robot when it becomes radioactive. The solution uses elliptical shapes to improve the tolerances to positioning errors and a rotary tool station to store the retrieved samples and tools. A set of experiments showed that with a positioning error of ± 3 mm they achieved a 90 % success rate and were able to change the tool with up to 10 mm positioning error. The proposed design meets most of the expectations, yet it has no electric or data connection between the robot and the tool.

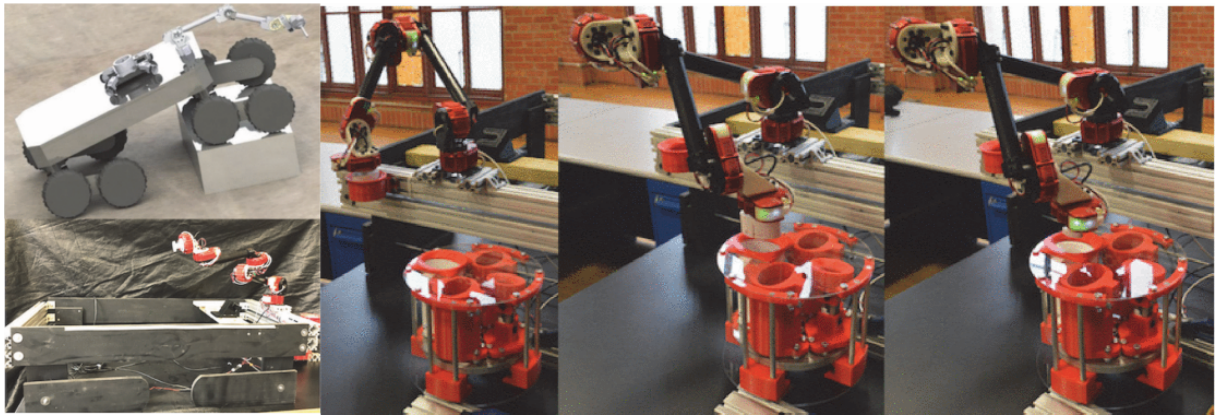


FIGURE 1.3 – *©2019 IEEE Tool-changer from Pettinger et al. (2019). Left: wheeled robotic plateform with robotic arm used on which the TC will be installed; Right: 3 steps to store a tool in the tool station.*

In Mourtzis et al. (2022), a new TC design for robotic arms is proposed. It is 3D printed and has no electrical connections. It is specifically designed for UR-10 robots. The authors evaluated its capabilities with finite element analysis and showed that it should match the expected values of a UR-10 robot.

Li et al. (2022) presented a highly customizable modular end-effector system with pneumatic, data, and electric connections (Fig. 1.4). It is designed for a mobile manipulation platform and integrates a camera to help locate tools and tool holders. This TC uses a solenoid as part of its locking mechanism. It has an integrated *Raspberry Pi 4B* streaming the camera on a ROS topic, controlling the solenoid to unlock the mechanism, and, the airflow passing through using valves. Given all the features of this TC, it is big, and its size may greatly increase the wrist torques.

In Iqbal et al. (2021), the authors used a TC to integrate a collaborative gripper into their robotic arm. The gripper is an autonomous gripper that can function detached from the robot as it is battery-powered. Thus, the gripper was able to open and close while detached from the robot. The tool-changers as a two-step attachment procedure that requires human intervention. Electromagnets align and hold the gripper in position w.r.t. the end-effector while the operator pulls mechanical latches to lock it. The operator has a control ring with buttons and vibration feedback that lets him control the electromagnets. This TC is designed for human-robot interaction. It greatly simplifies gripper change, yet, it does not let the robot have full autonomy. Furthermore, the presented version did not have electrical or data pass-through.

TC is also found in docking for modular robotics Dokuyucu and Özmen (2023). Self-reconfigurable modular robots use simple modular robots (usually with only 1 or 2 Degrees of Freedom (DoF)) that dock together to generate more complex behaviors and reconfigure



FIGURE 1.4 – *©2022 IEEE Tool-changer from Li et al. (2022). The tool station is mounted in the Freight robotic platform (left) and the tool-changer is used on the Fetch robotic plateform (right)*

their structure to adapt to different situations. In Saab et al. (2019), the authors reviewed existing docking mechanisms and methods. Due to the simple structure of each robot module, most docking systems rely on an inner actuation (motors for mechanical coupling, electromagnets for magnetic coupling...) to lock and unlock. Similarly to TC, coupling tolerances are of major importance.

In Pau Muñoz-Benavent and Tornero (2019), the authors used the task of tool changing to implement a robust closed-loop control based on visual servoing. The type of tool-changers is not detailed as the paper focuses on the development of a control law accounting for the limits of the robot. In Wei et al. (2021), the authors illustrated the benefits of automatic tool changes with visual servoing. Using a 6-DoF robot arm equipped with an ATI QC-11 automatic tool-changer (see Sec. 4.2.2), they compare the time to engage a tool automatically using visual servoing and the time it takes an operator to have the robot do the same task. The results show that using visual servoing, tool engagement is on average two times faster than when the operator guides the robot. The RGB-D camera used is mounted on the end-effector close to the robot side of the TC. It is used to align the end-effector in the horizontal plane with a precision close to 1 mm of translation and 1° of rotation error when both sides of the TC are 30 cm away from each other.

In Hu et al. (2019), the authors opposed the use of a tool-changer to the use of an easy-to-grasp tool. They design a simple tool that can be grasped by any gripper with two parallel fingers. The tool is based on a 4-bar mechanism with torsion springs that keep it open and let the robot grasp it. The authors optimized the spring and geometry to maximize the weight that can be lifted. Because the tool is cheap to reproduce, several tools with different tooltips for different tasks can be available in the robot workspace thus enhancing the versatility of the robot. However, this approach is mostly limited to grasping (an example for cutting is also given) as the tool is not powered, and no sensor feedback can be given by the tool.

In Friedmann and Fleischer (2022), the authors proposed a method to design modular gripper fingers. Their method uses modules from a gripper finger construction kit and automatically designs gripper fingers able to grasp a wide variety of parts. Sensors are integrated into the fingers to have quality control during the handling of parts. They claim using modular and flexible gripper fingers is better than TC or swivel wrist mechanism, however they only focus on grasping tasks.

According to several works, the lack of tolerance for positioning errors is a major flaw in the design of tool-changers. However, not all works propose an extensive evaluation of the robustness through repetitive testing Berenstein et al. (2018); Pettinger et al. (2019) In Li et al. (2022), the TC is evaluated by showing the working principle of their solution with three different end-effectors and two different robot platforms. Few works detail the physical resistance of their prototypes Gyimothy and Toth (2011). In Li et al. (2016), the authors simulated a TC during its engagement and locking phases using a rigid-flexible model. The TC design is similar to commercially available ones and uses air pressure to lock and unlock. It is concluded that the design should not break during the engagement phase. Finally, although it is mentioned as a requirement for many design Berenstein et al. (2018); Pettinger et al. (2019), having electric, data or fluid pass-throughs has only been done by a minority of works Gyimothy and Toth (2011); Li et al. (2022).

1.3 Other related litterature

As mentioned in the introduction, special attention was given to preparing real-life demonstrations on robotic platforms. The previous two sections gave references related to the major contributions of this thesis namely multi-contact balance and end-effector changers. However, to integrate them into our robotic platform several other technologies are required. The related works used to integrate these technics are developed in this section.

First of all, for all demonstrations of this thesis, we use the humanoid robot platform HRP-4 built by Kawada in Kaneko et al. (2011) showed in Fig. 1.5. It has 32 degrees of freedom and weighs around 40 kg. The control of the robot is done externally on another computer and sent to the robot via UDP on an ethernet cable.

1.3.1 Whole Body QP

In this thesis, all the controllers are implemented in the *mc_rtc* framework [1]. The development of the Whole Body Quadratic Program (WBQP) implemented in this framework results from many previous works Bouyarmane and Kheddar (2011); Vaillant et al. (2016, 2017); Bouyarmane et al. (2019b).

The formulation and workflow of *mc_rtc* are illustrated in Fig. 1.6. As several contributions of this thesis are related and rely on the formulation of the whole body QP, its

1. https://jrl.cnrs.fr/mc_rtc/



FIGURE 1.5 – *©2011 IEEE - Humanoid Robotic Platform 4 (HRP-4) as presented in Kaneko et al. (2011)*

formulation from Bouyarmane et al. (2019b) and Vaillant et al. (2017) is recalled here:

$$\min_{\ddot{\mathbf{q}},\tau,\lambda} \quad \sum_{k=1}^{M} w_k \|\ddot{g}_k - \ddot{g}_k^d\|^2 \tag{1.2}$$

$$\text{s.t.} \quad M(\mathbf{q})\ddot{\mathbf{q}} + N(\mathbf{q},\dot{\mathbf{q}}) = J^T\mathbf{f} + S\tau \tag{1.3}$$

$$\frac{\dot{\mathbf{q}}_{\min} - \dot{\mathbf{q}}}{\Delta t} \leq \ddot{\mathbf{q}} \leq \frac{\dot{\mathbf{q}}_{\max} - \dot{\mathbf{q}}}{\Delta t} \tag{1.4}$$

$$\frac{\mathbf{q}_{\min} - \mathbf{q} - \dot{\mathbf{q}}\Delta t}{\frac{1}{2}\Delta t^2} \leq \ddot{\mathbf{q}} \leq \frac{\mathbf{q}_{\max} - \mathbf{q} - \dot{\mathbf{q}}\Delta t}{\frac{1}{2}\Delta t^2} \tag{1.5}$$

$$J\dot{\mathbf{q}} = 0 \tag{1.6}$$

$$\lambda \geq 0 \tag{1.7}$$

$$\tau_{\min} \leq \tau \leq \tau_{\max} \tag{1.8}$$

The WBQP outputs the values for the robot's joint acceleration $\ddot{\mathbf{q}}$, the robot's joint torques τ, and the contact forces λ that minimize a sum of weighted quadratic functions modeling tasks (Eq. 1.2). The contact forces are modeled using the vertices description of the friction cones. Let $\mathbf{f}_i$ be the force applied by the environment on the robot, then:'

$$\mathbf{f}_i = \sum_{k=1}^{n_i} \lambda_{i,k}\mathbf{f}_{i,k} \tag{1.9}$$

where $\mathbf{f}_{i,k}$ are the n_i generators of the ith contact's friction cone. Then Eq. 1.6 represents

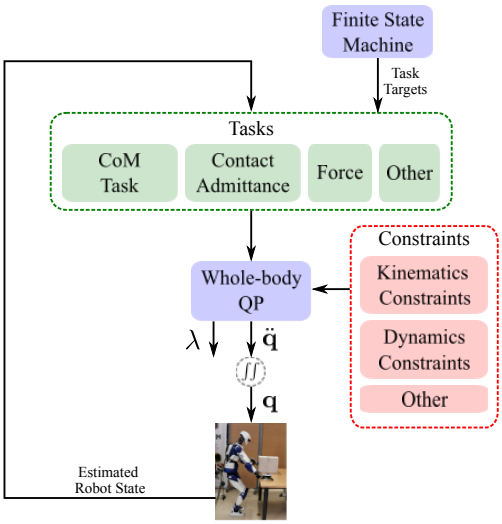


FIGURE 1.6 – *Schematics of the WBQP*

the unilaterality of the contacts by imposing positive coefficients. Eq. 1.3 is the dynamic constraint not accounting for the multi-robot interaction with $M(\mathbf{q})$ the mass matrix, $N(\mathbf{q},\dot{\mathbf{q}})$ the matrix representing Coriolis and gravity effects. J is the jacobian of the contact forces accounting for contact points where the forces are applied. Thus, Eq. 1.6 impose the contacts to be fixed kinematically. Eq. 1.4 and 1.4 represent velocity and position limits of the joints that is $\dot{\mathbf{q}}_{\text{min}} \leq \dot{\mathbf{q}} \leq \dot{\mathbf{q}}_{\text{max}}$ and $\mathbf{q}_{\text{min}} \leq \mathbf{q} \leq \mathbf{q}_{\text{max}}$. Eq. 1.8 implements the joint torques limits. Δt is the time interval between two iterations of the QP.

The tasks are represented by the g_k functions. The costs are computed from task error and task Jacobian. In Vaillant et al. (2017) and Bouyarmane et al. (2019b), the desired tasks acceleration are given by

$$\ddot{g}_k^d = \ddot{g}_k^{\text{ref}} - P_k(g_k - g_k^{\text{ref}}) - D_k(\dot{g}_k - \dot{g}_k^{\text{ref}}) \tag{1.10}$$

with g_k^{ref}, $\dot{g}_k^{\text{ref}}$, and $\ddot{g}_k^{\text{ref}}$ references given by the Finite State Machine (FSM). For the position and trajectory task, g_k corresponds to the world pose of the controlled frame. For the CoM task, g_k corresponds to the pose of the CoM.

The implementation of force control in this framework is also discussed in Bouyarmane et al. (2019a). An admittance task is formulated as follows: a velocity proportional to the error between the measured and target force at the target frame is derived and integrated to give position task references.

Given the implementation of tasks and how they are used in the WBQP, it is clear that they are interfering with each other. For example, it is possible to have two position tasks controlling the same frame but with different targets. Then, the resulting generated acceleration will move the frame closer to the target with the larger weight without reaching it because of the second one. In particular, a CoM task is particularly restrictive when considering reaching tasks for the hands. Thus, balance regions are implemented as a constraint similar to collision avoidance in this framework.

A balance regions $\mathcal{Y}$ is formulated as the intersection of n halfspaces such that the CoM $\mathbf{c}$ belongs to $\mathcal{Y}$ when

$$\forall k \in [1,n], \mathbf{a}_k^T \mathbf{c} \leq b_k \tag{1.11}$$

where $\mathbf{a}_k$ and b_k are the normal and offset of each halfspace. The distance between $\mathbf{c}$ and the kth halfspace is then

$$d_k = \mathbf{a}_k^T \mathbf{c} - b_k \tag{1.12}$$

The constraint for halfspace k is then given by

$$\ddot{d} \geq \frac{1}{\Delta t}\left(-\xi \frac{d_k - \delta_s}{\delta_i - \delta_s} - \dot{d}_k\right) \tag{1.13}$$

with ξ the damping coefficient, δ_i is the interaction distance and δ_s is the safety distance. This constraint acts as a damper when the CoM comes close to the kth halfspace such that its velocity is null when it reaches a distance δ_s. The damping starts when the CoM when $d_k \leq \delta_s$.

The main limitation of this constraint is that if the CoM velocity $\mathbf{c}_k$ is too important, the robot won't be able to generate a CoM acceleration $\ddot{\mathbf{c}}_k$ to respect this condition. To avoid this, the condition should be tested for more than just one step.

1.3.2 Vision

To locate the robot w.r.t its environment, the visual feedback from the cameras on the head of the robot is used. First, locating the robot in the environment can be done using the Simultaneous Localization And Mapping (SLAM) technique as in Kheddar et al. (2019); Chappellet et al. (2023). The robot is located in the environment which is in turn mapped. In Tsuru et al. (2021), the authors expended the SLAM technique by simultaneously planning for the optimal pose to further explore the surroundings. It lets the robot locate the apparatus for its next task.

Once the robot is in range of its target, SLAM-based techniques may lack the precision to execute closed-range tasks. However, object tracking and visual servoing methods as in Agravante et al. (2017); Wei et al. (2021) offer better precision for positioning tasks. Several libraries are available for this purpose Trinh et al. (2018); Chappellet et al. (2023); Stoiber (2023).

Finally, to ensure that the visual method is precise enough, the location of the camera w.r.t. the robot's kinematics frame has to be precise. Authors in Enebuse et al. (2021); Jiang et al. (2022), review hand-eye calibration methods to solve this problem.

1.3.3 Parameters estimation

Most approaches presented in this thesis, use model-based approaches.

In Orin et al. (2013), the authors studied the structure of the Centroidal Momentum Matrix and proposed an improved algorithm to compute it. This is to improve the computation of the centroidal dynamics.

To have a good approximation of the behavior of the controlled robot, it is important a good estimation of dynamic parameters such as the mass and inertia matrix of each limb of the robot. The field of dynamic parameter identification for fixed base robots has been

widely studied as suggested in Leboutet et al. (2021). When considering a humanoid robot, one has to also account for and identify the floating base dynamic parameters as in Ayusawa et al. (2008); Ogawa et al. (2014); Jovic et al. (2016)

1.3.4 Contact estimation

Finally, contact initiation, contact detection, and contact force control are a major part of this thesis.

The subject of contact state has been studied. In the field, the current contact state of the robot is described by its contact formation. The contact formation describes how two bodies interact (planar face to planar face, or corner vertice to planar face). Then, the objective is to determine the current contact formation. In Jasim and Plapper (2014), the authors used Gaussian mixtures to model the probability of different classes of contact formation based on the current measured wrench and pose of the end-effector. This study is limited as it has two major drawbacks in our topic. First, the pose of the end-effector is used to predict the contact state. It is difficult from the paper if this pose is relative to the environment. In this thesis, we try to assume that the exact pose of the robot is not known precisely and can be estimated using contacts. Second, they only use a cube interacting with three orthogonal planes making the possible contact formations simple.

In Van Wyk et al. (2018), the authors proposed a benchmark to compare peg-in-hole strategies. They implemented two systems to illustrate the benefits of the proposed benchmark. First, a soft dextrous hand is mounted on a robotic arm with a stiff control. Second, a stiff gripper is used, and the compliance comes from the robotic arm control. They also evaluated different trajectories to apply during the insertion phase. For instance, a spiral trajectory gave good results. However, the dextrous hand overall performed better for this task.

1.4 Conclusion

In section 1.1, an outline of the many works on humanoid balance has been given. First, walking motions based on the linear inverted pendulum are discussed. They allow the generation of dynamic motions important for legged locomotion. However, these methods are harder to apply on irregular non-coplanar surfaces. Often, they separate upper and lower limbs to account for perturbations. Then, methods to generate multi-contact behaviors are introduced. They display robust behavior even as the contacts are non-coplanar. However, they struggle to generate dynamic motions. Finally, region-based approaches are discussed. They are time-consuming and often cannot be computed online. However, they offer a wider range of motions compared to target CoM trajectories.

Chap. 2 and 3 of this thesis aim at devising methods to compute dynamic behaviors in multi-contact scenarios. In particular, Chap. 2 presents a faster method to calculate the robust CoM region introduced in Audren and Kheddar (2018), allowing it to be used online for the robot to accommodate for non-fixed contacts. Chap. 3 introduces a way to optimize the wrench distribution and CoM placement while having sliding non-coplanar contacts.

Section 1.2 presents most works on the topic of tool-changers applied to robotics. We found no work in that domain directly using humanoid robots. However, summarizing these contributions leads to outlining several requirements for the design of a proper end-effector changer for humanoids.

Chap. 4 and 5 build upon this by first implementing existing commercial tool-changers in humanoids and then by proposing a new design aimed at filling a gap in the existing solutions.

Section 1.3 adds details about related works and the framework used to prepare the demonstration of each contribution presented in this thesis and used in all chapters.

CHAPTER 2

CoM Balance Region for Scenarios with Multiple Non Coplanar Fixed and Moving Contacts

In our daily life, we often use our weight to move heavy objects such as doors or furniture. By doing so, our balance relies on the moving contact, and we may fall if the object moves unexpectedly. This chapter aims at implementing such behavior for the humanoid robot. It will balance while relying on non-coplanar and moving contacts. For this purpose, we propose to use a Center of Mass (CoM) balance region.

A balance region is defined such that the CoM of the robot should stay inside during its motion to avoid falling. The balance region computation is based on centroidal dynamics derived from the Newton-Euler equations with the friction cones inequalities constraints. It is computed exactly and is passed to the whole-body controller to constrain to CoM within the computed region, without explicitly specifying its target.

As discussed in Sec. 1.1, the balance region has a major role in the balance of humanoid robots in multi-contact scenarios. In particular, Bretl and Lall (2008) proposed an iterative ray shooting (projection) algorithm to compute an inclusion region for the Center of Mass (CoM) position. The CoM acceleration is considered null in Bretl and Lall (2008) and bounded in a convex-set in Audren and Kheddar (2018). We recall these methods in Section 2.1.

The main disadvantage of geometric balance regions lies in their computational cost; they are not useful when any contact location undergoes motion tasks, e.g. pushing, sliding, etc. The main contribution of this chapter is to propose a faster version of the iterative algorithms from Audren and Kheddar (2018) to use it as a balance criterion when contacts move, see Section 2.2.

Experiments demonstrating the capabilities of our approach are shown in Section 2.4; see Fig. 2.1 illustrating the experimental set-up.

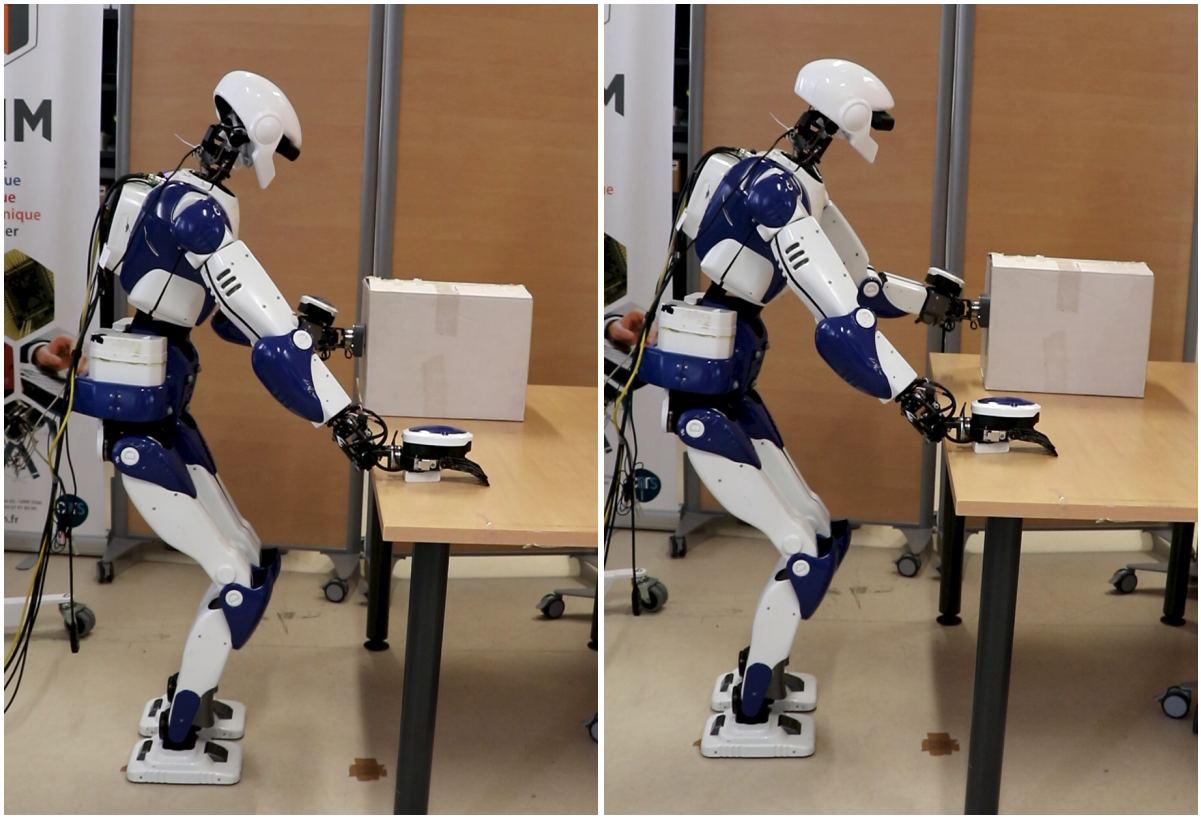


FIGURE 2.1 – *HRP-4 pushing a* 12.5 *kg box. It produced a force of up to* 30 *N horizontally with its left hand while controlling the remaining fixed contacts during the pushing phase.*

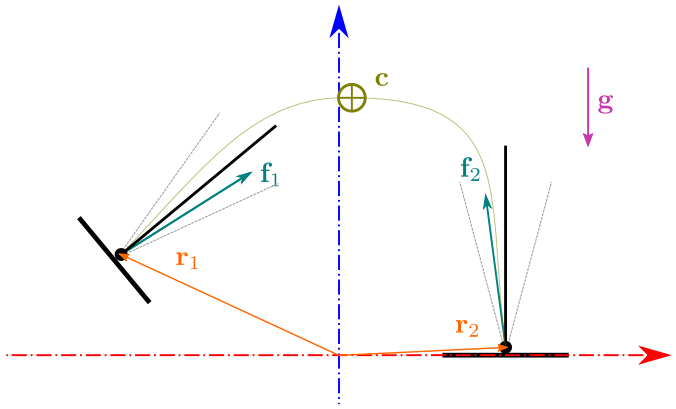


FIGURE 2.2 – *Schematic of the centroidal model of the robot*

2.1 Centroidal Model and CoM Balance Region

Let $\mathbf{c} \in \mathbb{R}^3$ be the CoM position of a floating base multi-limb robot and $m > 0$ its mass (See Fig. 2.2). The robot interacts with its environment through a contact set $\mathcal{F}$ of n unilateral contact points. For each contact point i, $\mathbf{r}_i \in \mathbb{R}^3$ is its contact position, $\mathbf{u}_i \in \mathbb{R}^3$ its normal, $\mathbf{f}_i \in \mathbb{R}^3$ the force applied by the environment on the robot at the contact point and μ_i its friction coefficient. Each contact is supposed to be ponctual, thus locally no torque is transmitted. Surface contacts are modeled by several ponctual contacts, one at each vertice. Newton-Euler

equations governing the CoM motion writes,

$$\sum_{i=1}^{n}\mathbf{f}_i+m\mathbf{g}=m\ddot{\mathbf{c}} \tag{2.1}$$

$$\sum_{i=1}^{n}\mathbf{r}_i\times\mathbf{f}_i+\mathbf{c}\times m\mathbf{g}=\mathbf{c}\times m\ddot{\mathbf{c}} \tag{2.2}$$

where g is the gravity vector; the moment equation is expressed w.r.t. the world's origin and assumes neglectible angular momentum.

The friction force for the contact point i is given by:

$$\mathbf{u}_i^T\mathbf{f}_i\geq 0 \tag{2.3}$$

$$\|(\mathbb{I}-\mathbf{u}_i\mathbf{u}_i^T)\mathbf{f}_i\|\leq\mu_i\mathbf{u}_i^T\mathbf{f}_i \tag{2.4}$$

An equilibrium can be described by the vector $(\mathbf{f}^T,\mathbf{c}^T)^T$ with $\mathbf{f}=(\mathbf{f}_1^T,...,\mathbf{f}_n^T)^T\in\mathbb{R}^{3n}$ such that equations (2.1) to (2.4) hold. Let $\mathcal{X}$ be the set of all equilibrium:

$$\mathcal{X}=\left\{\begin{pmatrix}\mathbf{f}\\ \mathbf{c}\end{pmatrix}\in\mathbb{R}^{3n}\times\mathbb{R}^3|(2.1)-(2.4)\right\} \tag{2.5}$$

Implementing $\mathcal{X}$ as constraints in planning or control frameworks enforces no sliding (even in a moving contact) and no contact break. Notice that the set of constraints is non-linear because of the cross product in (2.2). We aim to compute fast the boundaries of the equilibrium set in the 3D CoM space, i.e., finding the edges of the convex set:

$$\mathcal{Y}=\{\mathbf{c}\in\mathbb{R}^3|\exists\mathbf{f}\in\mathbb{R}^{3n},(\mathbf{c},\mathbf{f})\in\mathcal{X}\}$$

$\mathcal{Y}$ is image (also referred as projection) of $\mathcal{X}$ through the projection operator:

$$\begin{array}{ccc}\mathbb{R}^{3n}\times\mathbb{R}^3 & \rightarrow & \mathbb{R}^3\\ \begin{pmatrix}\mathbf{f}\\ \mathbf{c}\end{pmatrix} & \mapsto & \begin{pmatrix}\mathbf{0}_{3\times 3n} & \mathbb{I}_{3\times 3}\end{pmatrix}\begin{pmatrix}\mathbf{f}\\ \mathbf{c}\end{pmatrix}=\mathbf{c}\end{array} \tag{2.6}$$

This problem is studied in Bretl and Lall (2008) for static balance, i.e., $\ddot{\mathbf{c}}=\mathbf{0}$. The resulting region does not depend on the CoM's height c_z, nor on the mass of the robot Caron and Kheddar (2016). The equilibrium region is infinite in the direction parallel to gravity (noted z) with a constant projection on the horizontal plane (this is because bounds on torques and kinematic constraints are not accounted for in the centroidal model). They proposed to compute the cross-section only. The resulting equilibrium region limits the CoM position in the horizontal plane.

In Samy et al. (2017), the static balance region is extended to account for the effector torque limits that is also implemented in Orsolino et al. (2020). Using the projection algorithm in Bretl and Lall (2008), the balance region is computed at each control-loop iteration of their quadruped robot having at most four-point contacts. However, in almost all practical situations, the contacting surfaces cannot be approximated as a single point contact. Instead,

the contacting surface can be modeled as a set of contact points representing the vertices of the contacting surface discrete boundaries Caron et al. (2015a).

In Audren and Kheddar (2018), the followings are considered: (i) the CoM acceleration is bounded by a predefined convex set; (ii) each component of the contact force is constrained to be lower than the weight of the robot ($\|\mathbf{f}_i\|_\infty \leq m\mathbf{g}$); and (iii) the CoM position is constrained in a sphere of radius $c_{\max}$. That case is referred to as the robust equilibrium case as the CoM acceleration can correspond to external perturbations. The resulting 3D polytope depends on the mass of the robot and is used to constrain the CoM of a humanoid robot in motion.

In these works, the equilibrium balance region is obtained through an iterative algorithm based on a cutting plane method to approach $\mathcal{Y}$ using two convex polytopes $\mathcal{Y}_{\text{inner}}$ and $\mathcal{Y}_{\text{outer}}$, such that $\mathcal{Y}_{\text{inner}} \subseteq \mathcal{Y} \subseteq \mathcal{Y}_{\text{outer}}$ and all vertices of $\mathcal{Y}_{\text{inner}}$ lie on the boundary of $\mathcal{Y}$ and all faces of $\mathcal{Y}_{\text{outer}}$ are supporting planes of $\mathcal{Y}$ at the vertices of $\mathcal{Y}_{\text{inner}}$.

In order to find points that are on the boundary of $\mathcal{Y}$ and the corresponding supporting planes, the following Second Order Cone Programming (SOCP) problem is formulated:

$$\begin{aligned} \max_{\mathbf{c},\mathbf{f}} \quad & \mathbf{d}^T\mathbf{c} \\ \text{s.t.} \quad & \begin{pmatrix}\mathbf{f}\\ \mathbf{c}\end{pmatrix} \in \mathcal{X} \end{aligned} \tag{2.7}$$

where $\mathbf{d} \in \mathbb{R}^3$ is a search direction. The iterative projection algorithm (Algorithm 1) returns the inner approximation polytope $\mathcal{Y}_{\text{inner}}$ of the convex set $\mathcal{Y}$, see e.g. Fig. 2.3. Then its Half-space Representation (H-Rep) is employed to constrain the CoM position in a planning or control framework. The iterative algorithm is further detailled in next section.

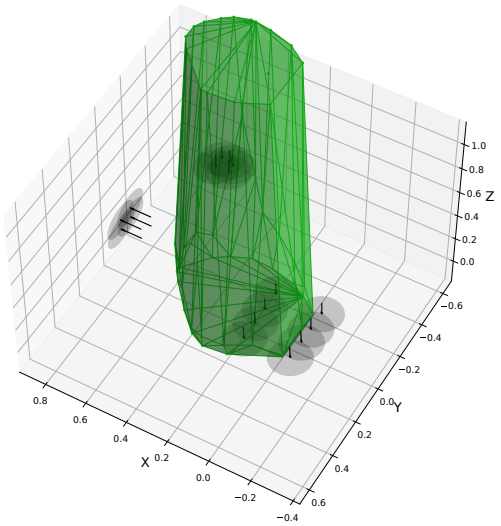
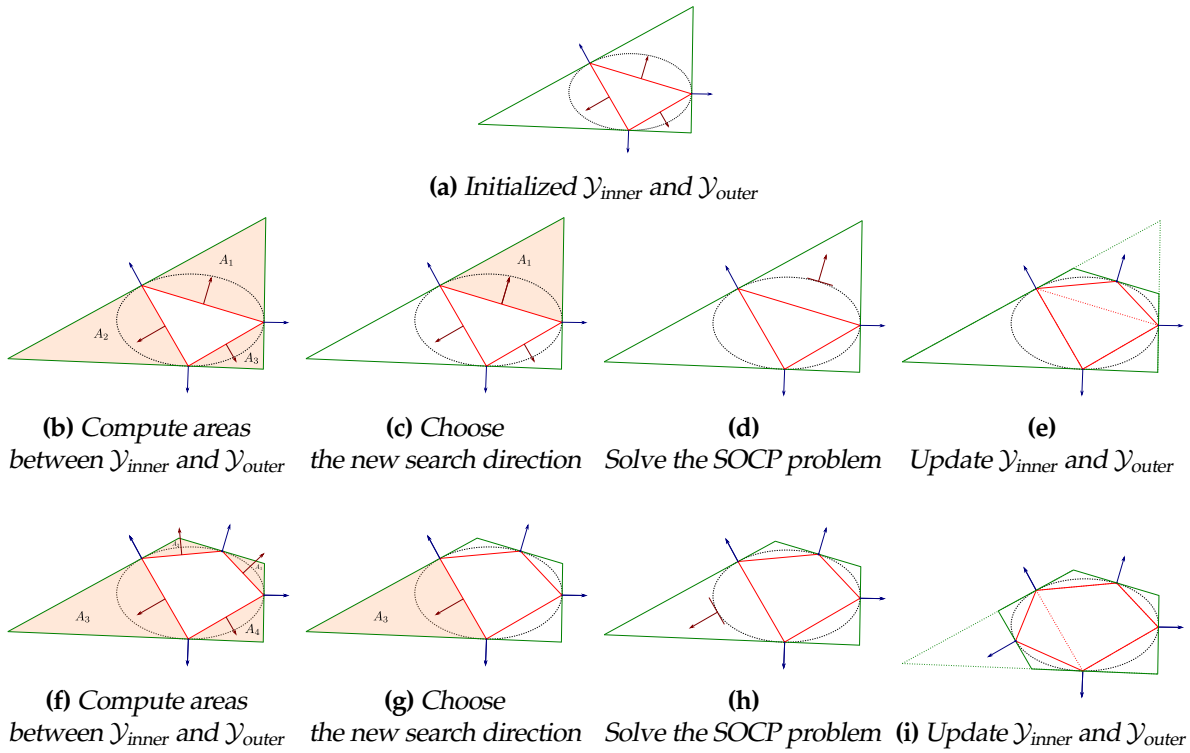


FIGURE 2.3 – *The balance CoM region for a robot in a contact configuration similar to Fig. 2.1. The CoM acceleration is bounded to* ± 0.4 *m.s*$^{-2}$ *in* x*-axis,* ± 0.3 *m.s*$^{-2}$ *in* y*-axis and* -9.81 ± 0.3 *m.s*$^{-2}$ *in* z*-axis. The black arrows are the contact used for the computation, each one of them is a vertex of a contact surface. The region is truncated at* $0 \leq z \leq 2$ *m*

2.2 Iterative Projection Algorithm

The iterative algorithm presented in Audren and Kheddar (2018) is recalled in Algorithm 1. Figure 2.4 illustrates it in the static case presented in Bretl and Lall (2008). In the static case, the surface area is used as a measure instead of the volume, and search directions and CoM positions are 2D instead of 3D.

The initialization happens as follows. First, 4 non-colinear directions d_i are chosen. Using the SOCP problem (Eq. 2.7), 4 initial CoM positions $\mathbf{c}_i$ are found on the boundary of $\mathcal{Y}$. The initial inner approximation $\mathcal{Y}_{\text{inner}}$ of $\mathcal{Y}$ is the convex hull of the initial vertices. The Vertice representation (V-rep) is obtained by linking the vertices to form a tetrahedron. The outer approximation $\mathcal{Y}_{\text{outer}}$ is obtained in its H-rep with the search directions d_i as the faces normals and the corresponding $\mathbf{c}_i$ to compute the faces offsets. The result is also a tetrahedron. Both V-rep and H-rep are computed for both $\mathcal{Y}_{\text{inner}}$ and $\mathcal{Y}_{\text{outer}}$ as each iteration uses them. Also, the final return approximation of $\mathcal{Y}$ is $\mathcal{Y}_{\text{inner}}$, which is usually used under its H-rep form. The initialization of the double description in both cases is straightforward, as both are tetrahedrons. The initial $\mathcal{Y}_{\text{inner}}$ and $\mathcal{Y}_{\text{outer}}$ are illustrated in Fig. 2.4a.



(a) *Initialized $\mathcal{Y}_{inner}$ and $\mathcal{Y}_{outer}$*

(b) *Compute areas between $\mathcal{Y}_{inner}$ and $\mathcal{Y}_{outer}$*

(c) *Choose the new search direction*

(d) *Solve the SOCP problem*

(e) *Update $\mathcal{Y}_{inner}$ and $\mathcal{Y}_{outer}$*

(f) *Compute areas between $\mathcal{Y}_{inner}$ and $\mathcal{Y}_{outer}$*

(g) *Choose the new search direction*

(h) *Solve the SOCP problem*

(i) *Update $\mathcal{Y}_{inner}$ and $\mathcal{Y}_{outer}$*

FIGURE 2.4 – *Illustration of the projection algorithm for the static case with initial state and the following two iteration (One per line). In dotted black lines is $\mathcal{Y}$, in red is $\mathcal{Y}_{inner}$ and in green is $\mathcal{Y}_{outer}$.*

We describe the improvements made in the computation of the dynamic equilibrium region to achieve higher time performances. The proposed modifications for Algorithm 1 are the followings:

- Step $*$: Linearizing the constraints and use Linear Programming (LP) instead of SOCP (Section 2.2.1);
- Computing support functions instead of volumes $\mathcal{V}$ (Section 2.2.2);

Algorithm 1 Iterative Projection Algorithm

Require: Robot's contact set $\mathcal{F}$
Require: $\mathcal{V}(\S)$ computes volume of $\S$
Initialize $\mathcal{Y}_{\text{inner}}$ and $\mathcal{Y}_{\text{outer}}$
while $\mathcal{V}(\mathcal{Y}_{\text{outer}}) - \mathcal{V}(\mathcal{Y}_{\text{inner}}) > \epsilon$ **do**
– $\forall$ face $H_i \in \mathcal{Y}_{\text{inner}}$, compute $\mathcal{V}_i(\mathcal{Y}_{\text{outer}}) \cap H_i$
– face $i \leftarrow \max \mathcal{V}_i$.
– new search direction $\mathbf{d}^*$ normal to the face i.
$*$ find CoM $\mathbf{c}^*$ from SOCP problem (2.7) for $\mathbf{d}^*$:

$$\max_{\mathbf{f},\mathbf{c}} \mathbf{d}^{*T}\mathbf{c} \quad \text{s.t.} \begin{pmatrix} \mathbf{f} \\ \mathbf{c} \end{pmatrix} \in \mathcal{X}$$

$+$ Update $\mathcal{Y}_{\text{inner}}$ by adding $\mathbf{c}^*$
$\times$ Update $\mathcal{Y}_{\text{outer}}$ by adding the plane of normal $\mathbf{d}^*$ and passing through $\mathbf{c}^*$
end while
return $\mathcal{Y}_{\text{inner}}$

- Step $+$: Warm starting the inner update (Section 2.2.3);
- Step $\times$: Warm starting the outer update (Section 2.2.3).

2.2.1 Linearization of the constraints

Linearization is a rather common practice and has already been applied to Algorithm 1 in Caron and Kheddar (2016); Orsolino et al. (2020). Linearizing the constraint (2.4) allows using Linear Programming (LP) instead of Second Order Cone Programming (SOCP) proposed in Bretl and Lall (2008); Audren and Kheddar (2018). The linearized friction cones are given by:

$$\frac{(\mathbb{I} - \mathbf{u}_i \mathbf{u}_i^T)\mathbf{f}_i}{\mu_i \mathbf{u}_i^T \mathbf{f}_i} \in \mathcal{P}_n \tag{2.8}$$

where $\mathcal{P}_n$ is the regular n-sided polygon inscribed in the unit circle. In addition, the maximal normal contact forces are bounded by $f_{\max}$:

$$\mathbf{u}_i^T \mathbf{f_i} \leq f_{\max} \tag{2.9}$$

$\mathbf{u}_i$ is the normal of the i-th contact. $f_{\max}$ depends on the configuration as it is related to joint torques limits Orsolino et al. (2020). In this study, $f_{\max}$ is taken constant.

Moreover, contrarily to bounding the CoM to a sphere Audren and Kheddar (2018), it can be determined using linear constraints:

$$A\mathbf{c} \leq \mathbf{b} \tag{2.10}$$

This new constraint results in a convex set $\mathcal{X}_{\text{linear}} \subseteq \mathcal{X}$:

$$\mathcal{X}_{\text{linear}} = \left\{ \begin{pmatrix} \mathbf{f} \\ \mathbf{c} \end{pmatrix} \in \mathbb{R}^{3n+3} | \text{(2.1),(2.2),(2.3),(2.8)} \right\} \tag{2.11}$$

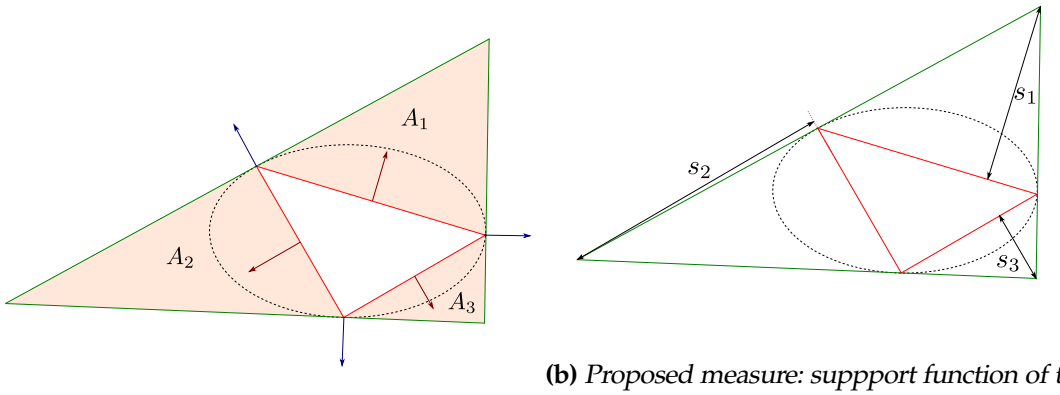
The projection in the CoM position space becomes:

$$\mathcal{Y}_{\text{linear}} = \left\{ \mathbf{c} \in \mathbb{R}^3 | \exists \mathbf{f} \in \mathbb{R}^{3n}, \begin{pmatrix} \mathbf{f} \\ \mathbf{c} \end{pmatrix} \in \mathcal{X}_{\text{linear}} \right\} \tag{2.12}$$

Constraint (2.8) being stricter than constraint (2.4) (i.e. for all $\mathbf{f}$ such that (2.8) holds then (2.4) holds as well), we have $\mathcal{X}_{\text{linear}} \subset \mathcal{X}$ and $\mathcal{Y}_{\text{linear}} \subset \mathcal{Y}$. Finally we replace the SOCP with the following LP:

$$\begin{aligned} \max_{\mathbf{c},\mathbf{f}} \quad & \mathbf{d}^T \mathbf{c} \\ \text{s.t.} \quad & \begin{pmatrix} \mathbf{f} \\ \mathbf{c} \end{pmatrix} \in \mathcal{X}_{\text{linear}} \end{aligned} \tag{2.13}$$

2.2.2 Support Point and Support Functions



(a) *Original measure: area of the regions*

(b) *Proposed measure: suppport function of the regions*

FIGURE 2.5 – *Illustration of the measures used to choose the next cutting direction. The regions corresponds to the cuts of $\mathcal{Y}_{outer}$ with the different faces H_i of $\mathcal{Y}_{inner}$.*

The first step of each iteration in Audren and Kheddar (2018) consists of computing the volumes cut off $\mathcal{Y}_{\text{outer}}$ by the faces of $\mathcal{Y}_{\text{inner}}$ (Fig. 2.5a). However, computing the volume of a polytope is time-consuming Büeler et al. (2000).

Let H_i be a face of $\mathcal{Y}_{\text{inner}}$. $\mathbf{d}_{H_i} \in \mathbb{R}^3$ and $b_{H_i} \in \mathbb{R}$ are respectively the normal and offset of face H_i such that

$$\mathcal{Y}_{\text{inner}} \subset H_i^- = \{ \mathbf{c} \in \mathbb{R}^3 | \mathbf{d}_{H_i}^T \mathbf{c} \leq b_{H_i} \} \tag{2.14}$$

$\mathbf{c}_{H_i} \in \mathcal{Y}_{\text{outer}}$ is a *support point* w.r.t. face H_i if

$$\forall \mathbf{c} \in \mathcal{Y}_{\text{outer}}, \quad \mathbf{d}_{H_i}^T \mathbf{c} \leq \mathbf{d}_{H_i}^T \mathbf{c}_{H_i} \tag{2.15}$$

$\mathbf{c}_{H_i} \in \mathcal{Y}_{\text{outer}}$ is the furthest point outside H_i^- and inside $\mathcal{Y}_{\text{outer}}$. Then, the *support function* s_i (Fig. 2.5b) is the euclidean distance from a face of $\mathcal{Y}_{\text{inner}}$ to its corresponding support point:

$$s_i = \mathbf{d}_{H_i}^T \mathbf{c}_{H_i} - b_{H_i} \tag{2.16}$$

As $\mathcal{Y}_{\text{outer}}$ is a bounded convex polytope described by a finite amount of vertices, $\mathbf{c}_{H_i}$ is one of its vertices. It can be found using a hill climbing method. The search direction for the next iteration of the algorithm is the normal of the inner approximation's face with the maximum support function.

In Bretl and Lall (2008); Audren and Kheddar (2018) proof of convergence using area/volume difference respectively between the inner and outer regions are provided. They use this difference as a stopping criterion. Instead, we propose a different metric that uses support functions and surfaces area instead of volumes:

$$\sum_{i \in [1,m]} \frac{1}{3} s_i \mathcal{A}_i \tag{2.17}$$

where m is the number of faces of $\mathcal{Y}_{\text{inner}}$, s_i and $\mathcal{A}_i$ are respectively the support function and the area of i-th face. By construction, each face of $\mathcal{Y}_{\text{inner}}$ is a triangle thus $\mathcal{A}_i$ is straighforward to compute. For each face i, $\frac{1}{3} s_i \mathcal{A}_i$ corresponds to the volume of the tetrahedron of the base and height of the support function.

This method is at a time fast and easy to compute. However, as only a lower bound of the error is provided, it cannot be used to prove convergence explicitly. Nonetheless, extensive simulations show good results and similar rates of convergence w.r.t the previous methods.

2.2.3 Updating the inner and outer approximations

Updating the inner and outer polytopes corresponds to iterative face enumeration and iterative vertex enumeration problems.

$\mathcal{Y}_{\text{inner}}$ is updated by removing its faces that rule the new vertex as outside. If an edge is between two removed faces, it is also removed. Then, new triangular faces are built using the edges of the removed faces and the new vertex. Because $\mathcal{Y}_{\text{inner}}$ is convex, all the faces to be removed are neighbors (i.e. connected by an edge). The face with the normal used to find the new vertex is tested first.

Conversely, $\mathcal{Y}_{\text{outer}}$ is updated by removing the vertices outside the new face and adding new ones where an edge would pierce the new face. Again, because $\mathcal{Y}_{\text{outer}}$ is convex, the vertices to be removed are neighbors, and the support point corresponding to the search direction is the first tested.

Warm starting the updates with the support face and support point allows us to implement efficiently the double description in this application.

2.2.4 Performance Comparison

To compare fairly our method to the original one Audren and Kheddar (2018), both are implemented in Python. The contact sets obtained during the simulation of a scenario of the robot pushing a box is used. In total, 893 different contact sets were used. The number of contacts is from 2 to 4 surface contacts corresponding to 8 to 17 point contacts. For instance

Fig. 2.1 represents a robot with 4 surface contacts, but the model used for the computation consists of 17 contacts points as shown in Fig. 2.3. Indeed, the right hand surface contact is not a square and has 5 vertices. The computation performances depend on the number of contact points; see Table 2.1.

TABLE 2.1 – *Timings in ms*

Method	SOCP and Volumes			Linearized and Support		
Nbr of contacts	8	13	17	8	13	17
Solve	438	1219	1977	44	202	332
Measure	24	35	32	3	7	7
Update	9	12	12	6	11	12
Total	473	1268	2022	54	222	352

First of all, our implementation of the original method in Audren and Kheddar (2018) (SOCP and Volumes in Table 2.1) show results that are even slightly better than those reported in Audren and Kheddar (2018), which is in favor of a fair comparison.

Overall, our proposed method is 6 to 9 times faster w.r.t the original one. The main gain comes from the use of LP instead of SOCP. Changing the measure decreased its computation time from 5 to 8 times. As for updating the inner and outer approximation, our method gives slightly better results w.r.t the original one. Finally, for the final version, a `C++` implementation of the algorithm is used, and the computation time performances are halved: 25 ms for two surface contacts, 80 ms for three surface contacts, and 175 ms for four surface contact.

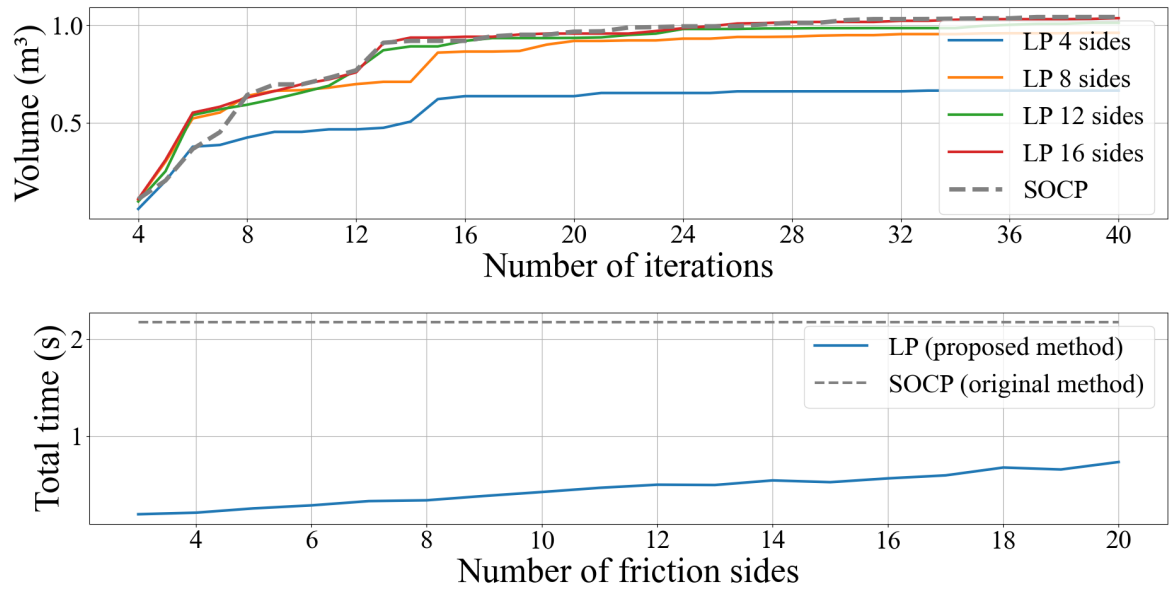


FIGURE 2.6 – *Impact of linearization; Top: evolution of the volume of the approximation w.r.t. the number of iteration of the projection algorithm for different number of sides for the linearization of the friction cone. Bottom: Evolution of the computation time w.r.t. the number of friction sides for the linearization of the friction cone*

Furthermore, The region generated with linearized friction cones tends to be smaller in volume compared to the original one when the number iterations increase (Fig. 2.6 top).

Thus the result from our method is slightly more conservative than the previous methods. Increasing the number of side for the linearization of the friction cones make the linearized region tend to the non-linearized one but the computation time will increase (Fig. 2.6 bottom). We found that a number of side between 6 and 8 is a good compromise.

The `C++` library implementing the iterative projection algorithm is available online [1]

2.3 Integration in the Whole-Body Control

Our proposed CoM balance region computation is introduced as a constraint in a task-space whole-body control framework introduced in Sec. 1.3.1. However, as discussed in Section 2.3.1, this constraint alone does not warrant that contact forces regulation tasks are achieved. Thus, in Section 2.3.2, a CoM admittance task is introduced such that the CoM moves to a position allowing for better tracking of the desired contact forces.

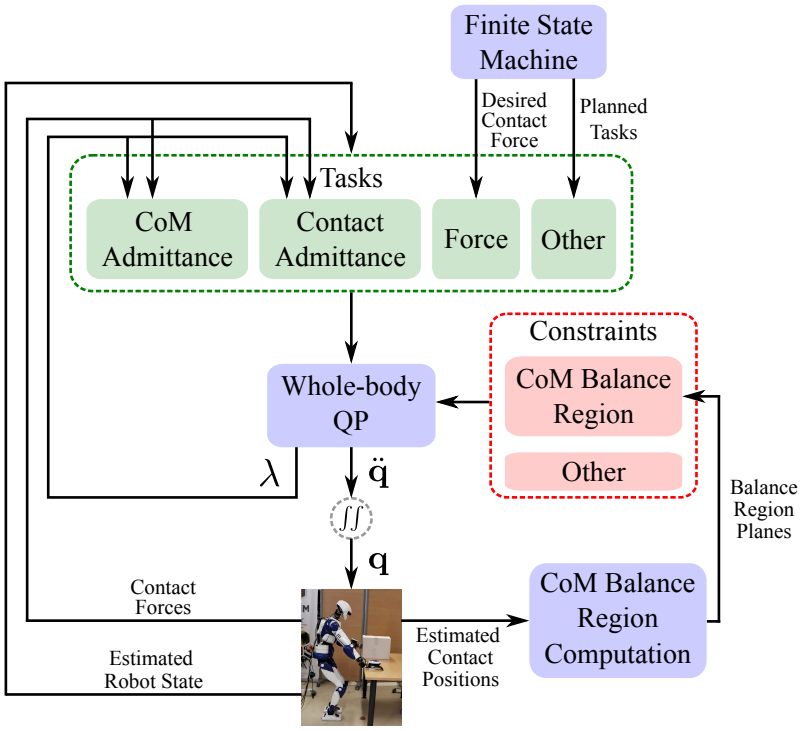


FIGURE 2.7 – *Tasks and constraints are formulated as function of the generalized acceleration coordinates $\ddot{\mathbf{q}}$ and contact force λ. The targets of the tasks are given by a FSM or the WBQP's output from previous iteration and are compared to the estimated state of the robot. Using the estimated contact position the balance region is computed and used as constraints.*

2.3.1 Whole Body Balance Framework

Our task-space Whole-Body Quadratic Programming (WBQP) framework Vaillant et al. (2016) formulates quadratic task objectives and constraints as a function of the generalized

1. https://github.com/JulienRo7/stabiliplus

acceleration coordinates and contact forces. Figure 2.7 illustrates our global framework: a Finite State Machine (FSM) schedules tasks on demand of the objectives and the used robot, and parametrize these tasks such as contact force, end-effector position, posture... with their respective targets. As our robot is controlled in a high gain position, only the computed joint accelerations are used as commands after being integrated twice. Thus, for the robot to track the desired contact forces, we use admittance tasks as in Bouyarmane et al. (2019b). The balance region is integrated as planar inequality constraints in the CoM position as in Audren et al. (2016). Other concurrent constraints include joint limits (position and velocity), non-desired collision avoidance, and contact constraints.

The CoM is allowed to live freely within the computed balance region. In practice, a more refined control of CoM position is required, e.g. Samadi et al. (2021); Abi-Farraj et al. (2019); Caron et al. (2019) and Chap. 3. Consider a task requiring the robot to apply a contact force $\mathbf{f}$ to its environment. To generate such a force and fulfill the dynamic constraint, the WBQP will adopt two strategies: (i) compensating $\mathbf{f}$ with internal forces and other contacts forces, and (ii) generating a CoM acceleration $\ddot{\mathbf{c}}$ in the opposite direction of $\mathbf{f}$. The former solution drives the contact wrenches close to their corresponding friction cones, leading to undesired contact slips. The latter can only be applied for a short amount of time before the CoM reaches the limit of the balance region.

A more suitable solution is moving the CoM to a position where the moment from gravity compensates for $\mathbf{f}$. However, that solution cannot be generated by the WBQP since it does not immediately decrease the error on the force task. For example, when the robot is pushing a box, as in Fig. 2.1 to generate enough force on the box, the robot needs to move its CoM forward. It has to decrease the pressure on the box, which is in opposition to reaching the target $\mathbf{f}$ and increasing the forces from the feet to move the CoM forward. Thus, a task is added for the WBQP to favor this behavior.

2.3.2 Formulation of a CoM Admittance Task

The admittance control is a common strategy to regulate dynamic force values for position-controlled robots, such as contact force Bouyarmane et al. (2019b) or ZMP Caron et al. (2019). We formulate a CoM admittance task applicable to a given measured force as,

$$\dot{\mathbf{c}} = \mathbf{K}(\mathbf{f}_{\text{tar}} - \mathbf{f}_{\text{mes}}) \tag{2.18}$$

where $\mathbf{f}_{\text{tar}}$ and $\mathbf{f}_{\text{mes}}$ are the target and measured contact forces in the global frame, respectively. The objective is to express the matrix $\mathbf{K}$ such that the generated CoM velocity moves the CoM to decrease the force error.

Starting from equation (2.2) and considering no CoM acceleration, the moment generated only by the supposed contact and the momentum from gravity, we get:

$$\mathbf{r} \times \mathbf{f} + \mathbf{c} \times m\mathbf{g} = \mathbf{0} \tag{2.19}$$

where $r = [x,y,z]^T$ the contact position, $\mathbf{f} = [f_x,f_y,f_z]^T$ the contact force, $\mathbf{c} = [c_x,c_y,c_z]^T$ the CoM position and $\mathbf{g} = [0,0,-g]^T$ with $g = 9.81$ m.s^{-2}. Then by projecting on the x and y axes:

$$yf_z - zf_y = mc_y g \tag{2.20}$$
$$zf_x - xf_z = -mc_x g \tag{2.21}$$

and writing,

$$\Delta \mathbf{f} = \mathbf{f}_{\text{tar}} - \mathbf{f}_{\text{mes}} = \begin{bmatrix} \Delta f_x \\ \Delta f_y \\ \Delta f_z \end{bmatrix} \text{ and } \Delta \mathbf{c} = \begin{bmatrix} \Delta c_x \\ \Delta c_y \\ \Delta c_z \end{bmatrix} \tag{2.22}$$

where $\Delta \mathbf{c}$ is the desired CoM displacement, we derive:

$$\Delta c_y = \frac{y}{mg} \Delta f_z - \frac{z}{mg} \Delta f_y \tag{2.23}$$

$$\Delta c_x = \frac{x}{mg} \Delta f_z - \frac{z}{mg} \Delta f_x \tag{2.24}$$

then $\mathbf{K}$ is given by

$$\mathbf{K} = k_{\text{ad}} \begin{bmatrix} -\frac{z}{mg} & 0 & \frac{y}{mg} \\ 0 & -\frac{z}{mg} & \frac{x}{mg} \\ 0 & 0 & 0 \end{bmatrix} \tag{2.25}$$

where k_{ad} is tuned from experimental data.

The target CoM velocity computed in equation (2.18) is then employed as reference velocity in a CoM task with no stiffness but damping alone (which could be considered as a CoM velocity task).

2.4 Experiments

We assess our implementations with multi-contact experiments using the HRP-4 humanoid robot. Two experiments are conducted: (i) regulation of the right-hand contact force (Section 2.4.1) and (ii) pushing a 12.5 kg box on a table (Section 2.4.2). A video of the following experiments is available online [2].

2.4.1 Regulation of contact force

In this experiment, HRP-4 establishes contact on a table with its right hand, and the normal target force is smoothly increased to 50 N. Then the desired force is set to a cosine curve down to 17 N and back up to 50 N. Finally, the normal target force is evenly decreased to 0 N to remove the contact. Figure 2.8 illustrates the tracking of the normal force as well as the CoM trajectory and velocity. Here the right-hand surface is horizontal, and the contact's normal is aligned with the gravity vector. The contact is in front of the robot and off-centered to the right. Thus, notice that when the target force increases and becomes higher than the measured force, the CoM moves forward and to the right. On the other hand, when the target force decreases and becomes lower than the measured force, the CoM moves backwards and to the left.

2. https://youtu.be/QLtO3rb4_NY

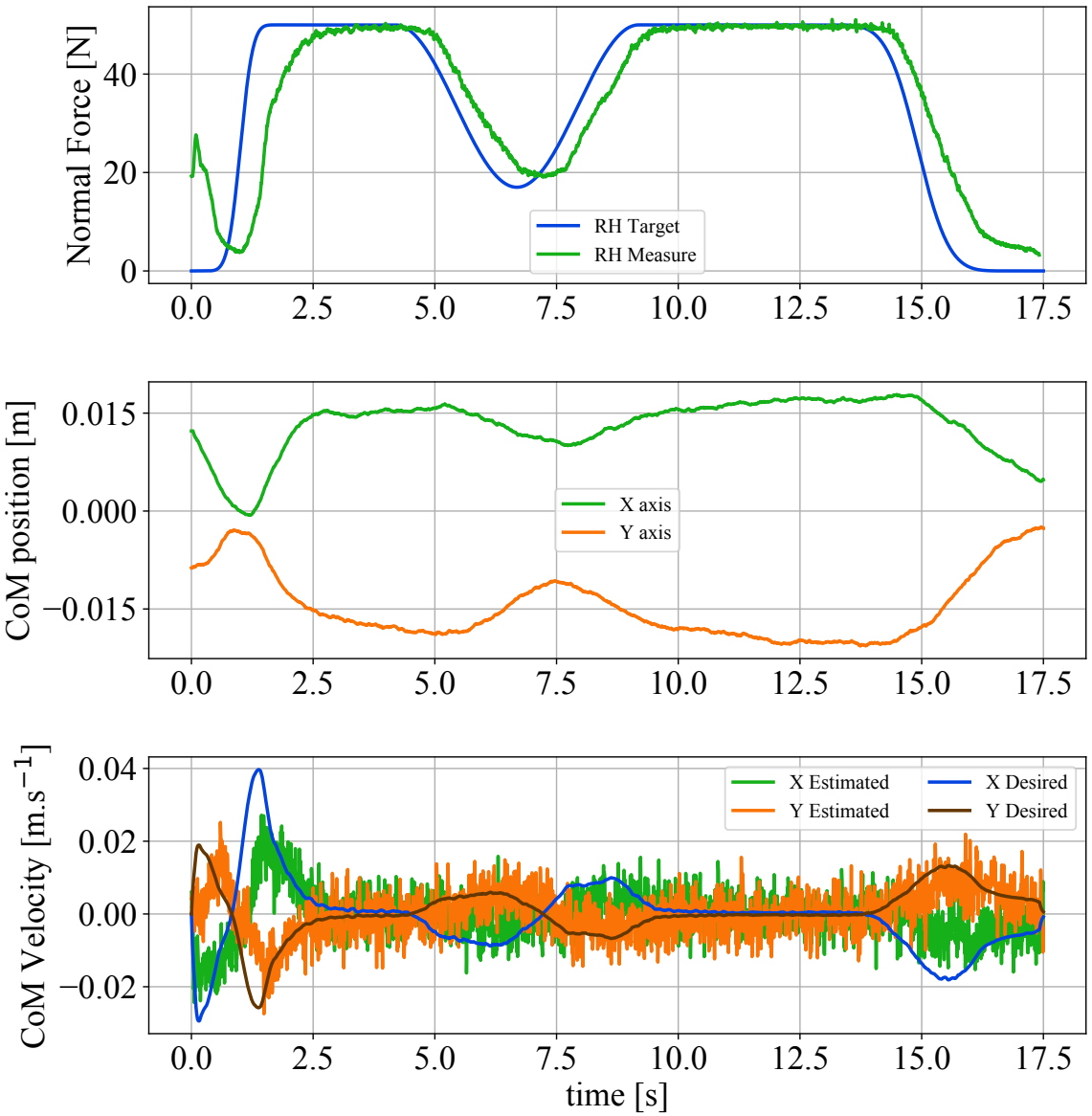


FIGURE 2.8 – *Right-hand force regulation. HRP-4 right hand is on a* 70 *cm high horizontal table around* 40 *cm in front of the standing robot and off centered by* 30 *cm to the right. The left hand is not in contact.*

2.4.2 Box pushing in multi-contact

Figure 2.1 illustrates the second experiment. The robot first establishes contact with its right hand on the table as in the previous experiment, but regulate its force to 37.5 N. Then it establishes contact with the box of mass 12.5 kg using its left hand. The normal force is regulated to 26 N. Then a position task is used to move the box 15 cm forward. Ultimately, both contacts are removed (first left hand then right hand) after their normal forces have been decreased to 0 N. Figure 2.9 shows the resulting force tracking, CoM trajectory and CoM velocity. The behavior concerning the right hand is similar to the previous scenario. However, for the left-hand contact, the contact normal is parallel to the global frame x-axis. Thus when controlling the normal force, motion of the CoM on the y-axis has no impact,

and only motions along the x-axis are used (as a result of (2.25)). This is noticeable in the CoM velocity curve in Fig. 2.9: from 7.5 to 19 s the desired velocity on the y-axis is almost zero while the velocity on the x-axis varies a lot. Notice that the regulation of the right hand is correct, and thus almost no CoM motion is generated by the right-hand contact. Figure 2.10 shows this experiment performed with a lighter box (7.5 kg). In that case, the box

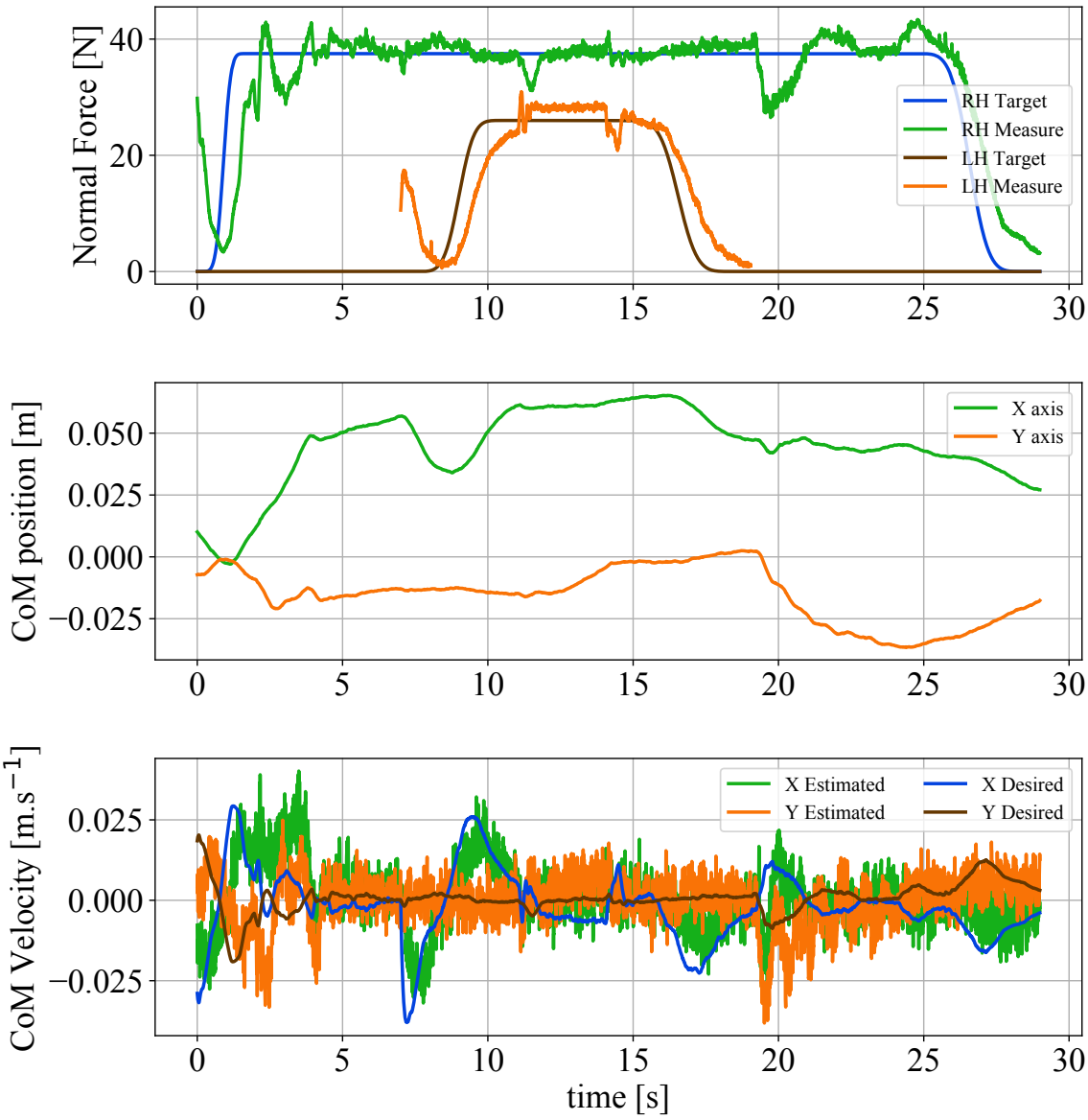


FIGURE 2.9 – *Box pushing in multi-contact (Fig. 2.1): the right hand is in the same position as the first experiment (Fig. 2.8); the left hand is in contact with a box on the table. When the contact is established, the hand is* 85 *cm high,* 40 *cm in front of the robot and* 30 *cm to its left. The box is* 12.5 *kg, and during motion, it moves* 15 *cm forward.*

starts to slide before the normal target force is reached. As a consequence, the CoM moved forward during the whole phase, where the left contact was established until the target was decreased. This happens because we do not identify the force required to move the box online. It is estimated from a prior experiment that in turn used an empirical value as target.

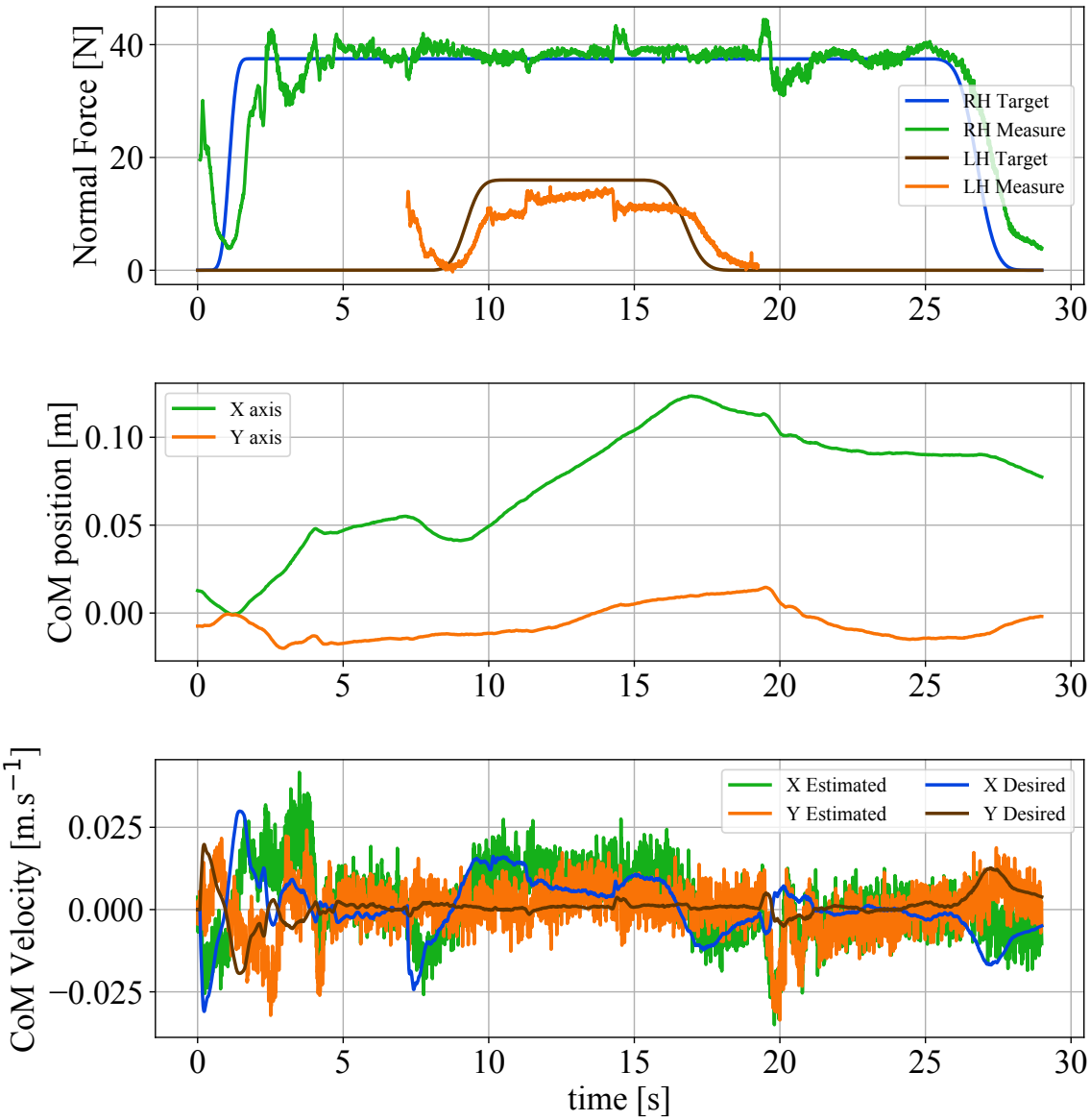


FIGURE 2.10 – *The same experiment as Fig. 2.9 with a box mass of 7.5 kg: the target force on the left hand is not reached (since the box can be moved with a lower force); thus, the CoM keeps moving forward until the target is decreased to remove the contact.*

2.4.3 Evolution of the balance region

When there is a discrete change in the contact set, i.e. when a contact is added or removed, the balance region changes abruptly. To generate a smooth transition, the upper and lower bounds for each contact's normal force is increased or decreased progressively as suggested in Park and Khatib (2006). This allows the balance region to evolve almost continuously. Figure 2.11 illustrate the evolution of the balance region when adding the right-hand contact.

Our task-based WBQP runs at 5 ms. Meanwhile, the computation of the balance region takes between 10 ms to 200 ms (depending on the number of contacts). To avoid slowing down the calculation in the control loop, the computation of the balance region is done in

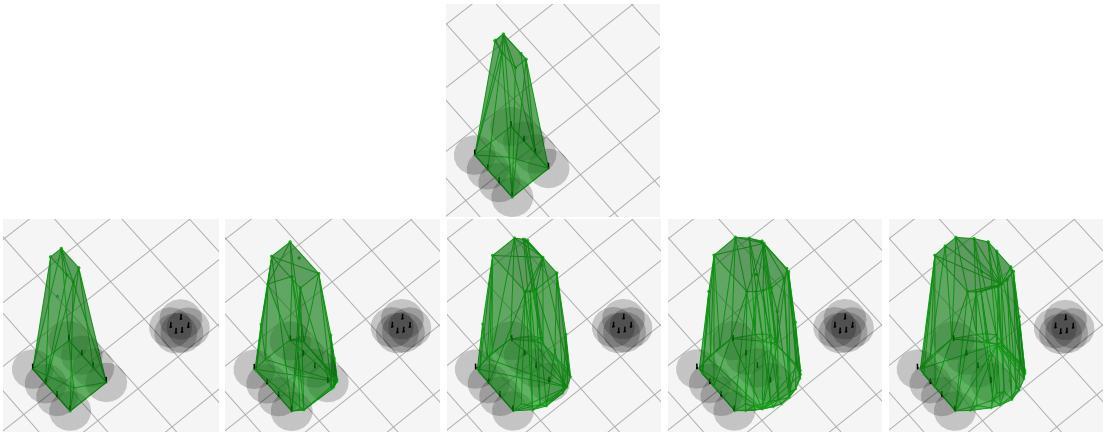


FIGURE 2.11 – *Evolution of the balance region when adding the right-hand contact with the table and increasing the normal force target. For the computation of the balance region, the contact normal force for each contact is bounded. Here, only the bounds on the right-hand contact are changing. From left to right:* $[0.0, 0.0]$ *[N](no right-hand contact),* $[5e-4, 1e-2]$ *[N],* $[7.0, 20.5]$ *[N],* $[15.7, 47.2]$ *[N],* $[22.0, 66.5]$ *[N],* $[25.0, 75.0]$ *[N](transition finished)*

a separate thread. Every time a balance region's computation has finished, the next one starts. As soon as a new region has been constructed, it is used to update the CoM constraint within the WBQP.

2.5 Conclusion

Previous works introduced CoM regions to account for the robot's balance while planning CoM trajectories. In this chapter, we built upon the iterative method to calculate an approximation of the robust CoM balance region proposed in Audren and Kheddar (2018). We propose a set of changes to reduce the computation time. We illustrated the consequences of the linearised friction cones in terms of computation times and the impact on the region's shape. New measures of approximation errors for the iterative algorithm allowed for improved computation speeds without sacrificing the quality of the result. As a result, the region is computed and used online to control forces for moving contacts.

The balance region is used as a constraint in the WBQP. However, it is not enough to have correct tracking of the forces. Thus, a CoM admittance task is proposed. It improves the tracking by generating CoM velocities in an adequate direction. Generating a CoM velocity instead of a CoM target position allows for better interaction with position-based tasks.

A major drawback of this approach is that it does not account for CoM velocity directly in the constraint. Indeed, if the CoM velocity is too important, the CoM may cross the region's boundary without the constraint being able to generate large enough deceleration to stop it (See Sec. 1.3.1). In the next chapter, we consider a different approach that computes an optimal CoM pose and contact wrench by maximizing the margin with the friction cones. No region is directly computed, but a conservative zone is defined as the Chebyshev sphere.

The library implementing the robust balance region in C++ is available online and independent of the *mc_rtc* framework. By changing the formulation of the problem, it is

possible to compute a 3d region from a higher dimension set for other cases as demonstrated in Dehio et al. (2022); Wang et al. (2023) where it is applied to impacts in robotics.

CHAPTER 3

Optimal CoM and Wrench Distribution for Scenarios with Multiple Non Coplanar Sliding Contacts

In the previous chapter, we considered the balance of humanoids with moving contacts. In this chapter, we study a different contact modality: sliding.

There are several scenarios where generating the sliding of a contact is necessary for the robot. The first example is wiping. The robot has to slide its hands along a surface while applying a desired normal force. It can be to clean a whiteboard or to sand a part. Then, during multi-contact locomotion scenarios where the robot cannot remove its weight from a contact, sliding it instead of breaking the contact allows it to move. It is important to note that in both cases, sliding the contact is controlled by the robot. The robot is not reacting to an unexpected contact slide. It generates a sliding motion.

We propose a novel approach to compute optimal CoM position and wrench distribution with respect to target normal force and sliding speed for non-coplanar contacts. This optimal distribution relies on the Chebyshev center of the set of admissible wrenches. A Quadratic Programming (QP) problem is formulated on the centroidal dynamics of the robot in Sec. 3.1. Then, it is integrated into a whole body controller as the first stage of a two-stage optimization in Sec. 3.2. The second stage is the WBQP presented in Sec. 1.3.1. In the second part of Sec. 3.2, wiping with both hands of the robot at the same time and shuffling the foot is demonstrated on the HRP-4 platform. Finally, the benefits of this method are discussed and compared to the use of the balance region in Sec. 3.3.

The results presented in this chapter were published in Samadi et al. (2021) (second author). A video of the resulting experiments is available online [1].

1. https://www.youtube.com/watch?v=cFYd9oQueRE

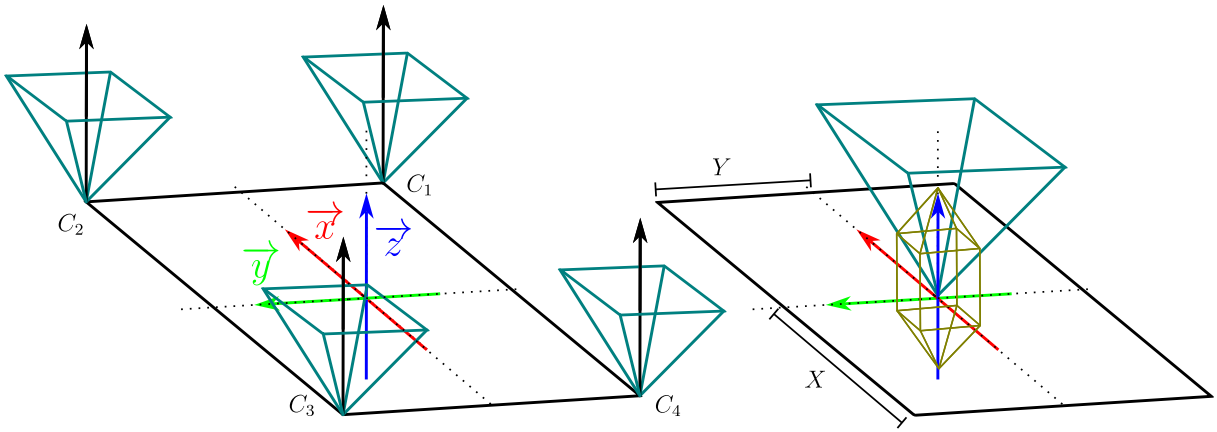


FIGURE 3.1 – *Representation of the friction inequalities for a rectangular planar surface according to Caron et al. (2015b). Left the surface contact is modeled with punctual contact at its corner. Right: the contact friction boundaries are expressed as the admissible wrenches expressed at the origin of the surface.*

3.1 Formulation of the Chebichev QP

In this section we formulate the balance problem with a different representation of the contacts: we use surface contact instead of contact point.

We will talk in terms of wrench distribution and compute a target CoM position $\mathbf{c}$ and the n contact wrenches W_i in the world frame expressed at the origin. We formulate an optimization problem on the state:

$$Y = [\mathbf{c}, W_1, ..., W_n] \tag{3.1}$$

of dimension $N = 3+6n$ The section is divided as follows: first section 3.1.1 expresses the formulation of the constraints on fixed and sliding contacts. Then, the formulation of the centroidal QP problem as formulated in Collette et al. (2007); Samadi et al. (2020) is recalled. Finally, QP is extended to find the Chebichev center of the contact wrench cone to increase the robustness of the robot's balance.

3.1.1 Contact formulation

First, let's recall the results from Caron et al. (2015b) on the computation of a contact wrench cone for a rectangular planar surface and then formulate the conditions for a sliding contact. In the previous chapter, surface contact was modeled with pontual contact at the vertices of the surface (Fig. 3.1 Left). This model is only equivalent to a pressure field on the surface when Dirac fields are authorized(i.e. when considering the solution as a distribution). Otherwise, a set of forces at the vertices of the surface can be computed from any pressure field but the converse is not guaranteed to exist. Then, the force distribution at each vertice of the surface can be expressed as a wrench at the center of the surface.

W_i represents the ith contact wrench expressed at the world's origin and in the global frame. Let's denote $\mathbf{w}_i = (\mathbf{f}_i, \tau_i)$ the contact wrench expressed in the global frame at the

contact point. Then, we have:

$$W_i = \begin{pmatrix} \mathbf{f}_i \\ \mathbf{p}_i \times \mathbf{f}_i + \tau_i \end{pmatrix} = \begin{bmatrix} \mathbb{I}_{3\times3} & \mathbb{O}_{3\times3} \\ [\mathbf{p}_{i\times}] & \mathbb{I}_{3\times3} \end{bmatrix} \mathbf{w}_i = \mathbf{V}_i^{-1}\mathbf{w}_i \tag{3.2}$$

with $\mathbf{p}_i$ the position of the ith contact and $[\mathbf{p}_{i\times}]$ the anti-symetric matrix corresponding to the cross product. Then, $w_{i,l}$ is the contact wrench expressed at contacts' origin in the local frame and $[\mathbf{R}]_i \in \mathbb{R}^{6\times6}$ is the rotation matrix mapping the local frame to the global frame such that:

$$\mathbf{w}_i = [\mathbf{R}]_i \mathbf{w}_{i,l} \tag{3.3}$$

Then

$$W_i = V_i^{-1}[\mathbf{R}]_i \mathbf{w}_{i,l} \quad \text{and} \quad \mathbf{w}_{i,l} = [\mathbf{R}]_i^T \mathbf{V}_i W_i \tag{3.4}$$

with

$$\mathbf{V}_i = \begin{bmatrix} \mathbb{I}_{3\times3} & \mathbb{O}_{3\times3} \\ [-\mathbf{p}_{i\times}] & \mathbb{I}_{3\times3} \end{bmatrix} \tag{3.5}$$

Fixed contacts under no sliding condition Then, the contact wrench can be formulated as $\mathbf{w}_l = (f_x, f_y, f_z, \tau_x, \tau_y, \tau_z)^T$ at the origin of the surface and in the local coordinate of the surface under the conditions:

$$|f_x| \leq \mu f_z \tag{3.6}$$
$$|f_y| \leq \mu f_z \tag{3.7}$$
$$f_{min} \leq f_z \leq f_{max} \tag{3.8}$$
$$|\tau_x| \leq Y f_z \tag{3.9}$$
$$|\tau_y| \leq X f_z \tag{3.10}$$
$$\tau_{min} \leq \tau_z \leq \tau_{max} \tag{3.11}$$
$$\tag{3.12}$$

with X and Y the half-length and half-width of the planar surface (Fig. 3.1), μ the friction coefficient corresponding to the contact and τ_{min} and τ_{max} defined as follow:

$$\tau_{min} = -\mu(X+Y)f_z + |Y f_x - \mu\tau_x| + |X f_y - \mu\tau_y| \tag{3.13}$$
$$\tau_{max} = +\mu(X+Y)f_z - |Y f_x + \mu\tau_x| - |X f_y + \mu\tau_y| \tag{3.14}$$

These constraints are linear with respect to the contact wrench expressed in the local frame and thus can be formulated in a matrix form.

Equations 3.7 to 3.11 are given by

$$G_l^1 \mathbf{w}_l = \begin{bmatrix} 1 & 0 & -\mu & 0 & 0 & 0 \\ -1 & 0 & -\mu & 0 & 0 & 0 \\ 0 & 1 & -\mu & 0 & 0 & 0 \\ 0 & -1 & -\mu & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -Y & 1 & 0 & 0 \\ 0 & 0 & -Y & -1 & 0 & 0 \\ 0 & 0 & -X & 0 & 1 & 0 \\ 0 & 0 & -X & 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} f_x \\ f_y \\ f_z \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ f_{max} \\ -f_{min} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \mathbf{h}^1 \tag{3.15}$$

Then, because of the absolute value in the formulation of τ_{min} and τ_{max}, equation 3.12 result in 8 inequalities that can be formulated in matrix form as:

$$G_l^2\mathbf{w}_l = \begin{bmatrix} Y & X & -\mu(X+Y) & \mu & \mu & 1 \\ Y & -X & -\mu(X+Y) & \mu & -\mu & 1 \\ -Y & X & -\mu(X+Y) & -\mu & \mu & 1 \\ -Y & -X & -\mu(X+Y) & -\mu & -\mu & 1 \\ Y & X & -\mu(X+Y) & -\mu & mu & -1 \\ Y & -X & -\mu(X+Y) & -\mu & \mu & -1 \\ -Y & X & -\mu(X+Y) & \mu & -\mu & -1 \\ -Y & -X & -\mu(X+Y) & \mu & \mu & -1 \end{bmatrix} \begin{bmatrix} f_x \\ f_y \\ f_z \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \mathbf{h}^2 \tag{3.16}$$

Then, in the local frame, the contact wrench for a fixed contact is constrained by the following inequality:

$$G_l\mathbf{w}_l = \begin{bmatrix} G_l^1 \\ G_l^2 \end{bmatrix}^l \mathbf{w} \leq \begin{bmatrix} h^1 \\ h^2 \end{bmatrix} = \mathbf{h} \tag{3.17}$$

Finally, the inequalities can be formulated in the global frame with the wrench expressed at the origin using equation 3.4 as:

$$G_iW_i = G_{l,i}[\mathbf{R}]_i^T\mathbf{V}_iW_i \leq \mathbf{h}_i \tag{3.18}$$

That definition of contact friction is limited to rectangular surfaces. It is harder to generalize.

Sliding contact The sliding contact corresponds to an equality constraint as the wrench for this contact is on the friction cone. It is a mix of equalities and inequalities depending on the velocity of the sliding contact. In the local frame for i^{th} contact the desired contact sliding velocity is $\mathbf{v_i} = \mathbf{v_{x,i}x_i} + \mathbf{v_{y,i}y_i}$. Then let's define

$$\alpha_{x,i} = \frac{v_{x,i}}{\|\mathbf{v_i}\|} \tag{3.19}$$

$$\alpha_{y,i} = \frac{v_{y,i}}{\|\mathbf{v_i}\|} \tag{3.20}$$

$$\mu_{x,i} = \alpha_{x,i}\mu_i \tag{3.21}$$

$$\mu_{y,i} = \alpha_{y,i}\mu_i \tag{3.22}$$

$$\tag{3.23}$$

μ_i is the dynamic friction coefficient different from the static friction coefficient used before. Then the sliding constraints are:

$$f_{x,i} = \mu_{x,i}f_{z,i} \tag{3.24}$$

$$f_{y,i} = \mu_{y,i}f_{z,i} \tag{3.25}$$

Under matrix form, we have:

$$A_{l,i}\mathbf{w}_{l,i} = \begin{bmatrix} 1 & 0 & -\mu_{x,i} & 0 & 0 & 0 \\ 0 & 1 & -\mu_{y,i} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} f_x \\ f_y \\ f_z \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ n_i \end{bmatrix} = \mathbf{b}_i \tag{3.26}$$

where n_i is the target normal force in the local frame. These equalities are completed by the non-sliding inequalities applied on the torques as given by:

$$G_{l,i}\mathbf{w}_{l,i} = \begin{bmatrix} 0 & 0 & -Y & 1 & 0 & 0 \\ 0 & 0 & -Y & -1 & 0 & 0 \\ 0 & 0 & -X & 0 & 1 & 0 \\ 0 & 0 & -X & 0 & -1 & 0 \\ Y & X & -\mu(X+Y) & \mu & \mu & 1 \\ Y & -X & -\mu(X+Y) & \mu & -\mu & 1 \\ -Y & X & -\mu(X+Y) & -\mu & \mu & 1 \\ -Y & -X & -\mu(X+Y) & -\mu & -\mu & 1 \\ Y & X & -\mu(X+Y) & -\mu & mu & -1 \\ Y & -X & -\mu(X+Y) & -\mu & \mu & -1 \\ -Y & X & -\mu(X+Y) & \mu & -\mu & -1 \\ -Y & -X & -\mu(X+Y) & \mu & \mu & -1 \end{bmatrix} \begin{bmatrix} f_x \\ f_y \\ f_z \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} \leq \mathbb{O}_{10\times 1} \tag{3.27}$$

Both equalities and inequalities can be formulated at the origin in the global frame with:

$$A_i W_i = A_{l,i}[\mathbf{R}]_i^T \mathbf{V}_i W_i = \mathbf{b}_i G_i W_i = G_{l,i}[\mathbf{R}]_i^T \mathbf{V}_i W_i \leq \mathbb{O}_{10\times 1} = \mathbf{h}_i \tag{3.28}$$

3.1.2 Centroidal QP

As proposed in Collette et al. (2007) and Samadi et al. (2020), we formulate the problem of finding an optimal CoM position and wrench distribution as a quadratic programming minimization problem (QP). First, this QP is constrained by the contact equalities and inequalities formulated in the previous section. The problem is also constrained by the Newton-Euler equations in the static case given by:

$$\sum_{i=1}^{i} W_i + W_g = 0 \tag{3.29}$$

with $W_g = (-m\mathbf{g}, -\mathbf{c}\times m\mathbf{g})^T$ the gravity wrench. In matrix form, we have:

$$A_{NE}Y = \begin{bmatrix} \begin{bmatrix} 0 & -mg & 0 \\ mg & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} & \mathbb{O}_{3\times 3} & \mathbb{I}_3 & \mathbb{O}_3 & \dots & \mathbb{I}_3 & \mathbb{O}_3 \\ \mathbb{O}_3 & \mathbb{I}_3 & \mathbb{O}_3 & \dots & \mathbb{O}_3 & \mathbb{I}_3 \end{bmatrix} \begin{bmatrix} \mathbf{c} \\ W_1 \\ \vdots \\ W_n \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -mg \\ 0 \\ 0 \\ 0 \end{bmatrix} = b_{NE} \tag{3.30}$$

Then the total system can be formulated as:

$$AY = \mathbf{b} \tag{3.31}$$

$$GY \leq \mathbf{h} \tag{3.32}$$

with

$$A=\begin{bmatrix} & & A_{NE} & & \\ & \dots & A_{s_1} & \dots & \mathbb{O}_{10\times 6} & \dots \\ \mathbb{O}_{10m\times 3} & & & \ddots & & \\ & \dots & \mathbb{O}_{10\times 6} & \dots & A_{s_m} & \dots \end{bmatrix}, \quad \mathbf{b}=\begin{bmatrix} b_{NE} \\ b_{s_1} \\ \vdots \\ b_{s_m} \end{bmatrix} \tag{3.33}$$

$$G=\begin{bmatrix} & G_1 & & \mathbf{O} \\ \mathbb{O}_{M\times 3} & & \ddots & \\ & \mathbb{O} & & G_n \end{bmatrix}, \quad \mathbf{h}=\begin{bmatrix} h_1 \\ \vdots \\ h_n \end{bmatrix} \tag{3.34}$$

with $0\leq s_1,\dots,s_m\leq n$ the indexes of the m sliding contacts and $M=10m+18(n-m)$ the total number of inqualities.

The centroidal QP minmize $\|Y-Y_{des}\|_2$ and is formulated as:

$$\begin{aligned} \min_{Y} \quad & \tfrac{1}{2}Y^T\mathbf{P}Y+\mathbf{q}^TY \\ s.t. \quad & AY=\mathbf{b} \\ & GY\leq\mathbf{h} \end{aligned} \tag{3.35}$$

with $P=2\mathbb{I}_{N\times N}$ and $\mathbf{q}=-2Y_{des}$. Y_{des} is chosen according to the current task the robot has to execute. The target CoM is given as the previous computer CoM. For the sliding contacts, the target normal force is set (as well as in the sliding constraint) and for the other wrenches, their target is set to 0 so that they are minimized.

This Centroidal gives a valid solution. However, there are no incentives for the CoM to move away from the border of the balance region defined in Chap. 2. Thus, we introduce the concept of the Chebyshev center and integrate it into the centroidal QP to improve the quality of the result.

3.1.3 Chebichev center

From the previous section, we have a set of equations constraining the contact forces. Because they are linear inequities, they define a convex set (each inequality gives a halfspace which is convex and the intersection of convex is convex).

Chebichev center of a set To avoid sliding, we want the point furthest away from any boundary of this high-dimensional set. This is encompassed in the concept of the Chebyshev center. As in Boyd and Vandenberghe (2004), let's first define the depth of a point $\mathbf{x}$ of the bounded set $C\subseteq\mathbb{R}^n$ as the distance to the closest point in the exterior of C:

$$\text{depth}(\mathbf{x},C)=\text{dist}(\mathbf{x},\mathbb{R}^n\backslash C) \tag{3.36}$$

Then, the furthest point from the boundary can be seen as the deepest point inside C:

$$\mathbf{x}_{\text{cheb}}(C)=\underset{\mathbf{x}}{\text{argmax}}\,\text{depth}(\mathbf{x},C)=\underset{\mathbf{x}}{\text{argmax}}\,\text{dist}(\mathbf{x},\mathbb{R}^n\backslash C) \tag{3.37}$$

Also, the depth $\mathbf{x}$ corresponds to the radius r of the biggest ball centered on $\mathbf{x}$ and is included in C. $r_{cheb} = \text{depth}(\mathbf{x}_{cheb}$ is called the Chebyshev radius. In this case, the set C is a polyhedron defined by a set of linear inequalities $\mathbf{a}_i^T\mathbf{x} \leq b_i$. Then, from Boyd and Vandenberghe (2004), the following value is computed:

$$g_i(\mathbf{x},r) = \sup_{\|\mathbf{u}\|\leq 1} \mathbf{a}_i^T(\mathbf{x}+r\mathbf{u}) - b_i = \mathbf{a}_i^T\mathbf{x} + R\|\mathbf{a}_i\| - b_i \tag{3.38}$$

It represents the algebraic distance between the ball centered on $\mathbf{x}$ and radius r and the boundary of the ith halfspace. If it is negative or null then the ball is included in the halfspace. Then, the Chebichev center can be found by solving the LP:

$$\underset{\mathbf{x},r}{\text{maximize}} \quad r \tag{3.39}$$

$$s.t. \quad \mathbf{a}_i^T\mathbf{x} + r\|\mathbf{a}_i\| \leq b_i, \quad i = 1,...,n \tag{3.40}$$

$$r \geq 0 \tag{3.41}$$

Wrench distribution and Chebichev center From there, we derive the QP that lets us optimize wrench distribution and CoM placement. To have the wrenches as far as possible from the boundaries of their local contact wrench cone, we introduce the Chebyshev in the inequalities constraints of Centroidal QP 3.35

The inequalities become:

$$GY + r\xi \leq \mathbf{h} \tag{3.42}$$

where r is the Chebyshev radius and ξ is the vector of the norms of the lines of G.

The state is then extended as:

$$X = \begin{bmatrix} Y \\ r \end{bmatrix} \tag{3.43}$$

which gives the equalities and inequalities matrix:

$$A^* = \begin{bmatrix} A & \mathbb{O}_{6+3m\times 1} \end{bmatrix} \tag{3.44}$$

$$G^* = \begin{bmatrix} G & \xi \\ \mathbb{O}_{1\times 3+6n} & -1 \end{bmatrix}, \text{ and, } \quad \mathbf{h}^* = \begin{bmatrix} \mathbf{h} \\ 0 \end{bmatrix} \tag{3.45}$$

The cost to minimize then becomes:

$$\begin{aligned} \|X - X_{des}\|^2 - r &= (X - X_{des})^T(X - X_{des}) - r && (3.46)\\ &= X^TX - 2X_{des}^TX + \underbrace{X_{des}^TX_{des}}_{\text{=constant}} - r && (3.47)\\ &\equiv \frac{1}{2}X^T\mathbf{P}X + \mathbf{q}^TX && (3.48) \end{aligned}$$

where $P = 2\mathcal{Q}$ with $\mathcal{Q} \in \mathbb{R}^{(3+6n+1)\times(3+6n+1)}$ a weight matrix to prioritize solution, and $\mathbf{q}[Y_{des}^T, -1]^T$. As no target is given for r, the corresponding weight in $\mathcal{Q}$ is small.

3.2 Integration in the whole body QP

In this section, we present the use of the Chebichev QP inside the WBQP and the resulting experiments.

3.2.1 Whole body QP integration

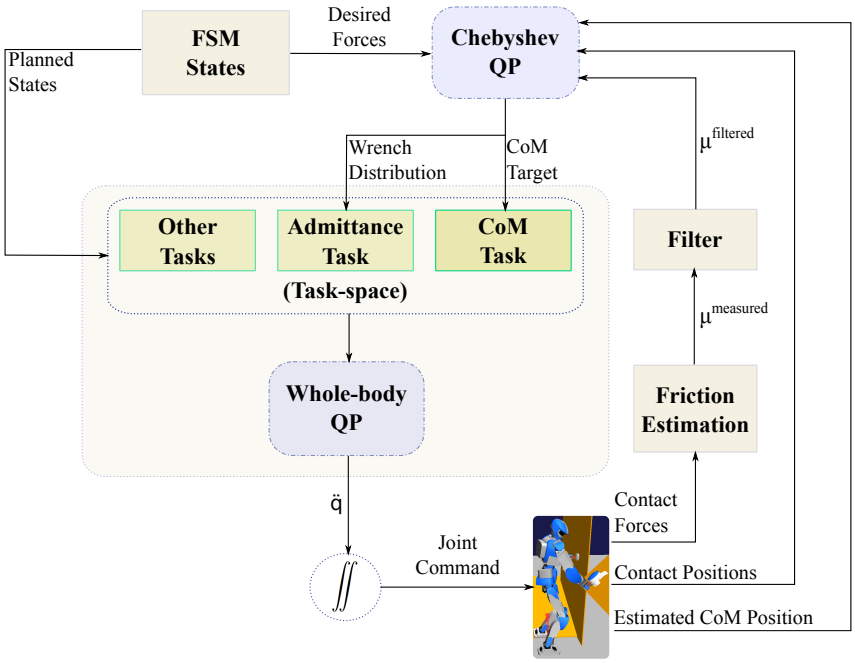


FIGURE 3.2 – *Integration of the Chebyshev QP to the WBQP*

The details of the whole body controller are given in Sec. 1.3.1. Fig. 3.2 illustrates how the Centroidal QP framework is used to balance the robot in scenarios with non-coplanar fixed and sliding contacts.

The Chebyshev QP outputs a target CoM and contact wrench distribution for the robot to track. A CoM task and an admittance task for each contact are used in the WBQP. A finite state machine (FSM) provides a target for other WBQP tasks such as trajectories for the limbs that do not interact with the environment.

The FSM also provides several inputs for the centroidal QP:

- Target wrench for the cost function;
- Target normal force for the sliding contacts (Eq. 3.26);
- Minimal and maximal normal forces for non-sliding contact(Eq. 3.15);
- Sliding contact trajectories (Eq. 3.20 and 3.21).

Finally, the Chebyshev QP requires information on the current state of the robot:

- estimation of the CoM pose from the kinematics and IMU;
- estimation of the contacts pose from the kinematics;
- dynamic friction coefficient for sliding contacts from measured contact forces.

The friction coefficient depends on the contact surfaces. It is complex to estimate a value in advance, hence, it is estimated online. In the contact frame, according to Eq. 3.24 and 3.25, we have

$$\underbrace{\sqrt{f_x^2+f_y^2}}_{\text{Norm of the tangential force}} = \mu_{dyn} \underbrace{f_z}_{\text{normal force}} \tag{3.49}$$

Thus, the dynamic friction coefficient can be expressed as

$$\mu_{dyn} = \frac{\sqrt{f_x^2+f_y^2}}{f_z} \tag{3.50}$$

However, as force measurement is noisy, the result must be filtered. We implemented a simple low-pass filter:

$$\mu_t^{\text{filtered}} = \gamma \mu_{t-1}^{\text{filtered}} + (1-\gamma)\mu_t^{\text{measured}} \tag{3.51}$$

with $0 \leq \gamma \leq 1$.

In the controller, the Chebichev QP can be computed on average in 0.2 ms, thus it can be calculated at each iteration.

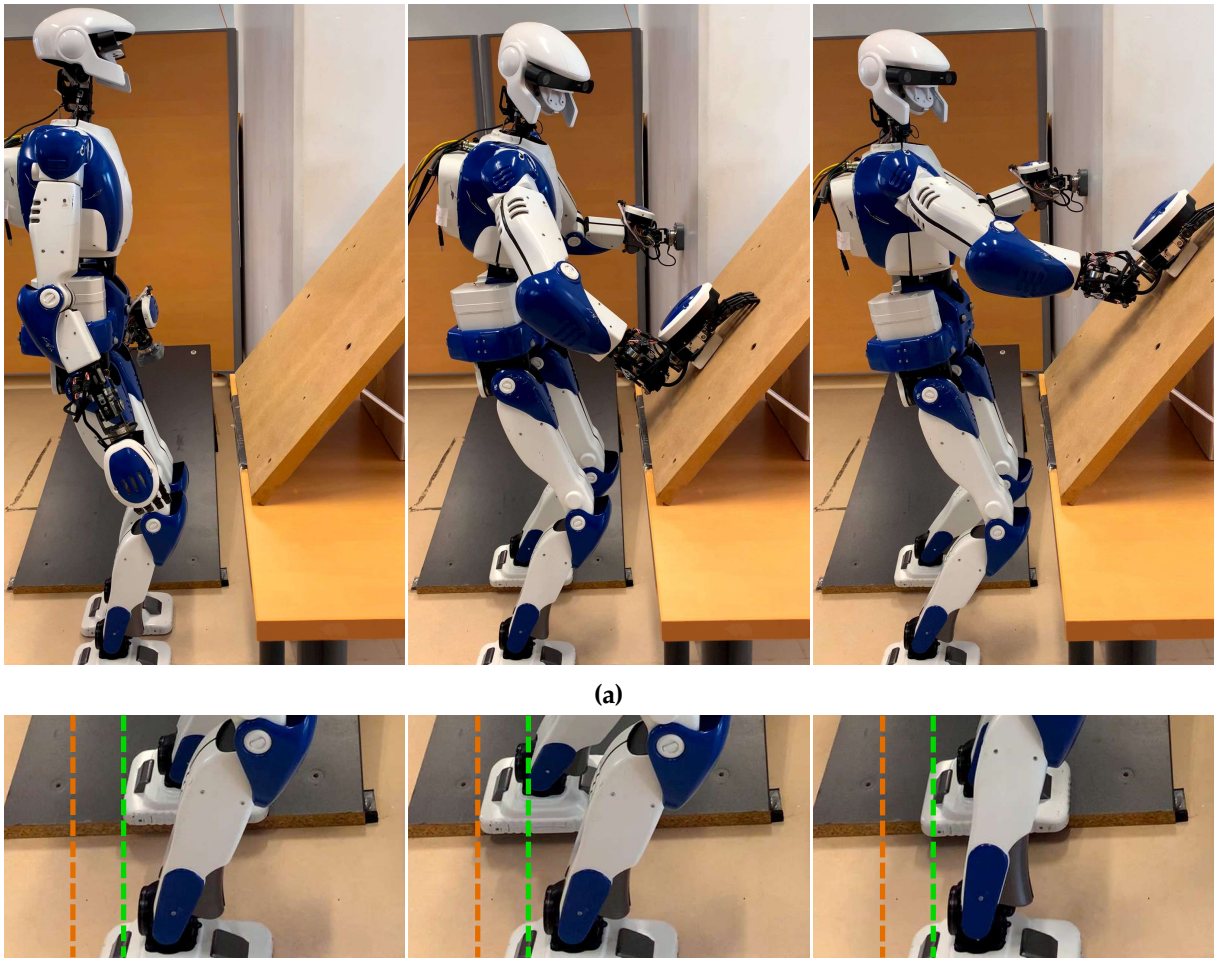
3.2.2 Resulting experiments

The approach is demonstrated in an experimental setup where the robot HRP-4 will execute wiping and shuffling tasks sequentially. During the demonstration, the robot uses 4 non-coplanar contacts with different materials and thus different friction coefficients. The principal position is illustrated in Fig. 3.3. The left hand is against a vertical wall. The right hand is on a tilted board ($\approx 50°$ w.r.t. horizontal). Its left foot is not on the ground as in previous work Samadi et al. (2020) but on a $20°$ slope. Meanwhile, the right foot is considered as an anchor and stays on the ground.

The first motion executed is co-wiping (Fig. 3.3a). Both hands slide along their respective contact surfaces drawing circles of different radii and applying different normal forces. The target and measured normal force are shown in Fig. 3.4. Before wiping, the target normal force is kept at a low positive value to keep the contact but with little impact on the balance. They are increased progressively to the desired normal wrench to smooth out the CoM trajectory (Fig. 3.6 top). During the wiping motions, the force has a constant target. The tracking on the left hand is correct while important variations are seen for the right hand. It can be explained by the limited kinematics of the robot. Several of the right arm's 7 DoF are close to their limits (e.g. the last joint: wrist roll). It is not the case for the left hand which has a different end-effector.

The second motion is shuffling: the left foot slides back a few centimeters and then comes back to its initial position (Fig. 3.3b). This motion has seldom been demonstrated in the literature as most of the time the foot is fully lifted when moved. During the motion, the robot shifts its weight from its left foot to its right foot and its hands as shown in Fig. 3.5.

Fig. 3.6 illustrates the CoM displacement during the wiping and shuffling motions. Because of the weakness of the robot's wrists, it is not possible to apply high hand forces. This results in a small CoM displacement on the x-axis. Yet, the CoM moves forward almost



(a)

(b)

FIGURE 3.3 – *Wiping and shuffling demonstrations*

10 cm when pushing on the hands during both motions. When shuffling, the left foot normal force decreases by 200 N resulting in a 20 cm sideway shift of the CoM.

The motions are demonstrated one after the other, showing the capability of this approach to switch between fixed and moving contacts.

The experiments computations were made by a laptop computer having Intel(R) Xeon(R) E-2276M CPU at 2.80 GHz×12. With this setup, the average computation times for the Chebyshev QP and the whole framework were noted as 0.2 and 1 ms respectively. These values are largely within the implemented real-time control loop with `mc_rtc` framework [2] that is 5 ms.

3.3 Discussion

In this section, the results of the Chebyshev approach are discussed and compared to the static balance region.

2. https://github.com/jrl-umi3218/mc_rtc

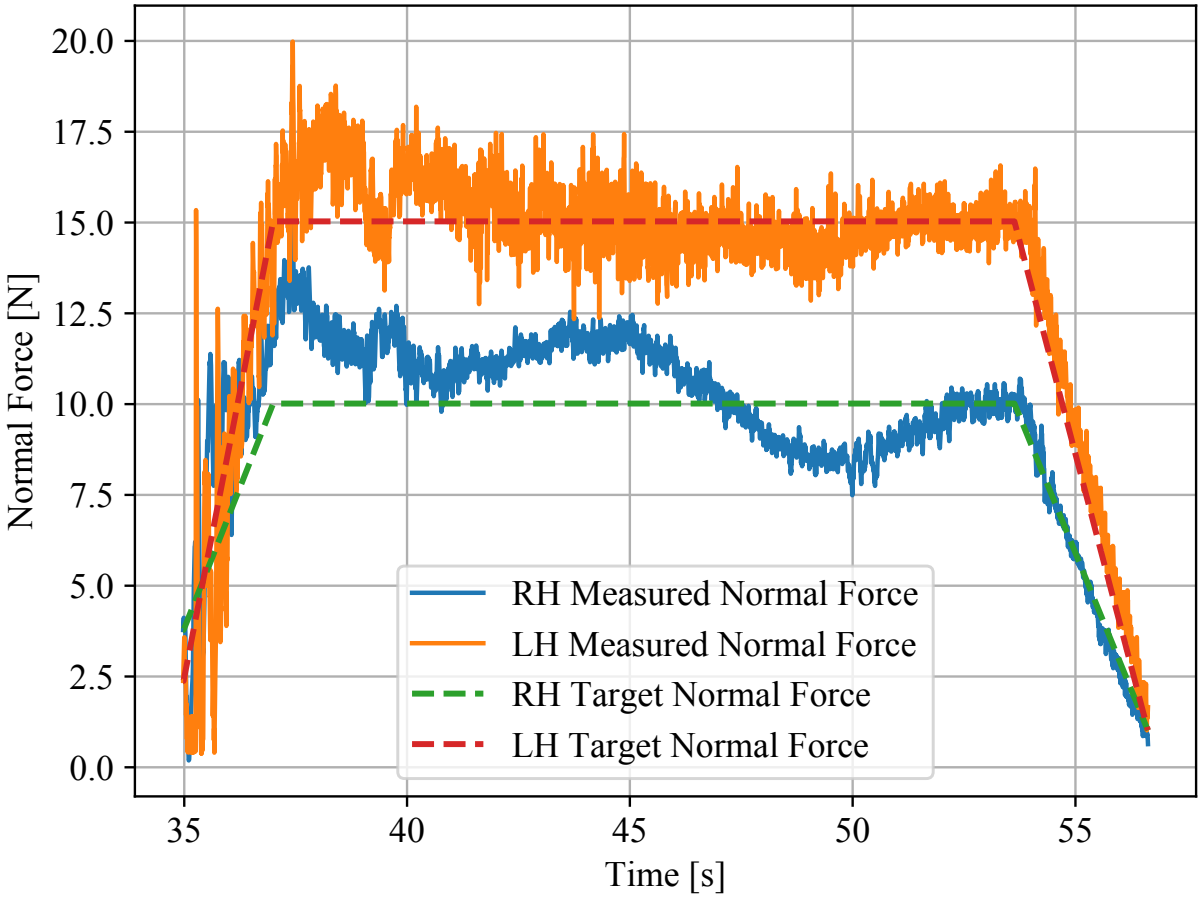


FIGURE 3.4 – *Hands normal forces during the co-wiping demonstration (As the hand's contact surfaces are non-coplanar, the normal are considered in the local frames)*

3.3.1 Interpretation of the Chebyshev radius

Given the formulation of the problem, the Chebyshev radius r has no direct interpretation. It has the meaning of a "distance" in a mixed high-dimensional space of the wrenches (mix of force and torque dimensions). It could be used as a representation of the contact stability margin. A larger radius means the wrench closest to its local boundary is further away than for a smaller radius. Thus, a bigger radius means it is less likely that a contact slides in an undesired fashion.

In Samadi (2021), Samadi proposed a demonstration to interpret the Chebyshev radius. First, a sphere in the Contact Wrench Cone (CWC) expressed at the CoM is given by

$$r_{\text{CWC}} = n \times r \tag{3.52}$$

with n the number of contact and r the Chebychev radius. The demonstration uses the fact that the CWC can be expressed as the Minkowski sum of all the contact friction cones expressed at CoM.

Then, given that the Gravito Inertial Wrench Cone is opposed to the CWC, a radius around the CoM was given as

$$r_{\text{CoM}} = \frac{r_{\text{CWC}}}{mg} \tag{3.53}$$

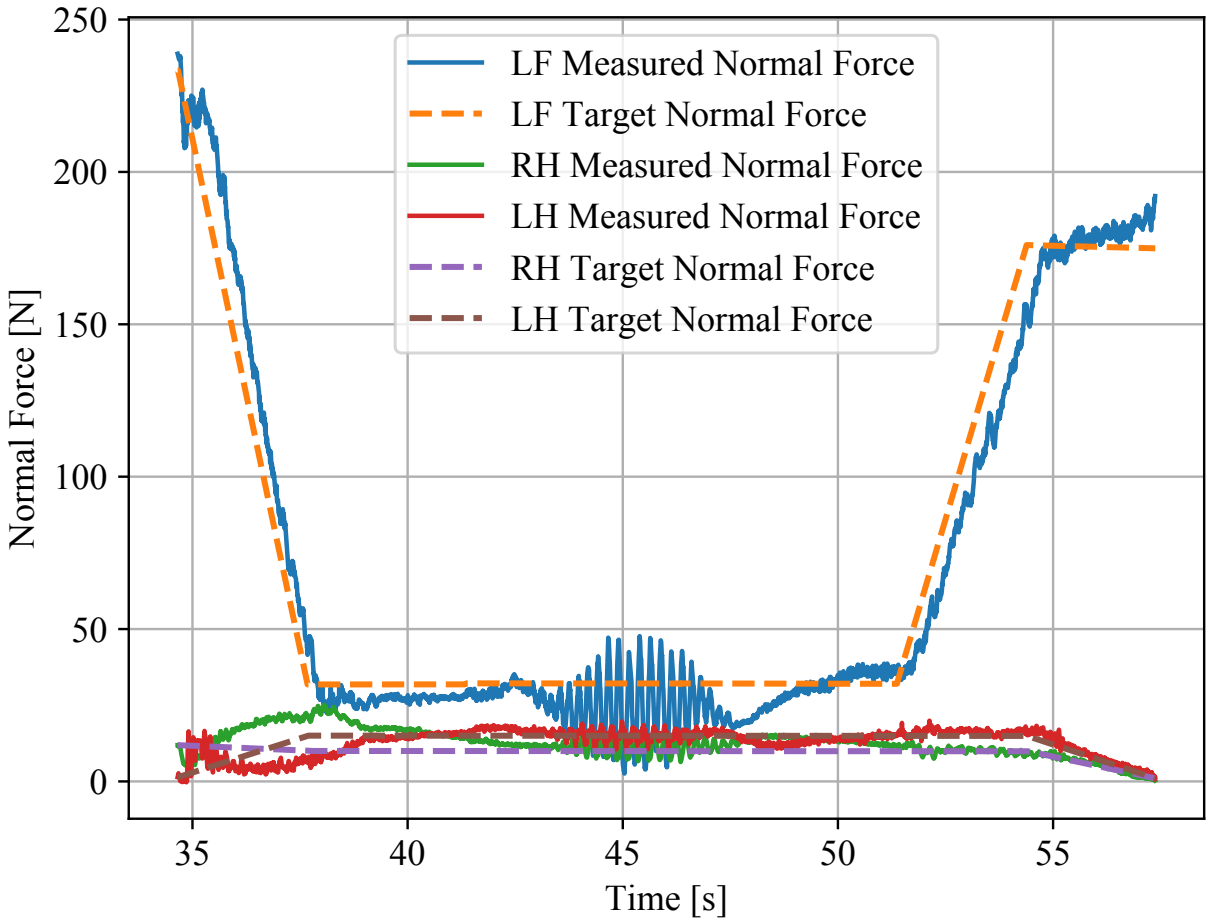


FIGURE 3.5 – *Hands and shuffled foot normal forces during the shuffling demonstration (As the hands' contact surfaces are non-coplanar, the normal are considered in the local frames)*

with m the mass of the robot and g the gravity acceleration. r_{CoM} is given as the lower bound of the distance between the CoM and the boundary of the static balance region. Using this definition, it is then possible to compare the static balance region with the Chebiyshev sphere at the CoM.

3.3.2 Comparison whith CoM balance region

In this section, we compare the approach proposed in this chapter to the balance region studied in Chap. 2. For this purpose, four regions are considered:

Region A the original static balance region presented in Bretl and Lall (2008);

Region B the static balance region using the contact wrench and dynamic constraints defined by equations 3.31 and 3.32;

Region C the static balance region built by constraining each contact wrench within the local Chebichev sphere;

Region D the circle corresponding to the Chebyshev CoM sphere centered on the target CoM with radius given in 3.53.

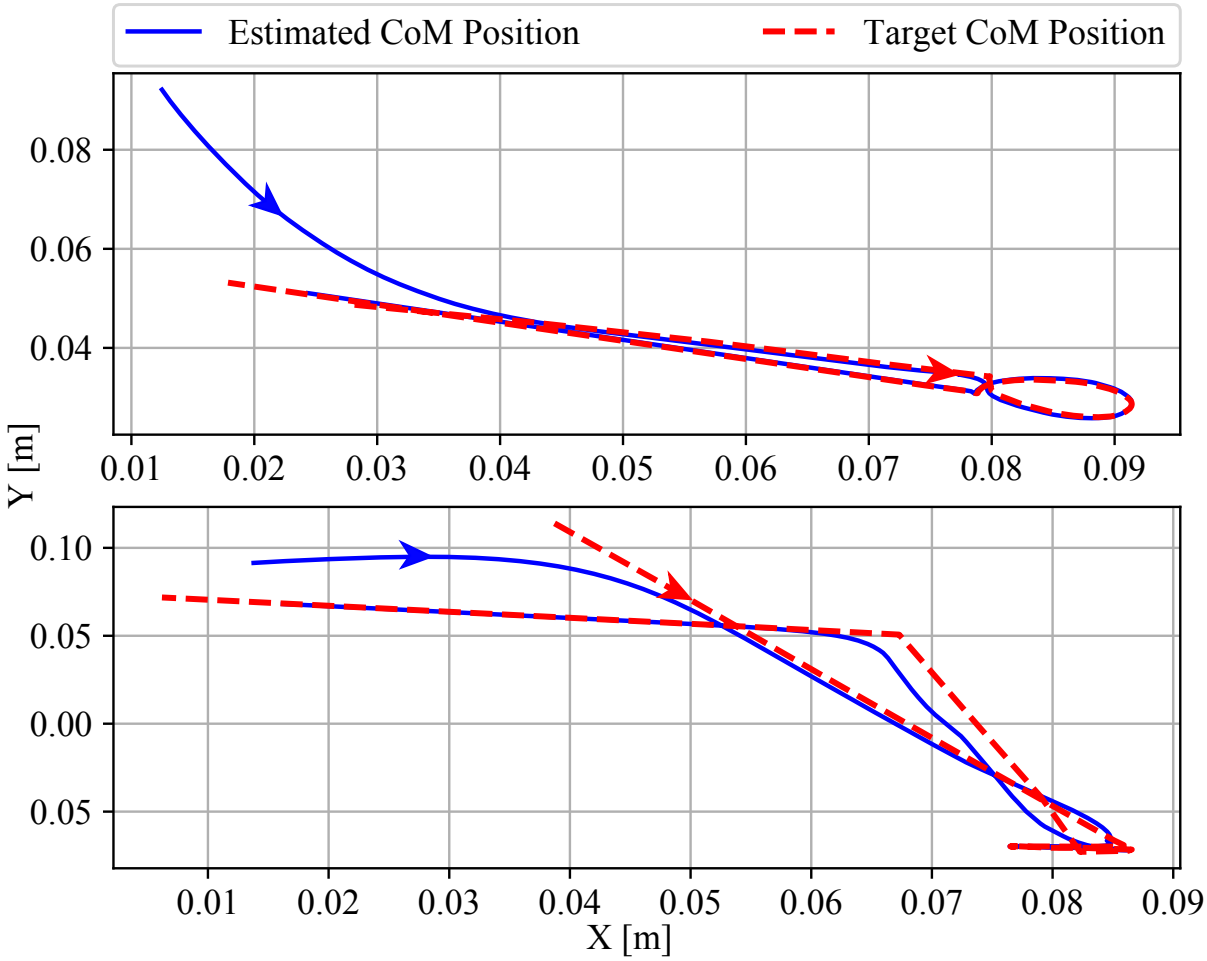


FIGURE 3.6 – *CoM trajectory projected in the horizontal plane during the co-wiping (top) and shuffling (bottom) experiments*

Regions A, B, and C are built using the iterative projection algorithm discussed in Chap. 2. They differ in the representation of the contacts as in Fig. 3.1: region A uses several punctual contacts per surface, and region B uses only one wrench per contact. Region C also only uses one wrench per contact yet the inequalities constraining each wrench are given by:

$$f_{\text{Cheb},i,x} - r \leq f_{i,x} \leq f_{\text{Cheb},i,x} + r \tag{3.54}$$
$$f_{\text{Cheb},i,y} - r \leq f_{i,y} \leq f_{\text{Cheb},i,y} + r \tag{3.55}$$
$$f_{\text{Cheb},i,z} - r \leq f_{i,z} \leq f_{\text{Cheb},i,z} + r \tag{3.56}$$
$$\tau_{\text{Cheb},i,x} - r \leq \tau_{i,x} \leq \tau_{\text{Cheb},i,x} + r \tag{3.57}$$
$$\tau_{\text{Cheb},i,y} - r \leq \tau_{i,y} \leq \tau_{\text{Cheb},i,y} + r \tag{3.58}$$
$$\tau_{\text{Cheb},i,z} - r \leq \tau_{i,z} \leq \tau_{\text{Cheb},i,z} + r \tag{3.59}$$
$$\tag{3.60}$$

where $f_{\text{Cheb},i}$ and $\tau_{\text{Cheb},i}$ are the target force and torque of the ith contact obtained from the Chebyshev QP. As the Chebyshev QP is a quasi-static approach that considers no CoM acceleration, the static balance region is used instead of the robust balance region.

Fig. 3.7 represents these regions. First, regions A and B are similar in shape and position. There are two origins for the difference:

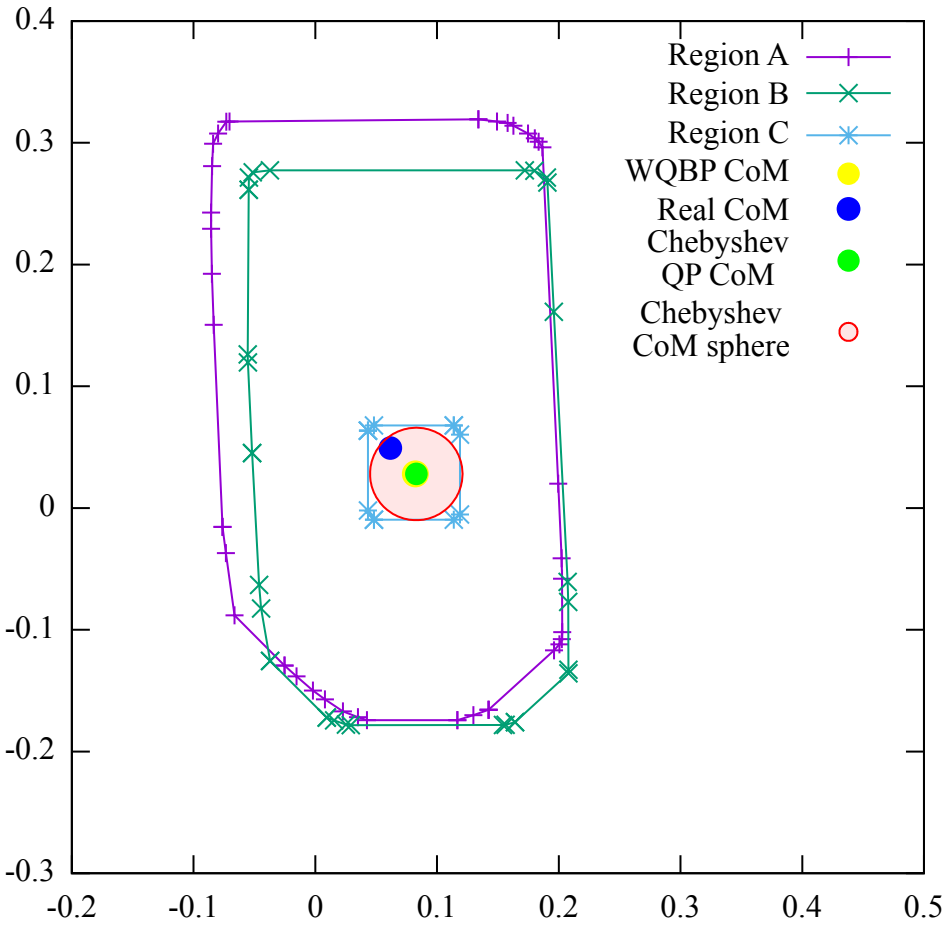


FIGURE 3.7 – *Plot of different CoM balance regions during the co-wiping motion of HRP-4 (Fig. 3.3a). Region A is the static balance region from Chap. 2. Region B is the static balance region using the fixed contact equations presented in Sec. 3.1.1. Region C (blue) is the static balance with the wrenches constrained to the Chebyshev sphere. The red circle is the Chebyshev CoM sphere with a radius defined in Eq. 3.53.*

- the right-hand contact is not a rectangular surface, the region A contact model accounts for non-rectangular surface contacts, which is not the case for region B;
- region B's definition accounts for the sliding direction of each surface, which is not directly possible in the contact models of region A. At best, it is possible to give a target normal force f_{tar} for region A by constraining the normal force f with $f_{\text{tar}} \leq f \leq f_{\text{tar}}$.

Region C and the Chebyshev CoM sphere are almost overlapping confirming the validity of Eq. 3.53 introduced in Samadi (2021). The squared shape of region C is due to simple linear bounds for the contact wrenches. Furthermore, reproducing this illustration at different moments of the motions illustrates that the distance between the target CoM given by Chebichev Qp is at least r_{CoM} as the Chebychev CoM sphere never touches the boundary of the static balance region.

As the computation of the Chebyshev QP is fast enough to happen at each iteration of the WBQP, it is possible to use the Chebyshev CoM sphere as a CoM balance region. However, the Chebyshev CoM sphere is conservative compared to the CoM balance region

and highly restricts the CoM motions.

3.3.3 Target CoM versus CoM region

The WBQP has two levels of hierarchy: the constraint level and the task level. The constraints are conditions always verified in the result. They defined the space of possible solutions. Meanwhile, the tasks are represented by cost functions for the optimization problem: a larger task error means a higher cost. Then, the QP outputs a compromise between all the tasks that minimizes the cost inside the region of possible solutions. Hence, when a CoM Task keeps the CoM back while another task aims at moving an end-effector forward, these tasks will interfere with each other. Their respective task errors will increase based on the ponderation. The CoM balance region constrains the CoM to be in a place with a valid contact wrench distribution. As such, it does not interact with other tasks as a CoM task would. A limitation is that it does not account for the CoM velocity, so the CoM needs to be controlled to improve tracking results and avoid constraint violation in some cases.

Compared to the speed requirement of the WBQP, the CoM balance region is still too long to compute for the constraint to be updated at each iteration. In Chap. 2, the update is fast enough relative to the motions that the region's shape changes between two updates are not significant. On the contrary, the CoM target given by Chebyshev QP is fast to compute and computes a new solution at each iteration of the WBQP. However, the size of the Chebyshev CoM sphere generated limits its benefits. In Samadi (2021), a solution with a different radius for each component of the contact wrenches was proposed. It increases the size of the region.

When considering both approaches from a balance point of view, similar behaviors are observed. When adding a contact in front of the robot with a normal maximal force of 10 N will increase the size of the balance region a lot (Fig. 2.11). Similarly, giving a target to the normal force of contact in front of the robot will move Chebyshev CoM forward a lot. However, there are diminishing returns: further increasing the force won't change the region as much and won't impact the Chebyshev CoM as much.

Compared to other literature such as Abi-Farraj et al. (2019) or inverted pendulum methods, the contacts are all considered in the same manner without a separation between supports and tasks oriented.

In Samadi (2021), the Chebyshev approach is modified in two manners. First, several radii are considered, each corresponding to one inequality, thus increasing the size of the resulting region and the stability margin of contacts. Second, the use of two QPs as proposed in Collette et al. (2008) is transformed in a one-step optimization by integrating the constraints of Chebyshev QP to the WBQP.

To further improve the result, faster computation of the balance region would still be beneficial. In Audren and Kheddar (2017), the balance region was computed offline and used online when reaching the corresponding contact state to constrain an MPC. The MPC was running at ≈ 20 ms. Using the results presented in Chap. 2, and depending on the number of contacts, it would be possible to compute the balance region online by anticipating the contact stance for the next MPC calculation. This would allow us to account for errors in the contact positions. Ongoing works in the team explore the analytic computation to have

it even faster based on the works presented in Delos and Teissandier (2014a,b).

Furthermore, extending the balance region by accounting for angular momentum variations is also to be considered. Indeed, if CoM is on the boundary of the balance region, it theoretically cannot come back in the region. However, using angular momentum variation to regain balance is possible.

In both Chap. 2 and 3, the objective was to track a desired contact force. However, in many applications, having a range of force between a minimum and a maximum is better suited. The next part of the thesis presents such an application end-effector change.

3.4 Conclusion

In this chapter, we first recalled a different formulation of the contact friction for rectangular surfaces. Combined with the sliding condition for some contacts, we defined a Chebyshev QP to optimize the CoM placement and the wrench distribution that maximizes the contact stability margins.

The Chebyshev QP was then applied in quasi-static scenarios where the humanoid robot HRP-4 demonstrated the capability to slide several end-effectors at the same time, as well as to shuffle a foot. Finally, we compared this new approach to balance in multi-contact to the use of the balance region presented in Chap. 2.

This chapter concludes the works of this thesis on the topic of multi-contact balance. To further improve the capabilities of humanoid robots, contributions to their design are presented in the following chapters.

CHAPTER 4

End-Effector Changers for Humanoid Robots: Case Study and Limitation Definitions

Previously, we discussed the balance of humanoids in multi-contact scenarios. Its impact on improving humanoid robots' autonomy is considerable. Yet, the previous two chapters only consider simple end-effectors designed for balance. Such end-effectors may be beneficial for the mobility of the humanoid, but they are probably not fit for more complex tasks that require better handling. For instance, some tasks may require a gripper, a power tool, or a sensor. Contraversively, high-end fragile end-effectors would limit the robot's motion specifically in complex unstructured environments.

As an example, one can look at Boston Dynamics Atlas: when it does dynamic motions and parkour, its end-effectors are simple spheres [1]. However, when it has tasks, it is equipped with grippers [2].

Thus, we propose to integrate tool and end-effector changes to humanoid robots. In robotic industry and manufacturing, this idea is not new. As described in Sec. 1.2 of this thesis, Tool-Changers (TC) have been used in robotics for a long time. TCs are usefull apparatus that increase manufacturing efficiency and performance. Many works study their limits and how to integrate them into robotic arms, mobile platforms, or how to automate tool change.

Even though some TCs from the literature are open source, most are difficult to access or not convenient to use with humanoid robots. This chapter presents the integration of two commercially available tool-changers. We will explain all our efforts to integrate them into our framework and platform. We will also focus on each end-effector (i.e. hands, feet and head) as they all have particularities. Then, from our experiments, we will list the limitations of the existing solutions and outline a set of constraints and requirements for end-effector-changers applied to humanoid robots.

This chapter is as follows: first, Sec. 4.1 defines the different terms used in this thesis to describe a tool change. Then, in Sec. 4.2, the different TCs available in the commerce, and two of them are chosen to be integrated into our robotic platform. How the humanoid platform

1. https://www.youtube.com/watch?v=tF4DML7FIWk
2. https://www.youtube.com/watch?v=LeeiN9smjjY

is modified and how the TCs are integrated is described in Sec. 4.3. Then autonomous tool change is implemented in our framework in Sec. 4.4. Following the initial tests, Sec. 4.5 formulates a set of limitations and design constraints for TCs to be used in humanoid robots before Sec. 4.6 concludes this chapter.

4.1 Terminology

This section clarifies the definitions and choices of words used in this chapter and the next one. First, we differentiate *tool* and *end-effector*. According to the Cambridge dictionary, the definition of a tool [3] is *a piece of equipment that you use with your hands to make or repair something*. It is a broad definition, and at the same time, the Cambridge dictionary does not define end-effector. From *Dictionnary.com* a tool [4] is "*an implement, especially one held in the hand, as a hammer, saw, or file, for performing or facilitating mechanical operations*" or "*the cutting or machining part of a lathe, planer, drill, or similar machine*". Meanwhile, an end-effector [5] is "*a tool, such as a gripper, vacuum, or drill, affixed to the end of a robotic arm to enable the machine to fulfill a specific purpose*". From the general literature in robotics, end-effector is a term used to designate the section at the end of a robotic limb interacting with the environment. An end-effector can be a tool, a gripper, a support (e.g., a foot), or a sensor.

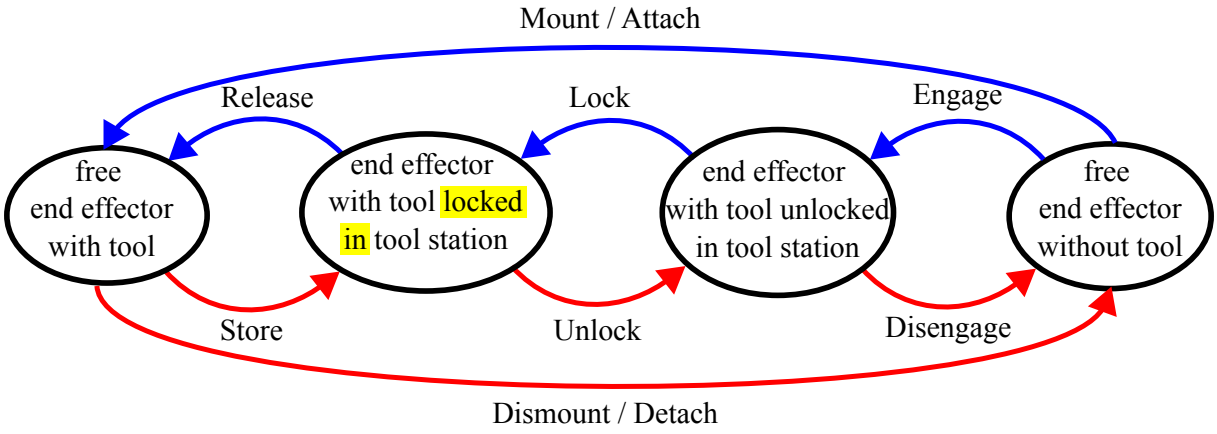
A *tool-changer* (TC) is a mechanism to change a tool. They are largely employed in the manufacturing industry with CNC machines where machining operations often require more than one tool (reamer, drillbit, etc.). Then, automatic tool-changers (ATC) are implemented to reduce the number of operator interventions during the manufacturing process. In the robotic arms industry, the term tool-changer is also used. Some TCs are coined as manual tool-changers because an operator is required to change the tool. In the context of this work, we explore the concept of tool changing applied to humanoid robots. As we consider all the robot's limbs and a wide variety of end-effectors, from grippers to cameras or feet, we will use the term *end-effector changer* (EEC). In this chapter, the term TC will be used as they have been so-called by the manufacturer. However, in the next chapter, the proposed design is for an EEC as the design considers all kinds of end-effectors and limbs for the robot.

Next, we call *tool station* the place where we store the end-effectors. Specifically, the *foot station* is where the lower limbs' end-effectors will be stored. The part of the TC attached to the robot that is not removed when changing the tool is the *robot side*. The part attached to each tool is called the *tool side*.

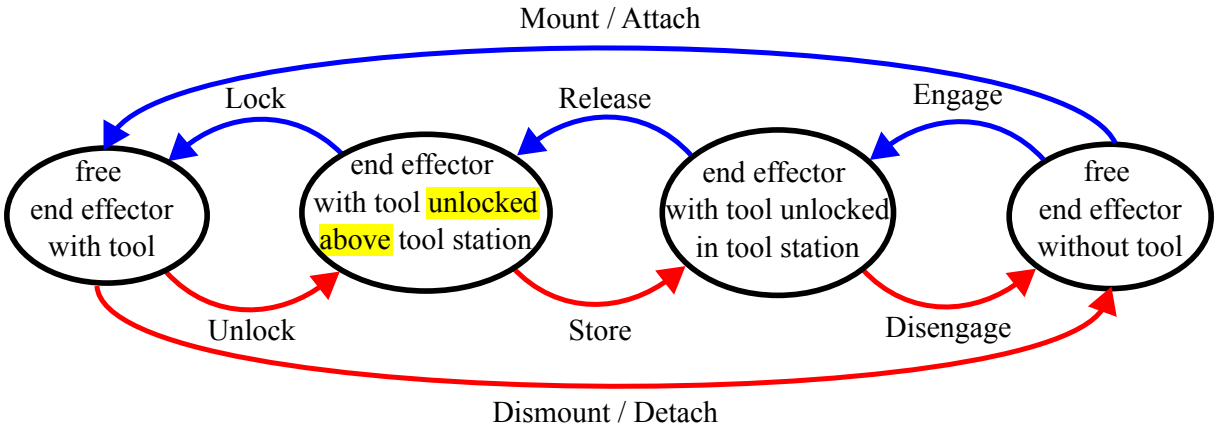
Changing an end-effector happens in several states. As they differ in terms of control behavior (contact force control or position-based control), we differentiate them. Fig. 4.1 gives the naming of these different phases.

We propose two versions of the Finite State Machine (FSM) depending on when the locking/unlocking of the TC happens. In most cases, the locking/unlocking of the TC happens when it is engaged with the tool in the tool station (Fig. 4.1a). Usually, this is the case when the TC is actuated as in Li et al. (2022) or with many commercial TCs discussed

3. https://dictionary.cambridge.org/dictionary/english/tool
4. https://www.dictionary.com/browse/tool
5. https://www.dictionary.com/browse/end-effector



(a) *Solution 1: the locking happens while the tool is still in the tool station.*



(b) *Solution 2: the locking happens after the tool has been released from the tool station.*

FIGURE 4.1 – *Words used to describe the steps of the tool-change. The top and bottom arrows represent the full sequence. The word* free *stands for not in contact with the tool station as this contact restrains end-effector motions.*

in Sec. 4.2.2 that use air to lock/unlock. A second solution is to lock the TC once the tool has been released from the tool station (Fig. 4.1b). This solution is proposed in the design presented in the next chapter because we chose not to use an internal actuator and did not want to rely on a particular type of tool station. Also, in some instances, the locking phase can happen at the same time as the release phase as in Berenstein et al. (2018): releasing the tool from the tool station mechanically triggers the locking of the TC. Then, it requires the tool station to be designed with the tool-changer.

4.2 Choosing initial tool-changers

In this section, an overview of commercially available TCs and how they work is carried out. First, subsection 4.2.1 describes manual tool-changers and their working principles. Then subsection 4.2.2 presents TCs that are considered as automatic as they do not only rely on operator handling. We tested two (a manual and an automatic)on our robotic platform. The following sections of this chapter will explain how they were integrated into our humanoid.

4.2.1 Manual Tool Changer

Many tool-changers are designed for the tool to be mounted and dismounted by an operator. Figures 4.2 and 4.3 show such tool-changers.



FIGURE 4.2 – *From left to right: (a) Schunk SHS manual change system with a payload of up to* 11 *kg and up to 6 pneumatic feed-throughs depending on the variant of the model; (b) Kosmek SXR manual robotic hand changer with a payload of up to* 10 *kg; (c) Stäubli SR series with a lifting force of up to 1000N and up to 10 pneumatic feed-throughs.*

The *Shunk* manual tool-changer [6] (Fig. 4.2) is unlocked by rotating its arc-like handle for more than 90 deg. *Kosmek* [7] provides a small manual tool-changer that unlocks by pushing on both sides of the robot sides. To unlock the *Stäubli* manual robotic tool-changer SR-Series [8] a handle is used to rotate the outer part of the robot side. In each of these examples, once unlocked, the tool disengages by sliding away from the end-effector. This is the major drawback of manual TCs: if the tool is not held suitably when the TC is unlocked, it will fall.

For the *OnRobot* [9] manual tool-changer (Fig. 4.3), the locking mechanism uses a spring, and the tool is unlocked by pressing a button. Then, the tool side rotates around a bar on the opposite side of the locking mechanism on the robot side. To fully detach the tool, a small translation breaks the pivot link by sliding the bar outside its slot. Meanwhile, the tool must be held strongly so as not to fall. We tested the *OnRobot* TC with our humanoid (*Kawada* Robotics HRP-4).

For each mentioned TC, if the orientation is wrong or the tool is not held properly by the operator, it will fall. Furthermore, the operator needs both hands: one to operate the TC and one to hold the tool. This requirement for high dexterity makes it extremely difficult for a robot to operate the TC by itself. Moreover, the humanoid might have end-effectors embedded with tools/grippers that cannot operate the release and mounting of the manual tool-changer mechanism as a human operator does.

6. https://schunk.com/us/en/automation-technology/change-systems/shs/c/PGR_3365
7. https://www.kosmek.co.jp/php_file/eng_product_page.php?no=310_01_01&lang=2
8. https://www.staubli.com/us/en/fluid-connectors/products/end-of-arm-tooling/manual-robotic-tool-changers/sr-series.html
9. https://onrobot.com/en/products/quick-changer

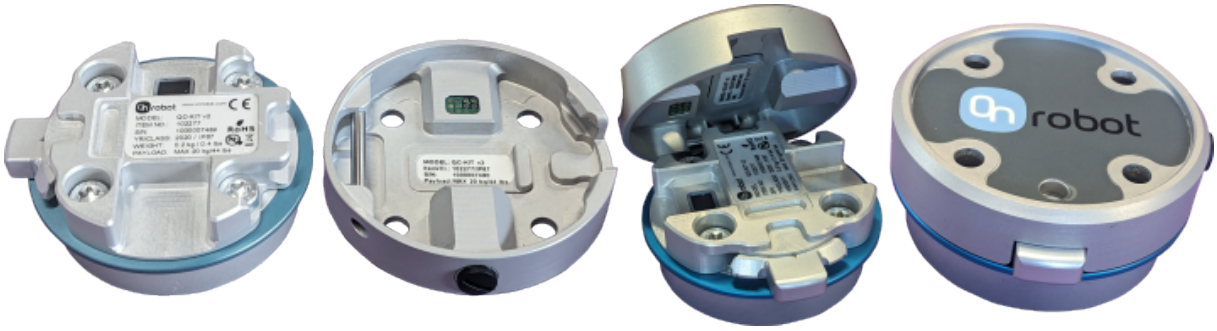


FIGURE 4.3 – *OnRobot TC with a rated payload of* 30 *kg*

With manual TCs, it is challenging for the robot to be fully autonomous. Although the operator is still required to interact with it, they provide convenience when the end-effector has to be changed often. For example, as the HRP-4 robot cannot reach its head, the *OnRobot* manual TC is used. In the team, we experiment with several vision-based approaches. Thus, we need different cameras (see Sec. 4.4.1), for which the *OnRobot* TC is a convenient interface for switching between the solutions on the humanoid head.

Thus, to pursue a better independence of the robot, we have to consider TC designed to be automatic.

4.2.2 Automatic Tool Changer

For the robot to change tools without the action of an operator, automatic tool-changers are to be considered. They can be actuated in several ways: pneumatically, mechanically, or electrically.

Most commercially available ATCs, such as the examples illustrated in Fig. 4.4, use pneumatic actuation. Using air, a piston rod is retracted inside the robot side of the TC. Because of its shape, it pushes outward steel balls that enter grooves in the tool side. Because the grooves are at an angle, the tool side is pulled against the robot side realizing a robust, precise, and repeatable lock of the TC.



FIGURE 4.4 – *Set of automatic tool-changers with similar payload and connectivity options. All of them are actuated by air. From left to right: Stäubli MPS series, Shcunk SWS series, Destaco RHC series, and ATI QC series.*

Then, similarly to the solution presented in Berenstein et al. (2018), some TCs use the motion of the end-effector to release the tool and the interaction between the TC and the tool station to lock the TC. Likewise, the TC is unlocked during the storing motion. We call them *mecanical* TC. Several solutions are illustrated in Fig. 4.5.

The principal advantage of these solutions is that they do not require an extra power source to be actuated as they rely on the robot's high number of actuators and degrees of freedom. Also, with some products, the operator can change the tool, without the robot's intervention. However, mechanical TC presents several drawbacks. To lock/unlock a mechanical TC, the robot has to generate a force against its environment. This required force can be greater than what the robot can provide in the locking/unlocking configuration (up to 75 N in some cases). It would also be possible for a robot to accidentally unlock the TC during an operation if an unexpected contact with the environment were to happen.



FIGURE 4.5 – *Set of automatic tool-changers with similar payload and connectivity options. From left to right: Triple A wingman, smartshift ATC, Kosmek.*

Finally, we can also consider electrically actuated TC. The main disadvantage of an extra actuator is the added weight to the end-effector, which will reduce the robot's payload. Most importantly, this actuator will have to be integrated from an electronic and firmware point of view into the robot's framework. Thus, almost no manufacturer offers this solution.

However, when considering humanoid robots, it is not convenient to use pressurized air to actuate a TC as they are not usually connected to a pressurized air network or have an air pump or reservoir equipped. When this study on TCs for humanoid robots started, few brands offered a mechanical TC solution. Initially, to integrate an ATC into our humanoid platform, we chose a prototype solution, modifying Kosmek's TC with the THK seed interface to create an electrically actuated TC. Fig. 4.6 shows the kosmek tool-changer with the seed modification that enables it to be electronically actuated. We were able to perform both hand and foot changes with this TC. This is described in the following sections.

For all the presented commercial solutions, the power transmission issue has also been answered with a multitude of options, from pneumatic feed-troughs to springloaded connectors. In our case, we need only electronic connections to power extra actuators and sensors for the hands and feet of the robot, as well as connection allowing for high data pass-through to plug different cameras on the head of the robot.



FIGURE 4.6 – *Kosmek tool-changer, from left to right: robot and tool side engaged together but not locked, tool side, robot side. The black parts on each side contain springloaded electric connectors to power tools on the tool side. These parts can be removed if the electric connections are not required to reduce the space occupancy.*

4.3 Hardware integration

The TC integration into the humanoid robot is described in this section. Instead of having a simple interface, the TC has to be installed between the last link of the robot limb and the end-effector. In the industry, the interfaces are standardized. However, the chosen tested TCs did not match the force sensor interfaces on HRP-4. Furthermore, the interfaces between the robot and TC, and between TC and the end-effector are designed to simplify the change of the end-effector.

This section is organized as follows: first, the physical parts to mount the TC on the robot and tools are presented for the OnRobot and Kosmek TCs. Then, the electronic integration of the Kosmek TC is designed.

4.3.1 On Robot

Fig. 4.7 shows different interfaces and ways to mount the *OnRobot* TC on our humanoid. Even though the *OnRobot* TC has standardized interfaces, they do not correspond to the interfaces of the robot's FT sensors. Thus, new interface parts were designed. Due to the alignment of mounting holes on the TC and FT sensor, the interface had to be split into two. It avoids interference between the screws and ensures the correct orientation of the TC with respect to the end-effector. The first part is fastened to the FT sensor on the robot side, while the second one is secured to the TC. Then, both parts of the interface are screwed together. This is the method used for the interface at the robot's wrist.

When then the different mounting holes do not interfere, it is possible to have a simple design such that the interface is fixed on the FT sensor, and the TC is then screwed on the interface (e.g. for the ankles and head)

The interfaces between the OnRobot TC and the end-effectors were designed similarly.

However, in some cases, such as the feet, as we devised the end-effector, the TC was mounted directly on the end-effector.

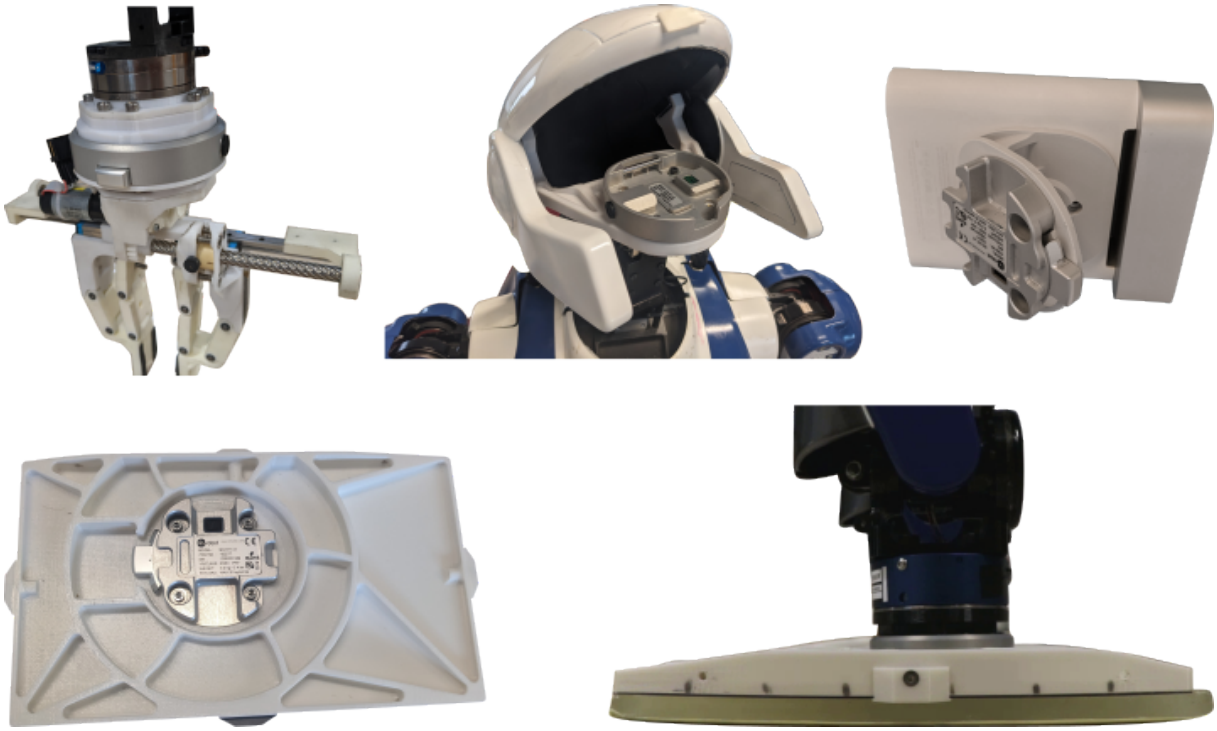


FIGURE 4.7 – *Top line, left to right: hand gripper attached to the wrist, head without camera, Azure Kinect with tool side. Bottom line, left to right: foot with tool side, foot attached to ankle*

The interfaces are 3D printed. It is a convenient solution in most cases, especially for the hands and the head. However, the printed interfaces introduce a lot of flexibility at the ankles as the torques are larger.

4.3.2 Kosmek hardware interfaces

The basis for the Kosmek TC interfaces is similar to the OnRobot TC with 2-part, 3d printed assemblies because the TC and robot/tool interfaces do not match. Additionally, the Kosmek TC is automated. Thus, the interface parts can fulfill two other functions. First, a way to store the tool when detached from the robot is needed. It has to be on the tool side interface. Second, the interface should help locate the end-effector with respect to the tool station by providing visual cues or surfaces for simpler environment interaction. The shape and constraints of humanoid robots also require a distinction between the hand and the foot end-effectors.

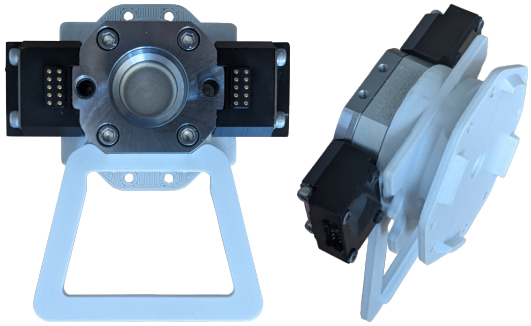
Wrist TC interfaces

For the hands, Fig. 4.8 shows the robot side, the tool side, and the tool station of the hand TC. The interface between the TC and the tool (Fig. 4.8a) has an overall shape of a flat cylinder with fins on its side. The interface has to be slim so as not to decrease the payload of the arm. It fits in the tool station (Fig. 4.8b) that has a U-shape with two slots to let the fins slide in. The top of the tool station has a funnel shape. It guides the tool side interface

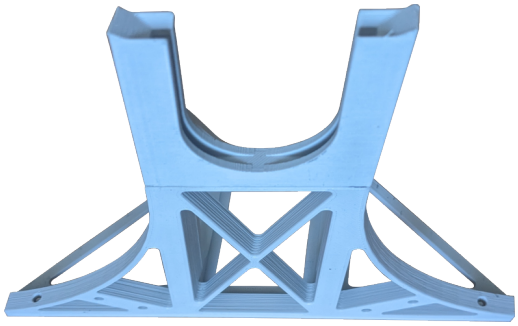
to the correct position, and increase the tolerable positioning error when storing the tool. In the configuration used in the initial test, the expected error in the horizontal directions is of ± 5 mm. These tolerances are within the range of the initially used vision system and are larger than the ones required by the tool-changers to lock.

Two other implements are considered to simplify the initial approach and positioning. First, under the interface part (Fig. 4.8a), a trapezoidal shape helps position the tool before storing it in the tool station. This shape makes touches with the tool station first. This contact removes two degrees of freedom when positioning the end-effector w.r.t. tool station: the horizontal translation along the z-axis (normal of the tool station front face) and preventing the rotation around the vertical axis. It does not significantly increase the collision shape of the TC because once the TC is engaged, it overlaps the actuator.

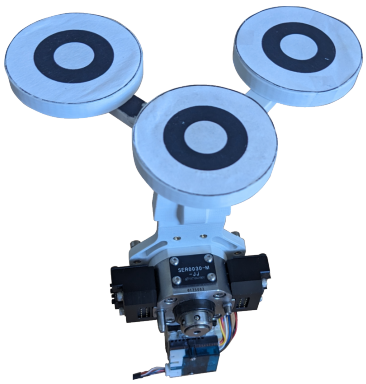
Second, to use precise and simple visual servoing for the initial tests, Whycon markers from the whycon library Ulrich et al. (2022) have been integrated into the design of the robot side wrist interface and the tool station. They are positioned in L-shapes of different side lengths to distinguish between the markers on the wrist of the robot and the markers on the tool station. Details of their usage for position-based visual servoing are given in the next section. It is expected from the vision to be precise enough to be within the tool station tolerances along the remaining 4 axes (2 translations and 2 rotations)



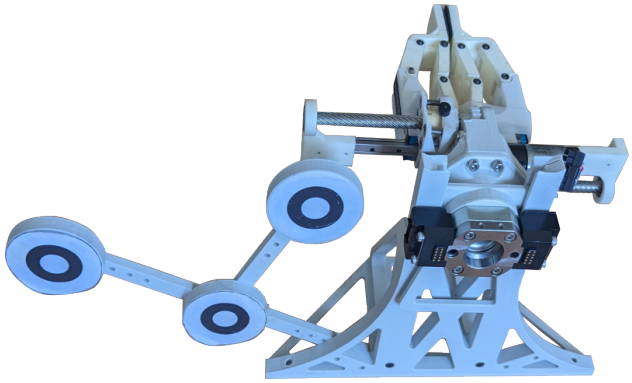
(a) *Tool side and interface between TC and tool*



(b) *Tool station without visual markers*



(c) *Robot side and interface between TC and robot with whycon L-shape markers*



(d) *gripper stored in the tool station with whycon markers*

FIGURE 4.8 – *Wrist integration of the Kosmek tool-changer*

Foot TC interfaces

When considering the feet, two issues arise. First, because the robot's camera robot is on its head, its body and arms often prevent the feet from being within its line of sight. Thus, when changing a foot, it is often impractical to rely on vision. Second, the feet's payload is much higher than the hands as they support the whole weight of the robot. So, to avoid damaging the TC, the interface has been designed to transmit part of the forces and torques instead of the TC. Fig. 4.9 shows the proposed solution.

In this case, the foot and the tool side interface were merged similarly to the OnRobot TC. The TC is off-centered, so the efforts are transmitted through a planar contact between the robot side interface and the foot (circular surfaces). Theoretically, the TC only transmits the moments and the pulling forces along the vertical axis. Residual efforts due to manufacturing tolerances and part flexing under the load were not considered in the design but existed and prevented walking with this TC. The robot side interface has two contact surfaces to probe its surroundings and locate its ankle relative to the tool station. Additionally, a funnel shape is added on the tool side to compensate for positioning errors and guide the ankle when engaging the new foot.



FIGURE 4.9 – *design of the foot parts for the kosmek TC*

4.3.3 Electronic integration of the Kosmek TC

The Kosmek electric TC uses a custom control interface from *THK* called Seed [10]. The Seed interface's programming language is a custom visual language working under the Windows operating system. As such, it was impossible to connect it to the robot's controller directly. An interface was built using an Arduino Mega card. The Arduino communicates with the Kosmek TC control card using its input/output pin and connects to the computer controlling the robot using a serial port. A mc_rtc module was created to have the communication with the Arduino card handled directly in the robot's controller. Fig. 4.10 shows the electrical montage. Furthermore, the Seed card and motor controllers require an extra 24 V power supply. Due to the number of the number of connections required with the computer as well as the extra power cable, this montage was not mounted on the robot.

10. https://www.seed-solutions.net/

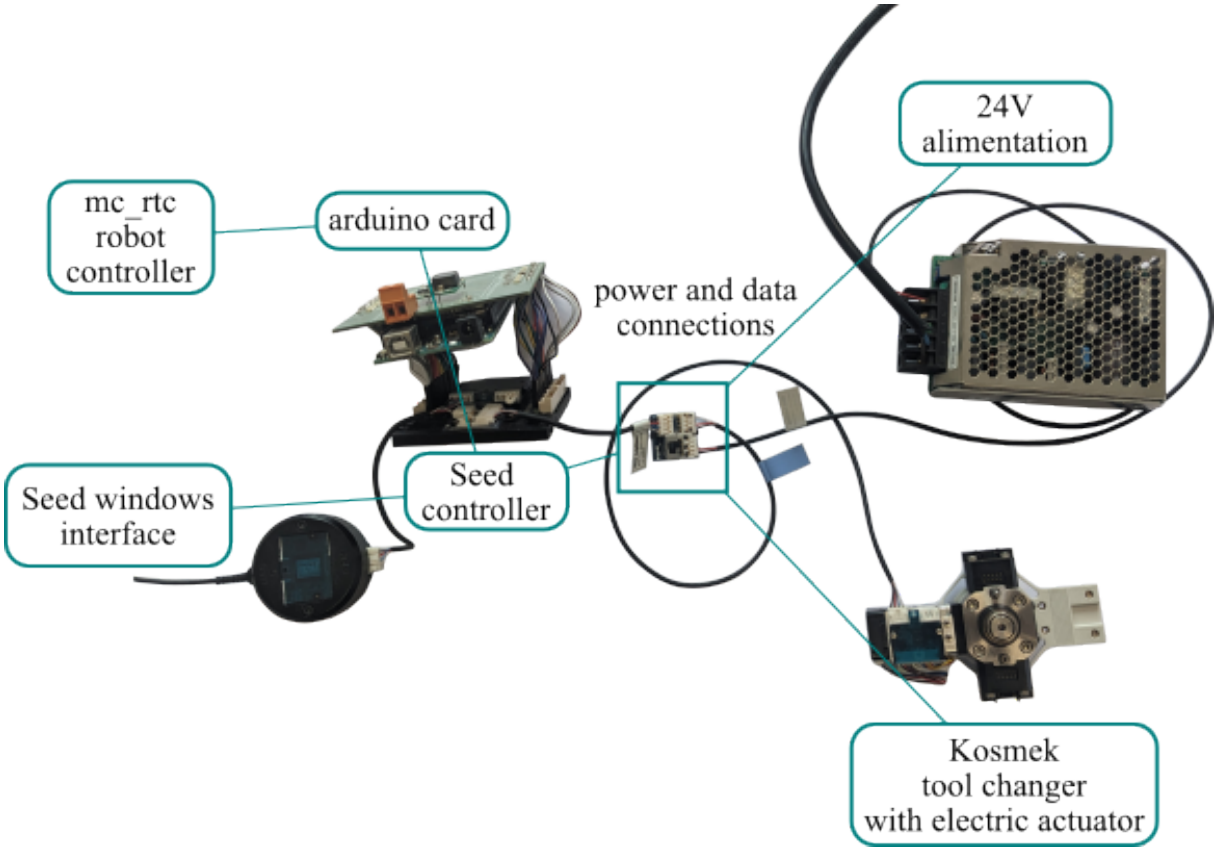


FIGURE 4.10 – *Kosmek TC electrical connection to make it work*

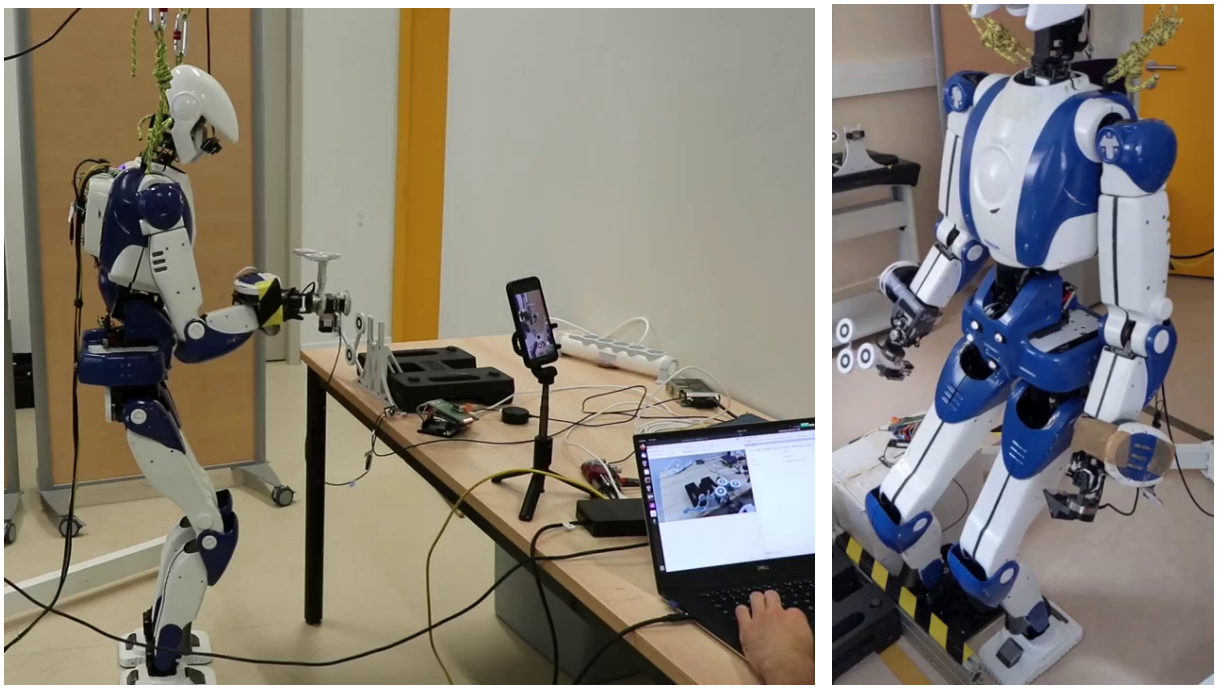
4.4 Behavior integration on HRP-4

In this section, the details of the integration of TC on HRP-4 and the initial tests are presented. It is assumed the robot starts in a configuration where it can reach the tool station. This configuration can, for example, be reached with SLAM. Then, to start the end-effector change, it has to move its end-effector close enough to the tool station. Depending on the limb and whether its terminal point can be seen by the robot, several approaches are used. Subsection 4.4.1 details the implementation of visual servoing in tool change. This allows moving the end-effector close enough to the tool station to make contact when the limb can be seen by a camera. Then, when the limb cannot be seen, or once the end-effector makes contact with the tool station, contact forces must be controlled to properly store, disengage, engage, and release the tool as explained in subsection 4.4.2. Finally, it is necessary to update the model of the robot in the control software once the end-effector is changed as described in subsecsion 4.4.3.

The details discussed here are illustrated with simple test scenarios of hand and foot changes as shown in Fig. 4.11.

Head change is not discussed in this section. The OnRobot solution has been used on the HRP-4 robot for some time as it provides a lot of convenience, and autonomous head change has only been tested with the new proposed EEC design as detailed in Sec. 5.2.4.

The behaviors implemented in this section will be built upon in the next chapter to integrate the proposed end-effector changer.



(a) *Kosmek hand change* **(b)** *Kosmek foot change*

FIGURE 4.11 – *Experiment with Kosmek tool-changer*

4.4.1 Vision integration

In the topic of visual servoing, two approaches are possible: Image-Based Visual Servoing (IBVS) and Position-Based Visual Servoing (PBVS). When positioning a frame from the robot w.r.t. a target frame, a position task is formulated. In IBVS, the error driving the position task is formulated based on image features. When using PBVS, the relative positions of the controlled frame and target frame w.r.t the camera are computed. In this work, we chose the latter approach as it was initially available in our framework. Furthermore, visual servoing is used only for arm end-effector change, as the other end-effectors are not in the line of sight of the camera on top of the robot's head. Then, vision requires extracting relative poses from camera features.

Initially, the positions of the robot's wrist and the tool station in the camera frame are estimated using whycon markers. The whycon library Ulrich et al. (2022) provides the position of each circular marker in the camera frame (Fig. 4.12a). However, only one marker is not enough to estimate a body pose as only its position is computed. Thus, markers are grouped by 3 in L-shape configurations. Each L-shape has a unique length for its orthogonal segments, identifying it [11].

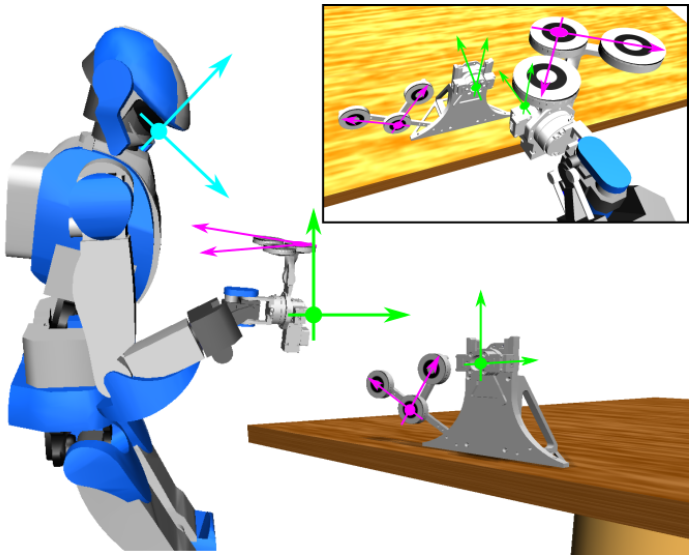
Figure 4.12b illustrates the different frames used in the visual servoing task. The PBVS tasks move the purple frame on the robot's writs to a target position. This target position is computed from the desired offset between the green frames and the transform between the purple and green on the robot and the tool station. Combined with the approaches presented in the next section, we can perform tool changes reliably. However, as seen in Fig 5, the markers in a L-shape considerably increase the collision shape of the wrist. They also add

11. https://github.com/mc-rtc/whycon_lshape

weight that reduces the arm payload. Several other solutions were considered to circumvent these issues.



(a) *Output of the whycon library with the Stereolab Zed 2 camera)*



(b) *Representation of the different frames used for visual servoing (picture from Choreonoid simulation). Light blue on the head of the robot is camera frame; Green on the wrist of the robot is the controlled frame; Green on the tool station is the target frame; Purple on both the robot wrist and the tool station are the frame tracked by the vision.*

FIGURE 4.12 – *Visual servoing using the whycon library*

First, using markers only on the tool station, the pose of the tool station w.r.t. the robot is estimated. From the whycon markers, the position of the tool station w.r.t. the camera frame is found. Then, using hand-eye calibration, the position of the camera frame w.r.t. the robot head link was estimated. It is then possible to compute the position of the end-effector frame w.r.t the tool station with the robot's kinematics. This method does not use visual servoing and is limited. The hand-eye calibration precision is not high enough. For example, given an average distance between the camera and the end-effector of 1 m, if the orientation of the camera frame w.r.t. the robot is wrong by about 1 deg then the positioning of the end-effector using the vision will have an error of about 100 cm$*\sin(\pi/180) = 1.74$ cm. Experimentally, the observed errors reached up to 10 cm after several attempts at calibration. We concluded that position-based visual servoing is much more robust, and we looked for other libraries

to attach frames to target bodies in the camera frame other than whycon markers.

When not using markers, most vision solutions need the CAD model of the tracked object. Several solutions using different types of cameras and modalities exist and were tested:

VISP from INRIA Trinh et al. (2018) the included trackers function with any monocular camera and showed promising results with simple shapes. However, when using more complex models, such as the tool station or the robot's wrist, the computations slowed, and the tracking became not robust enough for our application.

BigObjectTracking Chappellet et al. (2023) A dense 3D point cloud is used with strategies similar to ICP. It requires a lot of sampled points on the target object, and working with a wide-angle depth camera such as Azure Kinect does not provide enough points on the tool station and robot wrist as they are too small.

M3T from DLR Stoiber (2023) Robotic limbs and end-effectors are tracked using RGB-D cameras. It uses both depth and color images to track the object. It is remarkably resilient to occlusion.

The solution proposed in Stoiber (2023) is implemented in our framework with the new end-effector changer presented in Chap. 5 (Fig.4.13).



FIGURE 4.13 – *Robot's point of view using the Kinect Azure camera with the models of the parts track by the M3T library superimposed.*

The main limitation of these approaches is the initialization: they rely on local optimization, assuming that the current pose of the object is close to its last estimation. Thus, when the pose error becomes too important, the tracking algorithm will converge to a local minimum. Then, they need to be re-initialized. Often, the initialization step or detection step is done manually. The user has to click on 4 points on the image corresponding to predefined vertices of the model. From these points, the initial pose of the object can be estimated. A detector could be trained to find the object automatically and initialize the tracking algorithm.

4.4.2 Contact control

The vision is not precise enough to position the end-effector for the TC to lock itself, as we reach a precision of around 1 cm while the required precision is 1 mm (see Table 4.1). The tool cannot be changed using only a position-based control strategy, and a force-based control strategy must be used.

To control the contact forces inside mc_rtc, admittance tasks are used. As described in the first two chapters of this thesis, applying contact and controlling the contact forces requires accounting for the balance of the robot. It is done using the balancer presented in Murooka et al. (2021).

Hand Change

The strategy used for the hands to engage the TC or to store the tool in the station is to have the hand move in a direction while regulating the forces to zero in the other directions until a threshold force is detected. The implementation relies on admittance tasks.

When storing the tool, the geometry of the station and the interface between the tool and TC guide the tool. With visual servoing, the end-effector moves close to the tool station without making contact. Then, the end-effector moves forward ($x+$ axis) until a force is detected on that axis. It happens when the trapezoidal part on the interface part (Fig. 4.8a) makes contact with the tool station. Then, a target velocity downward ($z-$ axis) is given while the force and torques in the other directions have a zero target. The geometry then guides the tool to its storage position using the compliance given by the admittance task. When releasing the tool or disengaging the TC, the strategy is similar, with only the motion direction that changes (up for releasing the tool, back for disengaging the TC).

However, when engaging the TC with the Kosmek TC, its geometry does not allow sufficient positioning tolerances to lock it when the initial contact is created. Thus, simple peg-in-hole strategies Van Wyk et al. (2018) are used. Once contact has been established, to ensure that the TC is in the correct position before it is locked, the end-effector follows a simple spiral trajectory sliding against the surface as shown in Fig. 4.14. On each axis, the resulting motions and contact forces are shown in Fig. 4.15. Detecting the proper engagement is challenging: it is not possible using vision as there is not enough precision; it is not possible to consider the pose because of the flexibility of both the tool station and the robot. One can consider the contact forces and the norm of the component in the spiral plane: in Fig. 4.15, proper engagement can be observed to happen just after 20 s. The spiral was big enough to account for most of the vision limits while avoiding too big of contact forces (max 6 N in Fig. 4.15).

This initial strategy led to repeated tool changes without issues. However, because of friction, such strategies may damage the TC and the connections. Furthermore, this approach shows a major concern when changing tools: how to ensure the TC is engaged properly? This issue is barely tackled in the literature. In Gyimothy and Toth (2011), the authors add sensors to their design. In the other works cited in Sec. 1.2, the success of TC engagement was observed by the operator.

Finally, with these strategies, the robot performed multiple hand changes in a row where

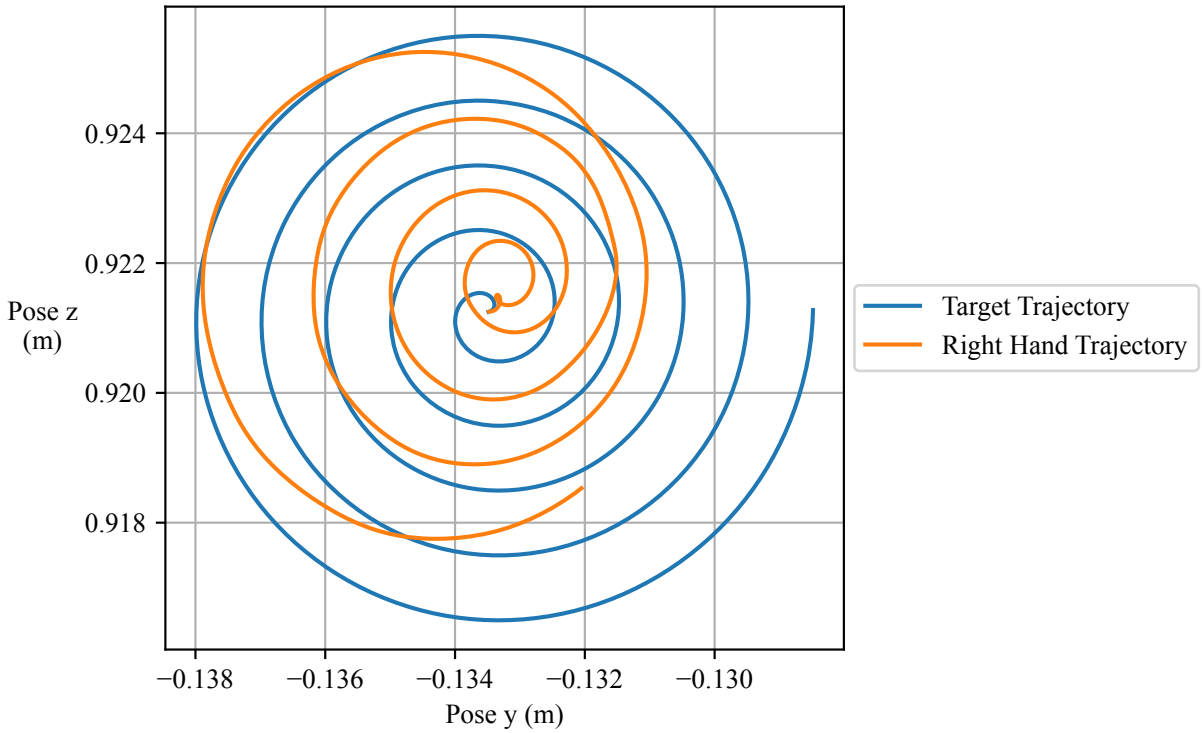


FIGURE 4.14 – *Spiraling hand trajectory to adjust the TC position when engaging it.*

the tool station was moved each time the robot was not in direct contact with the tool station.

Foot Change

When changing foot, the vision cannot locate the tool station. Thus, it is found by probing the environment with the foot and the ankle. Assuming the support foot, on which the robot is standing, is not slipping. We first find the position of the corner frame by pressing the front detection frame against the front wall (blue surfaces in Fig. 4.16). It gives the x coordinates of the corner. Then, pressing the side detection surface (green in Fig. 4.16) against the side wall gives the y coordinate of the corner. Knowing the position of the tool w.r.t. the corner, its position w.r.t. the robot can be estimated (see Fig. 4.9). The admittance task ensures that each contact is planar. Thus, as a planar contact blocks rotation around the 2 orthogonal axes in the contact plane and both probed planes are orthogonal, the orientation is also estimated. Finally, we use admittance on the z-axis above the tool-changer to insert the ankle into the foot. For the final insertion step, the funnel shape of the tool-changer is used to compensate for any lack of positioning precision. This approach can also be used to leave the tool in the correct position when removing it.

4.4.3 Model update

Once the TC has been engaged or disengaged, its kinematic and dynamic properties change.

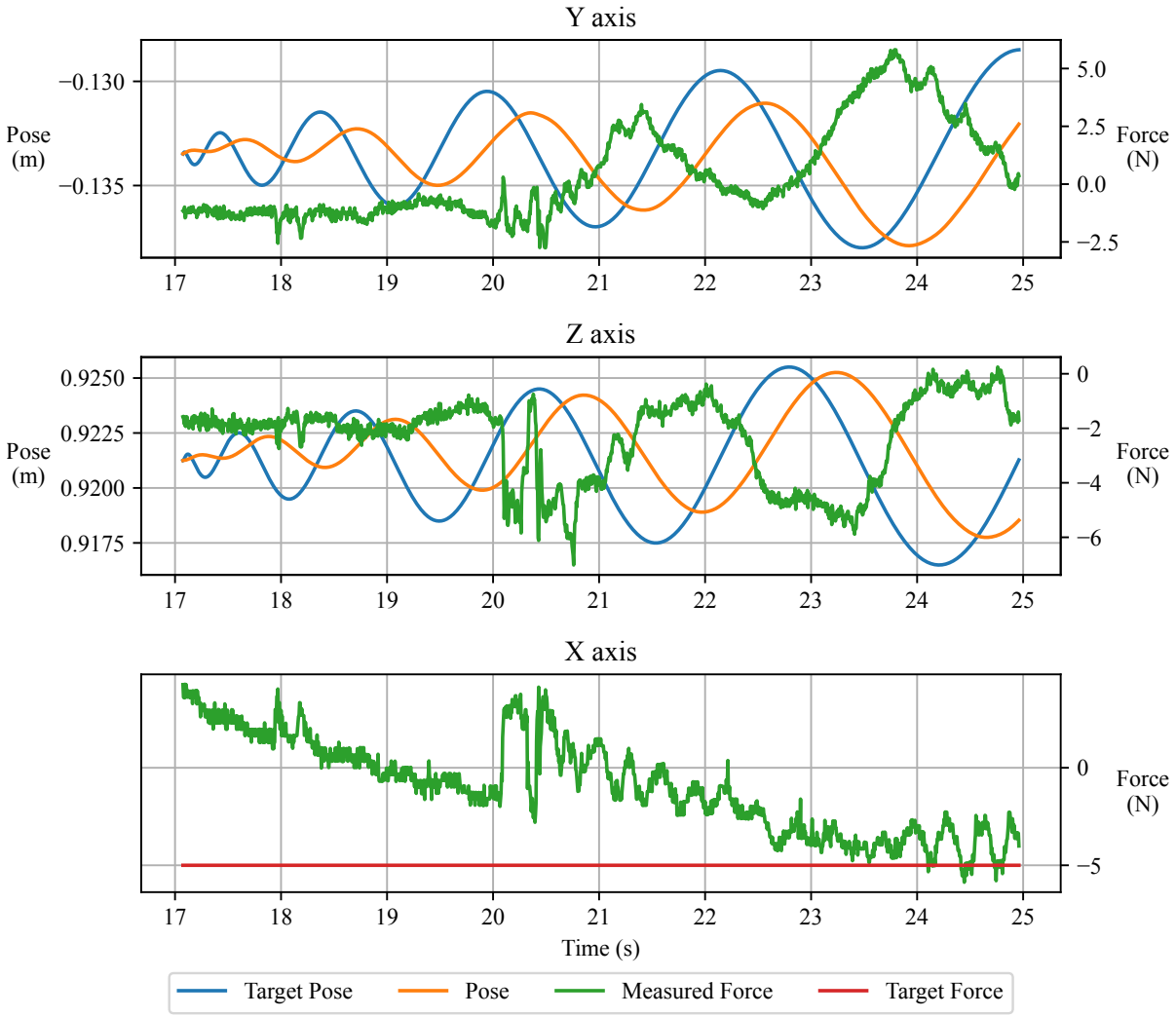


FIGURE 4.15 – *Trajectory and contact force at TC level when engaging the TC.*

After locking the TC, any added actuator has to be turned on. Respectively, before unlocking the TC, any removed actuator must be powered off. Turning on (resp. off) must happen after locking (resp. before unlocking) instead of after engagement (resp. before engagement) because when the TC is locked, the electric connections are guaranteed, and no shortcut can happen.

These added or removed actuators and corresponding links must be accounted for in the robot's controller. Thus, its model has to be updated online. It is done by reloading the URDF model of the robot in the controlled with the applied modifications. Additionally, from the kinematic chain update, collision shapes, and usable contact surfaces are changed to avoid undesired collisions with the environment.

Finally, it is important to consider changes in dynamic properties as they will directly impact forces measured by the force sensors.

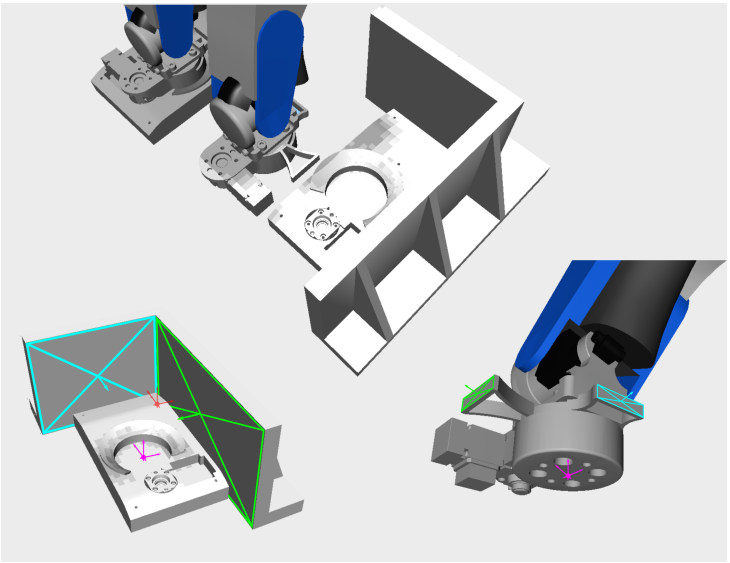


FIGURE 4.16 – *design of the foot parts for the kosmek TC*

4.5 Limitations and design constraints

Based on (i) the literature review (Sec. 1.2), (ii) the listed commercial tool changing solution (Sec. 4.2), and (iii) the initial tests done in Sec. 4.3 and 4.4, the limits of the existing tool-changing solutions are summarized in this section. Meanwhile, a set of design constraints for end-effector changers to be used with humanoid robots is established. Table 4.1 summarizes the properties of tool-changers presented in Sec. 1.2 and 4.2.

HRP-4 weighs less than 40 kg. Adding weight, especially at the tip of its limbs, impacts its dynamics. Also, the added weight will increase torques at the limbs' base and reduce their payload. Thus, the EEC has to be lightweight and flat (as lengthening the limb also reduces its payload).

Considering the tests with the Kosmek and OnRobot TC, autonomous tool change is a must. Yet, keeping the TC useable by an operator is an advantage. Not having an actuator is essential to simplify the integration into existing robots. An extra actuator adds considerable weight, and integrating it into the robot's electronics, drivers, and command can be complex.

As a humanoid robot has multiple limbs, a different end-effector can be used to operate the TC. For instance, the TC on the right arm can be locked by the left arm end-effector. The design would have to account for the operating limb to have a variety of possible end-effectors. Not relying on the design of the tool station for locking or unlocking would increase the versatility of the TC as it would not be as environment-dependent. Consequently, such a design should also be easily human-operated.

Tolerance to initial positioning error is an issue often highlighted in the literature that we could observe firsthand. Yet, it does not seem to be an issue for commercial stuff. Preliminary tests showed a realistic positioning error using visual servoing to be around ± 5 mm in position and ± 5 deg in orientation. Hence, the design should allow for such positioning errors when initiating the engagement of the TC.

TABLE 4.1 – *Existing TC data (extracted from data-sheets). Weights account for both robot and tool side. Required locking precision is the allowed positioning error when locking the TC.*

Model	Weight / Payload	Required Precision	Advantages	Disadvantages	Pneumatic / Electric Connexions
Schunk manual (SHS 050)	350 g / 11 kg	1 mm	manual	hard to handle	4 / no
OnRobot Quick Changer	200 g / 20 kg	-	manual	engagement trajectory	no / yes
Kosmek manual (SXR)	≈230 g / 10 kg	-	manual	pressing force to unlock	no / no
Kosmek mechanical (SMR)	≈650 g / 12 kg	±1.0 mm 1.2 deg	manual and autonomous, actuatorless	important contact forces to operate	up to 4 / optionnal
Kosmek electrical	≈400 g / 3 kg	±1.0 mm 1.2 deg	automatic	electronic integration	up to 4 / optionnal
Proposed in Gyimothy and Toth (2011)	- / 200 N, 10 N.m	±4.0 mm 1.0 deg	engagement detector	complex actuated design	0 / yes
Proposed in Berenstein et al. (2018)	-	-	actuatorless, large tolerances	no electric connection	0 / no
Proposed in Pettinger et al. (2019)	-	±3.0 mm (up to ±10 mm)	actuatorless, robust	no electric connection	0 / no
Proposed in Li et al. (2022)	-	-	Many connectors	important size powered by RPi	2 / yes
Proposed hand EEC	≈130 g Sec.5.1.3	±5.0 mm >5 deg		prototype	0 / yes
Proposed foot EEC	≈120 g Sec.5.1.3	±5.0 mm >5 deg		prototype	0 / yes

Furthermore, TCs should withstand the forces and torques of the robot's normal operation. The limits of the force/torque sensors to which the TC will be attached are reasonable targets.

Then, the TC connects the tool and the robot at a power and data level, so pass-throughs for electrical, data, and fluid connections have to be considered in the design.

For these last two constraints, we account for three different use cases of the TC:

hands relatively low-payload end-effectors that often accommodate a wide variety of pos-

sible grippers, tools, and sensors;

feet high-payload end–effectors that support the weight of the robot and its dynamics during walking;

head much lower payload end-effectors equipped with sensors that send large amounts of data, such as cameras. The data connections tend to use normative cables such as the USB-C norm.

4.6 Conclusion

Tool-changers for robots are a mature technology, as shown by the many market solutions. Initial tests demonstrated that some TC models can be fully integrated into humanoid robots with the technologies developed over the last decade (visual servoing, force control...) However, not all TCs are suitable for humanoids: some are too complex to operate autonomously, require high positioning precision, or are too heavy.

In the context of a research lab, where we experimented with different technologies, using TCs is highly beneficial. The case of the *OnRobot* TC used for the head camera on HRP-4 illustrates this need, as various specific cameras are regularly installed on the robot as they are required by different tested vision libraries.

The initial tests also revealed the limits of the existing TCs. We outlined a set of constraints to improve their use in humanoids. In the next chapter, we propose and integrate an end-effector changer that tries to answer this need. This EEC relies on the extra DoF of a humanoid to lock/unlock it mechanically.

The next chapter will capitalize on the lesson learned in these tests to propose a novel design of an end-effector changer based on the capabilities and constraints of humanoids. This new design accounts for all the shortcomings enumerated in the following sections, e.g., lightweight, no-actuator embedded, large precision tolerance in mounting and dismounting, no need for human operator assistance, etc.

CHAPTER 5

New Design of an End-Effector Changer for Humanoids

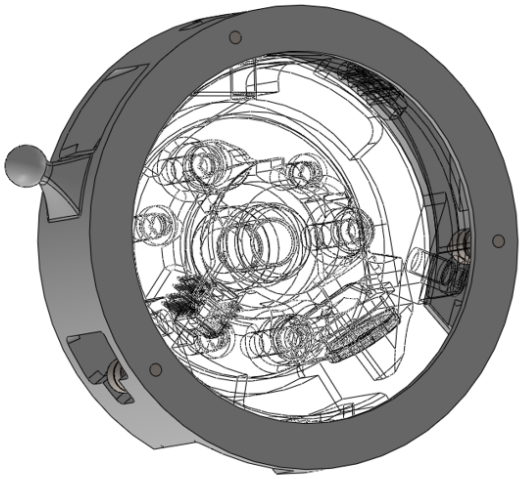
In the previous chapter, we explored the use of tool-changers to improve the capabilities of humanoid robots. After reviewing existing models from the literature and robotic accessories manufacturers, we tested two on our platform. As a result, tool-changers, or in this case, End-Effector Changers (EEC), were showed to be viable and beneficial for humanoids. However, the limits of the tested models were clear, and we outlined a set of design guidelines to improve them: mainly, they need to be lightweight, without actuators and offer a large tolerance to positioning error.

In this chapter, we propose a new design for an end-effector changer mechanism suitable for humanoids. It was a collaboration with Jean Baptiste Izard from Alted who prepared most of the design while we analyzed it and integrated it to the humanoid HRP-4.

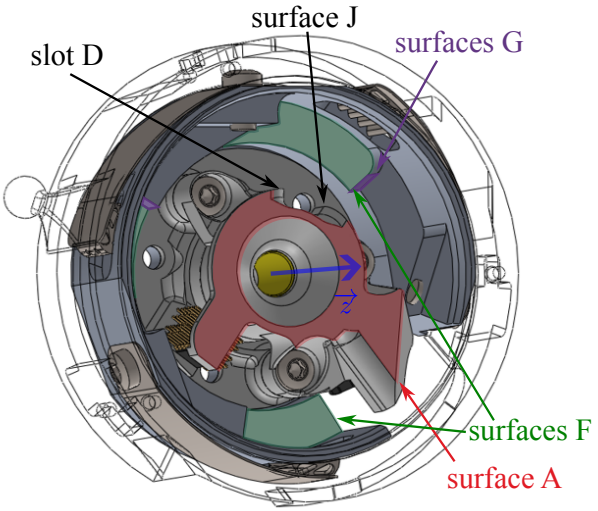
This chapter is organized as follows: in Sec. 5.1, the working principle of the proposed EEC is presented, and then the variation to use it for hand, foot, and head end-effector is detailed before it is analyzed using finite element analysis (FEA). Then, in Sec. 5.2, the efforts to integrate it into the different end-effectors of the robot are presented along with the other tool stations used to store tools. Finally, Sec. 5.3 and 5.4 discuss the results and possible improvements and conclude this chapter.

5.1 Design of the end-effector changer

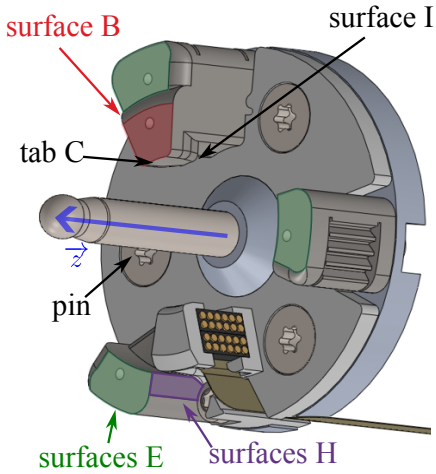
In this section, we describe the design and working principle of the proposed end-effector changer (EEC). After describing the general solution (targeted at the hands), we present two variations for the feet and head. These variations account for the requirements of different end-effectors suggested in the previous chapter. The design strain-stress is initially evaluated using finite element analysis (FEA).



(a) *locking bell*



(b) *robot side*



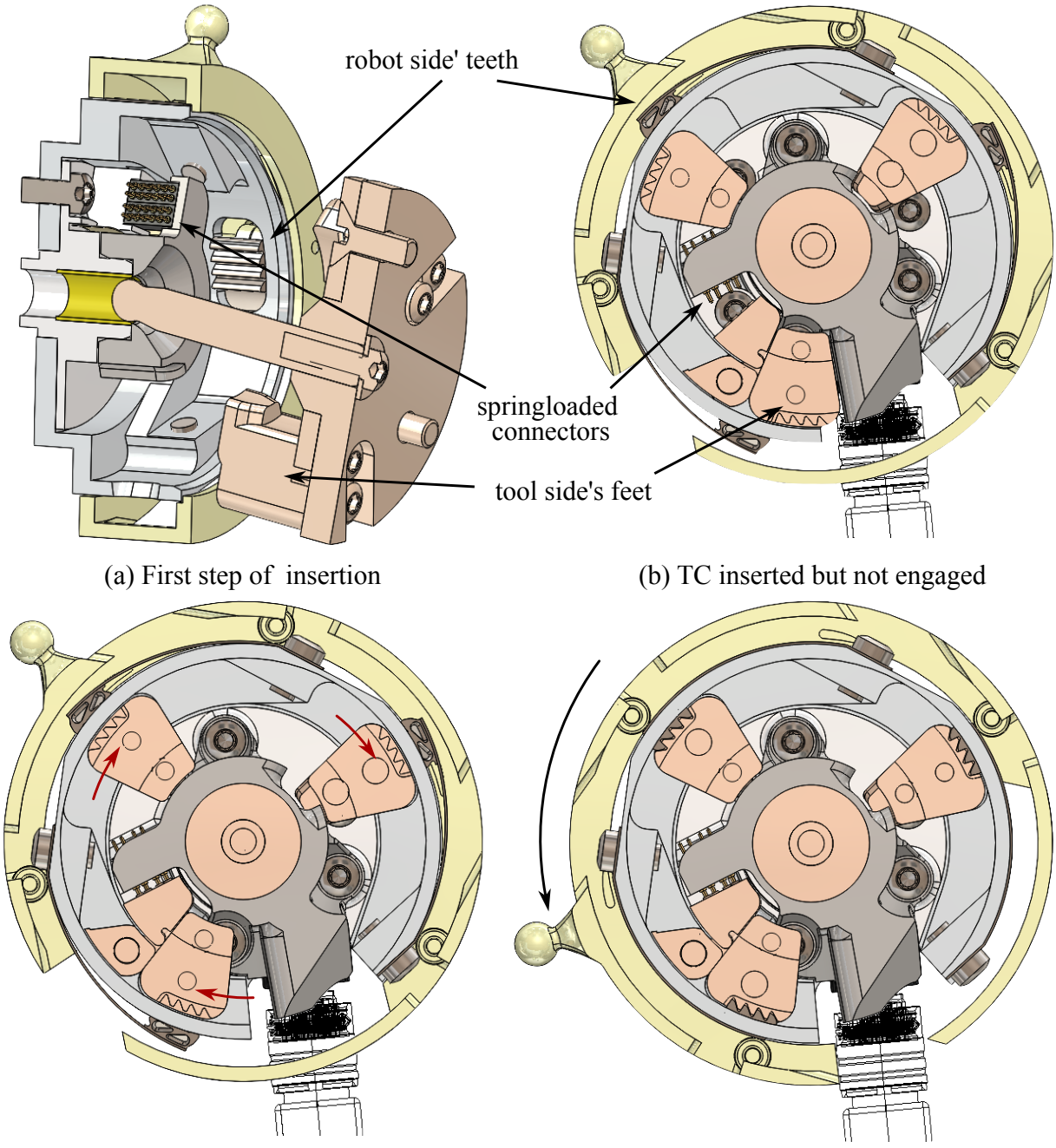
(c) *tool side*

FIGURE 5.1 – *Subassemblies of the proposed design. On the bell, the small sphere is used to lock/unlock the EEC. The surfaces are color-coded so to highlight that surfaces of the same color on both the robot and end-effector/tool sides will match once the ECC is locked. Surfaces E, F, G, and H are each a set of 3 similar surfaces spread around the ECC.*

5.1.1 Mechanism and operation phases

The EEC is made of 3 assemblies (Fig. 5.1): (i) the tool side (Fig. 5.1c with the pin coming out of it, attached to each end-effector/tool, (ii) the robot side (Fig. 5.1b) mounted on the robot instead of the regular end-effector, and (iii) the locking bell (Fig. 5.1a) rotating around the robot side to lock and unlock the EEC when the tool and robot sides are engaged.

The engagement of the EEC is as follows: first, the spherical tip of the pin on the tool side makes contact with the funnel on the robot side. Then, the tip slides along the surface of the funnel until the center is reached (Fig. 5.2a), locking the relative translations of the two sides along the plane normal to the z-axis. As the pin is pushed inside a plain bearing both z-axis of the tool and robot sides become colinear, letting only rotation and translation around the common z-axis possible. The pin is inserted until surfaces A and B are in contact.



(a) First step of insertion

(b) TC inserted but not engaged

(c) Final motion to engage the TC

(d) Motion of the bell to lock the TC

FIGURE 5.2 – *Different steps of EEC engagement. Sub-figure 5.2a is a slice in a plane containing the z-axis. Sub-figures 5.2b, 5.2c, and 5.2d are slices normal to the z-axis at the bottom of the pin looking from the end-effector/tool side toward the robot (effector) side. In each sub-figure the parts of the TC are color-coded in the same way.*

Next, both sides rotate relatively to each other until tab C enters slot D. This ensures the correct orientation for the final translation along the z-axis without damaging the electrical connectors. At this point, surfaces E and F are in contact (Fig. 5.2b). A rotation until contact between surfaces G and H will end the engagement of the two sides by preventing the tool side from moving back along the z-axis (contact between surfaces I and J) (Fig. 5.2c). The locking of the mechanism is done by rotating the bell around the robot side which pushes the teeth of the robot sides in the feet of the tool side (Fig. 5.2d). This locks the mechanisms

by preventing the final rotation along the z-axis.

The length of the pin is computed such that in the worst case scenario where the positioning error is 5 mm and the orientation error is 20 deg then there is no interference between the feet on the tool side and the robot side (Fig. 5.2a). However, to permit this means that once inserted, the pin sticks out the back of the robot side. Thanks to a hole in the middle of the wrist F/T sensor, it is not an issue.

It is important to note that the bell bears no load as all efforts are transmitted through the tool and robot sides.

For both standard and high payload versions, the electrical connection supports up to 24 wire connections and is designed to provide EtherCat-like connection capabilities. On the robot side, a standard connection using Fisher MiniMax connectors [1]. The data and power are rooted through each side of the TC using flexible PCBs. When the EEC is locked, spring-loaded pins link both sides electronically [2].

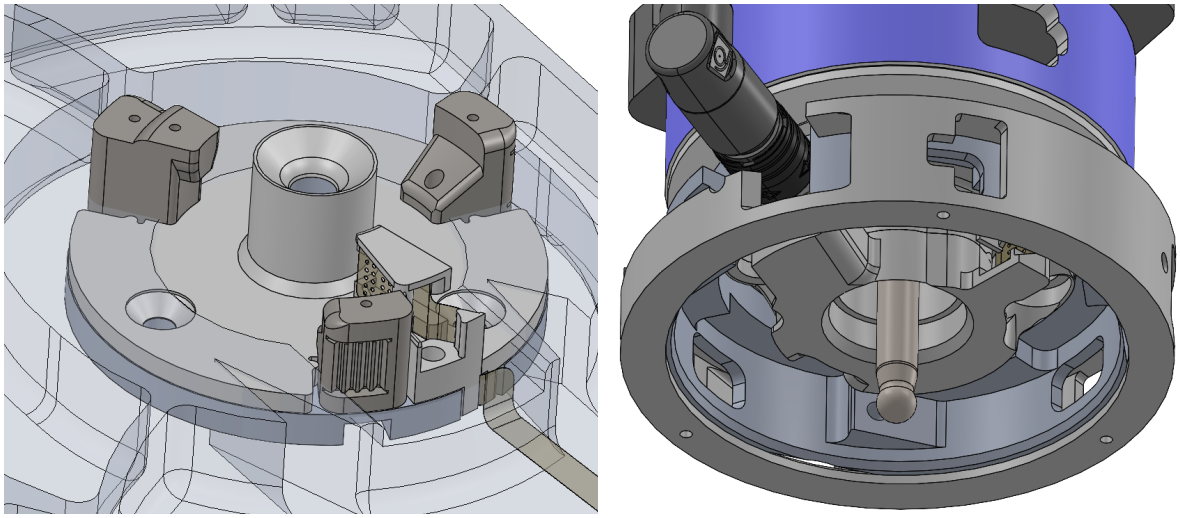
5.1.2 EEC Variations

Our design comes in 3 variations: (i) a standard version for lower load limbs as the hands of a humanoid robot (Fig. 5.1), (ii) a larger version to bear greater loads as the ankles of a humanoid robot (Fig. 5.3), and, (iii) a final one allowing the passing through of a USB type-C cable which allows for more information passing through like a camera output with limited loads (Fig. 5.4).

Foot variation The feet of a humanoid robot have a higher payload as their main purpose is locomotion in various environments. This first variant has to support the robot's weight and transmit the ankle torques. Yet the payload acts rather in compression and torque stresses and shall be strengthened accordingly. Thus, it has a bigger diameter and is slightly thicker. The diameter of the foot EEC is 64.4 mm (outer diameter of the tool side), while the diameter of the hand EEC is 50.4 mm. Also, because the foot F/T sensor has no hole, the pin is on the robot side and the funnel is on the tool side. The design of the foot or the foot interface has to account for the pin sticking out of that side.

Head Variation The head of a humanoid robot usually carries sensors that require large-bandwidth data transmission (e.g., cameras, LiDARs, etc.). It can also be used for social interactions and could be a sophisticated mechanism embedding several actuators, sensors, or displays. Thus, for head changes, several specificities are considered. The pin is removed and replaced by a pass-through that has a USB type-C connector through it. Until recently, many different connectors existed depending on such devices' brands. This year, the European Parliament (and highly likely other international regulation bodies will follow) imposed USB-C as the standard for all electronic devices to meet sustainability and ecological trends

1. https://fischerconnectors.com/en/circular-connectors/minimax-series/
2. https://www.mill-max.com/products/spring-loaded/pogo-pin-header-strip/855/10-002101/855-22-012-10-002101



(a) *tool side* **(b)** *robot side*

FIGURE 5.3 – *Foot end-effector changer*

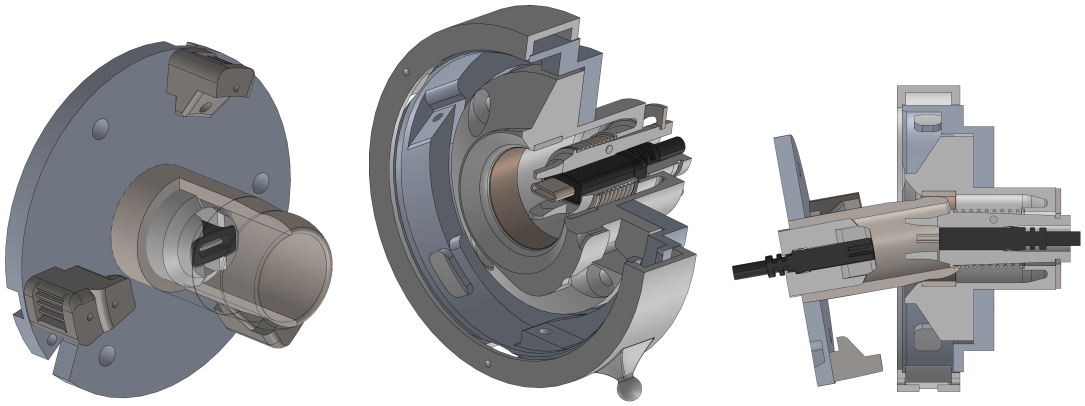


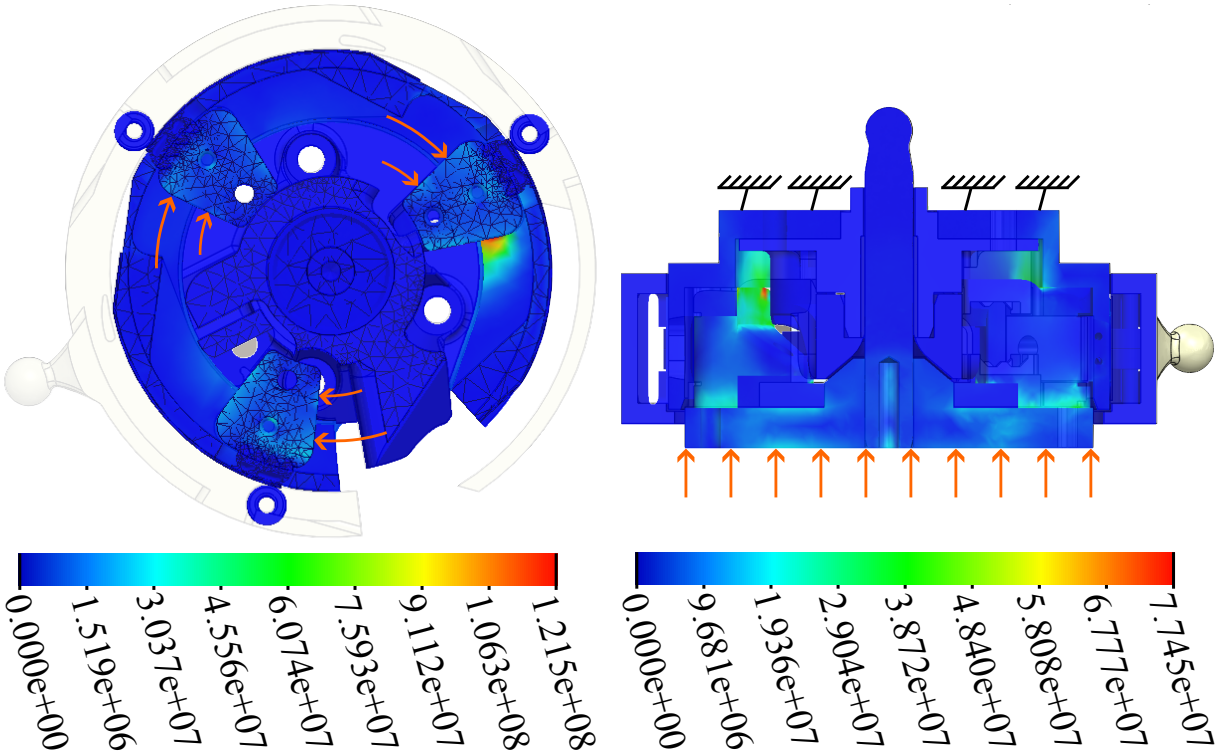
FIGURE 5.4 – *CAD of the head end-effector changer with USB-C pass-through. From left to right: tool side, robot side and engagement step. On the robot side, the USB-C cable can rotate with respect to the EEC as during the last rotation of the engagement, the cable is already plugged-in.*

by reducing waste. Thus, we integrate this standard into our head EEC design, shown in Fig. 5.4. The resulting EEC takes up more space on the robot side, as the required positioning tolerance was deemed more important than the reduced payload for this type of end-effector.

5.1.3 Stress evaluation

To evaluate the design of the EEC, we first realize a Finite Element Analysis (FEA) with target loads corresponding to the 2 times the limits of the force sensor on which the TC will be mounted. We studied the following cases for both the hand and foot end-effector changes (we do not expect the head to have a high payload and substantial external forces applied to it):

- Traction/Pulling along the z-axis is transmitted through the contact between surfaces I and J;
- Compression/Pushing along the z-axis is spread across the three E and F surface



(a) *Von Mises constrains (N/m^2), slice view hand TC Z+ torsion.*

(b) *Von Mises constrains (N/m^2) Slice view hand TC compression.*

FIGURE 5.5 – *FEA analysis results: in both case the robot side is considered fixed (black ground symbol) while the load is applied on the tool side (orange arrows).*

TABLE 5.1 – *Result from FEA on hand/head EEC.*

Load case	Target load	Material	Max load
Traction Z	1160 N	Polyamide	80 N
		Aluminium	550 N
Compression Z	1160 N	Polyamide	650 N
		Aluminium	$>$1160 N
Torsion Z+	20 N.m	Polyamide	11.2 N.m
		Aluminium	$>$20 N.m
Torsion Z-	20 N.m	Polyamide	$\approx$2 N.m
		Aluminium	$>$20 N.m

contacts;

- Torque along the z-axis is transmitted, depending on the direction, by the teeth in the feet (Z-) or the contact between surfaces H and G (Z+);
- Forces and torques in the plane normal to the z-axis are transmitted through the pin.

As the effort transmission is complex and depends on the load, we have to identify which part is the most solicited for each loading. Results are given in Table 5.1 for the hands and in Table 5.2 for the feet.

The initial results assumed the parts were made of 3D printed plastic material (polyamide

TABLE 5.2 – *Result from FEA for foot EEC.*

Load case	Target load	Material	Max load
Traction Z	800 N	Polyamide	100 N
		Aluminum	100 N
Compression Z	800 N	Polyamide	$>$800 N
		Aluminum	$>$800 N
Torsion Z+	24 N.m	Polyamide	10 N.m
		Aluminum	$>$24 N.m
Torsion Z-	24 N.m	Polyamide	5 N.m
		Aluminum	$>$24 N.m

SLS), a convenient solution for quick manufacturing. However, according to the FEA, this material is not strong enough for the EEC to support the target load of twice the force torque sensor limits. FEA was also conducted with aluminum as the main material for the EEC as a solution for an improved payload without increasing the weight too much. Considering aluminum greatly improved the results, yet not all the targets were reached. In particular, the target of traction on the Z axis was not reached. The efforts are mostly transmitted through surfaces I and J, which are too small, resulting in a high-stress concentration. Thus, to improve the bearing of the TC in the pulling/traction case, the design has to be modified.

In the meantime, the simulations are also limited as the system has a complex geometry, making it difficult to understand how it reacts to torsion around the X and Y axes.

Overall, these limits are still above the usual loads on our humanoid platform capabilities. Furthermore, for the feet changer, the case of traction seldom appears. Thus, we manufactured and tested prototypes for the hand and foot variation of this design to evaluate its performance relative to the other design constraints.

5.2 Evaluation and experiments on a humanoid HRP-4

Our proposed end-effector changer design is integrated to our humanoid as described in Sec. 5.2.1. Sections 5.2.2, 5.2.3 and 5.2.4 detail the implementation of the EEC to change hand, foot and, head, respectively.

5.2.1 Mechatronics integration and mounting

As shown in Fig. 5.6, our end-effector changer design was prototyped using 3D printing. Most parts were 3D-printed in polyamide using the selective laser sintering (SLS) method (dark grey). The feet on the tool side are also 3D-printed in steel (shiny grey). As the pin has a simple shape, it was machined. Only the hands and feet versions of the EEC were prepared for these tests. The hand EEC was also used to showcase a simple example of head change. Figure 5.7 illustrates how the EEC is mounted on the robot. For both the hands and the feet, the EEC's robot side was designed considering the F/T sensor interface. As such,

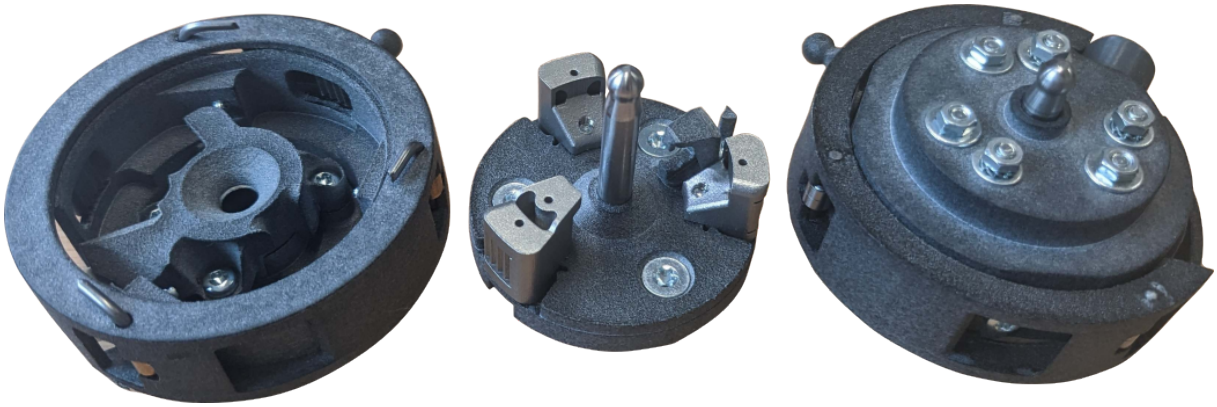


FIGURE 5.6 – *Prototype of the end-effector changer. From left to right: robot side, end-effector/tool side, mounted and locked tool-changer (both robot and tool side).*

the robot sides are directly screwed to the F/T sensors, thus saving space and weight an interface part would have required. For the head, a simple interface part was devised. On the end-effector/tool side, an interface part is added in the same way as presented for the Kosmek TC in Chap. 4. Details are given about this part and the design of the corresponding tool stations in Secs. 5.2.2, 5.2.3 and 5.2.4.

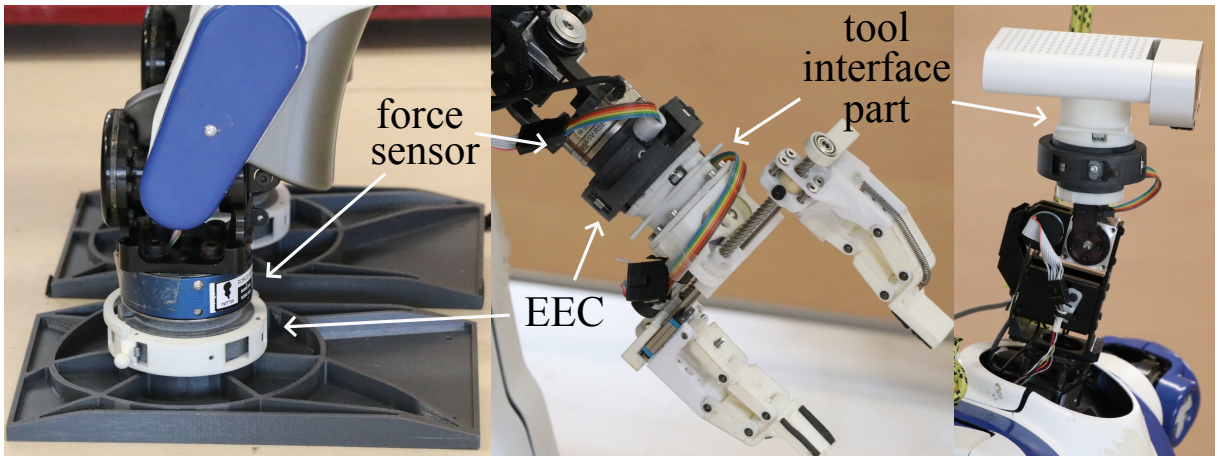


FIGURE 5.7 – *Prototype EEC mounted on humanoid HRP-4 (from left to right: feet, hand, head)*

The electrical and data connections are soldered directly to both sides of the changer mechanism. In our case, the used gripper has one actuator that uses 6 cable connections (2 for the power supply and 4 for the encoder data). The design can offer up to 24 wires using an internal flexible PCB.

The behavior implemented for the change of end-effector is similar to the ones presented in Chap. 4. Again, the robot is assumed to be close enough to the tool station to interact with it. The main difference lies in the motions to unlock and lock the mechanism. Also, the order of the operation is slightly different as it has to be operated outside the tool station. First, the current tool must be unlocked, stored in the tool station, and disengaged. Then, a new tool can be attached, released from the tool station, and locked.

The following sections detail some specificities for each end-effector.

5.2.2 Arms' end-effectors changing

Arm change (or hand change) is similar to the solution proposed in Chap. 4. The first difference comes from the interfaces between the robot, the EEC, and the end-effector. By design, the EEC is mounted directly on the F/T sensor at the terminal point of the arm, and the interface between the EEC and the end-effector is a simplified version of the one introduced for the Kosmek TC in Chap. 4. These interfaces do not include makers for vision or the surface previously used to make contact when storing the end-effector, as they would increase the collision shape of the wrist and decrease its payload. Thus, some initial steps when changing the end-effector require more tuning. The proposed interfaces are not definitive and should be adapted to the preference of the user, the application, and a different end-effector docking station. The most important feature is allowing for positioning tolerances when storing the end-effectors inside the docking station of the same order as those required to engage the EEC.

We consider 2 different effectors, one with a motor and one without:

- a simple un-actuated end-effector providing flat contact that the robot can use to balance (Fig. 5.8 and Sec. 5.2.3);
- a gripper with one actuator and a complex shape that was used in a previous project Kheddar et al. (2019).

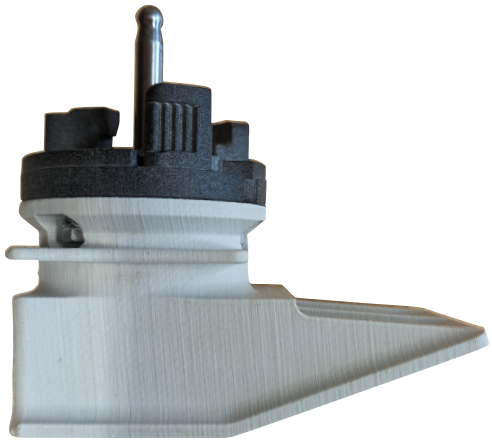


FIGURE 5.8 – *Arm end-effector use to support the robot with the tool side of the EEC*

Using such different end-effectors allows us to illustrate the electronic connection and how different end-effectors are usable. For the support end-effector, the interface is directly integrated into the design (Fig. 5.8). For the gripper, it is a simple variation of the Kosmek interface. The different end-effectors are mounted on the robot, and their inertial parameters are identified [3].

Then, the locking and unlocking of the EEC introduces some extra motions. Due to the kinematics and active self-collisions of the robot, it is not possible to execute these motions while the end-effector is stored in a station. Therefore, the design makes it possible to remove the end-effector from the docking station while unlocked. The robot then locks the EEC using the other arm (Fig. 5.9).

3. https://github.com/gergondet/mc_force_sensor_calibration_controller



FIGURE 5.9 – *Robot locking the EEC after releasing the gripper from the tool station*

5.2.3 Feet changing

The proposed foot EEC design allowed for better foot change behaviors comparatively to the automated Kosmek TC presented in Chap.4. The Kosmek TC was not strong enough to carry the robot's weight, and the initial implementation was limited. In the preliminary tests, the foot was not stored in the right place for it to be retrieved later. Given the design of the foot with a large funnel shape (Fig. 4.9) that allowed for large positioning tolerances (up to 15 mm), probing the environment was implemented in a simple form. The results, while demonstrating the possibility of using TC for the feet of the robot, were lacking in robustness.

Contrarily, with the proposed foot EEC designed to better handle the payload for the feet, we implemented much more robust foot change behaviors. As explained before, the changing of a foot introduces two challenges compared to the upper body end-effectors: (i) keeping the balance and (ii) engaging and disengaging the feet blindly because of the occlusion of the foot by the robot's body. These two issues are respectively discussed in the following two sections.

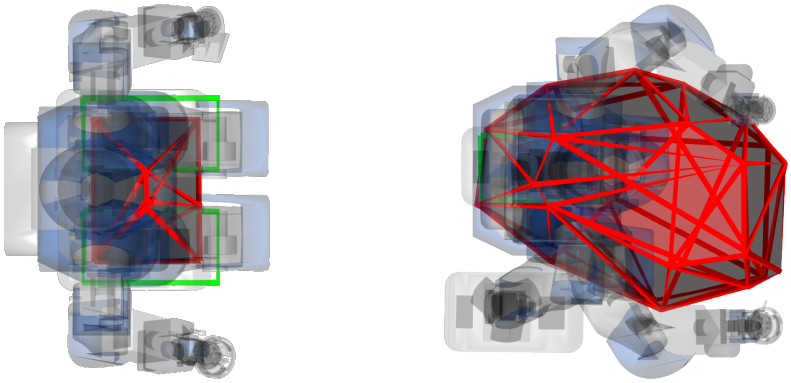


FIGURE 5.10 – *Balance region of the robot standing on two feet (left) and standing one foot and using its arms (right). The green rectangles correspond to the feet contacts and the red and black volume is the CoM balance region accounting for CoM acceleration up to* 0.5 $m.s^{-2}$ *in all horizontal directions.*

Balancing the robot while changing a foot

During the motion, the balance is kept using the stabilizer presented in Murooka et al. (2021). This stabilizer is based on a linear inverted pendulum model. It accounts for external wrenches from supplementary contacts by adding offsets derived from the applied external wrenches to the CoM and ZMP. When the robot stands on one foot while changing the other one, the force applied by the changing foot is considered an external wrench. When possible, to reduce the torques at the ankles level, we can use additional arms' contacts (here on the tables). This also enlarges the CoM balance region of the robot as in Roux et al. (2021), see Fig. 5.10. The wrenches applied by the hands are limited due to a lesser payload for these limbs compared to the feet.

Design of the tool station

Most of the time, parts of the robot body occludes the motion of the feet while changing them. Thus, changing the feet does not rely on vision. We use the foot force sensors to locate precisely the position of the foot station.

The station is designed to be simple to build and use. However, its design can differ depending on the application, and some improvements are proposed at the end of this section. The detached feet are stored close together so that the robot can reach both at a time. Indeed, once the robot has disengaged a foot, it cannot move, contrarily to arms. Also, their relative position is known precisely so that the robot can move its leg from one to the other with low error in positioning.

Figure 5.11 shows the approach we adopted for the foot station for the right foot. Initially, the new foot to be equipped is stored on one side (here right) of the foot station while the robot deposits its equipped one on the other (here left) side. With its ankle force torque sensor, the robot makes contact with the aluminum profile at the front. Then, the foot slides along the station guide until contacting its front and then right sides (Motion from Ⓐ to Ⓑ). The relative pose between the station and the robot is estimated using the robot's kinematic chain. The following motions account for the estimated pose of the station, in particular for its orientation, as the robot may not initially be aligned with it. Then, the foot slides back

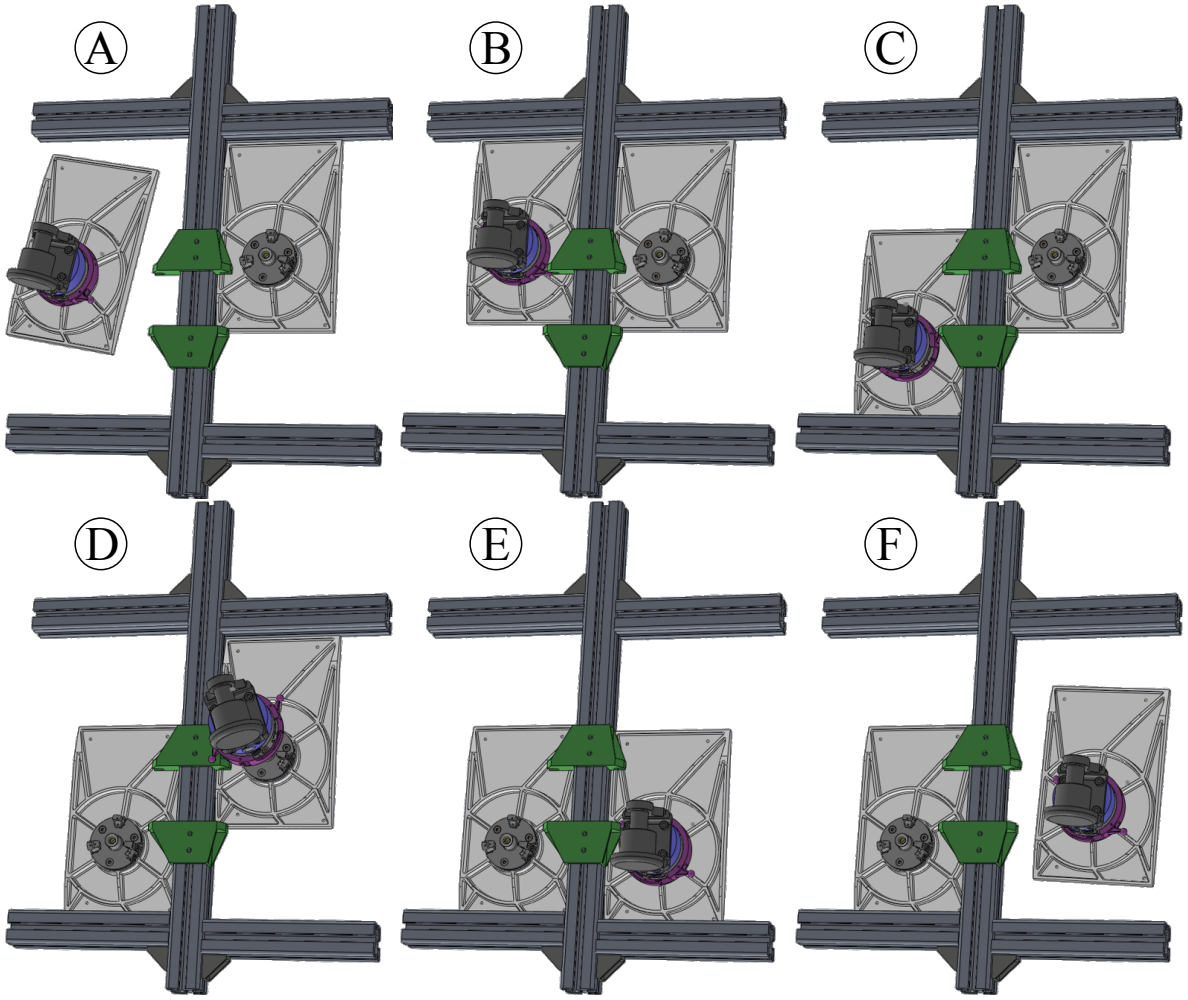


FIGURE 5.11 – *Top view of the (right) foot docking/storage station (dark gray aluminium profile and the 3D printed green part). The light grey rectangles are the feet of the robot. The one on the right is stored at the foot station while the left one is still attached to the robot (black part is the ankle link of the robot and purple part is the locking bell of the EEC).*

against the station until it touches its back. During this motion, the green parts of the station make contact with the little sphere on the bell of the EEC and unlock it. (Motion from Ⓑ to Ⓒ). Once the EEC is unlocked, the robot disengages the current foot by twisting its ankle to the left (Rotation around Z+) until contact is detected in the EEC. Then, it lifts its ankle without the foot. The robot then moves its ankle above the new foot (Motion from Ⓒ to Ⓓ). To engage its new foot, the robot lowers its leg and turns it to the left. Then, similarly to how the EEC has been unlocked, it is locked again by sliding the foot backward. (Motion form Ⓓ to Ⓔ). Finally, the robot can move its foot away.

During these motions, keeping contact is ensured using different target wrenches enforced using admittance tasks in the control of the robot's leg. These wrenches ensure non-zero normal forces along the sliding surface to have a precise motion Samadi et al. (2021). Motions are generated by a target body velocity along the desired sliding directions. Each motion ends when a predefined threshold contact force is detected. For the rotation when engaging and disengaging the EEC, the motion stops when a predefined threshold torque is detected. Fig. 5.12 shows the target and measured contact forces at ankle level for the right foot during foot change.

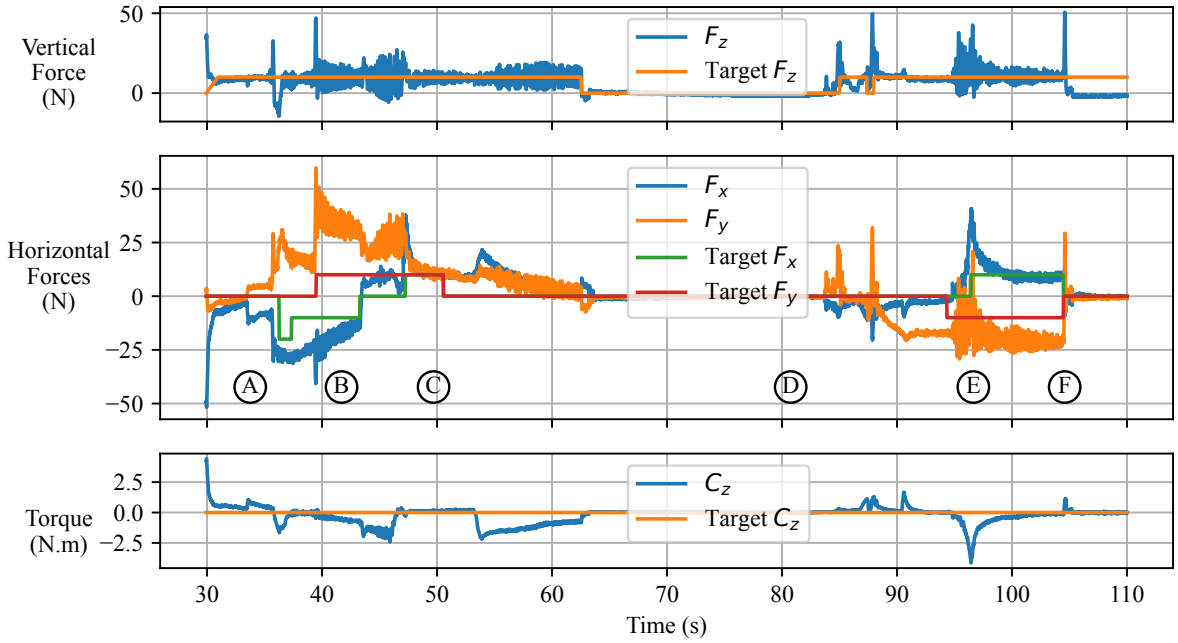


FIGURE 5.12 – *Tracking of the forces and torque from the ground of the robot at the sole of the right foot during the motions showed in Figure 5.11. Form top to bottom there is (i) the normal vertical force applied by the foot on the floor (along the Z axis) (ii) The horizontal forces applied by the foot, the X-axis being forward and Y-axis from right to left when looking from the robot point of view, (iii) the torque around the Z-axis. The legend hides the part of the graphs were the foot is in the air without tool and moving from one tool to the other and thus all measures and targets are at 0. The letters correspond to Figure 5.11 and are positionned horizontally at the corresponding time*

On the z-axis a constant force is applied on the ground to keep the contact. It is still low enough to have the foot slide on the ground without issues. There are two main reasons why the force target for the x-axis and y-axis in the horizontal plane can be 0 N. First, a target speed is given on these axes. Thus, there is no point in controlling the contact force on these axes using an admittance task. In that case, the measured force corresponds to the friction of the foot on the ground. For example this happens (Fig. 5.12):

- Between 33 s and 36 s, the foot slides diagonally toward the tool station until the contact is made;
- Between 43 s and 47 s, the foot slides backward along the x-axis to unlock the TC until the back contact is detected (high jump on the x-axis force at 47 s). The measured non-zero force on the x-axis corresponds to the friction of the foot against the ground and the tool station.

The second reason is that the force is monitored to 0 N in that direction. It ensures a smooth motion with reduced friction force in the other direction, for instance, when engaging or disengaging the EEC:

- From 50 s to 52 s, the ankle rotates for the first part of disengaging the EEC;

- From 52 s to 62 s, the horizontal forces are monitored to 0 N so the ankle can be lifted without the foot moving. It ensures the foot will stay in the proper position while stored for the next foot change.

In the horizontal plane, a non-zero target is to ensure continuous unbroken contact during the motion:

- From 36 s to 37 s, the increased target on the x-axis ensures the foot is pressed against the station by increasing the torque around the z-axis;
- From 37 s to 40 s, the constant target on the x-axis keeps the foot against the station while it slides sideways until the side contact is made;
- From 40 s to 43 s, all the contact forces are maintained while the position of the tool station is estimated w.r.t. the robot;
- From 43 s to 47 s, the force on the y-axis keeps the foot against the tool station while it is sliding.

The torques in the horizontal plane are monitored to 0 N.m to keep the foot flat against the ground. Thus, the CoP of the foot is below the force sensor frame. The torque around the z-axis is also monitored to 0 N.m most of the time to keep the foot aligned with the foot station. But, there are some exceptions:

- From 50 s to 54 s, when the EEC is disengaging;
- From 85 s to 87 s, when looking for the correct orientation to engage the EEC;
- From 88 s to 91 s, when the EEC finishes its engagement.

An angular velocity target is given to the ankle frame around the z-axis. In the first and third cases, the motion stops when the measured torque on the z-axis reaches a predefined threshold. When searching for the EEC's correct orientation, the motion stops when detecting a drop in the vertical force (Fig 5.1 tab C enters slot D).

Another difficulty during this motion lies in the kinematics of the robot: when reaching position D, some joints of the robot as the right knee are close to their limits. Control of the contact forces is then much more difficult (Fig.5.12 from 95 s to 105 s). To mitigate this issue, during the final experiment, the length needed for the new foot to move was shortened (Right side of the tool station in Fig. 5.13). The robot was set as close as possible to the foot station which in turn led to the foot not having enough space to be put down after foot change (See experiment video).

In contrast to hand change, there is no extra part on the foot to store the foot in the station. Thus, the design of the foot station must account for different foot sizes. Initially, there is only one size possible. The design of the EEC's bell accounts for the assembly of the EEC. Thus, the locked and unlocked position of the sphere, used to interact with the EEC, is not symmetrical w.r.t the yz-plane of the F/T sensor. Some additional efforts in the design of the tool station

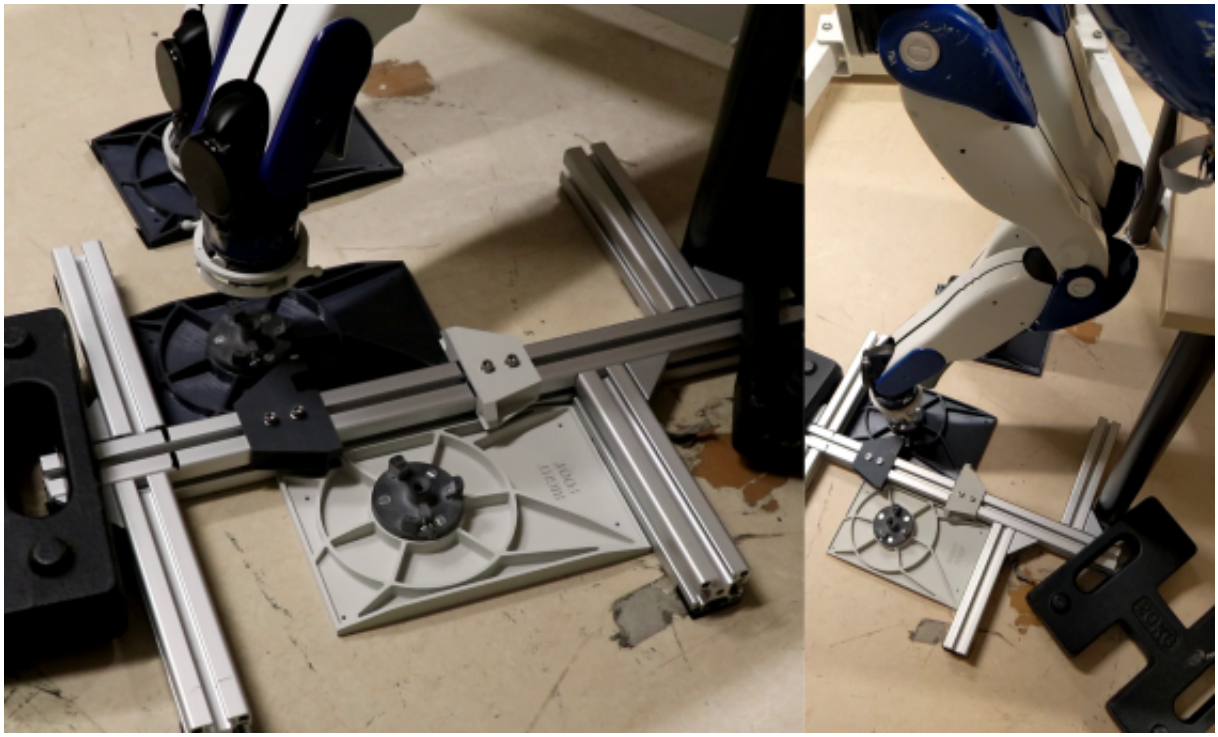
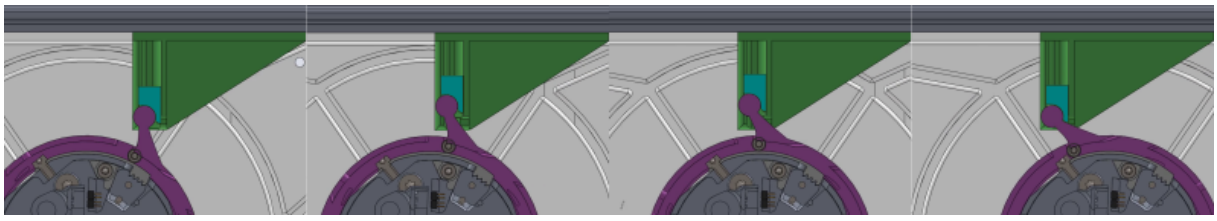
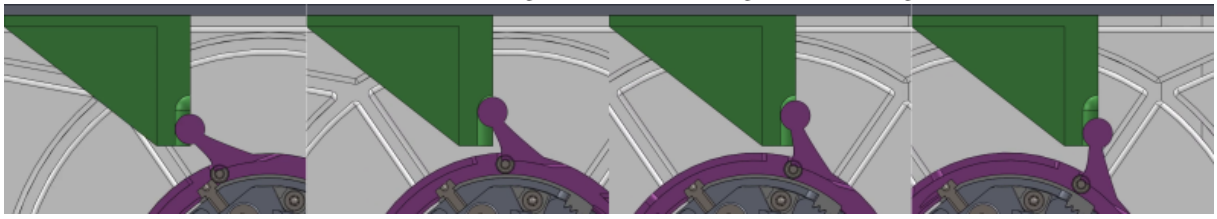


FIGURE 5.13 – *Foot change from the grey one to the white one*



(a) *Slice of the foot unlocking mechanism during the unlocking motion*



(b) *Slice of the foot locking mechanism during the locking motion*

FIGURE 5.14 – *Locking and unlocking of the foot EEC*

part that unlocks the EEC were required. This part was designed with a sliding cart to avoid collision (Fig. 5.14a). In contrast, the part that locks the EEC is much simpler (Fig. 5.14b).

To allow for a foot station to receive a variety of foot heights, the first step is to change the locking part with a flat surface (Fig. 5.15). Then, the next height limiting factor is the distance between the locking sphere and the highest part of the foot. In the design of the EEC's bell, moving the sphere up would increase this distance. The sphere would then also be placed at a position that lets the locking and unlocking of the EEC be symmetric without preventing its assembly. Consequently, the tool station part that unlocks the TC would be the same as the locking one. These improvements would allow for foot heights varying up to 20 mm. This leaves potential improvement for the foot-changing mechanism, especially

if the user needs a wider variety of feet.

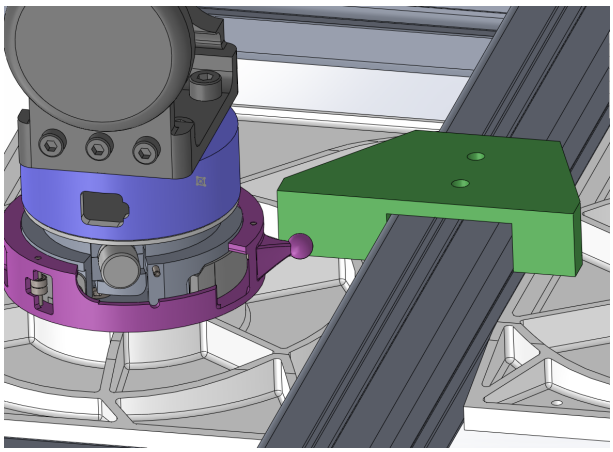


FIGURE 5.15 – *Modification of the locking mechanism to have larger variation of feet.*

5.2.4 Head changing

Head changes present problems different from hand or foot changes. First, because changing the head implies changing a sensor essential for positioning, the robot would be 'blind' during the operation. A solution is to use an external camera mounted on the environment close to the head tool station or another part of the robot, such as its torso. Such a camera would then communicate with the controller (WiFi) to servo the humanoid to the appropriate replacement head location.

Second, Because there is usually no force-sensing device between the neck and torso, head EEC contact control uses force sensing available at the other end-effectors. For example, those available at the feet, or when kinematically possible, use the arms to perform or assist the head change.

To make a satisfactory head EEC integration, substantial mechanical engineering efforts are needed (some changes would be irreversible). Therefore, we demonstrate the head-changing process, considering the head in its simplest form: a camera. The head EEC prototype used is the same as the one for the hands. The tool station design is similar in shape to the wrist (hand EEC) tool station with the added geometry to actuate the EEC in the same fashion as foot change (Fig. 5.16). During the changing motions, the locking (resp. unlocking) happens during the releasing (resp. storing) of the end-effector. The head-changing motions are generated by the same task-space controller operating the humanoid, specifying needed translations and rotation at the neck level, inducing whole-body motions (see attached multimedia and Fig. 5.17).

5.3 Discussion

In a manufacturing industrial context, using EEC is to be compared (in terms of cost, size, flexibility, weight, etc.) to a sophisticated fixed gripper with a dexterity comparable to

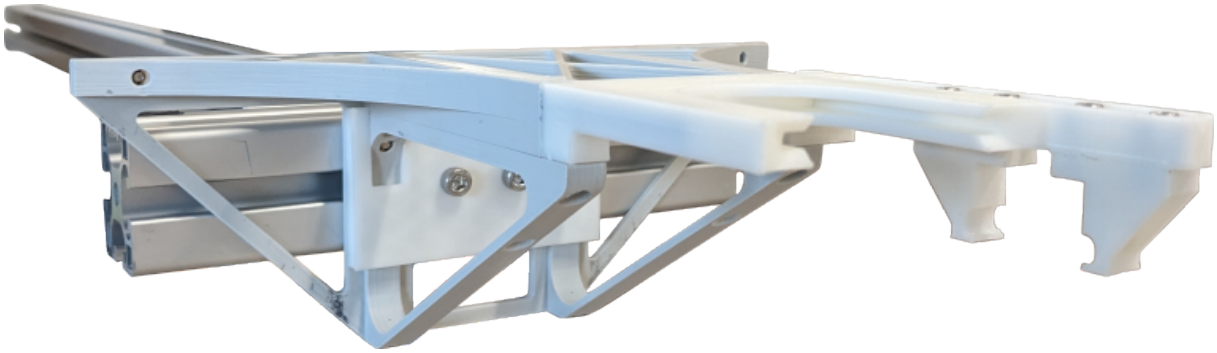


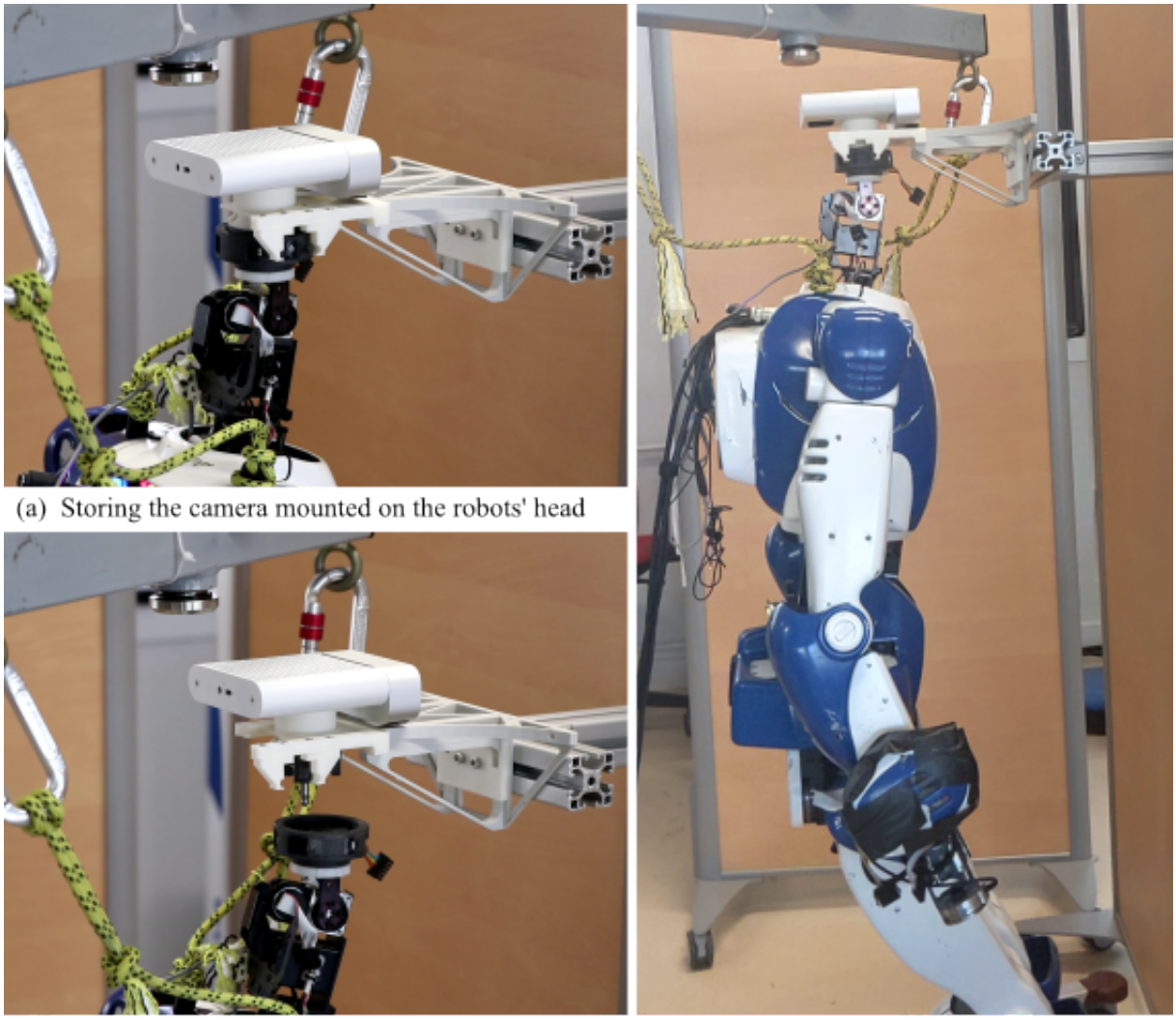
FIGURE 5.16 – *Head station*

human hands. As shown by our demonstrations, EECs offer direct control of the end-effector from the robot's software without an extra interface through the electric connection. Robust and repetitive handling of any attached end-effectors is guaranteed by the direct mechanical link to the wrist. Furthermore, it also ensures the precise use of the end-effectors as their relative position to the limb is fixed and known. Finally, the use of an EEC does not exclude the use of a complex dexterous gripper. Relative to modular robotics connectors Saab et al. (2019), our design is gendered, asymmetric, with no inner actuation, and offers a relatively larger tolerance to misalignment. However, as it has been designed for humanoid robots that generally have kinematics redundancy and various positioning possibilities (floating base), it will be complex to use this EEC as a docking mechanism in modular robotics. In the literature Berenstein et al. (2018); Pettinger et al. (2019); Wei et al. (2021), tool-changers are evaluated by repetition of the tool changing motion with various initial positions and measuring the number of successes and failures. To show the robustness of our design, we also repeated the motion of the EEC but introduced variability by alternating the changed end-effector. During the demonstrations, no particular issues arose for the hands as our robotic platform is limited in the efforts that can be transmitted through the arms.

In the previous chapter, the design constraints were outlined. Regarding the impact of the EEC on the robot's payload, the solution is flatter than the Kosmek solution. This is partly due to the lack of an interface between the EEC and the robot F/T sensor. 3D printing most parts of the initial prototypes reduced the weight by a wide margin compared to other solutions (See Table. 4.1). However, regarding the payload and stress resistance, the FEA showed that the design is quite limited. It was enough for initial proof of concept. For the feet, one of the parts broke (as expected in the simulation). After fixing it, we conducted experiments with limited ankle torques by imposing extracontacts. A major design flaw results from the size of the surfaces holding together the two halves of the EEC when pulling on it.

Thanks to the simple mechanical locking mechanism, the EEC can be operated by the robot in autonomy, by a human, and by interacting with the tool station. Some changes to the bell would still benefit the design and the interaction with the different tool stations. We are also devising a system with an inner locking mechanism for comparison.

The pin and funnel approach allows for wide positioning error tolerances that were decisive in the success of the demonstrations. However, they should be evaluated more precisely. Then, we designed several versions of the EEC for different end-effectors to fulfill the needs of the different effectors. The electric and data connection is demonstrated by switching from a simple support end-effector to a gripper with a motor and an encoder for



(a) Storing the camera mounted on the robots' head

(b) Releasing the EEC with Azure Kinect camera stored in head station

(c) Full body view of head change

FIGURE 5.17 – *Head change on HRP-4 using the hand EEC*

the arms' terminal point.

On top of electric and data connections, fluid connections would be a plus. As for the proposed head EEC design, a fluid connection could pass through the center of the EEC without needing major revision of the concept. As illustrated by the head changer design (Fig. 7), the center of the EEC would also serve the purpose of conducting fluid or air pressure in applications that require it. Instead of a USB-C connection, a push-to-connect fluid coupling with automatic shut-off valves could convey fluids without leakage through the EEC. In this case, the shaft is hollowed to integrate commercially available fluid/air connectors; this is part of our future work for applications beyond humanoids (e.g., modular robotics).

Our demonstrations show that in some postures, our humanoid robot is limited in motion range (especially the feet), yet motion ranges are advised in Kheddar et al. (2019); Zhu et al. (2023) as a must-consider issue, for other purposes, in future humanoid robots design. A short-term solution would be to consider improving the tool station. For instance, tilting it could make all the positions to change foot reachable with minor changes to the foot change behavior.

Future developments can be considered along three main axes. First, the state of engagement of the EEC and whether the tool has been stored properly in the tool station is not detected. Although the EEC design allows for large positioning errors when engaging an end-effector, no solution currently exists to estimate whether each step was successful. Even though progress in visual servoing offers better and better solutions, a proper way to detect failure relying on another modality is necessary to ensure robust end-effector change. Indeed, precisely estimating the failure modes would benefit deciding which recovery measure is the best in the robot's control. The topic of contact state estimation has been developed in Jasim and Plapper (2014) and offers an approach to detect contact state only relying on the end effector pose and measured contact wrenches. It could be done electronically. For now, the system could be unlocked by accident without the robot being aware, a digital sensor could solve this issue. However, such a sensor would only give binary information on the success or failure of the full motion.

Secondly, more efforts in the development of end-effector stations would allow for more complex behaviors. A simple design for the locking mechanism allows for a lot of freedom in the design of the tool stations. They can be simple and affordable as proposed in this work. It highly decreases the integration efforts for new scenarios. One can also consider more complex storing systems that communicate with the robot and provide a wider range of tools. Or, to the contrary, we could consider having the end-effectors disposed of anywhere in the environment without any tool station needed.

Finally, end-effector changers can be integrated as "changers" in the humanoid: we can imagine the humanoid changing larger parts of its body, such as a limb. The first benefit is the maintenance of the robot and the reduced downtime when parts of the robot break. It would further increase the modularity of humanoids as with modular robots, where the robot can even change its kinematic architecture.

5.4 Conclusion

Following Chap. 4 where we tested and outlined the limits of tool-changers design for robots, we devised a novel end-effector changer for humanoid robots. Its passive locking mechanism showed its benefits for humanoid robots: it is lightweight, viable for different end-effectors, and human-friendly. Although the initial version, lacks in terms of payload the different variants show that the design can easily adapted to fit diverse situations. Also, the relevance of the end-effector changing to a humanoid robot has been proven again through a wider set of experiments than Chap. 4 by demonstrating the electric connection between the robot and the as well as head change. Finally, further research for this application have been discuss to make end-effector changer a standard in commercial humanoids.

Conclusion

In this thesis, we approached the development of humanoids in two aspects. First, we discussed the multi-contact applications where the robots have to use indiscriminately their limbs for both support and executing tasks. Then, we proposed the use of end-effector changers to expand the modularity of humanoids. Both aspects were treated while keeping in mind an industrial setup.

Considering multi-contact balance, we first improved the computation of the robust CoM balance region to be used as part of the online control of the robot and not just for planning robot motion. In Chap. 2, the computation of the balance region was improved. The new version is fast enough to be used online so that the control of the robot can account for variations of the contact stance. In addition, a CoM admittance task was introduced to improve the tracking of the contact force by generating target CoM velocities. The capabilities of these contributions were illustrated by moving a heavy box relative to the robot's weight. Moreover, the library implemented in Chap. 2 is also useful to project other regions as it is independent of the control framework. Some ongoing works also look into analytical methods of computation to obtain similar results.

Then, the proposed approach in Chap. 3 allows for an even faster approach as the optimal wrench distribution and CoM position are computed at every iteration of the controller. They also allow for robust quasi-static motions by maximizing the margin of the contact wrenches w.r.t. their friction cones. We demonstrated simultaneous sliding on non-coplanar surfaces of the robot's end-effectors.

However, both approaches cannot fully account for highly dynamic motions. As a result, in Chap. 4, the results from Murooka et al. (2021) were used instead as they provide better dynamic behavior required for foot change. Indeed, during foot change, the robot has to be standing on one foot which is a challenging situation for the balancing methods proposed in this thesis. Thus, a short-term application of Chap. 2 and 3 would be to adapt their results in end-effector changing. Such an application would allow us to understand fully their shortcoming as the end-effector changes offer all the challenges initially studied.

When considering humanoid robots, end-effector changers had never been integrated into their design before. In Chap. 4, we tested existing solutions and outlined their limits. Taking into consideration previous works from the literature, we established a set of constraints for the conception of EEC for humanoids. Chap. 5 introduces a corresponding new design of EEC better suited for humanoids. Our demonstrations, although in a lab setting, revealed the high potential of embedding future humanoid robots with EEC mechanisms. There are many possible improvements in the mechanism, the engagement and disengagement processes, and the software. First, in the short term, experiments to quantify the tolerances to

position would be beneficial to allow improving the design. Then, as the EEC showed it lacks in terms of strength for end-effectors regularly under heavy loads, a re-design of the EEC to improve its resistance is required. Finally, to further show their benefits, experimenting with different tool stations or no tool station at all would illustrate the versatility of our design.

BIBLIOGRAPHIE

Abi-Farraj, F., Henze, B., Ott, C., Giordano, P. R., and Roa, M. A. (2019). Torque-Based Balancing for a Humanoid Robot Performing High-Force Interaction Tasks. *IEEE Robotics and Automation Letters*, 4(2):2023–2030. 8, 31, 53

Agravante, D. J., Cherubini, A., Sherikov, A., Wieber, P.-B., and Kheddar, A. (2019). Human-humanoid collaborative carrying. *IEEE Transactions on Robotics*, 35(4):833–846. 6

Agravante, D. J., Claudio, G., Spindler, F., and Chaumette, F. (2017). Visual servoing in an optimization framework for the whole-body control of humanoid robots. *IEEE Robotics and Automation Letters*, 2(2):608–615. 17

Audren, H. and Kheddar, A. (2017). Model-predictive control in multi-contact based on stability polyhedrons. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, pages 631–636. 9, 53

Audren, H. and Kheddar, A. (2018). 3-D Robust Stability Polyhedron in Multicontact. *IEEE Transactions on Robotics*, 34(2):388–403. 9, 18, 21, 24, 25, 26, 27, 28, 29, 36

Audren, H., Kheddar, A., and Gergondet, P. (2016). Stability polygons reshaping and morphing for smooth multi-contact transitions and force control of humanoid robots. In *IEEE-RAS International Conference on Humanoid Robots*, pages 1037–1044. 9, 31

Ayusawa, K., Venture, G., and Nakamura, Y. (2008). Identification of humanoid robots dynamics using floating-base motion dynamics. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2854–2859. 18

Berenstein, R., Wallach, A., Moudio, P. E., Cuellar, P., and Goldberg, K. (2018). An open-access passive modular tool changing system for mobile manipulation robots. In *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)*, pages 592–598. IEEE. III, 11, 14, 57, 60, 73, 91

Bouyarmane, K., Caron, S., Escande, A., and Kheddar, A. (2019a). *Multi-contact Motion Planning and Control*, pages 1763–1804. Springer Netherlands. 16

Bouyarmane, K., Chappellet, K., Vaillant, J., and Kheddar, A. (2019b). Quadratic Programming for Multirobot and Task-Space Force Control. *IEEE Transactions on Robotics*, 35(1):64–77. 14, 15, 16, 31

Bouyarmane, K. and Kheddar, A. (2011). Using a multi-objective controller to synthesize simulated humanoid robot motion with changing contact configurations. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4414–4419. 14

Boyd, S. and Vandenberghe, L. (2004). *Convex optimization*. Cambridge university press. 44, 45

Bretl, T. and Lall, S. (2008). Testing static equilibrium for legged robots. *IEEE Transactions on Robotics*, 24(4):794–807. 9, 21, 23, 25, 26, 28, 50

Brooks, R. A., Breazeal, C., Marjanović, M., Scassellati, B., and Williamson, M. M. (1998). The cog project: Building a humanoid robot. In *International workshop on computation for metaphors, analogy, and agents*, pages 52–87. Springer. 5

Büeler, B., Enge, A., and Fukuda, K. (2000). *Exact Volume Computation for Polytopes: A Practical Study*, pages 131–154. Birkhäuser. 27

Caron, S. and Kheddar, A. (2016). Multi-contact walking pattern generation based on model preview control of 3d com accelerations. In *IEEE-RAS International Conference on Humanoid Robots*, pages 550–557. 9, 23, 26

Caron, S., Kheddar, A., and Tempier, O. (2019). Stair climbing stabilization of the HRP-4 humanoid robot using whole-body admittance control. In *IEEE International Conference on Robotics and Automation*, pages 277–283. 6, 7, 31

Caron, S., Pham, Q.-C., and Nakamura, Y. (2015a). Leveraging cone double description for multi-contact stability of humanoids with applications to statics and dynamics. In *Robotics: Science and Systems*. 9, 24

Caron, S., Pham, Q.-C., and Nakamura, Y. (2015b). Stability of surface contacts for humanoid robots: Closed-form formulae of the contact wrench cone for rectangular support areas. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5107–5112. IV, 7, 40

Carpentier, J., Budhiraja, R., and Mansard, N. (2017). Learning feasibility constraints for multi-contact locomotion of legged robots. In *Robotics: Science and Systems*. 9

Chappellet, K., Murooka, M., Caron, G., Kanehiro, F., and Kheddar, A. (2023). Humanoid loco-manipulations using combined fast dense 3D tracking and SLAM with wide-angle depth-images. *IEEE Transactions on Automation Science and Engineering*, pages 1–14. 1, 17, 68

Chen, W. H., Wegener, K., and Dietrich, F. (2014). A robot assistant for unscrewing in hybrid human-robot disassembly. In *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*, pages 536–541. IEEE. 10

Clévy, C., Hubert, A., Agnus, J., and Chaillet, N. (2005). A micromanipulation cell including a tool changer. *Journal of Micromechanics and Microengineering*, 15(10):S292. 10

Clévy, C., Hubert, A., and Chaillet, N. (2008). Flexible micro-assembly system equipped with an automated tool changer. *Journal of micro-nano mechatronics*, 4(1):59–72. 10

Collette, C., Micaelli, A., Andriot, C., and Lemerle, P. (2007). Dynamic balance control of humanoids for multiple grasps and non coplanar frictional contacts. In *IEEE-RAS International Conference on Humanoid Robots*, pages 81–88. 7, 40, 43

Collette, C., Micaelli, A., Andriot, C., and Lemerle, P. (2008). Robust balance optimization control of humanoid robots with multiple non coplanar grasps and frictional contacts. In *IEEE International Conference on Robotics and Automation*, pages 3187–3193. 7, 53

Dehio, N., Wang, Y., and Kheddar, A. (2022). Dual-arm box grabbing with impact-aware mpc utilizing soft deformable end-effector pads. *IEEE Robotics and Automation Letters*, 7(2):5647–5654. 37

Delos, V. and Teissandier, D. (2014a). Minkowski sum of HV-polytopes in $\mathbb{R}^n$. *arXiv preprint arXiv:1412.2562*. 54

Delos, V. and Teissandier, D. (2014b). Minkowski sum of polytopes defined by their vertices. *arXiv preprint arXiv:1412.2564*. 54

Dokuyucu, H. İ. and Özmen, N. G. (2023). Achievements and future directions in self-reconfigurable modular robotic systems. *Journal of Field Robotics*, 40(3):701–746. 12

Enebuse, I., Foo, M., Ibrahim, B. S. K. K., Ahmed, H., Supmak, F., and Eyobu, O. S. (2021). A comparative review of hand-eye calibration techniques for vision guided robots. *IEEE Access*, 9:113143–113155. 17

Englsberger, J., Ott, C., and Albu-Schäffer, A. (2015). Three-dimensional bipedal walking control based on divergent component of motion. *IEEE Transactions on Robotics*, 31(2):355–368. 6

Focchi, M., Del Prete, A., Havoutis, I., Featherstone, R., Caldwell, D. G., and Semini, C. (2017). High-slope terrain locomotion for torque-controlled quadruped robots. *Autonomous Robots*, 41(1):259–272. 9

Friedmann, M. and Fleischer, J. (2022). Automated configuration of modular gripper fingers. *Procedia CIRP*, 106:70–75. 9th CIRP Conference on Assembly Technology and Systems. 14

Gyimothy, D. and Toth, A. (2011). Experimental evaluation of a novel automatic service robot tool changer. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 1046–1051. III, 10, 11, 14, 69, 73

Henze, B., Roa, M. A., and Ott, C. (2016). Passivity-based whole-body balancing for torque-controlled humanoid robots in multi-contact scenarios. *The International Journal of Robotics Research*, 35(12):1522–1543. 8

Hu, Z., Wan, W., and Harada, K. (2019). Designing a mechanical tool for robots with two-finger parallel grippers. *IEEE Robotics and Automation Letters*, 4(3):2981–2988. 13

Hyon, S.-H., Hale, J. G., and Cheng, G. (2007). Full-body compliant human–humanoid interaction: Balancing in the presence of unknown external forces. *IEEE Transactions on Robotics*, 23(5):884–898. 8

Iqbal, Z., Pozzi, M., Prattichizzo, D., and Salvietti, G. (2021). Detachable robotic grippers for human-robot collaboration. *Frontiers in Robotics and AI*, 8:644532. 12

Jasim, I. F. and Plapper, P. W. (2014). Contact-state recognition of compliant motion robots using expectation maximization-based gaussian mixtures. In *International Symposium on Robotics*, pages 1–8. 18, 93

Jiang, J., Luo, X., Luo, Q., Qiao, L., and Li, M. (2022). An overview of hand-eye calibration. *The International Journal of Advanced Manufacturing Technology*, 119(1):77–97. 17

Jovic, J., Escande, A., Ayusawa, K., Yoshida, E., Kheddar, A., and Venture, G. (2016). Humanoid and human inertia parameter identification using hierarchical optimization. *IEEE Transactions on Robotics*, 32(3):726–735. 18

Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K., and Hirukawa, H. (2003). Biped walking pattern generation by using preview control of zero-moment point. In *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, volume 2, pages 1620–1626 vol.2. 6, 7

Kajita, S., Matsumoto, O., and Saigo, M. (2001). Real-time 3d walking pattern generation for a biped robot with telescopic legs. In *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*, volume 3, pages 2299–2306 vol.3. 6

Kaneko, K., Kanehiro, F., Morisawa, M., Akachi, K., Miyamori, G., Hayashi, A., and Kanehira, N. (2011). Humanoid robot hrp-4 - humanoid robotics platform with lightweight and slim body. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4400–4407. III, 14, 15

Kheddar, A., Caron, S., Gergondet, P., Comport, A., Tanguy, A., Ott, C., Henze, B., Mesesan, G., Englsberger, J., Roa, M. A., et al. (2019). Humanoid robots in aircraft manufacturing: The airbus use cases. *IEEE Robotics & Automation Magazine*, 26(4):30–45. 1, 17, 83, 92

Koolen, T., Posa, M., and Tedrake, R. (2016). Balance control using center of mass height variation: Limitations imposed by unilateral contact. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 8–15. 6

Kumagai, I., Morisawa, M., Sakaguchi, T., Nakaoka, S., Kaneko, K., Kaminaga, H., Kajita, S., Benallegue, M., Cisneros, R., and Kanehiro, F. (2019). Toward industrialization of humanoid robots: Autonomous plasterboard installation to improve safety and efficiency. *IEEE Robotics & Automation Magazine*, 26(4):20–29. 1

Leboutet, Q., Roux, J., Janot, A., Guadarrama-Olvera, J. R., and Cheng, G. (2021). Inertial parameter identification in robotics: A survey. *Applied Sciences*, 11(9):4303. 18

Lengagne, S., Mathieu, P., Kheddar, A., and Yoshida, E. (2010). Generation of dynamic multi-contact motions: 2d case studies. In *2010 10th IEEE-RAS International Conference on Humanoid Robots*, pages 14–20. 7

Li, J. and Nguyen, Q. (2023). Multi-contact mpc for dynamic loco-manipulation on humanoid robots. In *2023 American Control Conference (ACC)*, pages 1215–1220. IEEE. 8

Li, J., Teeple, C., Wood, R. J., and Cappelleri, D. J. (2022). Modular end-effector system for autonomous robotic maintenance & repair. In *IEEE International Conference on Robotics and Automation*, pages 4510–4516. III, 12, 13, 14, 56, 73

Li, N., Yang, Z., Huang, H., and Zhang, G. (2016). The dynamic simulation of robotic tool changer based on adams and ansys. In *2016 International Conference on Cybernetics, Robotics and Control (CRC)*, pages 13–17. IEEE. 14

Lundberg, K. (2011). Automatic tool changer [25 years ago]. *IEEE Control Systems Magazine*, 31(6):18–18. 10

Mesesan, G., Schuller, R., Englsberger, J., Ott, C., and Albu-Schäffer, A. (2023). Unified motion planner for walking, running, and jumping using the three-dimensional divergent component of motion. *IEEE Transactions on Robotics*, 39(6):4443–4463. 6

Morisawa, M., Cisneros, R., Benallegue, M., Kumagai, I., Escande, A., and Kanehiro, F. (2018). Online 3D CoM Trajectory Generation for Multi-Contact Locomotion Synchronizing Contact. In *IEEE-RAS International Conference on Humanoid Robots*. 6

Mourtzis, D., Angelopoulos, J., Papadokostakis, M., and Panopoulos, N. (2022). Design for 3d printing of a robotic arm tool changer under the framework of industry 5.0. *Procedia CIRP*, 115:178–183. 12

Murooka, M., Chappellet, K., Tanguy, A., Benallegue, M., Kumagai, I., Morisawa, M., Kanehiro, F., and Kheddar, A. (2021). Humanoid loco-manipulations pattern generation and stabilization control. *IEEE Robotics and Automation Letters*, 6(3):5597–5604. 7, 69, 85, 95

Nozawa, S., Kanazawa, M., Kakiuchi, Y., Okada, K., Yoshiike, T., and Inaba, M. (2016). Three-dimensional humanoid motion planning using COM feasible region and its application to ladder climbing tasks. In *IEEE-RAS International Conference on Humanoid Robots*, pages 49–56. 9

Nozawa, S., Murooka, M., Noda, S., Kojima, K., Kojio, Y., Kakiuchi, Y., Okada, K., and Inaba, M. (2017). Unified humanoid manipulation of an object of unknown mass properties and friction based on online constraint estimation. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, pages 249–256. IEEE. 8

Ogawa, Y., Venture, G., and Ott, C. (2014). Dynamic parameters identification of a humanoid robot using joint torque sensors and/or contact forces. In *2014 IEEE-RAS International Conference on Humanoid Robots*, pages 457–462. IEEE. 18

Orin, D. E., Goswami, A., and Lee, S.-H. (2013). Centroidal dynamics of a humanoid robot. *Autonomous robots*, 35:161–176. 17

Orsolino, R., Focchi, M., Caron, S., Raiola, G., Barasuol, V., G. Caldwell, D., and Semini, C. (2020). Feasible region: An actuation-aware extension of the support region. *IEEE Transactions on Robotics*, 36(4):1239–1255. 9, 23, 26

Ozawa, R. and Ishizaki, J. (2011). Passivity-based balance control for a biped robot. In *2011 IEEE International Conference on Robotics and Automation*, pages 550–556. IEEE. 8

Ozawa, R. and Kamioka, T. (2022). Passivity-based multi-contact balancing of humanoid robots based on power-conserved transformations. In *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*, pages 17–23. IEEE. 8

Park, J. and Khatib, O. (2006). Contact consistent control framework for humanoid robots. In *IEEE International Conference on Robotics and Automation*, pages 1963–1969. 35

Pau Muñoz-Benavent, J. Ernesto Solanes, L. G. and Tornero, J. (2019). Robust auto tool change for industrial robots using visual servoing. *International Journal of Systems Science*, 50(2):432–449. 13

Pettinger, A., Dimoush, C., and Pryor, M. (2019). Passive tool changer development for an elastic and compliant manipulator. In *IEEE International Conference on Automation Science and Engineering*, pages 1200–1205. III, 11, 12, 14, 73, 91

Popovic, M. B., Goswami, A., and Herr, H. (2005). Ground reference points in legged locomotion: Definitions, biological trajectories and control implications. *The international journal of robotics research*, 24(12):1013–1032. 6

Pratt, J., Carff, J., Drakunov, S., and Goswami, A. (2006). Capture point: A step toward humanoid push recovery. In *2006 6th IEEE-RAS International Conference on Humanoid Robots*, pages 200–207. 6

Righetti, L., Buchli, J., Mistry, M., Kalakrishnan, M., and Schaal, S. (2013). Optimal distribution of contact forces with inverse-dynamics control. *The International Journal of Robotics Research*, 32(3):280–298. 8

Rogelio, J. P. and Baldovino, R. G. (2014). Development of an automatic tool changer (ATC) system for the 3-axis computer numerically-controlled (CNC) router machine: Support program for the productivity and competitiveness of the metals and engineering industries. In *IEEE International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management*, pages 1–5. 10

Roux, J., Samadi, S., Kuroiwa, E., Yoshiike, T., and Kheddar, A. (2021). Control of humanoid in multiple fixed and moving unilateral contacts. In *International Conference on Advanced Robotics*, pages 793–799. 9, 85

Ryuh, B.-S., Park, S. M., and Pennock, G. R. (2006). An automatic tool changer and integrated software for a robotic die polishing station. *Mechanism and Machine Theory*, 41(4):415–432. 10

Saab, W., Racioppo, P., and Ben-Tzvi, P. (2019). A review of coupling mechanism designs for modular reconfigurable robots. *Robotica*, 37(2):378–403. 13, 91

Sakagami, Y., Watanabe, R., Aoyama, C., Matsunaga, S., Higaki, N., and Fujimura, K. (2002). The intelligent asimo: system overview and integration. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 3, pages 2478–2483 vol.3. 5

Samadi, S. (2021). *Humanoid robots balance in multi-contact modes settings*. PhD thesis, Université Montpellier. 49, 52, 53

Samadi, S., Caron, S., Tanguy, A., and Kheddar, A. (2020). Balance of humanoid robots in a mix of fixed and sliding multi-contact scenarios. In *IEEE International Conference on Robotics and Automation*, pages 6590–6596. 9, 40, 43, 47

Samadi, S., Roux, J., Tanguy, A., Caron, S., and Kheddar, A. (2021). Humanoid control under interchangeable fixed and sliding unilateral contacts. *IEEE Robotics and Automation Letters*, 6(2):4032–4039. 9, 31, 39, 86

Samy, V., Caron, S., Bouyarmane, K., and Kheddar, A. (2017). Post-impact adaptive compliance for humanoid falls using predictive control of a reduced model. In *IEEE-RAS International Conference on Humanoid Robotics*, pages 655–660. 23

Sentis, L., Park, J., and Khatib, O. (2010). Compliant control of multi-contact and center of mass behaviors in humanoid robots. *IEEE Transactions on Robotics*, 26(3):483–501. 9

Sherikov, A., Dimitrov, D., and Wieber, P.-B. (2015). Balancing a humanoid robot with a prioritized contact force distribution. In *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pages 223–228. 6, 7

Stoiber, M. (2023). *Closing the Loop: 3D Object Tracking for Advanced Robotic Manipulation*. PhD thesis, Technische Universität München. 17, 68

Takenaka, T., Matsumoto, T., and Yoshiike, T. (2009). Real time motion generation and control for biped robot -1st report: Walking gait pattern generation-. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1084–1091. 6

Tian, H., Yang, Z., Chen, C., Jia, Y., Li, Y., Zhu, D., and Guo, J. (2017). Study on operational reliability of tool magazine and automatic tool changer based on tool-pulling force. In *IEEE International Conference on System Reliability and Safety*, pages 195–201. 10

Trinh, S., Spindler, F., Marchand, E., and Chaumette, F. (2018). A modular framework for model-based visual tracking using edge, texture and depth features. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'18*, Madrid, Spain. 17, 68

Tsuru, M., Escande, A., Tanguy, A., Chappellet, K., and Harad, K. (2021). Online object searching by a humanoid robot in an unknown environment. *IEEE Robotics and Automation Letters*, 6(2):2862–2869. 17

Ulrich, J., Alsayed, A., Arvin, F., and Krajník, T. (2022). Towards fast fiducial marker with full 6 dof pose estimation. In *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*, pages 723–730. 63, 66

Vaillant, J., Bouyarmane, K., and Kheddar, A. (2017). Multi-character physical and behavioral interactions controller. *IEEE Transactions on Visualization and Computer Graphics*, 23(6):1650–1662. 14, 15, 16

Vaillant, J., Kheddar, A., Audren, H., Keith, F., Brossette, S., Escande, A., Bouyarmane, K., Kaneko, K., Morisawa, M., Gergondet, P., et al. (2016). Multi-contact vertical ladder climbing with an HRP-2 humanoid. *Autonomous Robots*, 40(3):561–580. 14, 30

Van Wyk, K., Culleton, M., Falco, J., and Kelly, K. (2018). Comparative peg-in-hole testing of a force-based manipulation controlled robotic hand. *IEEE Transactions on Robotics*, 34(2):542–549. 18, 69

Wang, Y., Tanguy, A., and Kheddar, A. (2023). Impact-aware multi-contact balance criteria. *arXiv preprint arXiv:2308.06784*. 37

Wei, D., Trombley, C. M., Sherehiy, A., and Popa, D. O. (2021). Precise and effective robotic tool change strategy using visual servoing with rgb-d camera. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 85451. 13, 17, 91

Zhu, X., Gergondet, P., Cai, Z., Chen, X., Yu, Z., Huang, Q., and Kheddar, A. (2023). The development of a 7-dof humanoid arm for driving using a task-driven design method. *IEEE/ASME Transactions on Mechatronics*, pages 1–13. 92