University of Genova
PhD Program in Bioengineering and Robotics

Enabling Human-Robot Collaboration via Holistic Human Perception and Partner-Aware Control

Venkata Sai Yeshasvi Tirupachuri

Thesis submitted for the degree of *Doctor of Philosophy* (31° cycle)

April
2020

Supervisors
Daniele Pucci
Giulio Sandini

Jury & External Reviewers*

Rachid Alami*	Senior Scientist, LAAS-CNRS, France
Antonio Franchi	Assoc. Professor, University of Twente, Netherlands
Robert Griffin	Research Scientist, IHMC, USA
Dana Kulić*	Professor, Monash University, Australia
Ludovic Righetti	Assoc. Professor, New York University, USA
Oliver Stasse	Senior Researcher, LAAS-CNRS, France

Dynamic Interaction Control Lab
Italian Institute of Technology

Department of Informatics, Bioengineering,
Robotics and Systems Engineering

To my beloved grandmother and parents . . .

Acknowledgements

This research work would not have materialized without the trust and constant support from Dr. Daniele Pucci. Thank you for believing in me and giving me an incredible opportunity to be a part of our research team. To my dearest collaborator, Gabriele Nava, who has been a trusted confidant over the years, thank you for constantly inspiring and supporting me in carrying out this research. My comrades in the trenches of AnDy: Claudia, Silvio, Diego, Lorenzo, Kourosh, Ines, Francisco, thank you for always being there in exploring new ideas and materializing them to fruition. The last four years have been an enriching and rewarding experience which was made possible by many of the current and former colleagues and friends. Thanks to everyone, especially Stefano, Giulio, Nuno, Prashanth, Luca Fiorio, Enrico, Gianluca, Yue, Ugo, Vadim, Prof. Gabriel Baud-Bovy, Lisa, Mariacarmela, Chiara, Marco, Prof. Fulvio Mastrogiovanni, Marie, Naveen, Joan, Ali, Dimitris, Alberto, Michele, Stefano De Simone, Matteo, Julien, Marta, Lucia, Luca Recla. I would also like to thank Prof. Rachid Alami and Prof. Dana Kulić for providing their invaluable reviews of this doctoral thesis.

Finally, this journey would not have been easier and possible without the incredible love and support from my dearest Lizeth, my parents, and my dear friends Stefano Bracchi, Dara and Tagore.

Above all, I am forever grateful for all your kindness and compassion. Thank you!

Abstract

As robotic technology advances, the barriers to the coexistence of humans and robots are slowly coming down. Application domains like elderly care, collaborative manufacturing, collaborative manipulation, etc., are considered the need of the hour, and progress in robotics holds the potential to address many societal challenges. The future socio-technical systems constitute of blended workforce with a symbiotic relationship between human and robot partners working collaboratively. This thesis attempts to address some of the research challenges in enabling human-robot collaboration. In particular, the challenge of a holistic perception of a human partner to continuously communicate his intentions and needs in real-time to a robot partner is crucial for the successful realization of a collaborative task. Towards that end, we present a holistic human perception framework for real-time monitoring of whole-body human motion and dynamics. On the other hand, the challenge of leveraging assistance from a human partner will lead to improved human-robot collaboration. In this direction, we attempt at methodically defining what constitutes assistance from a human partner and propose partner-aware robot control strategies to endow robots with the capacity to meaningfully engage in a collaborative task.

Start where you are.

Use what you have.

Do what you can.

<div style="text-align: right">— Arthur R. Ashe Jr.</div>

Nomenclature

p	Scalar (non-bold small letter)
$\boldsymbol{p}$	Vector (bold small letter)
$\dot{\boldsymbol{p}}$	First-order time derivative
$\ddot{\boldsymbol{p}}$	Second-order time derivative
$\boldsymbol{A}$	Matrix (bold capital letter)
$(\cdot)^{\top}$	Transpose matrix
$|\cdot|$	Matrix determinant
$\mathbf{S}(\cdot)$	Skew-symmetric matrix
$\mathcal{A}$	Coordinate reference frame (calligraphic letter)
$O_{\mathcal{A}}$	Origin of a coordinate frame $\mathcal{A}$
$^{\mathcal{A}}\boldsymbol{p}$	Vector expressed in frame $\mathcal{A}$
$\mathcal{I}$	Inertial frame of reference
$\sum$	Summation operator
$(\cdot)^{\dagger}$	Pseudoinverse of a matrix
$(\cdot)^{-1}$	Inverse matrix
$rank(\cdot)$	Rank of a matrix
$\arg\max(\cdot)$	Maximizing argument
$\arg\min(\cdot)$	Minimizing argument
$diag(\cdot)$	Diagonal matrix
$\|\cdot\|$	Norm
$\boldsymbol{J}$	Jacobian matrix
$\boldsymbol{M}$	Mass matrix
$\boldsymbol{C}$	Coriolis matrix

Abbreviations and Acronyms

HRI	Human-Robot Interaction
HRC	Human-Robot Collaboration
DoF	Degrees of Freedom
1D	One-dimensional
2D	Two-dimensional
3D	Three-dimensional
6D	Six-dimensional
F/T	Force-Torque
IMU	Inertial Measurement Unit
IK	Inverse Kinematics
ID	Inverse Dynamics
RNEA	Recursive Newton-Euler Algorithm
URDF	Unified Robot Description Format
API	Application Programming Interface
YARP	Yet Another Robot Platform
ROS	Robot Operating System
RGB	Red-Green-Blue
CoM	Centre of Mass
PDF	Probability Density Function
MAP	Maximum-A-Posteriori
MSE	Mean Square Error
RMSE	Root Mean Square Error

Contents

Figures

Tables

Prologue

Ever since the creation of the first industrial robot Unimate from Unimation, the field of robotics made tremendous progress in developing a variety of robots with new venues for their use ranging from disaster response to elderly care (Gasparetto and Scalera, 2019; Zamalloa et al., 2017). Industrial Robots (IR) have been massively adopted in manufacturing industries. Historically, industrial robots have been employed in caged environments and are programmed to do heavy duty tasks repetitively with minimal supervisory control from humans. Though industrial robots typically offer higher operation speeds and payload capacities, they generally lack the flexibility to change the production lines quickly. This puts them at a disadvantage under emerging economic pressures of competitive markets that deem increased productivity and flexibility of production lines to meet the demand of rapid product changes.

Over the last two decades, the field of Human-Robot Interaction (HRI) emerged as an established independent research area that is dedicated to understanding, designing and evaluating robotic systems intended to be used by or with humans (Goodrich et al., 2008). In particular, the research area of Human-Robot Collaboration (HRC) focuses on the aspects of bringing together humans and robots as collaborative partners with a shared sense of purpose in achieving a common goal. Recent technological advances in hardware design, sensory and actuation capabilities paved way to a new class of robots called collaborative robots or cobots that are intended to be used safely along-side human partners either to assist them or augment their physical capabilities (Ogenyi et al., 2019).

The human partner is an integral part of human-robot collaboration scenarios (Pfeiffer, 2016). In general, the human partner establishes a common goal for collaboration and the robot partner needs to be communicated about the human intentions and needs during the entire duration of the collaborative task. Several communication interfaces are investigated in literature for various collaborative applications (Ajoudani et al., 2018). Interest in multi-modal interfaces is clearly on the rise owing to the new possibilities they enable for active collaboration. However, achieving a holistic human perception not limiting to just the kinematics but also the dynamics of the human partner helps in achieving sophisticated human as an actor models in collaborative

scenarios. Furthermore, it enables to quantitatively measure several human factors for complex tasks across different sectors.

Current generation of robotic platforms ranging from cobots to humanoids facilitate various control modalities for improved interactions with humans partners. The critical aspect of safety during physical human-robot interaction has been extensively investigated in literature and several reactive robot behaviors have been successfully tested (Haddadin, 2015; Haddadin et al., 2017). In contrast to unexpected collisions or disturbances from a human partner, collaboration scenarios involve sustained intentional physical interactions, that are often helpful for the robot partner. So, instead of exhibiting simple reactive behaviors, a robot partner can exploit the help from a human partner to perform the collaborative task efficiently. However, a clear definition of what constitutes a helpful interaction from a human partner is still an open question that needs to be methodically addressed.

In view of the above observations, we believe there is a strong potential benefit in establishing a holistic human perception framework towards partner-aware reactive robot behaviors on collaborative tasks. The research presented in this thesis is a minuscule contribution towards realizing the future socio-technical systems where human and robot partners constitute the blended workforce across different sectors. This research work has been carried out during my PhD within Dynamic Interaction Control (DIC) - a robotics research group at the Italian Institute of Technology (IIT). The doctoral program has been carried in accordance with the requirements of University of Genoa, Italy in order to obtain a PhD title. My research work is funded by the European projects: Perception and Action in Complex Environments[1] (PACE), and Advancing Anticipatory Behaviors in Dyadic Human-Robot Collaboration[2] (An.Dy). The present document is organized into three parts:

Part I: Background and Thesis Context

- **Chapter 1 Human and Robot Partners** presents a brief introduction of the current robotic technology trends in relation to humans. A brief literature review related to the challenges in human perception and robot control for collaboration scenarios is presented with a the motivation behind the research work in this doctoral thesis within the scope of An.Dy project.

---

[1] Marie Skłodowska-Curie grant agreement No. 642961
[2] Horizon 2020 Research and Innovation Programme under grant agreement No. 731540

- **Chapter 2 Rigid Multi-Body System** introduces the basic notation followed throughout this thesis, followed by rigid body kinematics and dynamics representation. Furthermore, the modeling and dynamics of a rigid multi-body system is presented in this chapter.

- **Chapter 3 Recall on Human Modeling** described the importance of digital human modeling in different fields and presents modeling details human as a system of articulated rigid bodies that is relevant to the context of robotics research.

- **Chapter 4 Enabling Technologies** describes the technologies used for human motion measurements and environmental interaction force and moment measurements. Furthermore, details of the humanoid robotic platform used for experiments in this research are presented in this chapter.

Part II: Holistic Human Perception

- **Chapter 5 Human Kinematics Estimation** introduces the problem of inverse kinematics in the context of real-time human motion tracking. Different approaches to solving an inverse kinematics problem for a highly articulated human model are presented.

- **Chapter 6 Human Dynamics Estimation** recalls the problem of inverse dynamics formulated as a stochastic estimation problem. A novel sensorless external force estimation approach is presented along with experimental validation of human dynamic variables estimation for a floating base human model.

- **Chapter 7 Software Architecture** explains our modular and extensible software infrastructure with wearable technology towards realizing a sophisticated holistic human perception framework.

Part III: Reactive Robot Control

- **Chapter 8 Partner-Aware Control** recalls classical feedback linearization control technique and describes the typical interaction characterization for human-robot collaboration scenarios. Further, this chapter presents a coupled dynamics formalism of an external agent and a robotic agent engaged in physical collaboration. More importantly, interaction characterization in terms of external agent

joint torques is presented and a partner-aware control law through Lyapunov stability analysis is proposed. Experimental validation with a whole-body humanoid robot controller for performing the task of sit-to-stand transition is carried using two complex humanoid robots.

- **Chapter 9 Trajectory Advancement** explores the idea of an intuitive reactive robot behavior through the trajectory advancement that endows a robot with the capacity to accomplish a task quicker by leveraging assistance from the human partner. The details of trajectory advancement proposition through the Lyapunov analysis is presented with experimental validation using a simple trajectory tracking controller for an end-effector of the robot and a more complex whole-body controller for performing the task of sit-to-stand transition.

- **Chapter 10 Whole-Body Retargeting & Teleoperation** presents our research towards telexistence with the main focus on whole-body human motion retargeting to robotic systems and teleoperation. Thanks to the flexibility of our software architecture presented in Chapter 7 that enables us to perform experiments with minimal changes. Our approach is validated through whole-body retargeting experiments with multiple human subjects and multiple robots. Furthermore, teleoperation experiments with two state-of-the-art whole-body controllers are presented in this chapter.

Research Contributions

- Research investigation into the coupled dynamics formalism and partner-aware control during a physical collaboration task between an external partner and a robot partner presented in Chapter 8 is published as a part of the conference proceedings and it secured a "Best Student Paper" award.

  > Tirupachuri, Y., Nava, G., Latella, C., Ferigo, D., Rapetti, L., Tagliapietra, L., Nori, F., & Pucci, D. (2019, September). Towards partner-aware humanoid robot control under physical interactions. In Proceedings of SAI Intelligent Systems Conference (pp. 1073-1092). Springer, Cham.
  >
  > **Video:** https://youtu.be/auHfyuTvkuY

- The concept of trajectory advancement and the experimental validation presented in Chapter 9 is published as a part of the conference proceedings and the extended experimental validation is accepted as a workshop dissemination.

Tirupachuri, Y., Nava, G., Rapetti, L., Latella, C., & Pucci, D. (2019). Trajectory Advancement during Human-Robot Collaboration, in press IEEE Ro-MAN 2019. arXiv preprint arXiv:1907.13445.

Video: https://youtu.be/mGHU5QTk9BI

Tirupachuri, Y., Nava, G., Rapetti, L., Latella, C., & Pucci, D. (2019). Trajectory Advancement for Robot Stand-up with Human Assistance, Italian Robotics and Intelligent Machines Conference (I-RIM 2019) 2019. arXiv preprint arXiv:1907.13445.

Video: https://youtu.be/OZ-cgzTm_pM

- The research on whole-body human motion retargeting to a robot platform and the experimental validation with state-of-the-art whole-body controllers for humanoid robots detailed in Chapter 10 is presented as a part of the conference proceedings.

  Tirupachuri, Y., Darvish, K., Romualdi, G., Rapetti, L., Ferigo, D., Chavez, F. J. A., & Pucci, D. (2019). Whole-Body Geometric Retargeting for Humanoid Robots, in press IEEE Humanoids 2019. arXiv preprint arXiv:1909.10080.

  Video: https://youtu.be/hUj83DMWxCo

- Investigations into extending our stochastic human dynamics estimation framework for a floating base human model laid out in Chapter 6 are presented as a journal contribution.

  Latella, C., Traversaro, S., Ferigo, D., Tirupachuri, Y., Rapetti, L., Andrade Chavez, F. J., Nori, F., & Pucci, D. (2019). Simultaneous Floating-Base Estimation of Human Kinematics and Joint Torques. Sensors, 19(12), 2794.

  Video: https://youtu.be/kLF4GS7tDxY, https://youtu.be/VuvVmXXiYEA

- Our work on real-time human motion tracking through various inverse kinematics algorithms and their benchmarking presented in Chapter 5 is under review for a conference.

  Rapetti, L., Tirupachuri, Y., Darvish, K., Latella, C., & Pucci, D. (2020). Model-Based Real-Time Motion Tracking using Dynamical Inverse Kinematics. arXiv preprint arXiv:1909.07669.

  Video: https://youtu.be/_-oe8F8UP7g

- The research effort on sensorless external force estimation through an updated formulation of our stochastic human dynamics estimation presented in Chapter 6 will soon be submitted to an upcoming conference. **Video:** https://youtu.be/4hLnP_6-rCs

PART I:

BACKGROUND AND THESIS CONTEXT

Human and Robot Partners

1.1 Introduction

The current decade (2010-2020) will be etched in history as the decade of many key successes in the field of robotics. The success stories are marked by some of the most practical and interesting research work witnessed through government-backed competitions like DARPA Robotics Challenge (DRC) (Krotkov et al., 2017; Norton et al., 2017). Also, the total amount of investments in robotics has been steadily increasing throughout the decade as highlighted in Fig. 1.1.
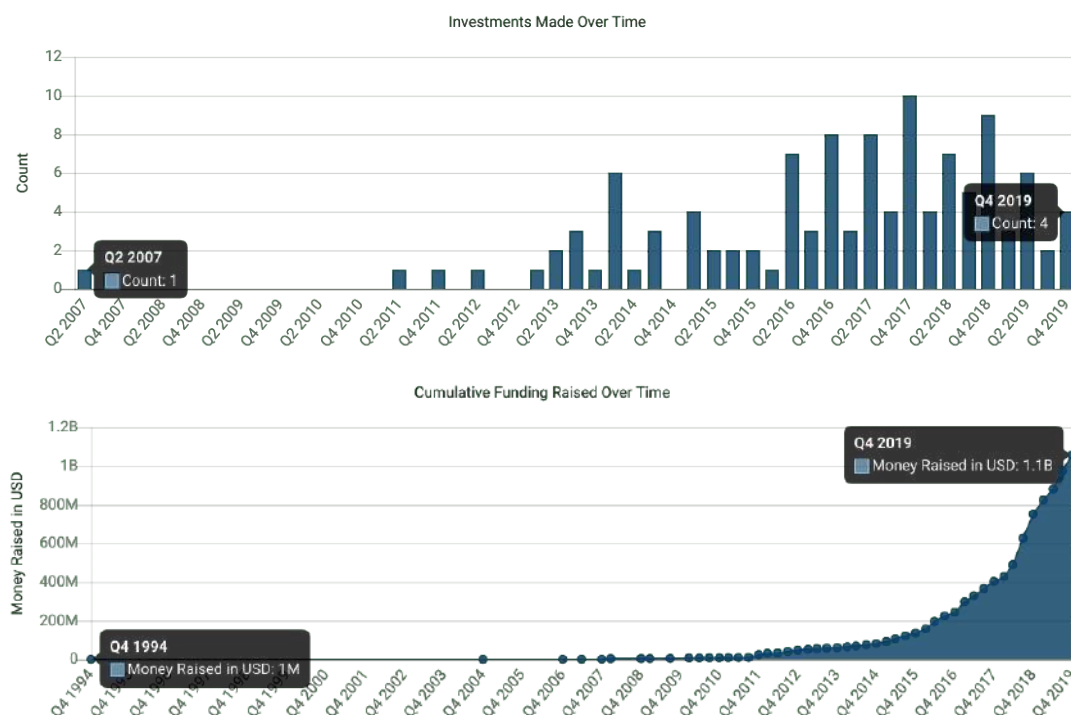
Fig. 1.1 Investment trends in robotics - Source: Crunchbase

A myriad of sophisticated products like Jibo, Mayfield Robotics Kuri, Anki Cozmo, Rethink Robotics Baxter, Franka Emika Panda, Universal Robotics U5, DJI Phantom from some of the most notable robotic startups are introduced in the market. However, as the decade draws in, streams of failures dawn on may robotic startups leading to eventual bankruptcy. On careful analysis, one can conclude that most of the startups that failed attempted to provide social robots or promised a higher emphasis on the social component in their products oriented to consumers. While on the other hand, successful companies focused more on delivering products that are function-oriented e.g., Franka Emika and Universal Robotics products are for collaborative manufacturing, DJI drones are for creative fields and entertainment.

An interesting thought exercise is to compare and contrast the smart devices market and consumer robotics market. At the hardware level, the key enabling technologies behind the smart devices and the current robots are similar e.g., interactive touch screens, high capacity graphics processing units, high bandwidth, and low latency communication devices, etc. However, the momentum sustained by the smart devices and the rapid cultural adaptation of them in the previous decade is markedly more prominent than that of robots in the current decade. Although built as general-purpose computing units, the promise, and the strength of smart devices lie in providing connectivity in the digital space, access to information and a myriad of tools to enhance creativity and productivity. In contrast, robots are expected to share the same physical space alongside humans and are expected to have many anthropomorphic traits ranging from physical structure to emotional intelligence that can assist or augment the quality of human life. So, to have more cultural acceptance and adoption, robots need to have relational skills (Damiano et al., 2015; Parviainen and Pirhonen, 2017) that ensure a social component in their existence. On the other hand, the need for standardized measurement tools and methodological principles to quantify the effect of a robot for interacting with humans reliably is backed in the research community (Bartneck et al., 2009; Belhassein et al., 2019).

Businesses are capitalistic entities by nature and they always try to lower the costs and increase the profit margins. The global trend of income inequality in western society is a direct result of decades of operating costs optimization by relocating the business operations from labor expensive regions to inexpensive regions. The latest advances in robotics and automation are perceived as a great threat that completely replaces human labor across different sectors (Frey and Osborne, 2017). Current sentiments assume a general rule of thumb that any routine work is on the brink of replacement by robots and automation. An example of routine work is the assembly task on a shop

floor. However, a recent qualitative study on assembly work highlights the intricacies of the "*perceived*" routine work of assembly and the importance of experiential knowledge (Pfeiffer, 2016).

> **Excerpt from (Pfeiffer, 2016)**
>
> *Automation always has aimed and always will aim to substitute for human labor, and with new robotic concepts ahead we will see many attempts to do so in the world of assembly. However, the limits of automation are to be seen after decades of robotic use in industry: coping with imponderables, the flexibility and ability to adapt, the unlimited variability of behavior and tacit skills and body intelligence — all these dimensions of genuinely human work cannot be replaced by robots.*

Humans will remain an integral part of the organic workforce, even as the complexity and composition of the work evolves to meet the demands of rapid market changes. Recently collaborative robots gained a lot of popularity and are increasingly adopted across several industries (Bragança et al., 2019) as the technology is becoming more flexible, versatile and safe (Saenz et al., 2018). However, current robotic technology is limited in terms of active collaboration through physical interactions with human partners. This limitation is directly linked to the deficiencies of human perception involving both whole-body motion and articular stress which leads to a blind spot for robots working alongside humans. Furthermore, a collaborative human partner aims at helping the robot partner through physical interactions. Under such circumstances instead of exhibiting simple reactive behaviors, a robot partner can actively use the help to accomplish a shared goal.

1.2 Brief Literature Review

Robots existed as separate entities till now but the horizons of a symbiotic human-robot partnership is impending. In particular, application domains like elderly care, collaborative manufacturing, collaborative manipulation, etc., are considered the need of the hour. Across all these domains, it is crucial for robots to physically interact with humans to either assist them or to augment their capabilities. Such human in the loop physical human-robot interaction (PHRI) or physical human-robot collaboration (PHRC) scenarios demand careful consideration of both the human and the robot systems while designing controllers to facilitate robust interaction strategies for successful task completion. More importantly, a generalized agent-robot interaction formalism is

needed to study the physical interaction adaptation and exploitation, where the *agent* can be a human or another robot.

The three main components of any physical agent-robot collaboration (PARC) scenario are 1) a *robotic agent*, 2) an *external agent* be it a human or another robot, and 3) the *environment* surrounding them. Over time, physical interactions are present between any of the two components. An intuitive conceptual representation of the interactions occurring during collaborative scenarios is presented in (Losey et al., 2018).

1.2.1   Challenges in Human Perception

The knowledge of human intent is a key element for the successful realization of PHRI tasks. The process of human intent detection is broadly divided into *intent information, intent measurement* and *intent interpretation* (Losey et al., 2018). The choice of a communication channel is directly related to intent measurement and affects the robot's ability to understand human intent. Accordingly, a myriad of technologies has been used as interfaces for different applications of PHRI. In the context of rehabilitation robotics, electroencephalography (EEG) (Mattar et al., 2018; McMullen et al., 2014; Sarac et al., 2013) and electromyography (EMG) (Au et al., 2008; Radmand et al., 2014; Song et al., 2008; Zhou et al., 2018) proved to be invaluable. Force myography (FMG) (Cho et al., 2016; Rasouli et al., 2016; Yap et al., 2016) is a relatively new technology that has been successfully used in rehabilitation. EMG has also been successfully used by (Peternel et al., 2018) to realize a collaborative application to continuously monitor human fatigue. Force-Torque (F/T) sensors mounted on the robots are often the most relied technology in PHRC scenarios for reactive robot control as they facilitate direct monitoring and regulation of the interaction wrenches[1] between the human and the robot partners (Bussy et al., 2012a,b; Donner and Buss, 2016; Ikemoto et al., 2012; Peternel and Babič, 2013).

Vision-based techniques like human skeletal tracking (Reily et al., 2018), human motion estimation (Kyrkjebø, 2018), and hand gesture recognition (Rautaray and Agrawal, 2015) are also used as interfaces. In general, the designer decides an interface, to communicate the human intent, depending on the application and often using a single interface mode is limiting. Hence, a combination of vision and haptic interfaces are used in literature to realize successful applications of human-robot collaboration (Agravante et al., 2014; De Santis et al., 2007).

---

[1] Wrench refers to forces and moments and is sometimes used in this thesis in place of forces and moments

We believe there is an impending change in the paradigm of human perception and the future technologies for human-robot collaboration will leverage on getting as much holistic information as possible from human partners, especially for domains like collaborative manufacturing. A step in this direction is Digital Human Models (DHM) simulation tools that are embraced currently in manufacturing systems (Demirel and Duffy, 2007; Endo et al., 2014; Fritzsche, 2010; Shahrokhi and Bernard, 2009). Ergonomics or human factors are a critical component in modern hybrid assembly lines that consists of a shared workspace between human and robot partners (Alexopoulos et al., 2013; Galin and Meshcheryakov, 2019; Irshad et al., 2019; Rajesh and Srinath, 2016). Several investigations into ergonomic analysis using digital human modeling tools are available in the literature. However, these tools are often used as offline standalone tools during the preliminary design phases of a product assembly line and reiterating or redesigning is a resource exhaustive process. Furthermore, lack of holistic human perception poses a big challenge to quantitatively evaluate and validate the performance metrics of passive or active assistive devices, which encouraged researchers to find interesting, albeit complex applications for humanoid robots (Imamura et al., 2018; Ito et al., 2018; Yoshida et al., 2017). The development of low-cost easy-to-use tools for real-time monitoring of human motion and dynamics is still an open challenge (Moes, 2010; Mourtzis, 2019). Having both the kinematic quantities like joint positions, velocities and accelerations; and dynamic quantities like joint torques of the human will enable real-time monitoring to build robust controllers for successful task completion taking into account the physical interactions.

1.2.2 Challenges in Robot Control

Collisions pose a significant challenge to robots in dynamic environments of human habitats and the capacity to detect, isolate, identify and react is fundamental for their coexistence alongside humans (Haddadin et al., 2017). Given the task of tracking a reference trajectory, impedance control (Hogan, 1984; Magrini et al., 2015) is one of the most exploited approaches to achieve stable robot behavior while maintaining contacts. It facilitates a compliant and safe physical interaction between the agents in HRC scenarios. Any interaction wrench from the human is handled safely by changing the robot's actual trajectory i.e. the forces and moments are controlled by acting on position and orientation changes. Novel adaptive control schemes are successfully implemented expanding the applicability of impedance control (Gopinathan et al., 2017; Gribovskaya et al., 2011; Li and Ge, 2013). The quality of interaction is further

augmented through online adaptive admittance controls schemes which consider human intent inside the control loop (Lecours et al., 2012; Ranatunga et al., 2016). Adaptive control schemes endow robots with compliant characteristics that are safe for physically interacting with them. An obvious outcome of such compliance is the momentary deviation from the reference trajectory to accommodate external interactions but the original trajectory is restored when the interaction stops (Geravand et al., 2013).

Interaction between an external agent and a humanoid robot is particularly challenging because of the complexity of the robotic system (Goodrich et al., 2008). Unlike traditional industrial robots which are fixed base by design, humanoid robots are designed as floating base systems to facilitate anthropomorphic navigational capabilities. The aspect of balancing has received a lot of attention in the humanoid robotics community and several prior efforts (Caux et al., 1998; Hirai et al., 1998; Hyon et al., 2007) went into building controllers that ensure stable robot behavior. More recently momentum-based control proved to be a robust approach and several successful applications have been realized (Herzog et al., 2014; Hofmann et al., 2009; Koolen et al., 2016; Stephens and Atkeson, 2010) ensuring contact stability (Nori et al., 2015b) with the environment by monitoring contact wrenches through quadratic programming (Nava et al., 2016; Ott et al., 2011; Wensing and Orin, 2013). In general, these controllers are built to ensure robustness to any external perturbations and hence they are often blind to any helpful interaction an external agent is trying to have with the robot to help achieve its task.

Collaboration scenarios involve sustained intentional physical interactions, that are often helpful for the robot partner. So, instead of exhibiting simple compliant reactive behaviors, a robot can exhibit partner-aware behavior by exploiting the help from the human partner to perform the collaborative task efficiently. However, a clear definition of what constitutes a helpful interaction of a human partner is still an open question that needs to be methodically addressed.

1.3 Motivation

The Horizon-2020 European project named An.Dy - Advancing Anticipatory Behaviors in Dyadic Human-Robot Collaboration (H2020-ICT-2016-2017, No.731540) (An.Dy, 2017) aims at advancing the current state-of-the-art in human-robot collaboration scenarios that require intentional physical interactions between the human partner and the robot partner. Successful physical collaboration needs sustained dynamic

wrench exchange between the interacting agents. Some example scenarios of physical human-robot collaboration are highlighted in Fig. 1.2 where an industrial manipulator helping an assembly line worker in transporting a heavy load (1.2a); an actuated exoskeleton assisting overhead tasks (1.2b); and a whole-body collaboration from a humanoid robot for assisted object transportation (1.2c).

| (a) | (b) | (c) |
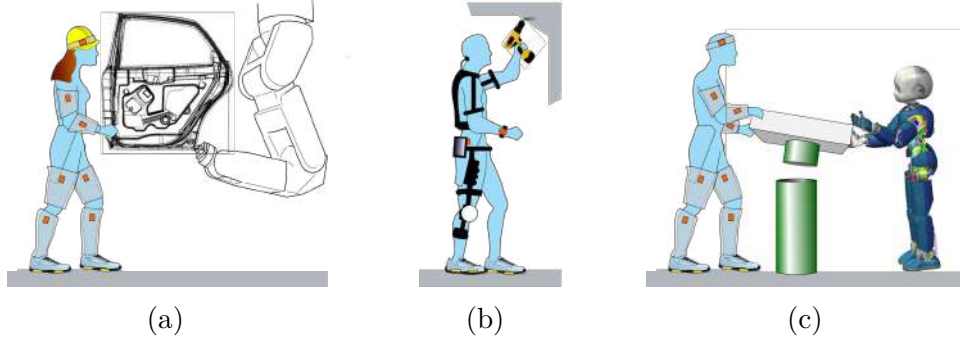
Fig. 1.2 Example scenarios of physical human-robot collaboration

Two of the main objectives of An.Dy project are:

- AnDy Suit: A novel wearable technology for real-time monitoring of human whole-body kinematics (position, velocity, acceleration), and dynamics (contact forces, joint torques, and muscle activations).

- AnDy Control: Development of partner-aware reactive and predictive robot control strategies for efficient physical human-robot collaboration leveraging AnDy suit technology.

A growing technological trend referred to as *Human 2.0* is about augmenting humans covering a wide range of cognitive and physical improvements. The two main categories of physical augmentation (source - Gartner tech trends 2020) are: Sensory augmentation (hearing, vision, perception); Appendage and biological function augmentation (exoskeletons, prosthetics). We envision a future where technologies like AnDy suit become an integral part of human augmentation and the advanced control strategies enhance the physical interaction experience with collaborative robot partners. A potential outcome of this is a future blended workforce where the robots are seamlessly incorporated in the human umwelt.

Towards the goal of enabling human-robot collaboration via holistic human perception and partner aware control, we consider both the robot and the human are a system of rigid bodies. So, the next chapter presents the basic notation and representational details relevant to rigid body systems.

Rigid Multi-Body System

2.1 Notation

The mathematical notation introduced in this section is followed throughout this thesis. Additional notation as required for a specific chapter will be introduced at the beginning of the chapter.

- $\mathcal{I}$ denoted the inertial frame of reference, with z-axis pointing against gravity.

- The constant g denotes the norm of the gravitational acceleration.

- $\mathbb{R}$ denotes the set of real numbers and $\mathbb{N}$ denotes the set of natural numbers.

- $\boldsymbol{p} \in \mathbb{R}^n$ denotes a n-dimensional column vector of real numbers and p denotes a scalar quantity.

- Given two n-dimensional column vectors of real numbers, i.e. $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$, $\boldsymbol{u}^\top \boldsymbol{v}$ denotes their inner product, with the transpose operator $\top$.

- Given two n-dimensional column vectors of real numbers, i.e. $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$, $\boldsymbol{u} \times \boldsymbol{v}$ denotes their cross product, where

$$\boldsymbol{u} \times := \begin{bmatrix} 0 & -u_z & u_y \\ u_z & 0 & -u_x \\ -u_y & u_x & 0 \end{bmatrix} \in \mathbb{R}^{3 \times 3} \tag{2.1}$$

- The operator $\|.\|$ indicates the squared norm of a vector, such that given a vector $\boldsymbol{u} \in \mathbb{R}^3$, it is defined as,

$$\|\boldsymbol{u}\| = \sqrt{\boldsymbol{u}_1^2 + \boldsymbol{u}_2^2 + \boldsymbol{u}_3^2} \tag{2.2}$$

- $\boldsymbol{I}_n \in \mathbb{R}^{n \times n}$ denotes an identity matrix of dimension n; $\boldsymbol{0}_n \in \mathbb{R}^n$ denotes the zero column vector of dimension n; and $\boldsymbol{0}_{n \times m} \in \mathbb{R}^{n \times m}$ denotes the zero matrix of dimension $n \times m$.

- Given a vector $\boldsymbol{p}$ and a reference frame $\mathcal{A}$, the notation $^{\mathcal{A}}\boldsymbol{p}$ denotes the vector $\boldsymbol{p}$ expressed in $\mathcal{A}$.

- Let $SO(3)$ be the set of $\mathbb{R}^{3 \times 3}$ orthogonal matrices with determinant equal to one, such that

$$SO(3) := \{\boldsymbol{R} \in \mathbb{R}^{3 \times 3} \mid \boldsymbol{R}^T \boldsymbol{R} = \boldsymbol{I}_3 , \quad |\boldsymbol{R}| = 1\} \tag{2.3}$$

- Let $so(3)$ be the set of the skew-symmetric matrices $\in \mathbb{R}^{3 \times 3}$, such that

$$so(3) := \{\mathbf{S} \in \mathbb{R}^{3 \times 3} \mid \mathbf{S}^T = -\mathbf{S}\} \tag{2.4}$$

- Let the set $SE(3)$ be defined as

$$SE(3) := \left\{ \begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ \boldsymbol{0}_{1 \times 3} & 1 \end{bmatrix} \in \mathbb{R}^{4 \times 4} \mid \boldsymbol{R} \in SO(3) , \quad \boldsymbol{p} \in \mathbb{R}^3 \right\} \tag{2.5}$$

- The operator $\mathbf{S}(.) : \mathbb{R}^3 \to so(3)$ denotes *skew-symmetric* vector operation, such that given two vectors $\boldsymbol{v}, \boldsymbol{u} \in \mathbb{R}^3$, it is defined as $\boldsymbol{v} \times \boldsymbol{u} = S(\boldsymbol{v})\boldsymbol{u}$.

- The *vee* operator $.^{\vee} : so(3) \to \mathbb{R}^3$ denotes the inverse of the *skew-symmetric* vector operator, such that given a matrix $\boldsymbol{A} \in so(3)$ and a vector $\boldsymbol{u} \in \mathbb{R}^3$, it is defined as $\boldsymbol{A}\boldsymbol{u} = \boldsymbol{A}^{\vee} \times \boldsymbol{u}$.

2.2   Rigid Body Representation

Definition

A rigid body is defined as an object that is non-deformable under the application of external forces and moments. Considering any physical object to be a collection of point masses, the distance between any two point masses of a rigid body does not change when an external wrench is applied.

Although a perfectly non-deformable object seems unrealistic, for all practical purposes a rigid body is considered to have *negligible* deformation.

The motion of a rigid body is well represented by associating a *coordinate frame* that is attached to its body, typically with the origin of the frame at the Center of Mass (CoM) of the rigid body. This frame is usually referred to as the *body frame* denoted by $\mathcal{B}$ (Siciliano and Khatib, 2007; Siciliano et al., 2008). Consider an inertial frame of reference $\mathcal{I}$, with the origin at a 3D point $O_\mathcal{I}$. Let $O_\mathcal{B}$ be a 3D point that is the origin of the body frame $\mathcal{B}$. The motion of the rigid body (translation, rotation or a combination of the two) can be expressed with respect to (w.r.t) the inertial frame $\mathcal{I}$.

2.2.1 Rigid Body Pose

The coordinates of the origin $O_\mathcal{B}$ w.r.t. $\mathcal{I}$ are elements of the position vector,

$$
{}^\mathcal{I}\boldsymbol{o}_\mathcal{B} = \begin{bmatrix} {}^\mathcal{I}o_x \\ {}^\mathcal{I}o_y \\ {}^\mathcal{I}o_z \end{bmatrix}_\mathcal{B} \in \mathbb{R}^3 \tag{2.6}
$$

The orientation of $\mathcal{B}$ w.r.t. $\mathcal{I}$ is described by a rotation matrix ${}^\mathcal{I}\boldsymbol{R}_\mathcal{B} \in SO(3)$, regardless of the positions of the origins $O_\mathcal{I}$ and $O_\mathcal{B}$.

Homogeneous Transformation

A compact representation of the pose combining both the position and the orientation of a rigid body is given by the homogeneous transformation matrix ${}^\mathcal{I}\boldsymbol{H}_\mathcal{B} \in SE(3)$.

$$
{}^\mathcal{I}\boldsymbol{H}_\mathcal{B} = \begin{bmatrix} {}^\mathcal{I}\boldsymbol{R}_\mathcal{B} & {}^\mathcal{I}\boldsymbol{o}_\mathcal{B} \\ \boldsymbol{0}_{1\times3} & 1 \end{bmatrix} \tag{2.7}
$$

Consider a 3D point P present in a rigid body as shown in Figure 2.1. The vector ${}^\mathcal{B}\boldsymbol{p} \in \mathbb{R}^3$ denotes the point P w.r.t frame $\mathcal{B}$. The vector ${}^\mathcal{I}\boldsymbol{p} \in \mathbb{R}^3$ denotes the point P w.r.t frame $\mathcal{I}$ can be computed using the homogeneous matrix ${}^\mathcal{I}\boldsymbol{H}_\mathcal{B}$ as following,

$$
{}^\mathcal{I}\boldsymbol{p} = \begin{bmatrix} {}^\mathcal{I}\boldsymbol{R}_\mathcal{B} & {}^\mathcal{I}\boldsymbol{o}_\mathcal{B} \\ \boldsymbol{0}_{1\times3} & 1 \end{bmatrix} \begin{bmatrix} {}^\mathcal{B}\boldsymbol{p} \\ 1 \end{bmatrix} \tag{2.8}
$$

$$
{}^\mathcal{I}\boldsymbol{p} = {}^\mathcal{I}\boldsymbol{o}_\mathcal{B} + {}^\mathcal{I}\boldsymbol{R}_\mathcal{B}\,{}^\mathcal{B}\boldsymbol{p} \tag{2.9}
$$

If $O_\mathcal{B} \equiv O_\mathcal{I}$ i.e., null position vector $^\mathcal{I}\boldsymbol{o}_\mathcal{B}$, Eq. (2.9) falls into the a pure rotational case and the transformation becomes,

$$^\mathcal{I}\boldsymbol{p} = {}^\mathcal{I}\boldsymbol{R}_\mathcal{B} \, {}^\mathcal{B}\boldsymbol{p} \tag{2.10}$$

If the frame $\mathcal{B}$ has the same orientation as that of the inertial frame of reference $\mathcal{I}$ i.e., the rotation matrix $^\mathcal{I}\boldsymbol{R}_\mathcal{B} = \boldsymbol{1}_3$, then (2.9) falls into a pure translation case and the transformation becomes,

$$^\mathcal{I}\boldsymbol{p} = {}^\mathcal{B}\boldsymbol{p} + {}^\mathcal{I}\boldsymbol{o}_\mathcal{B} \tag{2.11}$$
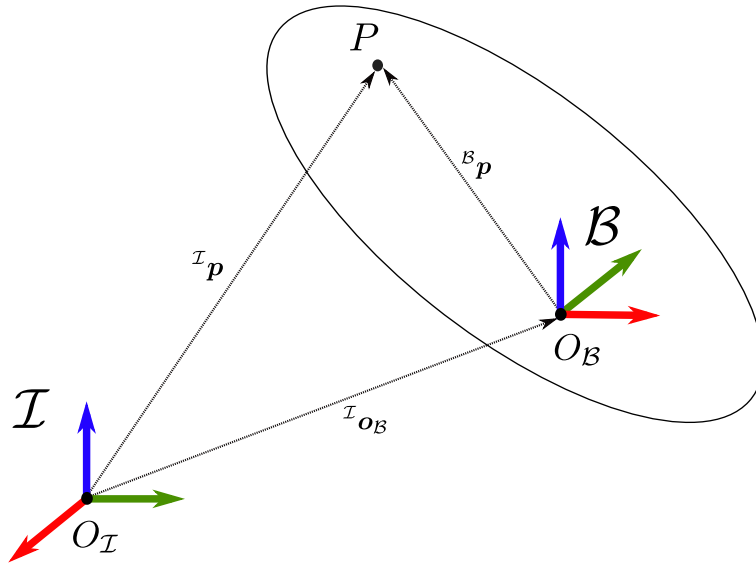
Fig. 2.1 Representation of a rigid-body with body frame $\mathcal{B}$ and a 3D point P in two different coordinate frames. The figure introduces RGB (Red-Green-Blue) convention for x-y-z axes.

> **Frame Convention**
>
> *Throughout the thesis, the reference frames are represented using a RGB (Red-Green-Blue) convention for x-y-z axes, respectively.*

2.2.2 The Derivative of a Rotation Matrix

Given a rotational motion of a rigid body, the rotational matrix $^\mathcal{I}\boldsymbol{R}_\mathcal{B}$ is a time-varying quantity. The angular velocity $^\mathcal{I}\boldsymbol{\omega}_\mathcal{B}$ of the body w.r.t. the inertial frame of

reference $\mathcal{I}$, the relation between the time derivative of the rotation matrix and the transformation of the rotation matrix can be established as following (refer to (Siciliano et al., 2008), Section 3.1.1, for a detailed description),

$$\mathbf{S}(^{\mathcal{I}}\boldsymbol{\omega}_{\mathcal{B}}) = {^{\mathcal{I}}}\dot{\boldsymbol{R}}_{\mathcal{B}} \, {^{\mathcal{I}}}\boldsymbol{R}_{\mathcal{B}}^{T} \tag{2.12}$$

where,

$$\mathbf{S}(^{\mathcal{I}}\boldsymbol{\omega}_{\mathcal{B}}) := {^{\mathcal{I}}}\boldsymbol{\omega}_{\mathcal{B}}\times \; = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}_{\mathcal{B}} \in so(3) \tag{2.13}$$

2.2.3 Velocity

The velocity of the 3D point P of a rigid body as shown in Figure 2.1 w.r.t. $\mathcal{I}$ can be obtained from the first-order time derivative of Eq. (2.9), such that

$$^{\mathcal{I}}\dot{\boldsymbol{p}} = \frac{d}{dt}\left(^{\mathcal{I}}\boldsymbol{p}\right) = {^{\mathcal{I}}}\dot{\boldsymbol{o}}_{\mathcal{B}} + {^{\mathcal{I}}}\dot{\boldsymbol{R}}_{\mathcal{B}} \, {^{\mathcal{B}}}\boldsymbol{p} \tag{2.14}$$

Using the relation from Eq. (2.13), the above relation becomes,

$$^{\mathcal{I}}\dot{\boldsymbol{p}} = {^{\mathcal{I}}}\dot{\boldsymbol{o}}_{\mathcal{B}} + \mathbf{S}(^{\mathcal{I}}\boldsymbol{\omega}_{\mathcal{B}}) \, {^{\mathcal{I}}}\boldsymbol{R}_{\mathcal{B}} \, {^{\mathcal{B}}}\boldsymbol{p} \tag{2.15}$$

$$^{\mathcal{I}}\dot{\boldsymbol{p}} = {^{\mathcal{I}}}\dot{\boldsymbol{o}}_{\mathcal{B}} + {^{\mathcal{I}}}\boldsymbol{\omega}_{\mathcal{B}}\times \, {^{\mathcal{I}}}\boldsymbol{R}_{\mathcal{B}} \, {^{\mathcal{B}}}\boldsymbol{p} \tag{2.16}$$

2.2.4 Acceleration

The velocity of the 3D point P of a rigid body as shown in Figure 2.1 w.r.t. $\mathcal{I}$ can be obtained from the second-order time derivative of Eq. (2.9), such that

$$\begin{aligned} ^{\mathcal{I}}\ddot{\boldsymbol{p}} &= \frac{d^2}{d^2t}\left(^{\mathcal{I}}\boldsymbol{p}\right) = {^{\mathcal{I}}}\ddot{\boldsymbol{o}}_{\mathcal{B}} + {^{\mathcal{I}}}\dot{\boldsymbol{\omega}}_{\mathcal{B}}\times \, {^{\mathcal{I}}}\boldsymbol{R}_{\mathcal{B}} \, {^{\mathcal{B}}}\boldsymbol{p} + {^{\mathcal{I}}}\boldsymbol{\omega}_{\mathcal{B}}\times \, {^{\mathcal{I}}}\dot{\boldsymbol{R}}_{\mathcal{B}} \, {^{\mathcal{B}}}\boldsymbol{p} \\ &= {^{\mathcal{I}}}\ddot{\boldsymbol{o}}_{\mathcal{B}} + {^{\mathcal{I}}}\dot{\boldsymbol{\omega}}_{\mathcal{B}}\times \, {^{\mathcal{I}}}\boldsymbol{R}_{\mathcal{B}} \, {^{\mathcal{B}}}\boldsymbol{p} + {^{\mathcal{I}}}\boldsymbol{\omega}_{\mathcal{B}}\times \left(^{\mathcal{I}}\boldsymbol{\omega}_{\mathcal{B}}\times \, {^{\mathcal{I}}}\boldsymbol{R}_{\mathcal{B}} \, {^{\mathcal{B}}}\boldsymbol{p}\right) \end{aligned} \tag{2.17}$$

where, $\dot{\boldsymbol{\omega}}_{\mathcal{B}}$ is the angular acceleration of the rigid body.

2.3 6D Vectors

A proper mathematical representation of the rigid body kinematics and dynamics becomes essential when dealing with a multi-body system to express it succinctly. 6D vectors are a handy tool for expressing rigid body dynamics in a compact form. A thorough material on the most commonly used 6D vector representations for dealing with rigid body dynamics is presented in (Traversaro, 2017).

2.3.1 6D Motion Vectors

The 6D velocity vector of a rigid body expressed w.r.t. the inertial frame of reference $\mathcal{I}$ is written as,

$$^{\mathcal{I}}\boldsymbol{v}_{\mathcal{B}} = \begin{bmatrix} ^{\mathcal{I}}\dot{\boldsymbol{p}} \\ ^{\mathcal{I}}\boldsymbol{\omega}_{\mathcal{B}} \end{bmatrix} \in \mathbb{R}^6 \tag{2.18}$$

The 6D acceleration vector of a rigid body expressed w.r.t. the inertial frame of reference $\mathcal{I}$ is written as,

$$^{\mathcal{I}}\boldsymbol{a}_{\mathcal{B}} = \begin{bmatrix} ^{\mathcal{I}}\ddot{\boldsymbol{p}} \\ ^{\mathcal{I}}\dot{\boldsymbol{\omega}}_{\mathcal{B}} \end{bmatrix} \in \mathbb{R}^6 \tag{2.19}$$

Adjoint Transformation for Motion Vectors

Homogeneous transformation introduced in Eq. (2.7) cannot be used directly with 6D motion vectors. The change of frame of reference for 6D motion vectors is achieved through a new transformation matrix called *adjoint matrix* denoted by $\boldsymbol{X} \in \mathbb{R}^{6\times 6}$. Given two generic frames of reference $\mathcal{A}$ and $\mathcal{B}$, and $^{\mathcal{B}}\boldsymbol{o}_{\mathcal{A}}$ the position vector of the origin of $\mathcal{A}$ w.r.t. $\mathcal{B}$, the adjoint transformation matrix is defined as,

$$^{\mathcal{B}}\boldsymbol{X}_{\mathcal{A}} = \begin{bmatrix} ^{\mathcal{B}}\boldsymbol{R}_{\mathcal{A}} & \boldsymbol{0}_3 \\ -^{\mathcal{B}}\boldsymbol{R}_{\mathcal{A}}\,\mathbf{S}\left(^{\mathcal{B}}\boldsymbol{o}_{\mathcal{A}}\right) & ^{\mathcal{B}}\boldsymbol{R}_{\mathcal{A}} \end{bmatrix} \tag{2.20}$$

and the transformation of 6D motion vectors is denoted as,

$$^{\mathcal{B}}\boldsymbol{v}_{\mathcal{A}} = {}^{\mathcal{B}}\boldsymbol{X}_{\mathcal{A}}\,{}^{\mathcal{A}}\boldsymbol{v}_{\mathcal{A}} \tag{2.21}$$

Cross Product for Motion Vectors

The cross product operation as defined in Eq. (2.1) is not applicable to 6D motion vectors. Consider a rigid body with a body frame $\mathcal{B}$ that is moving with a velocity $^{\mathcal{I}}\boldsymbol{v}_{\mathcal{B}}$, as defined in Eq. (2.18), the cross product operator is defined as,

$$^{\mathcal{I}}\boldsymbol{v}_{\mathcal{B}}\times \ := \ \begin{bmatrix} ^{\mathcal{I}}\boldsymbol{\omega}_{\mathcal{B}}\times & ^{\mathcal{I}}\dot{\boldsymbol{p}} \\ \boldsymbol{0}_3 & ^{\mathcal{I}}\boldsymbol{\omega}_{\mathcal{B}}\times \end{bmatrix} \in \mathbb{R}^{6\times 6} \tag{2.22}$$

2.3.2 6D Force Vectors

6D force vectors represent the forces and moments, with forces being the first three elements followed by the moments as the next three elements. A 6D force vector with respect to the inertial frame $\mathcal{I}$ is denoted as,

$$^{\mathcal{I}}\boldsymbol{f} = \begin{bmatrix} ^{\mathcal{I}}\mathbf{f} \\ ^{\mathcal{I}}\boldsymbol{m} \end{bmatrix} \in \mathbb{R}^6 \tag{2.23}$$

Note that the 6D force vector denoted by $^{\mathcal{I}}\boldsymbol{f}$ only expresses the forces and moments coordinates with respect to the inertial frame of reference $\mathcal{I}$ but they do not specify the frame (associated to any rigid body) on which they are acting up. This distinction is important to observe because the 6D motion vectors represent the motion associated with a rigid body and are associated with frame that is attached to the body. While in the case of 6D force vectors, the forces and moments are quantities that are external to the rigid body.

Adjoint Transformation Dual for Force Vectors

Given two generic frames of reference $\mathcal{A}$ and $\mathcal{B}$, and $^{\mathcal{B}}\boldsymbol{o}_{\mathcal{A}}$ the position vector of the origin of $\mathcal{A}$ w.r.t. $\mathcal{B}$, the adjoint transformation matrix for force vectors is defined as,

$$^{\mathcal{B}}\boldsymbol{X}^{*}_{\mathcal{A}} = \begin{bmatrix} ^{\mathcal{B}}\boldsymbol{R}_{\mathcal{A}} & -^{\mathcal{B}}\boldsymbol{R}_{\mathcal{A}}\,\mathbf{S}\left(^{\mathcal{B}}\boldsymbol{o}_{\mathcal{A}}\right) \\ \boldsymbol{0}_3 & ^{\mathcal{B}}\boldsymbol{R}_{\mathcal{A}} \end{bmatrix} \tag{2.24}$$

and the transformation of 6D force vectors is denoted as,

$$^{\mathcal{B}}\boldsymbol{f} = {}^{\mathcal{B}}\boldsymbol{X}^{*}_{\mathcal{A}}\,{}^{\mathcal{A}}\boldsymbol{f} \tag{2.25}$$

Cross Product Dual for Force Vectors

Similar to the cross product for the 6D motion vectors, we have a cross product for the 6D force vectors that is the dual version of Eq. (2.22) that is defined as,

$$
{}^{\mathcal{I}}\boldsymbol{v}_{\mathcal{B}}\times^{*} := \begin{bmatrix} {}^{\mathcal{I}}\boldsymbol{\omega}_{\mathcal{B}}\times & \boldsymbol{0}_3 \\ {}^{\mathcal{I}}\dot{\boldsymbol{p}} & {}^{\mathcal{I}}\boldsymbol{\omega}_{\mathcal{B}}\times \end{bmatrix} \in \mathbb{R}^{6\times 6} \tag{2.26}
$$

and the cross product dual operator to be used with the 6D motion vectors and 6D force vectors is denoted as,

$$
{}^{\mathcal{I}}\boldsymbol{v}_{\mathcal{B}}\times^{*}\, {}^{\mathcal{B}}\boldsymbol{f} \in \mathbb{R}^{6} \tag{2.27}
$$

2.4 Rigid Body Dynamics

The dynamics of a rigid body deals with understanding the motion induced by the application of an external force or torque on the rigid body. The equations of motion describe the mathematical model for studying the dynamics of a rigid body. Newton-Euler representation of equations of motion expressed in a compact form using 6D vectors is denoted as,

$$
{}^{\mathcal{I}}\boldsymbol{f}^{\mathcal{B}} = \frac{d}{dt}\Big(\boldsymbol{I}\, {}^{\mathcal{I}}\boldsymbol{v}_{\mathcal{B}}\Big) = \boldsymbol{I}\, {}^{\mathcal{I}}\boldsymbol{a}_{\mathcal{B}} + {}^{\mathcal{I}}\boldsymbol{v}_{\mathcal{B}}\times^{*}\boldsymbol{I}\, {}^{\mathcal{I}}\boldsymbol{v}_{\mathcal{B}} \tag{2.28}
$$

where,

- ${}^{\mathcal{I}}\boldsymbol{f}^{\mathcal{B}} \in \mathbb{R}^6$ in the net 6D force i.e., sum of all the forces and moments acting on the rigid body expressed in the inertial frame of reference

- $\boldsymbol{I}$ is the inertia matrix $\in \mathbb{R}^{6\times 6}$, such that

$$
\boldsymbol{I} = \begin{bmatrix} \boldsymbol{I}_c + \mathrm{m}\,\boldsymbol{c}\times\,\boldsymbol{c}^{\top} & \mathrm{m}\,\boldsymbol{c}\times \\ \mathrm{m}\,\boldsymbol{c}\times^{\top} & \mathrm{m}\,\boldsymbol{1}_3 \end{bmatrix} \tag{2.29}
$$

where m is the body mass, $\boldsymbol{I}_c$ is the rotational inertia w.r.t. the body center of mass (CoM), and $\boldsymbol{c}$ is the position vector from the CoM of the body to the origin of the body frame $\mathcal{B}$

- The term ${}^{\mathcal{I}}\boldsymbol{v}_{\mathcal{B}}\times^{*}$ is the operator that maps $\boldsymbol{I}$ to its derivative $\dot{\boldsymbol{I}}$

Eq. 2.28 describes the relation between the net forces and moments acting on a rigid body and the rate of change of its momentum $\boldsymbol{I}\ {}^{\mathcal{I}}\boldsymbol{v}_{\mathcal{B}}$.

2.5 Rigid Multi-Body System

A rigid multi-body system is composed of two or more interconnected rigid bodies. The connection between the rigid bodies is made through *joints*. The relative motion between two interconnected rigid bodies is determined by the motion subspace of the joint present between them. So, a combination of rigid bodies through joints compose an articulated multi-body system. The resulting motion of the entire articulated rigid multi-body system can be determined by accounting the elementary motions of each of the rigid bodies present, taking into account the constraints imposed by various joints.

2.5.1 Modeling

A robotic system is often considered as an articulated rigid multi-body system that can be represented as a kinematic tree (Featherstone, 2007; Siciliano et al., 2008). According to the technical jargon in robotics community, a rigid body of a robotic system is often referred to as a *link*. The links and the joints of a robotic system forms the edges and nodes of the kinematic tree topological representation of an articulated rigid multi-body system as shown in Figure. 2.2.
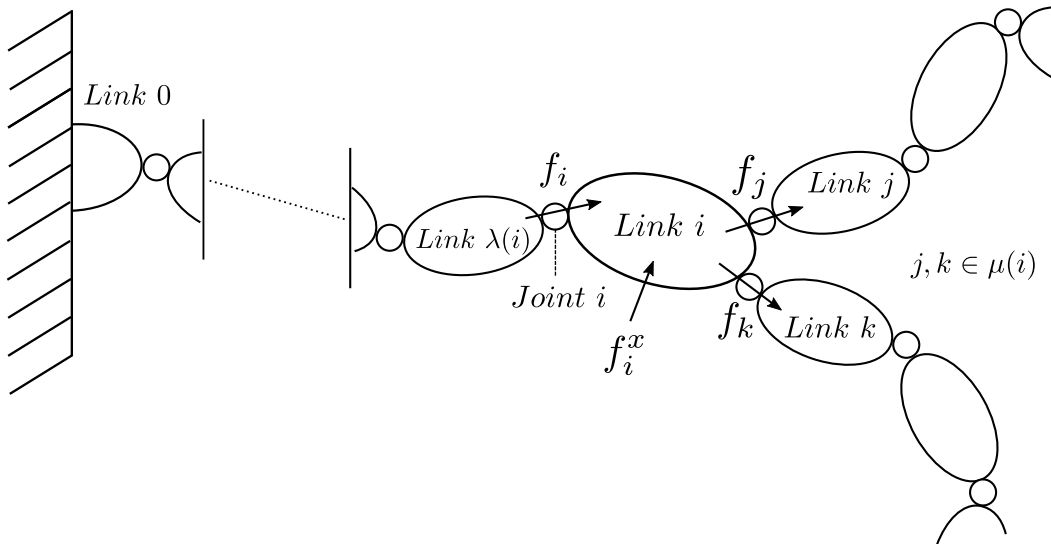
Fig. 2.2 Kinematic tree representation of an articulated rigid multi-body system.

Consider a robotic system that consists of $N_B \in \mathbb{N}$ (≥ 2) links. One of the links is consider to be a *base link* that is typically associated with the topological numbering of 0. Let i be the index of a generic link in the tree such that $1 < i < N_B$. The node numbers in the kinematic tree are always selected in a topological order, such that the i-th link has a higher number than its unique parent link $\lambda(i)$ and a smaller number than all the link in the set of its children $\mu(i)$.

The i-th link and its parent $\lambda(i)$ are coupled through joint i (Denavit and Hartenberg, 1955). The degrees of freedom (DoF) of the i-th joint is denoted as $n_i \in \mathbb{N}$ ($0 < n_i < 6$) and its motion subspace is denoted as $\bar{S}_i \in \mathbb{R}^{6 \times n_i}$. The total number of degrees of freedom present in the articulated systems is denoted by $n = n_1 + ... + n_{N_B}$.

The articulation of a robotic system is achieved through the control (actuation) of the various joints present in the system. Depending on the nature of the joint, actuation is achieved by applying either a torque (for rotary motion) or a force (for translatory motion). The configuration of the i-th joint is typically denoted by $s_i \in \mathbb{R}^{n_i}$ where n_i is the number of DoF of the i-th joint. The actuation of the i-th joint is typically denoted by $\tau_i \in \mathbb{R}^{n_i}$. The variable q_i is referred to as the *generalized coordinate* of the mechanical system and the topology of the entire system is represented by a vector s as,

$$s = \begin{bmatrix} s_1^\top & s_2^\top, & ..... & s_n^\top \end{bmatrix}^\top \in \mathbb{R}^n \tag{2.30}$$

Traditional robotic manipulators as shown in Figure. 2.3a are considered to be *fixed base systems* (Siciliano et al., 2008). It means that the base link of the robot is rigidly fixed to the ground and it does not move with respect to the inertial frame of reference. So, it is a general practice to consider the coordinate frame attached to the base link as the inertial frame of reference. On the contrary, robotic systems like *Humanoids* are designed on the principles of anthropomorphism to facilitate human like navigational capabilities. On such robotic systems, none of the links are assumed to have an *a-priori* pose with respect to the inertial frame of reference including the base link as shown in Figure. 2.3b. Such systems are referred to as *floating base systems* or *free-floating systems* (Siciliano et al., 2008) and the coordinate frame attached to the base link is referred to as *floating base frame* denoted by $\mathcal{F}$. So, to refer to a floating base robotic systems, the pose and velocity of the base link are also an integral part at the modeling level and are considered as generalized coordinates. The mechanical systems considered in this thesis are assumed to be floating base systems.
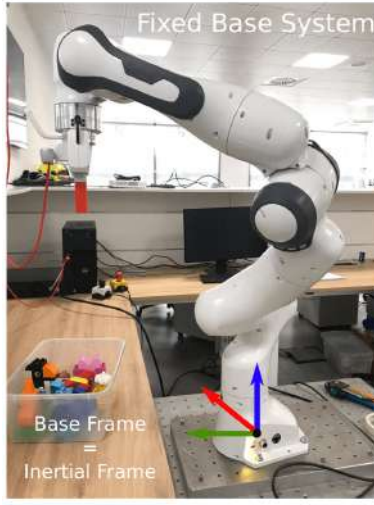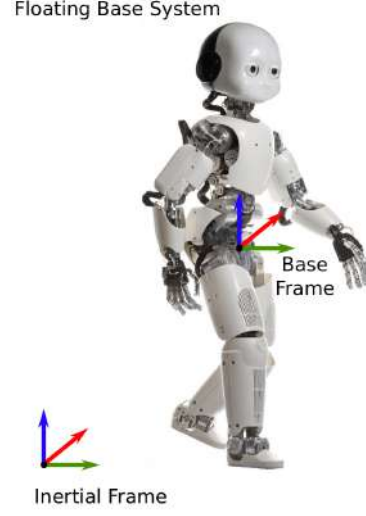
(a) Fixed Base System   (b) Floating Base System

Fig. 2.3 Examples of fixed base and floating base mechanical system

System Configuration

The configuration space of a *free-floating* mechanical system is characterized by the *floating base frame* $\mathcal{F}$ and the *joint positions*. It is defined as a set of elements with 6 dimensions representing the *floating base* pose and the total number of joints n. Hence, it lies on the Lie group $\mathbb{Q} = \mathbb{R}^3 \times SO(3) \times \mathbb{R}^n$.

An element in the configuration space is denoted by $\boldsymbol{q} = (\boldsymbol{q}_{\mathcal{B}}, \boldsymbol{s}) \in \mathbb{Q}$, which consists of the pose of the floating base frame $\boldsymbol{q}_{\mathcal{B}} = ({}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{F}}, {}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{F}}) \in \mathbb{R}^3 \times SO(3)$ where ${}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{F}} \in \mathbb{R}^3$ denotes the position of the floating base frame $\mathcal{F}$ with respect to the inertial frame $\mathcal{I}$; ${}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{F}} \in SO(3)$ denotes the rotation matrix representing the orientation of the floating base frame $\mathcal{F}$ with respect to the inertial frame $\mathcal{I}$; and the joint positions vector $\boldsymbol{s} \in \mathbb{R}^n$ captures the topology of the mechanical system.

System Velocity

The system velocity is characterized by the *linear and angular velocity* of the floating base frame along with the *joint velocities*. The configuration velocity space lies on the Lie group $\mathbb{V} = \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}^n$ and an element $\nu \in \mathbb{V}$ is defined as $\nu = ({}^{\mathcal{I}}\boldsymbol{v}_{\mathcal{F}}, \dot{\boldsymbol{s}})$ where ${}^{\mathcal{I}}\boldsymbol{v}_{\mathcal{F}} = ({}^{\mathcal{I}}\dot{\boldsymbol{p}}_{\mathcal{F}}, {}^{\mathcal{I}}\boldsymbol{\omega}_{\mathcal{F}}) \in \mathbb{R}^6$ denotes the linear and angular velocity of the floating base frame $\mathcal{F}$ expressed with respect to the inertial frame $\mathcal{I}$, and $\dot{\boldsymbol{s}} \in \mathbb{R}^n$ denotes the joint velocities.

2.5.2 Equations of Motion

A multi-body mechanical system with links connected through joints is a dynamical system that evolves in time through articulation. Joint actuation is one of the most relevant and prominent ways to produce a dynamic motion in a mechanical system. The dynamics of a mechanical system is typically described using a set of nonlinear, second-order, ordinary differential equations. The *state* of the system is composed of system configuration $\boldsymbol{q}$ and the system velocity ν as described in section 2.5.1. Two prominent representations of the dynamics of a rigid multi-body system are 1) Newton-Euler representation and 2) Euler-Poincaré Representation.

Newton-Euler Representation

Similar to the application of Newton-Euler representation for the equations of motion for a rigid body explained in section 2.4, we can express the dynamics of a rigid multi-body system based on the balance of forces acting on each of the rigid bodies present in the system. This approach is referred in robotics literature as the Recursive Newton-Euler Algorithm (RNEA) (refer Chapter 3 of (Featherstone, 2007)). Under the Newton-Euler representation, all the quantities, except the net external forces, are expressed with respect to the body frame. The external forces are expressed with respect to the inertial frame of reference $\mathcal{I}$.

Considering the modeling of a rigid multi body system presented in section 2.5, the velocity of the i-th link and the velocity of the i-th joint are defined recursively as,

$$\boldsymbol{v}_{Ji} \;=\; \bar{\boldsymbol{S}}_i \dot{\boldsymbol{q}}_i \tag{2.31}$$

$$\boldsymbol{v}_i \;=\; {}^i\boldsymbol{X}_{\lambda(i)} \boldsymbol{v}_{\lambda(i)} + \boldsymbol{v}_{Ji} \tag{2.32}$$

The acceleration of the i-th link is defined as,

$$\boldsymbol{a}_i \;=\; {}^i\boldsymbol{X}_{\lambda(i)} \boldsymbol{a}_{\lambda(i)} + \bar{\boldsymbol{S}}_i \ddot{\boldsymbol{q}}_i + \boldsymbol{v}_i \times \boldsymbol{v}_{Ji} \tag{2.33}$$

Equations (2.31), (2.32) and (2.33) are propagated throughout the kinematic tree with the initial boundary conditions $\boldsymbol{v}_0 = \boldsymbol{0}$ and $\boldsymbol{a}_0 = -\boldsymbol{g}$, which corresponds to

the gravitational acceleration vector expressed in the body frame 0, such that $\boldsymbol{g} = \begin{bmatrix} 0 & 0 & -9.81 & 0 & 0 & 0 \end{bmatrix}^\top$.

The net 6D force $\boldsymbol{f}_i^{\mathcal{B}}$ acting on the i-th link is related to the rate of change of momentum as described in Eq. (2.28)

$$\boldsymbol{f}_i^{\mathcal{B}} \;=\; \boldsymbol{I}_i \boldsymbol{a}_i + \boldsymbol{v}_i \times^* \boldsymbol{I}_i \boldsymbol{v}_i \tag{2.34}$$

The internal forces and moments transmitted through the i-th joint are denoted by the 6D force vector $\boldsymbol{f}_{Ji}$ and all the external forces and moments acting on the i-th link are denoted by the 6D force vector $\boldsymbol{f}_i^x$. Now, the balance of forces is give by the following relation,

$$\boldsymbol{f}_{Ji} \;=\; \boldsymbol{f}_i^{\mathcal{B}} - {}^i\boldsymbol{X}_0^* \, \boldsymbol{f}_i^x + \sum_{j \in \mu(i)} {}^i\boldsymbol{X}_j^* \, \boldsymbol{f}_{Jj} \tag{2.35}$$

On computing the forces and moments across each joint, the joint torques are computed through the joint motion subspace using the following relation,

$$\boldsymbol{\tau}_i \;=\; \bar{\boldsymbol{S}}_i^T \, \boldsymbol{f}_{Ji} \tag{2.36}$$

Euler-Poincaré Representation

Another most commonly used approach to derive the dynamic equations of motion of a mechanical systems is *Lagrangian Analysis*, that relies on the energy properties of the mechanical system under consideration. The detailed derivation is available in several robotics text books like (Siciliano et al., 2008), Chapter 7 or (Khalil and Dombre, 2004), Chapter 9. Most of the textbooks present the Euler-Lagrangian equations of motion for a fixed base mechanical systems. However, in this thesis, the mechanical systems considered are assumed to be a floating base systems and hence the equations of motions for such systems are slightly different than a fixed base system. Specifically, Euler-Poincaré equations (Marsden and Ratiu, 2010) describe the equations of motion for a floating base system. The reader is advised to refer to Chapter 3 of (Traversaro, 2017) for a detailed derivation of the equations of motion for a floating base system.

The equations of motion of a *floating base* mechanical system are described by,

$$\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{\nu}} + \boldsymbol{C}(\boldsymbol{q},\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{B}\boldsymbol{\tau} + \sum_{i=1}^{n_c} \boldsymbol{J}_{c_i}^{\top}\, \boldsymbol{f}_i \qquad (2.37)$$

where, $\boldsymbol{M} \in \mathbb{R}^{(n+6)\times(n+6)}$ is the mass matrix, $\boldsymbol{C} \in \mathbb{R}^{(n+6)\times(n+6)}$ is the Coriolis matrix, $\boldsymbol{G} \in \mathbb{R}^{n+6}$ is the gravity term, $\boldsymbol{B} = (\boldsymbol{0}_{n\times 6}, \boldsymbol{I}_n)^T$ is a selector matrix, $\boldsymbol{\tau} \in \mathbb{R}^n$ is a vector representing the joint torques, $\boldsymbol{f}_i \in \mathbb{R}^{6n_c}$ represents the 6D forces acting on i-th contact link expressed with respect to the body frame, and $\boldsymbol{J}_{c_i} \in \mathbb{R}^{(n+6)\times 6n_c}$ is the i-th contact jacobian.

3

Recall on Human Modeling

Digital Human Models (DHM) are becoming increasingly popular as enabling technologies in several domains. This chapter presents the significance of a faithfully articulated model for a human partner across different domains and lays the considerations we made in our human modeling aimed towards the goal of human-robot collaboration. At the modeling level, we consider the human partner to be a mechanical system of rigid bodies. More specifically, we consider the human to be a floating base mechanical systems.

3.1 Digital Human Modeling

A faithfully articulated full body human model is very useful across various domains (Alami et al., 2006; Duffy, 2016; Guo et al., 2020; Xiang et al., 2010). Human bodies are anatomically very complex that consists of bones, joints, muscles, tendons and skin. Capturing all of the details for digital representation is an enormous endeavor. On one hand, graphics community uses *virtual humans* to do character animation for entertainment or interactive virtual reality applications. On the other hand, biomechanical community uses human models to understanding the biomechanics of complex human movement. Furthermore, there is a growing interest in the human-robot interaction community towards using digital human model abstractions (Alami, 2013; Saikia and Hazarika, 2017).

An established standard for virtual humans in graphics industry is called *H-Anim* (Humanoid Animation) (H-Anim, 2006) which is a hierarchical representation of the human skeletal structure. A simplest skeletal structure is composed of joints and the bones. Naturally, all the joints are considered to be rotational type and the change in their configuration modifies the pose of various bones in the skeletal structure. So, the joints are the basis for the articulation of a skeleton. *Levels of Articulation* (LOA)

refers to the number of joints considered for the skeleton. Figure. 3.1 highlights the joints that are considered for Level of Articulation 1 (LOA1) in H-Anim standard.
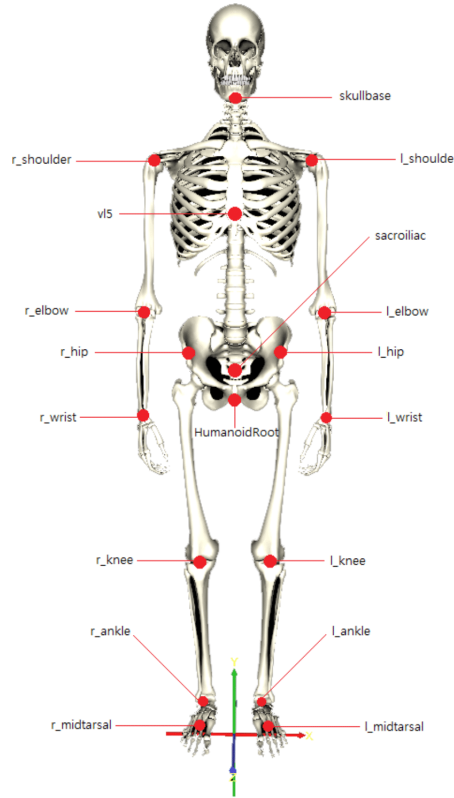
Fig. 3.1 Joints considered for Level of Articulation 1 (LOA1) in Humanoid Animation (H-Anim) standard (H-Anim, 2006). Skeletal model in anatomically neutral N-pose.

Coming to the field of biomechanics, the human model is more holistic with the inclusion of muscles and tendons. The articulation of the skeleton is achieved through the muscle activation rather than using the joints directly. Motions capture systems are an invaluable asset that captures motion data that can be used along with a complex human model for detailed simulation and analysis of anatomically accurate models.

Most commercial motion capture systems often define their models that facilitate motion data in a clinically meaningful manner without the full complexity of the human model. The models they employ are similar to the graphics community as a way to provide a real-time visualization of the human motion projected to a virtual avatar. However, they often adhere to International Society of Biomechanics (ISB) standards for their modeling. One such motion capturing system we consider as reference for our human model is Xsens MVN motion capture system (Roetenberg et al., 2009). The kinematic model consists of a total of 23 body segments whose names are as

indicated in Figure. 3.2a and the frames associated with each segment are as shown in Figure. 3.2b.

(a) Segment names

(b) Segment frame definitions when the model is in T-pose
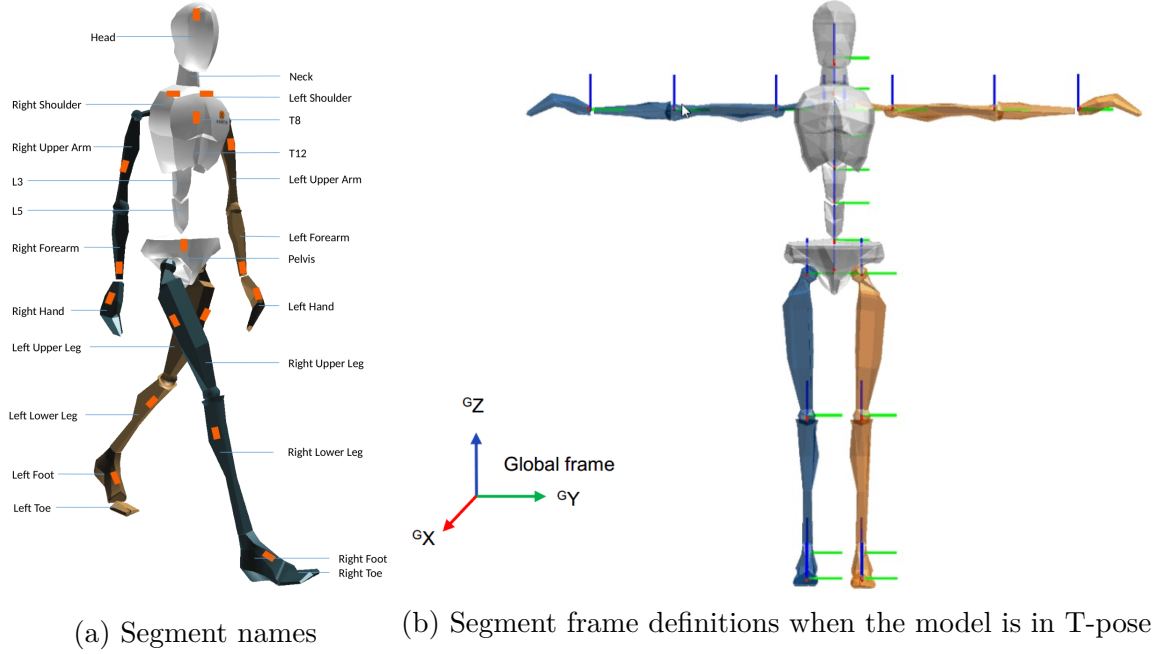
Fig. 3.2 The definition of the 23 segments in the kinematic model of Xsens MVN motion capturing system (Karatsidis et al., 2017)

In this thesis, the human body is assumed to be composed of a set of rigid bodies for various body segments. Following the details presented in Section 2.5, we model whole human body as a rigid multi-body system. A total of 23 rigid bodies or links constitute our human model i.e., $N_B = 23$. This choice is motivated by the use of Xsens MVN motion capture system at the initial stages of our research.

3.2 Link Modeling

The names associated with the links in our model are consistent with that of the Xsens model. Unlike graphical virtual humans that are in general very detailed from the level of facial expressions to the fine articulation at the hands, we are primarily interested in whole-body motion and dynamics of humans. So, the *geometry* of the links considered are very simple when compared to the sophisticated virtual humans used in graphics community.

3.2.1 Geometry

The main shapes we considered for the links are *parallelepiped boxes*, *cylinders* and *spheres*. This choice is motivated by our interest in representing the human model close to an animated avatar rather than a simple multi rigid-body structure. The dimensions of each of the body segment are obtained through a calibration procedure that is capable of scaling based on the dimensions of the actual human being we want to model. We rely on an anthropomorphic reference (Winter, 1990) that attributes a certain percentage of the total human mass to different body segments. Table 3.1 highlights the link names (similar to the Xsens model), link geometry considered and the link mass in terms of percentage of the total mass of the human.

Table 3.1 Link names, link geometry and the link mass in terms of percentage of the total mass of the human subject (extracted from (Winter, 1990))

Label	Shape	% Mass of human
Pelvis	parallelepiped	0.08
L5	parallelepiped	0.102
L3	parallelepiped	0.102
T12	parallelepiped	0.102
T8	parallelepiped	0.04
Neck	cylinder	0.012
Head	sphere	0.036
RightShoulder	cylinder	0.031
RightUpperArm	cylinder	0.030
RightForeArm	cylinder	0.020
RightHand	parallelepiped	0.006
LeftShoulder	cylinder	0.031
LeftUpperArm	cylinder	0.030
LeftForeArm	cylinder	0.020
LeftHand	parallelepiped	0.006
RightUpperLeg	cylinder	0.125
RightLowerLeg	cylinder	0.0365
RightFoot	parallelepiped	0.013
RightToe	parallelepiped	0.015
LeftUpperLeg	cylinder	0.125
LeftLowerLeg	cylinder	0.0365
LeftFoot	parallelepiped	0.013
LeftToe	parallelepiped	0.0015

3.2.2 Inertial Properties

Inertial properties of the body segments i.e., mass, center of mass and moments of inertial are important to understand and capture the dynamics of the human. A popular approach for human body segments inertial properties estimation is based on *geometric approximation* where a collection of measurements from the human are mapped to a geometric model to derive the segments inertial properties based on the geometry and density assumption. Unlike muscles of a human body that generally vary in density, we assumed density isotropy for all the body segments (Hanavan, 1964). Under these assumptions, the inertial tensor $\boldsymbol{I}$ is computed as,

$$\boldsymbol{I} = \begin{bmatrix} \mathrm{I}_{xx} & 0 & 0 \\ 0 & \mathrm{I}_{yy} & 0 \\ 0 & 0 & \mathrm{I}_{zz} \end{bmatrix} \tag{3.1}$$

where I_{xx}, I_{yy} and I_{zz} are the principal moments of inertia. Table 3.2 lists analytical formulas for the principal moments of inertia computation.

Table 3.2 Principal moments of inertia of three different geometric shapes considered for the links with mass m: (*on left column*) a rectangular parallelepiped of width α, height β and depth γ; (*on middle column*) a circular cylindrical of radius r and height h; (*on right column*) a sphere of radius r

Inertia	Parallelepiped	Cylinder	Sphere
I_{xx}	$\frac{1}{12}\mathrm{m}\left(\alpha^2 + \beta^2\right)$	$\frac{1}{12}\mathrm{m}\left(3r^2 + h^2\right)$	$\frac{2}{5}\mathrm{m}r^2$
I_{yy}	$\frac{1}{12}\mathrm{m}\left(\beta^2 + \gamma^2\right)$	$\frac{1}{2}\mathrm{m}r^2$	$\frac{2}{5}\mathrm{m}r^2$
I_{zz}	$\frac{1}{12}\mathrm{m}\left(\gamma^2 + \alpha^2\right)$	$\frac{1}{12}\mathrm{m}\left(3r^2 + h^2\right)$	$\frac{2}{5}\mathrm{m}r^2$

3.3 Joint Modeling

The links are coupled through joints and as with the link names, we use the same joint names as that of the Xsens model. Typically, a rotational joint like the shoulder is usually composed of three degrees of freedom (DoF) rotation, although some anatomical joints like elbow, knee and finger joints are less than three degrees of freedom rotation. For the sake of generality, we assumed all the joints present in our model to be a three

degrees of freedom joints. Given a total of 23 links considered in our model, a total of 22 three degree of freedom rotational joints are present in our model. In other words, the total number of internal degrees of freedom are $n = 22 \times 3 = 66$. A scaled down version of our model consists of only $n = 48$ degrees of freedom, by setting some of the joints to be of two degrees of freedom or one degree of freedom. Table 3.3 highlights the joint names, total degrees of freedom of a joint with the reduced DoF in parenthesis and the links connected through a joint.

Table 3.3 Joint names, DoFs per each joint (reduced DoFs) and the links connected through a joint

Label	DoF	Connected links		
jL5S1	3 (2)	Pelvis	$\longleftrightarrow$	L5
jL4L3	3 (2)	L5	$\longleftrightarrow$	L3
jL1T12	3 (2)	L3	$\longleftrightarrow$	T12
jT9T8	3	T12	$\longleftrightarrow$	T8
jT1C7	3	T8	$\longleftrightarrow$	Neck
jC1Head	3 (2)	Neck	$\longleftrightarrow$	Head
jRightHip	3	Pelvis	$\longleftrightarrow$	RightUpperLeg
jRightKnee	3 (2)	RightUpperLeg	$\longleftrightarrow$	RightLowerLeg
jRightAnkle	3	RightLowerLeg	$\longleftrightarrow$	RightFoot
jRightBallFoot	3 (1)	RightFoot	$\longleftrightarrow$	RightToe
jLeftHip	3	Pelvis	$\longleftrightarrow$	LeftUpperLeg
jLeftKnee	3 (2)	LeftUpperLeg	$\longleftrightarrow$	LeftLowerLeg
jLeftAnkle	3	LeftLowerLeg	$\longleftrightarrow$	LeftFoot
jLeftBallFoot	3 (1)	LeftFoot	$\longleftrightarrow$	LeftToe
jRightC7Shoulder	3 (1)	T8	$\longleftrightarrow$	RightShoulder
jRightShoulder	3	RightShoulder	$\longleftrightarrow$	RightUpperArm
jRightElbow	3 (2)	RightUpperArm	$\longleftrightarrow$	RightForeArm
jRightWrist	3 (2)	RightForeArm	$\longleftrightarrow$	RightHand
jLeftC7Shoulder	3 (1)	T8	$\longleftrightarrow$	LeftShoulder
jLeftShoulder	3	LeftShoulder	$\longleftrightarrow$	LeftUpperArm
jLeftElbow	3 (2)	LeftUpperArm	$\longleftrightarrow$	LeftForeArm
jLeftWrist	3 (2)	LeftForeArm	$\longleftrightarrow$	LeftHand

3.4 Model Representation

Both the graphics and biomechanics communities have an active developer base whose contributions lead to tools with many advanced features. Blender (Blender, 2019) is a free and open-source 3D computer graphics software used for character animation. COLLADA (with a .dae extension) is one of the most widely used file formats for sharing graphics related models. Typically, along with the geometry of the model, COLLADA facilitates storing appearance, scene and animation information that are essential for character animation. OpenSim (Delp et al., 2007) is an open source software for biomechanical modeling, simulation and analysis. Opensim Model (with .osim extension) is the file format used for sharing an anatomically accurate human model. In our observation we noticed that COLLADA and OpenSim Model formats are not compatible with each other as the purposes they serve are not overlapping.

Our main motivation behind developing a human model is towards facilitating human-robot interaction and human-robot collaboration research. Lately, *Robot Operating System* (ROS) (Robotic-Operating-System, 2018) emerged as the most popular open-source middle-ware for robotics research. Unified Robot Description Format (URDF) (with .urdf extension) is an Extensible Markup Language (XML) schema based file format that is widely adopted to represent robot models. URDF models are not as anatomical accurate as OpenSim models or as detailed as COLLADA models, but they capture all the details needed to perform the simulation and analysis of rigid multi-body systems. So, a natural choice for us to represent our human model is the URDF format. Currently, the process of generating a human URDF model relies on the Xsens motion capture data during the calibration procedure (Latella, 2018). Human models of subjects with varying body dimensions are available as an open source repository `human-gazebo`[1].

---

[1] https://github.com/robotology/human-gazebo
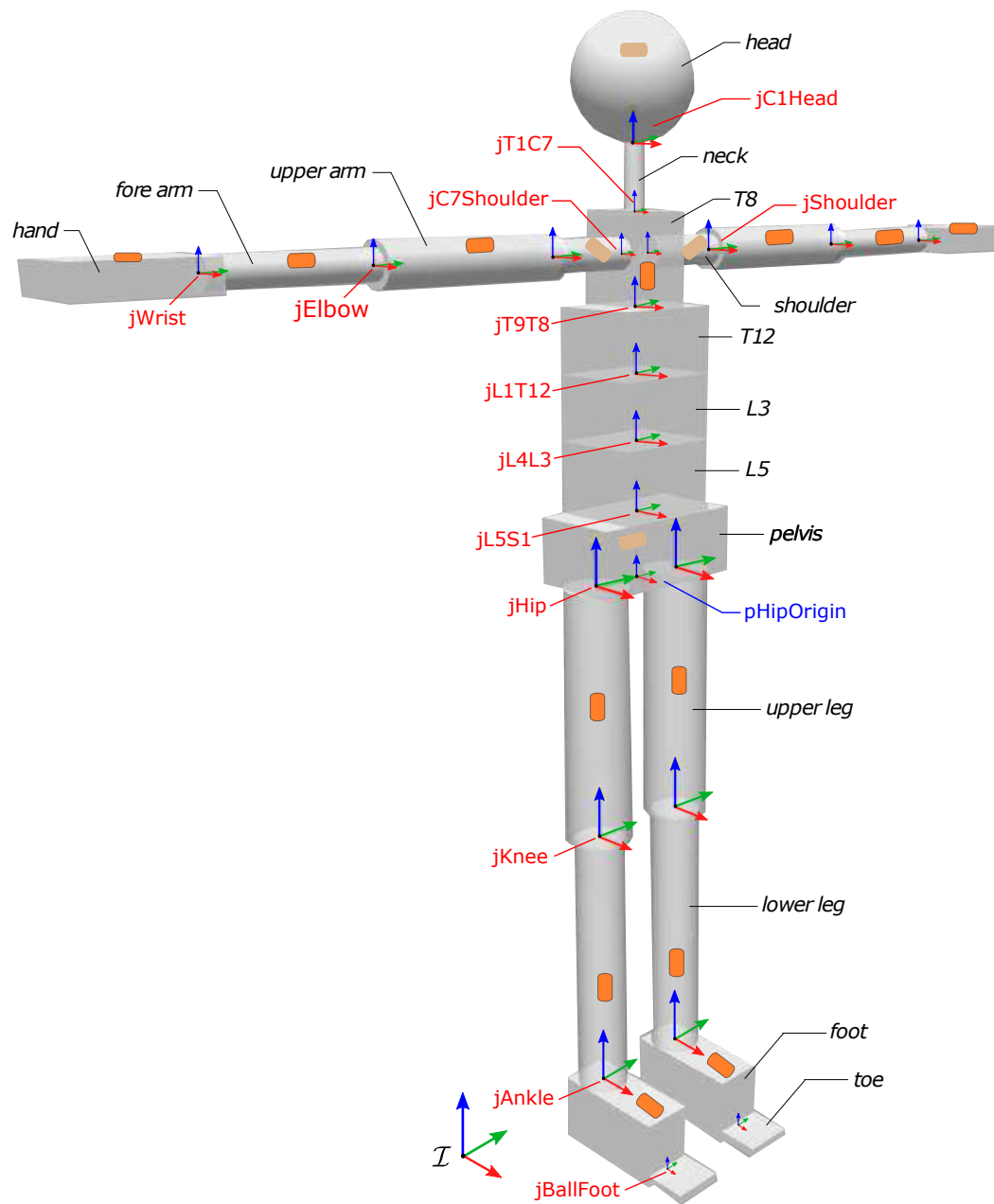
Fig. 3.3 URDF model of human body

4

Enabling Technologies

The outcome of any scientific endeavor depends significantly on the use of proper tools to address the research questions and accomplish the set objectives. This chapter presents the enabling technologies that serve as the main tools for building a holistic human perception framework and evaluating our partner-aware robot control techniques towards realizing our research goals of enabling human-robot collaboration.

4.1 Human Motion Perception

Human motion tracking is used in a variety of applications like gaming, animation, virtual reality etc. Different representations of human body with different level of complexity from contours (Leung and Yee-Hong Yang, 1995; Shio and Sklansky, 1991), stick figure (Bharatkumar et al., 1994; Niyogi and Adelson, 1994), and volumes (Gall et al., 2009; Wachter and Nagel, 1999) are investigated in literature. Motion tracking through optical tracking technology is one of the most mature fields with it roots starting from over two decades (Aggarwal and Cai, 1999; Wagh and Kanade, 2019). Several motion capture systems based on reflective markers also exists as commercial products such as Vicon and Qualisys. However, they impose space constraints and are not robust to occlusions.

Inertial tracking technologies quickly rose to popularity as they provide a convenient alternative to human motion tracking without any space constraints. Furthermore, they ensure low latency that is crucial for real-time applications (Zhu and Zhou, 2004). So, in this research, we chose to employ inertial tracking technology over optical tracking technology. At the time of this writing, we relied on Xsens MVN inertial tracking system (Roetenberg et al., 2009), a commercial platform of multiple inertial measurement units (IMUs) distrusted over the entire human body as shown in Figure. 4.1.

Fig. 4.1 Xsens whole-body motion tracking suit with distributed IMUs present at the locations of the green dots

As discussed in the human modeling chapter 3, the kinematic model considered by Xsens system is composed of 23 body segments. So, the motion capturing system provides, the pose of a body segment with respect to the inertial frame of reference. Additionally, one can also access the linear and angular velocity of a body segment, and the *sensor acceleration* (Refer to Chapter 2 of (Traversaro, 2017)). Although the joint configuration values are accessible, they are pertinent to the kinematic model considered by Xsens system. To reduce our reliance on the kinematic model of a commercial platform and have more control on our human model, we do not make use of the joint configuration data. Instead, we employ the motion measurement data in terms of link pose and velocity, and developed our inverse kinematic algorithms.

4.2 Force-Torque Sensing

Six axis force-torque (F/T) sensing is an important capability that has been exploited in robotics for regulating contact forces and torques (Siciliano and Villani, 2012), joint

torque estimation and whole-body control of complex humanoid robots (Nori et al., 2015b). Also, force-torque sensing is an invaluable tool in understanding human dynamics (Riemer and Hsiao-Wecksler, 2008; Skals et al., 2017). Strain gauge-based sensing is one of the most popular approaches. In the case industrial robots force-torque sensors are embedded at the end-effector and provides reliable measurements. However, in the case of complex humanoid robots, the reliability of the measurements deteriorate and procedures for quick and reliable calibration are highly important (Chavez et al., 2016). Coming to the case human dynamics analysis, ground fixed force plates provide very reliable and accurate information of ground reaction forces. However, as they are ground fixed, they limit the mobility of the human subject. This limits the use for force plates for tasks that consider the human model to be a floating base system.

We rely on the six axis force-torque sensor technology[1] developed at the Istituto Italiano di Tecnologia (IIT) shown in Fig. 4.2a.

(a)   (b)   (c)   (d)   (e)
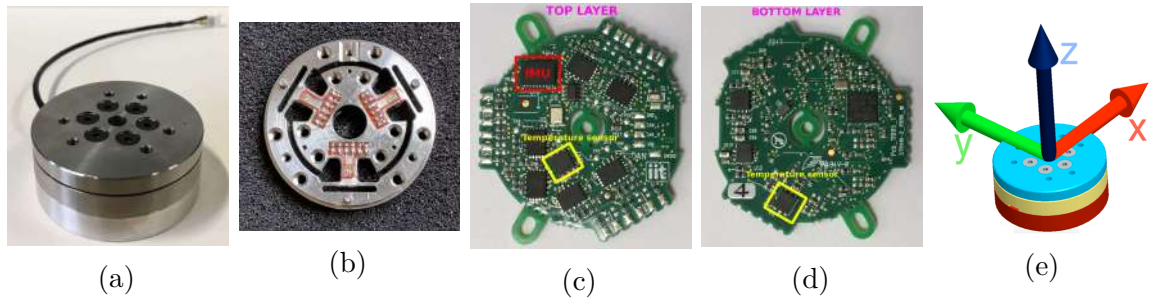
Fig. 4.2 (a) A six axis force-torque sensor developed at the Istituto Italiano di Tecnologia (IIT); (b) Strain gauge; (c) Top layer of internal electronics showing an inertial measurement unit (IMU) and a temperature sensor; (d) Bottom layer of internal electronics showing a temperature sensor; (e) Sensor reference frame;

4.3 Ground Reaction Force Monitoring

The six axis force-torque sensor technology presented in section 4.2 are small and provide reliable sensor measurements with proper calibration procedures (Chavez et al., 2016). Leveraging this sensor technology, we developed sensorized wearable shoes as shown in Fig. 4.3. They will be referred to as *ftShoes* and they do not pose the limitations of the force plates on human mobility. Each shoe is equipped with two force-torque sensors, one at the front and the other at the rear, as shown in Figure. 4.3. The sensor measurements from the front and the read sensors are combined and expressed

---

[1] http://wiki.icub.org/wiki/FT_sensor

with respect to the human foot frame that is shown in Figure. 4.3. Unlike the ground
fixed force platform, the *ftShoes* enable real-time ground reaction forces monitoring of
human subjects while doing dynamic tasks like walking.
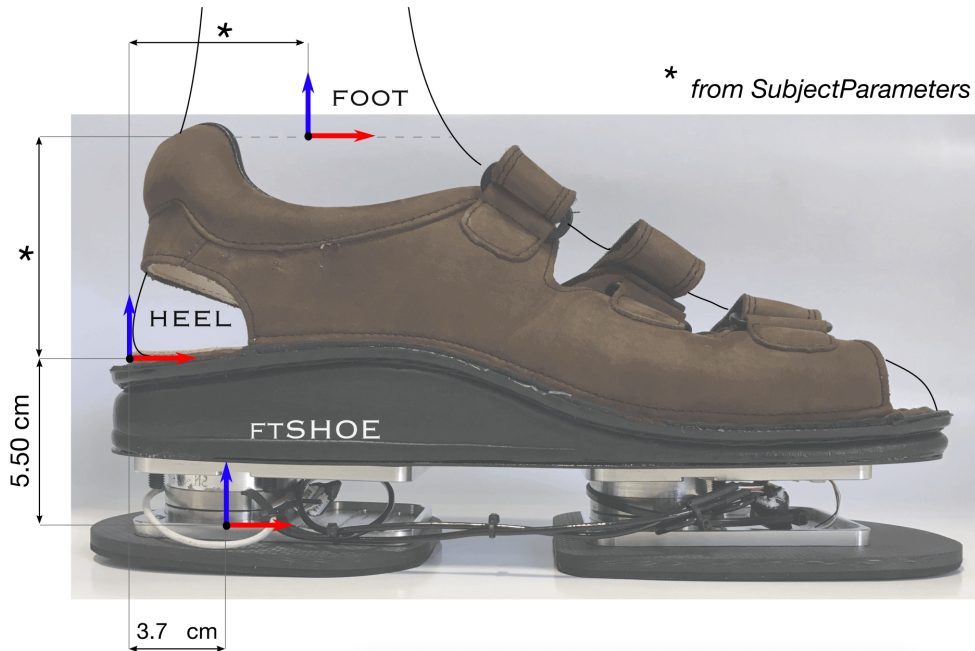
Fig. 4.3 Sensorized wearable shoe with two force-torque sensors placed at the front and
the rear, with frame references for measurements transformation from the sensors to
the human foot

4.4 iCub Humanoid Robot

The iCub[2] humanoid robot is the main robotic platform used for experiments
presented in this thesis. It is developed by the iCub Facility at the Italian Institute
of Technology (IIT). It is a child-sized humanoid robot originally developed by the
RobotCub[3] European Project for research in embodied cognition (Sandini et al., 2004).
Since its initial release in 2006, the platform has been continuously updated with
latest improvements and features both in terms of hardware and software. This
section introduces the key details concerning the latest "standard" version of the iCub,
informally referred hereafter as iCub 2.5, as of the beginning of 2017.

---

[2] http://www.icub.org/
[3] http://www.robotcub.org/

The iCub platform consists of 53 degrees-of-freedom (DOF) that are distributed as following: 3 for the head and 3 for the eyes; 7 for each arm and 9 for each hand; 3 for the torso and 6 for each leg. The actuation is achieved using Brushless DC electric motor (BLDC) with an Harmonic Drive transmission, making them suitable for joint torque control. More details on the actuation and mechanics of the iCub 2.5 can be found in (Parmiggiani et al., 2012).

Fig. 4.4 iCub humanoid robot (version 2.5)

Traditional robotic manipulators are fixed-base systems meaning that one of the known links of the robot is always in rigid contact with the environment. Alternatively, humanoid robots are floating-base systems (Featherstone, 2007) meaning that the contact information is typically not known a-priori. It is a direct reflection of their design to facilitate anthropomorphic navigational capabilities. One of the research goals of the iCub project is to endow humanoids with advanced physical interaction capabilities. This is motivated by the idea that future robots will be required to physically interact with the environment and, in the long run, with humans or other robotic agents. A key requirement for this interaction control is the capability to measure and control the forces that a humanoid robot is exchanging at its contacts, i.e. contact forces control. While in traditional industrial applications this is achieved by

placing a six axis force-torque sensor between the robot and the environment, this is not feasible in humanoid robots, in which the contact location is typically not known a-priori. To overcome this limitations, a unique set of dynamics-related sensors have been added during the years to the iCub.

(a)   (b)
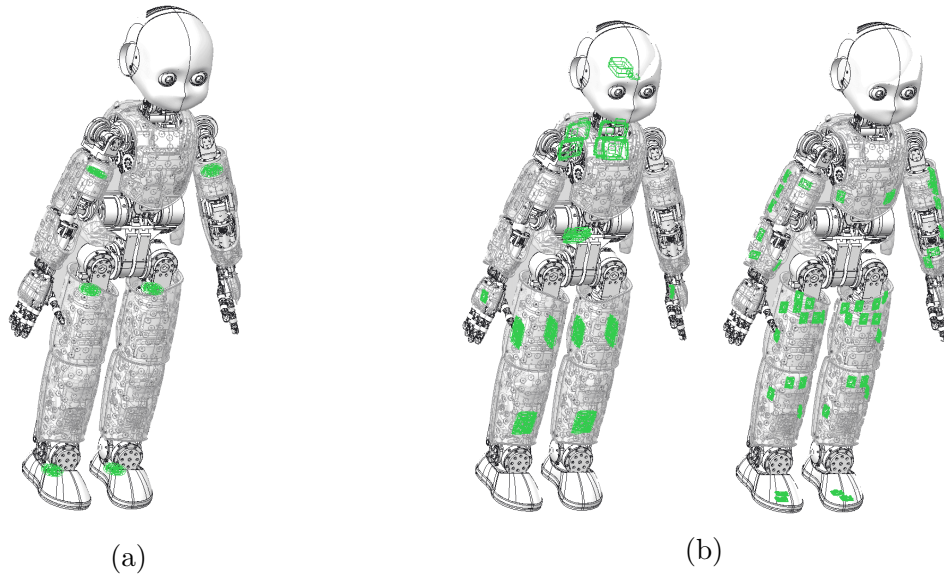
Fig. 4.5 Distribution of a) six embedded six axis force-torque sensors b) inertial sensors i.e. gyroscopes (left) and accelerometers (right) in iCub 2.5.

The main force sensors available on the iCub are six internal six axis force-torque sensors. Four of them are mounted at the base of each limb while two of them are mounted in feet right below the ankles. The locations of these sensors are highlighted in Figure 4.5a. iCub has a full-fledged Inertial Measurement Unit, equipped with a 3 DOFs magnetometer, accelerometer and gyroscope, that is mounted on the head of the robot. Additionally, several motor control boards are distributed in the robot structure, and each motor control board is equipped with both a 3 DOFs accelerometers and a 3 DOFs gyroscopes. Furthermore, some dedicated electronic boards to read distributed tactile sensors are equipped with a 3 Degree-of-Freedom (DOFs) accelerometers. These distributed inertial sensing is highlighted in Figure 4.5b.

The availability of these distributed sensors, coupled with a novel modeling formalism and estimation algorithms (Andrade Chavez et al., 2019; Nori et al., 2015a; Traversaro, 2017; Traversaro et al., 2015) enabled the development of whole-body controllers for the iCub humanoid robot capable of performing highly dynamic movements while balancing (Dafarra et al., 2016; Nori et al., 2015b; Pucci et al., 2016a).

PART II:

HOLISTIC HUMAN PERCEPTION

Human Kinematics Estimation

The problem of inverse kinematics is well addressed in robotics literature. This chapter presents a brief introduction to the problem of inverse kinematics in robotics and highlights the role of inverse kinematics in the context of real-time human motion tracking through the kinematics estimation of a human wearing a whole-body inertial motion tracking system. A description of two instantaneous optimization approaches and a dynamical optimization approach to real-time human motion tracking is presented with a benchmarking in terms of the computation time.

Kinematics deals with the description of the motion of a multi rigid-body system without considering the influence causing the motion or the inertial parameters of rigid bodies. A multi rigid-body system with a set of rigid bodies connected without any loops is referred as a *serial kinematic chain*. Robotic manipulators as shown in Figure. 5.1 are an ideal example of a serial kinematic chain. One of the links of the system is referred as an *End-Effector* that is present at the end of the kinematic chain.
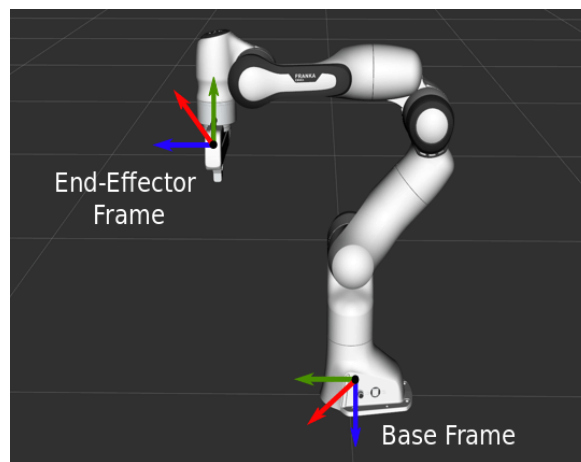
Fig. 5.1 Example of a robotic manipulator

End-effectors are designed to interact with the environment. So, the knowledge of the end-effector pose with respect to the base frame is crucial for realizing any task with the manipulator. Given the values of the joint configuration i.e., the topology s as indicated in Eq. 2.30, and the knowledge of the kinematic chain model, one can compute the end-effector pose with respect to the base. This is known as the *Forward Kinematics* (FK) problem. However, it is more intuitive to specify an interested pose for the end-effector in the Cartesian space and then compute the joint configuration that enables the end-effector placement correctly. This is referred as *Inverse Kinematics* (IK) problem. As an example, the desired pose of the end-effector along with the desired pose of the rest of the robot links is indicated in *orange* color in Figure. 5.2a. An example inverse kinematics solution of a series of joint configurations to realize the desired end-effector pose is highlighted in Figure. 5.2b.

| (a) | (b) |
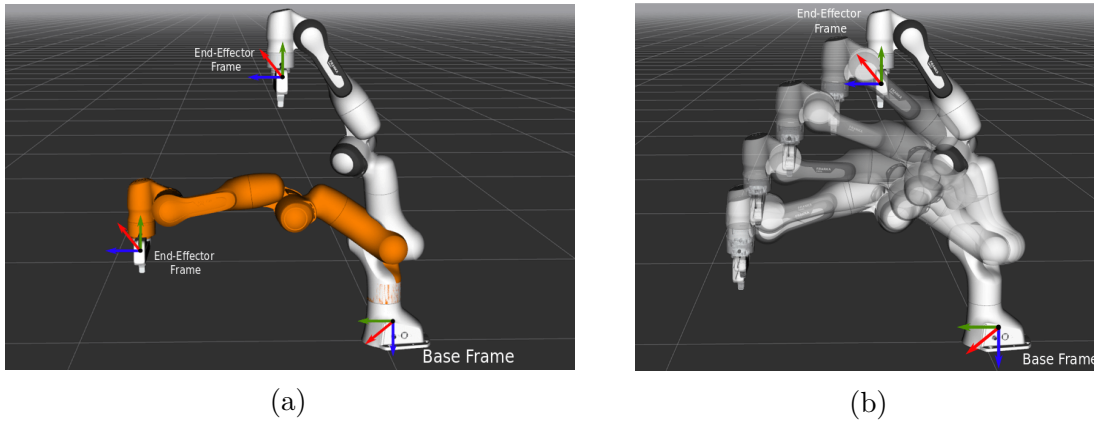
Fig. 5.2 Inverse kinematics example of a robotic manipulator

Investigations into inverse kinematics problem first originated in robotic community and significant research effort was put towards finding efficient algorithms that provide real-time performance. Apart from robotics community, IK is also a critical component in graphics community for animating articulated models (Aristidou et al., 2018) or for real-time human motion tracking (Pons-Moll et al., 2011). Analytical methods provide an exact solution and are usually faster to compute closed-form solution for mechanisms of lower degrees of freedom like robotic manipulators (Craig, 2009). However, when the considered model is highly redundant, as our human model with 66 DoFs, application of analytical methods is not feasible. In the subsequent sections we refer to inverse kinematics only in the context of human motion tracking.

5.1 Inverse Kinematics Problem Definition

This section presents the inverse kinematic problem in the context of real-time human motion tracking. A *posture* is defined as a particular configuration of human body parts that results from a particular set of human joints configuration. A human subject wearing a whole-body motion tracking suit moves various joints to attain a series of postures depending on the task. For example, when the human subject moves from an anatomically neutral *N-pose* to *T-pose*, there is a change in the configuration of the shoulder joints as shown in Figure. 5.3a. Similarly, when moving from N-pose to a squatting position, there is a change in the configuration of the knee and ankle joints as shown in Figure. 5.3b.

(a) Transition from N-pose to T-pose   (b) Transition from N-pose to Squat
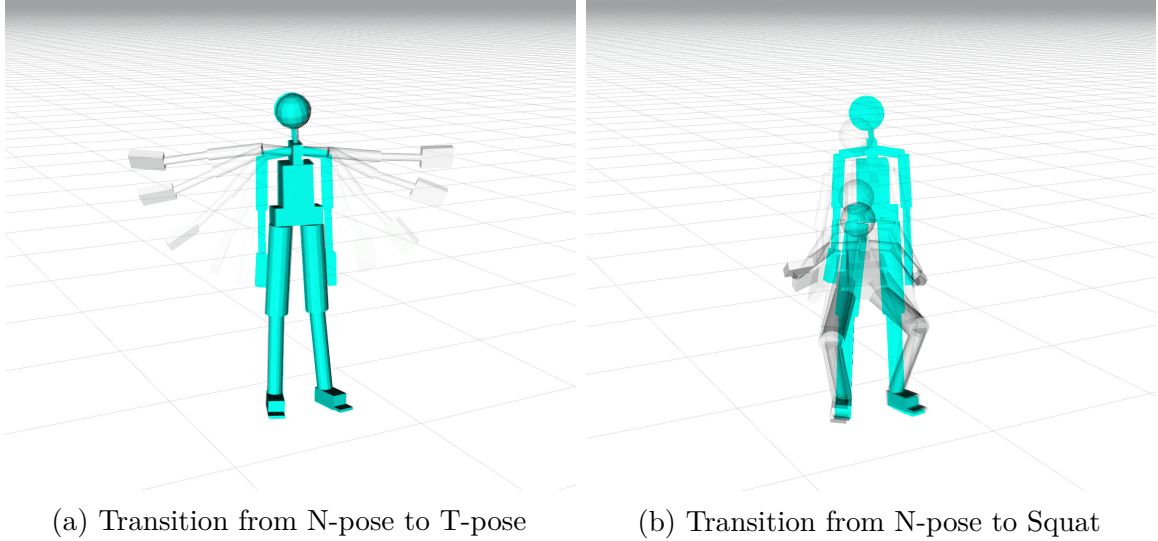
Fig. 5.3 Example of a series of human postures during different tasks

The motion tracking system provides position and orientation measurements of various body segments in real-time. These measurements are expressed with respect to the inertial frame of reference. Now, the inverse kinematics problem is defined as, given the position and orientation measurements of various body segments, we need to compute the joint configuration of the human.

The set of frames $\mathbb{L} = \{\mathcal{L}_0, \mathcal{L}_1, ...\mathcal{L}_{N_B-1}\}$ correspond to the number of body segments (links) considered in the human model with $N_B = 23$. The human model is considered as a floating base system i.e. $\mathcal{L}_1 = \mathcal{F}$. Given a system configuration at a time instant t, $\boldsymbol{q}(t) = (\boldsymbol{q}_{\mathcal{B}}(t), \boldsymbol{s}(t))$, the Cartesian pose of a link with respect to the inertial frame of reference can be computed as a combination of a series of

transformations obtained through the kinematic equations of the model. This geometric mapping through a series of transformations constitute the forward kinematics that is denoted as a function,

$$
{}^{\mathcal{I}}\boldsymbol{H}_{\mathcal{L}_i}(t) = f(\boldsymbol{q}(t)) \ \ \forall \ \ i = 1....,N_B - 1 \tag{5.1}
$$

The positions of all the links is put together as a vector denoted by,

$$
\boldsymbol{p}_{\mathbb{L}} =
\begin{bmatrix}
{}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{F}} \\
{}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{L}_1} \\
{}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{L}_2} \\
... \\
... \\
{}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{L}_{N_B-1}}
\end{bmatrix}
\in \mathbb{R}^{3N_B} \tag{5.2}
$$

The rotations of all the links is put together as a matrix denoted by,

$$
\boldsymbol{R}_{\mathbb{L}} =
\begin{bmatrix}
{}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{F}} \\
{}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{L}_1} \\
{}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{L}_2} \\
... \\
... \\
{}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{L}_{N_B-1}}
\end{bmatrix}
\in \mathbb{R}^{3N_B \times 3} \tag{5.3}
$$

The posture change of the human results in a new set of measurements for the position and orientation of the links, say at a time instant $t+1$. The new measurements are referred as *targets*. The vector of position targets is denoted as,

$$
\boldsymbol{p}_{\mathbb{L}}^d =
\begin{bmatrix}
{}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{F}}^d \\
{}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{L}_1}^d \\
{}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{L}_2}^d \\
... \\
... \\
{}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{L}_{N_B-1}}^d
\end{bmatrix}
\in \mathbb{R}^{3N_B} \tag{5.4}
$$

Similarly, the vector of rotational targets is denoted as,

$$\boldsymbol{R}_{\mathbb{L}}^{d} = \begin{bmatrix} {}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{F}}^{d} \\ {}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{L}_1}^{d} \\ {}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{L}_2}^{d} \\ ... \\ ... \\ {}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{L}_{N_B-1}}^{d} \end{bmatrix} \in \mathbb{R}^{3N_B \times 3} \tag{5.5}$$

A distance measure between the current link positions and target position is defined in terms of *Euclidean* distance as,

$$d_{pos}(\boldsymbol{p}_{\mathbb{L}}^{d}, \boldsymbol{p}_{\mathbb{L}}) = \begin{bmatrix} {}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{F}}^{d} - {}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{F}} \\ {}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{L}_1}^{d} - {}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{L}_1} \\ {}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{L}_2}^{d} - {}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{L}_2} \\ ... \\ ... \\ {}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{L}_{N_B-1}}^{d} - {}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{L}_{N_B-1}} \end{bmatrix} \tag{5.6}$$

> ### Position Targets Remark
>
> *The human model considered in this work consists of only rotational joints. So, the only relevant position target corresponds to the base frame $\mathcal{F}$.*

As orientation is parametrized in terms of rotation matrix, the distance measure is defined using the $\mathbf{S}(.)^{\vee}$ operator denoted as,

$$d_{ori}(\boldsymbol{R}_{\mathbb{L}}^{d}, \boldsymbol{R}_{\mathbb{L}}) = \begin{bmatrix} \mathbf{S}(({}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{F}}^{d})^{T}\, {}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{F}})^{\vee} \\ \mathbf{S}(({}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{L}_1}^{d})^{T}\, {}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{L}_1})^{\vee} \\ \mathbf{S}(({}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{L}_2}^{d})^{T}\, {}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{L}_2})^{\vee} \\ ... \\ ... \\ \mathbf{S}(({}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{L}_{N_B-1}}^{d})^{T}\, {}^{\mathcal{I}}\boldsymbol{R}_{\mathcal{L}_{N_B-1}})^{\vee} \end{bmatrix} \tag{5.7}$$

Now, the solution of the inverse kinematics problem is the joint configuration change that resulted in the posture change of human during a task. The following sections present different ways of formulating the inverse kinematics problem for real-time human motion tracking.

5.2 Instantaneous Optimization

Unlike analytical methods, *iterative methods* are ideal for systems with high degrees of freedom. One of the most common approaches is to formulate the inverse kinematics problem as a non-linear optimization problem and solve via iterative algorithmss (Buss, 2004; Goldenberg et al., 1985). This class of algorithms, referred to as *instantaneous optimization*, aims to converge to a stable solution for each time-step t_k. A general formulation of the optimization problem is defined as following:

$$\underset{\boldsymbol{q}(t_k)}{\text{minimize}} \qquad \left\| \boldsymbol{K}_{pos}\ d_{pos}(\boldsymbol{p}_{\mathbb{L}}^d, \boldsymbol{p}_{\mathbb{L}}) + \boldsymbol{K}_{ori}\ d_{ori}(\boldsymbol{R}_{\mathbb{L}}^d, \boldsymbol{R}_{\mathbb{L}}) \right\| \qquad (5.8a)$$

$$\text{subject to} \qquad\qquad\qquad \boldsymbol{A}\boldsymbol{q}(t_k) \leq \boldsymbol{b} \qquad (5.8b)$$

where $\boldsymbol{K}_{pos}$ and $\boldsymbol{K}_{ori}$ are diagonal weight matrices for position and orientation targets respectively, $\boldsymbol{A}$ and $\boldsymbol{b}$ are two parameters that represent the limits for the joint configuration of the model. A multibody kinematics library iDynTree (Nori et al., 2015b), and an open-source non-linear optimization software library called IPOPT (Wächter and Biegler, 2006) are used to implement two different inverse kinematics problems. The first implementation, referred as *whole-body instantaneous optimization*, considers the inverse kinematics problem of the full human model as a single non-linear optimization problem i.e., with 23 body segments and 66 DoFs. The second implementation, referred as *pair-wise instantaneous optimization*, decomposes the human model into multiple sub models with each model containing a pair of links. So, inverse kinematics of each pair is solved in parallel and combined to get the final joint configuration of the full human model.

5.3 Dynamical Optimization

Real-time human motion tracking is a time critical application and in our experience we observed that instantaneous optimization algorithms perform poorly for highly dynamic movements like running. So, we resort to an alternative approach of formulating the inverse kinematics as a control problem (Sciavicco and Siciliano,

1988). This work is originally presented in (Rapetti et al., 2019)[1]. In this approach we consider the state of the system to be composed of both the system configuration $\boldsymbol{q}$ and the system velocity ν. The pose targets are grouped into a vector denoted as,

$$\boldsymbol{x} = \begin{bmatrix} {}^{\mathcal{I}}\boldsymbol{p}^d_{\mathcal{F}} \\ {}^{\mathcal{I}}\boldsymbol{p}^d_{\mathcal{L}_1} \\ ... \\ {}^{\mathcal{I}}\boldsymbol{p}^d_{\mathcal{L}_{N_B-1}} \\ {}^{\mathcal{I}}\boldsymbol{R}^d_{\mathcal{F}} \\ {}^{\mathcal{I}}\boldsymbol{R}^d_{\mathcal{L}_1} \\ ... \\ {}^{\mathcal{I}}\boldsymbol{R}^d_{\mathcal{L}_{N_B-1}} \end{bmatrix} \tag{5.9}$$

The pose distance vectors are grouped as a pose residual vector denoted as,

$$\boldsymbol{r} = \begin{bmatrix} d_{pos} \\ d_{ori} \end{bmatrix} \tag{5.10}$$

The velocity of the i-th link ${}^{\mathcal{I}}\boldsymbol{v}_{\mathcal{L}_i} = ({}^{\mathcal{I}}\dot{\boldsymbol{p}}_{\mathcal{L}_i}, {}^{\mathcal{I}}\boldsymbol{\omega}_{\mathcal{L}_i})$ is related to the system velocity ν through the Jacobian matrix $\boldsymbol{J}_{\mathcal{L}_i}(\boldsymbol{q}) \in \mathbb{R}^{6 \times (n+6)}$ which is dependent on the system configuration $\boldsymbol{q}$ through the following relation,

$$ {}^{\mathcal{I}}\boldsymbol{v}_{\mathcal{L}_i} = \boldsymbol{J}_{\mathcal{L}_i}(\boldsymbol{q})\ \nu \tag{5.11}$$

Similar to the pose targets, the velocity targets are grouped as a vector denoted by,

$$\boldsymbol{v} = \begin{bmatrix} {}^{\mathcal{I}}\dot{\boldsymbol{p}}^d_{\mathcal{F}} \\ {}^{\mathcal{I}}\dot{\boldsymbol{p}}^d_{\mathcal{L}_1} \\ ... \\ {}^{\mathcal{I}}\dot{\boldsymbol{p}}^d_{\mathcal{L}_{N_B-1}} \\ {}^{\mathcal{I}}\boldsymbol{\omega}_{\mathcal{F}} \\ {}^{\mathcal{I}}\boldsymbol{\omega}_{\mathcal{L}_1} \\ ... \\ {}^{\mathcal{I}}\boldsymbol{\omega}_{\mathcal{L}_{N_B-1}} \end{bmatrix} \tag{5.12}$$

[1] Rapetti, L., Tirupachuri, Y., Darvish, K., Latella, C., & Pucci, D. (2020). Model-Based Real-Time Motion Tracking using Dynamical Inverse Kinematics. IEEE IROS 2020 under review arXiv:1909.07669.

The current link velocities are denoted as $\boldsymbol{J}(\boldsymbol{q})\ \nu$, where

$$\boldsymbol{J}(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{J}_{\mathcal{F}}(\boldsymbol{q}) \\ \boldsymbol{J}_{\mathcal{L}_1}(\boldsymbol{q}) \\ \boldsymbol{J}_{\mathcal{L}_2}(\boldsymbol{q}) \\ ... \\ ... \\ \boldsymbol{J}_{\mathcal{L}_{N_B-1}}(\boldsymbol{q}) \end{bmatrix} \tag{5.13}$$

Now, the velocity residual vector is denoted as,

$$\boldsymbol{u} = \boldsymbol{v} - \boldsymbol{J}(\boldsymbol{q})\ \nu \tag{5.14}$$

Given the residual vectors $\boldsymbol{u}, \boldsymbol{r}$, we consider the dynamic system,

$$\boldsymbol{u} + \boldsymbol{K}\ \boldsymbol{r} = 0 \tag{5.15}$$

where, $\boldsymbol{K} \in \mathbb{R}^{3N_B \times 3N_B}$ is a positive definite diagonal matrix. The system velocity ν is considered as a control input to drive the residual vectors towards zero. This approach is referred as *dynamical optimization* as the system configuration and velocity is controlled to dynamically converge to the targets. Fig. 5.4 presents the dynamical optimization algorithm. In contrast to the instantaneous optimization approach, here the solution is computed in a single iteration. So, the absence of iterations make this approach faster for solving the whole-body inverse kinematics for the human model.
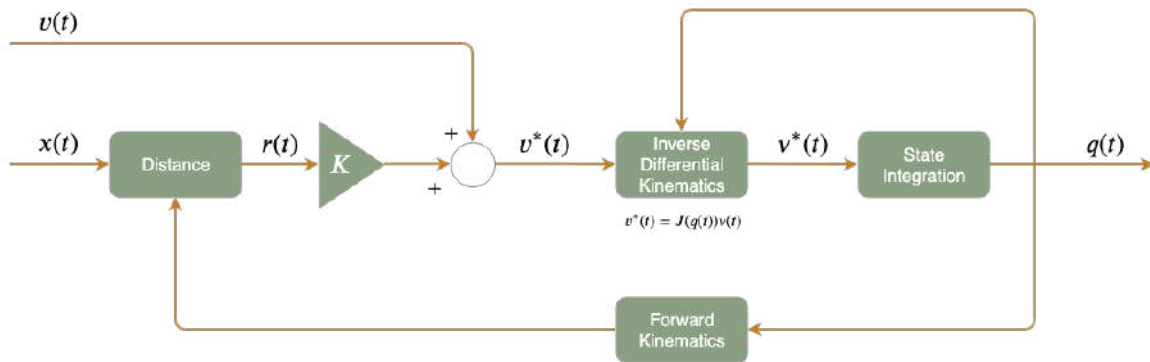
Fig. 5.4 Dynamical optimization for real-time inverse kinematics

5.4 Benchmarking

The proposed approaches of solving inverse kinematics for real-time human motion tracking are benchmarked in terms of *computational time* using three different dynamic tasks:

- **T-pose** : The human subject standing on two feet moves the arms parallel to the ground

- **Walking** : The human subject walks on a treadmill at a constant speed of $4\,km\,h^{-1}$

- **Running** : The human subject runs on a treadmill at a constant speed of $10\,km\,h^{-1}$

The motion data is acquired with the Xsens Awinda wearable suit (Roetenberg et al., 2009) with wireless IMU units distributed through out the human body. The motion capturing system provides the pose and velocity of the 23 body segments and are associated with the links of our human models at a sampling rate of 60 Hz. Furthermore, we considered the full human model with 66 DoFs referred as *Human66* and also the reduced human model with 48 DoFs referred as *Human48*. The motion data is streamed through YARP middleware (Metta et al., 2006) that facilitates recording and real-time playback of data. Box plots of computational time for the proposed inverse kinematics approaches with two different human models is highlighted in Figure. 5.5. It is evident that the dynamical optimization approach is the best way that solves the inverse kinematics of different human models across different tasks when compared with the instantaneous optimization approaches. Furthermore, the computational time of the dynamic approaches increase as the dynamicity of the task increases. So, the task of T-pose has the lowest computational time followed by walking and then running.

Between the two instantaneous approaches, pair-wise instantaneous optimization seems to perform poorly for the T-pose task. This is explained by the additional time taken for parallel computation of small IK problems and combining their output. However, for the tasks of high dynamicity like walking and running, the computational time of pair-wise instantaneous optimization is lower than the whole-body instantaneous optimization.
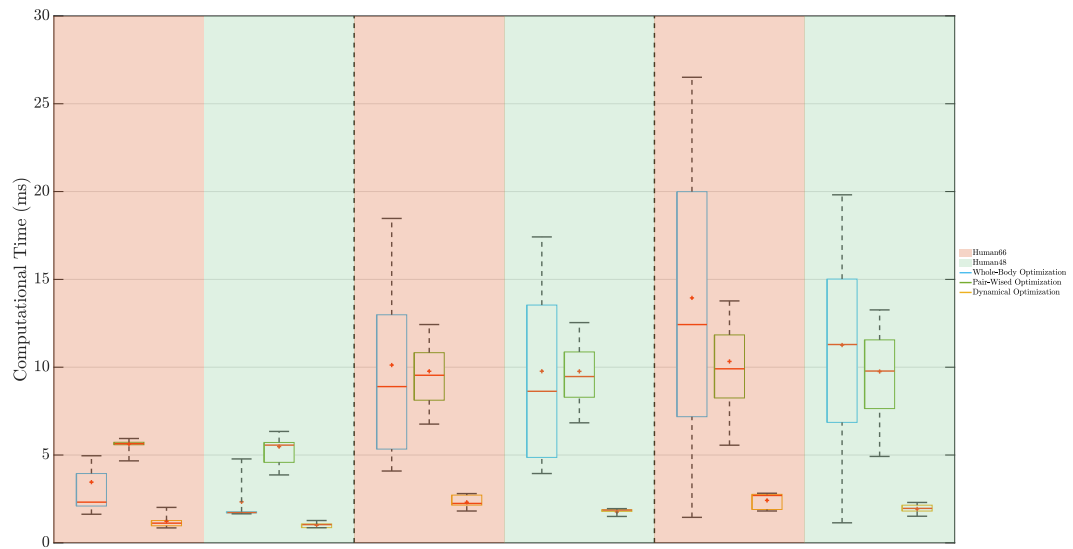
Fig. 5.5 Box plots of computational time for the proposed inverse kinematics approaches with two different human models

6

Human Dynamics Estimation

The knowledge of joint torques of a human body serves useful in various domains like ergonomic analysis, biomechanical studies, collaborative robot control such as exoskeletons. This chapter presents our stochastic framework for human dynamics estimation to perform a sensorless external force estimation on different links of the human. Furthermore, an experimental validation of our framework for a floating base fully articulated human model performing different dynamic tasks is presented.

The dynamics of a mechanical system describes the motion of the rigid bodies under external influence with the inertial parameters considered. In robotics literature, the two main problems concerning the dynamics are:

- *Forward Dynamics (FD)*: Given the actuation forces for joints of the robot, forward dynamics finds the resulting accelerations and motion.

- *Inverse Dynamics (ID)*: Give the desired motion of the robot, inverse dynamics finds the actuation forces for the joints that will induced the desired motion.

Unlike in robotics, human motion is generated through the joint articulation facilitated by sophisticated musculoskeletal structure. Internal body forces and joint torques are crucial in understanding the human dynamics. Although our model lacks the sophistication of biomechanical models, it enables us to model the joint torques i.e., articular stress. In the context of human dynamics, the inverse dynamics problem deals with estimation of the joint torques of human.

6.1 Recourse on Stochastic Dynamics Estimation

A stochastic formulation of the inverse dynamics estimation problem for a whole-body fixed base human model is first investigated in (Latella, 2018; Latella et al., 2019a) and later extended to a floating base human model in our recent work (Latella et al., 2019b). The central idea is to represent the equations of motion as a set of linear equations by defining a new human dynamic variables vector $\boldsymbol{d}$ and the measurement vector $\boldsymbol{y}$. The relation between the dynamic variables $\boldsymbol{d}$ and the model is encapsulated in a model constrains equation. The measurements equations represents the relation between the dynamic variables $\boldsymbol{d}$ and the measurements vector $\boldsymbol{y}$.

6.1.1 Model Constraints Equation

Considering the human model described in Chapter 3 with N_B rigid bodies and n internal DoFs, the dynamic variables vector is constructed as,

$$\boldsymbol{d} \;=\; \begin{bmatrix} \boldsymbol{d}_{link}^{\top} & \boldsymbol{d}_{joint}^{\top} \end{bmatrix}^{\top} \in \mathbb{R}^{12N_B+7n} \tag{6.1}$$

$$\boldsymbol{d}_{link} \;=\; \begin{bmatrix} \boldsymbol{\alpha}_0^g & \underline{\boldsymbol{f}}_0^x & \boldsymbol{\alpha}_1^g & \underline{\boldsymbol{f}}_1^x & \dots & \boldsymbol{\alpha}_{N_B-1}^g & \underline{\boldsymbol{f}}_{N_B-1}^x \end{bmatrix} \in \mathbb{R}^{12N_B} \tag{6.2a}$$

$$\boldsymbol{d}_{joint} \;=\; \begin{bmatrix} \underline{\boldsymbol{f}}_1 & \underline{\boldsymbol{f}}_2 & \dots & \underline{\boldsymbol{f}}_n & \ddot{s}_1 & \ddot{s}_2 & \dots & \ddot{s}_n \end{bmatrix} \in \mathbb{R}^{7n} \tag{6.2b}$$

where, $\boldsymbol{\alpha}^g \in \mathbb{R}^6$ is the proper sensor acceleration from the IMUs placed on human links (Latella et al., 2019b). $\underline{\boldsymbol{f}}_i^x$ denotes the external 6D force on the i-th link, $\underline{\boldsymbol{f}}_i$ denotes the 6D force exchanged through the i-th joint, and $\ddot{s}_i$ denotes the acceleration of the i-th joint. The joint torques $\boldsymbol{\tau}$ are retrieved by projecting the joint force and moments on the joint motion subspace as defined in Eq. (2.36). Given the definition of $\boldsymbol{d}$, the Newton-Euler equations of motion Eq.(2.33)-(2.36) are rearranged in a matrix form, presenting a set of model constraints of the mechanical system given as,

$$\boldsymbol{D}(\boldsymbol{q},\nu)\boldsymbol{d} + \boldsymbol{b}_D(\boldsymbol{q},\nu) = \boldsymbol{0} \tag{6.3}$$

where $\boldsymbol{D}$ is a block matrix $\in \mathbb{R}^{(18N_B+n)\times(12N_B+7n)}$ and $\boldsymbol{b}_D$ is a vector $\in \mathbb{R}^{18N_B+n}$ (refer to Chapter 4 of (Latella, 2018)).

6.1.2   Measurements Equation

Consider that the human is equipped with N_s number of sensors. Typically senors that are used for human dynamics estimation include inertial measurement units that provide the link acceleration or force-torque sensors that provide external forces acting on certain links of the human. The measurement equation is formulated as,

$$\boldsymbol{Y}(\boldsymbol{q},\nu)\boldsymbol{d} + \boldsymbol{b}_Y(\boldsymbol{q},\nu) = \boldsymbol{y} \tag{6.4}$$

where, $\boldsymbol{Y} \in \mathbb{R}^{N_s \times (12N_B + 7n)}$ is a block matrix defined as,

$$\boldsymbol{Y} = \begin{bmatrix} \boldsymbol{Y}_1 & \boldsymbol{Y}_2 & \ldots & \boldsymbol{Y}_{N_S} \end{bmatrix}^\top \in \mathbb{R}^{N_S \times (12N_B + 7n)} \tag{6.5}$$

and the bias vector $\boldsymbol{b}_Y$ is defined as,

$$\boldsymbol{b}_Y = \begin{bmatrix} \boldsymbol{b}_{Y_1} & \boldsymbol{b}_{Y_2} & \ldots & \boldsymbol{b}_{Y_{N_S}} \end{bmatrix}^\top \in \mathbb{R}^{N_S} \tag{6.6}$$

The dimensions depend on the type of the sensor e.g., in the case of an IMU the size of sensor reading is 3 for the linear acceleration and in the case of a force-torque sensor, it is 6 for the six axis forces and torques. An illustrative example of a simple three link 2 DoF mechanical system with and IMU and external 6D force measurements is presented in Chapter 4 of (Latella, 2018).

6.1.3   Stochastic Estimation

On putting together the model constraints equation Eq. (6.3) and the measurements equation Eq. 6.4, we obtain

$$\begin{bmatrix} \boldsymbol{D}(\boldsymbol{q},\nu) \\ \boldsymbol{Y}(\boldsymbol{q},\nu) \end{bmatrix} \boldsymbol{d} + \begin{bmatrix} \boldsymbol{b}_D(\boldsymbol{q},\nu) \\ \boldsymbol{b}_Y(\boldsymbol{q},\nu) \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{y} \end{bmatrix} \tag{6.7}$$

The central assumption is that the the number of available measurements in the vector $\boldsymbol{y}$ ensure enough constraints on estimating the dynamic variables vector $\boldsymbol{d}$. This condition is represented through the rank assumption as,

$$rank\left(\begin{bmatrix} \boldsymbol{D}(\boldsymbol{q}, \nu) \\ \boldsymbol{Y}(\boldsymbol{q}, \nu) \end{bmatrix}\right) = 12N_B + 7n \tag{6.8}$$

Give these set of equations, under the assumptions of all the model constraints having equal relevance and all the sensor measurements having equal accuracy, one way to solve the system represented by Eq. (6.7) is the least squares method using a Moore-Penrose pseudoinverse. Alternatively, a more pragmatic approach is to discard the equal relevance and accuracy assumptions and employ a weighted pseudoinverse to find a weighted least-squares solution. However, for a system of high dimension like the human considered here, it is a tedious task to find the proper weights.

We took an alternative approach and consider the dynamic variables $\boldsymbol{d}$ and the measurements $\boldsymbol{y}$ as stochastic variables with Gaussian distributions. The Maximum A-Posteriori (MAP) estimator tool is employed to estimate the dynamics variables $\boldsymbol{d}$. The problem definition can be summarized as,

> ### Stochastic Estimation Problem
>
> *Given the measurements $\boldsymbol{y}$ and the prior knowledge of the model constraints, estimate the vector $\boldsymbol{d}$ by maximizing the conditional probability distribution of $\boldsymbol{d}$ given $\boldsymbol{y}$, such that*
>
> $$\boldsymbol{d}^{\text{MAP}} = \arg\max_{\boldsymbol{d}} \; p(\boldsymbol{d}|\boldsymbol{y}) \propto \arg\max_{\boldsymbol{d}} \; p(\boldsymbol{d}, \boldsymbol{y}) \tag{6.9}$$

The MAP solution coincides with the mean of the probability density function (PDF) $p(\boldsymbol{d}|\boldsymbol{y})$ yielding to:

$$\boldsymbol{d}^{\text{MAP}} = \boldsymbol{\mu}_{d|y} \tag{6.10}$$

The reader is advised to refer Appendix A for the complete mathematical details in arriving at the MAP solution. In terms of optimization, the estimation problem can be presented as following,

$$\boldsymbol{d}^{\mathrm{MAP}} = \arg\min_{d}(||\boldsymbol{D}\boldsymbol{d} + \boldsymbol{b}_D||^2_{\boldsymbol{\Sigma}_D^{-1}} + ||\boldsymbol{Y}\boldsymbol{d} + \boldsymbol{b}_Y - \boldsymbol{y}||^2_{\boldsymbol{\Sigma}_y^{-1}} + ||\boldsymbol{d} - \boldsymbol{\mu}_d||^2_{\boldsymbol{\Sigma}_d^{-1}}) \qquad (6.11)$$

6.1.4 Covariance Tuning

The solution to the estimation problem from eq. (6.11) is influenced by three covariances: *i*) Prior covariance $\boldsymbol{\Sigma}_d$, *ii*) Measurement covariance $\boldsymbol{\Sigma}_y$, and *iii*) Model covariance $\boldsymbol{\Sigma}_D$. The effect of these covariances on the estimator can be summarized as following:

- Prior covariance $\boldsymbol{\Sigma}_d$ enables the estimator to change the dynamic variable estimates. The prior mean $\boldsymbol{\mu}_d$ for the dynamic variables is set to zero as we do not assume any prior information

- Measurement covariance $\boldsymbol{\Sigma}_y$ influences the estimator to push some of the dynamic variable estimates towards the sensor measurements in $\boldsymbol{y}$

- Model covariance $\boldsymbol{\Sigma}_D$ guides the estimator to ensure all the dynamic variable estimates to be consistent with the model constraints

The combined contribution of this set of covariances affects the final estimation of the dynamic variables.

6.2 Sensorless External Force Estimation

Currently, our stochastic formulation of estimating the dynamic variables considers the external 6D force measurements (inside the measurement vector $\boldsymbol{y}$) obtained from the sensorized *ftShoes* described in section 4.3. Apart from the feet links on which ground reaction forces are present, we assume all the other links do not have any external 6D force on them. So, the external 6D force measurement values on all the other links are set to $\mathbf{0}_6$. The measurement covariance values $\boldsymbol{\Sigma}_y$ for all the external 6D forces are set to a low value (10^{-6}) suggesting that the measurements are highly reliable. This choice implies that the estimation algorithm is forced to keep the feet external 6D force estimates to be as close as possible to the measurements coming from the *ftShoes*, while the estimation of the external 6D force estimates on all the other links are towards $\mathbf{0}_6$.

| (a) | (b) |
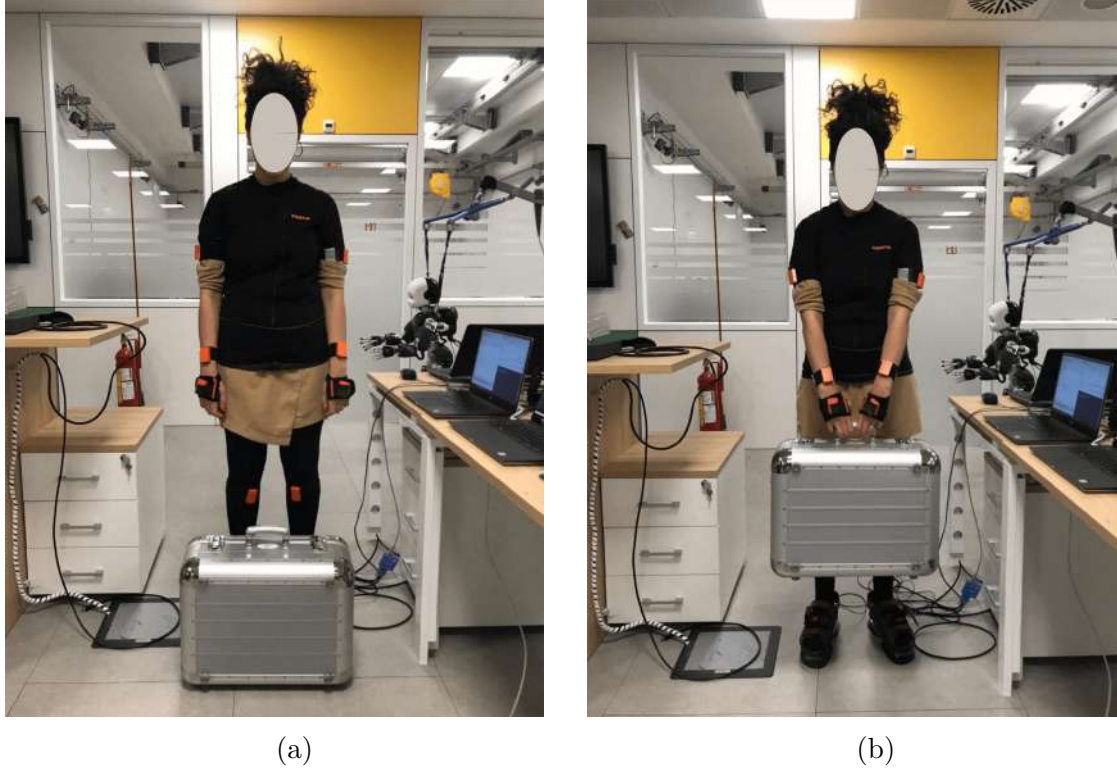
Fig. 6.1 Human subject lifting an external heavy object

Let us consider an example scenario where the human is lifting a heavy object as shown in Figure. 6.1. In particular, when the object is lifted, the additional weight of the object is measured by the *ftShoes* as shown by the yellow arrows in Figure. 6.2. Now, rather than pushing the 6D force estimates at the hand links towards their set measurement of $\mathbf{0}_6$, we consider the problem of reflecting the weight of the object as the 6D force estimates at the hand links. This problem is referred to as Non-collocated wrench estimation. Considering the current formulation of the estimation problem from eq. (6.11), the 6D force estimates at the hand links can be influenced by their measurement covariance value. However, the measurement covariance can only influence the estimator to push the estimates to be close to their set measurement of $\mathbf{0}_6$ or away from that but we do not have any additional constraint that guides the estimates to reflect the object weight. Moreover, given that the dynamic variables vector $\boldsymbol{d}$ from eq. (6.1) is comprised of both the external link wrench $\underline{\boldsymbol{f}}^x$ and the internal joint wrench $\underline{\boldsymbol{f}}$, their estimation is coupled through the consideration of the model constraints through model covariance $\boldsymbol{\Sigma}_D$. So, to achieve the goal of reflecting the object weight as the 6D force estimates at the hands, we need to decouple the estimation of external link wrench and the internal joint wrench quantities. This called

for a new formulation of the estimation problem which will be discussed in detail in the following section.

(a)   (b)
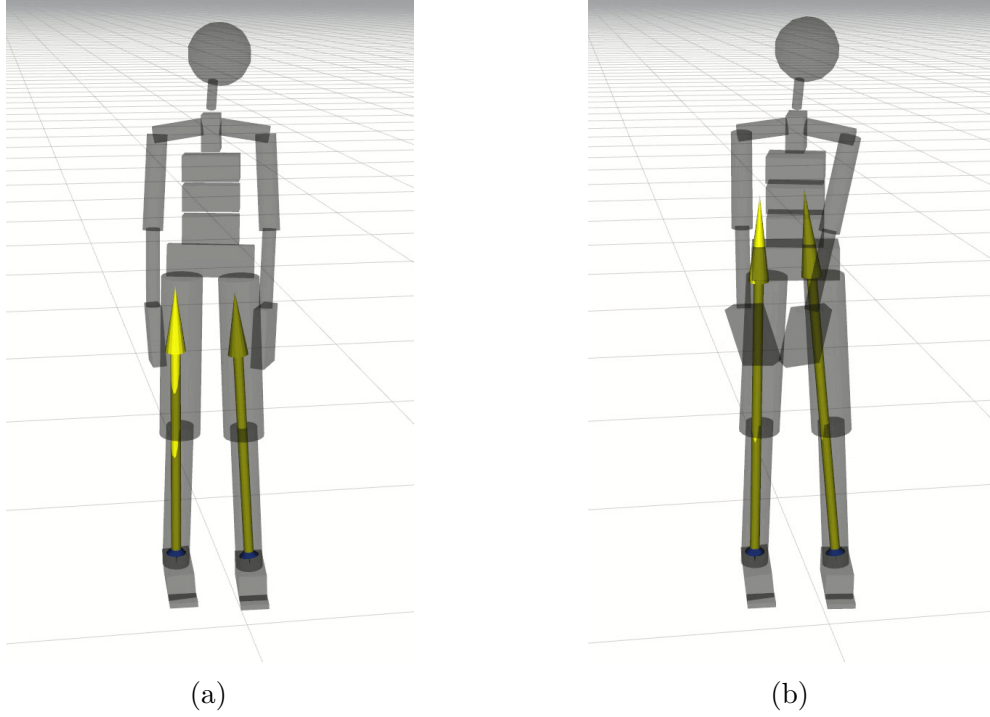
Fig. 6.2 Increase in ground reaction force measurements from *ftShoes* while lifting a heavy object

6.2.1 Stack of Tasks Stochastic Estimation

The stochastic estimation problem for the floating base human model described in section 6.1.3 is reformulated as a two step stack of tasks estimation problem. We introduce a new model constraint in the first task that relates the rate of the change of momentum to the sum of all the external 6D force i.e. the principle of momentum conservation. To illustrate this a new frame, called *centroidal* frame denoted by $\mathcal{G}[\mathcal{I}]$. The centroidal frame is a unique coordinate frame with its origin at the center of mass of the system and its orientation similar to the orientation of the inertial frame of reference. Fig. 6.3 highlights the centroidal frame along with the other frame definitions of a mechanical system.

Now the mathematical formulation of the new constraint when all the quantities are expressed in the centroidal frame (Saccon et al., 2017; Traversaro and Saccon, 2019) is defined as,
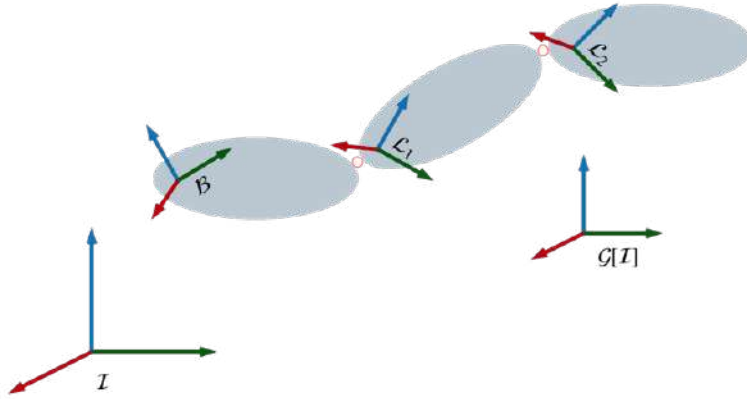
Fig. 6.3 Centroidal frame definition

$$
{}^{\mathcal{G}[\mathcal{I}]}\dot{\boldsymbol{L}} := \begin{bmatrix} m\,\ddot{\boldsymbol{x}}_{com} \\ {}^{\mathcal{G}[\mathcal{I}]}\dot{\boldsymbol{L}}_{ang} \end{bmatrix} = \sum_{i=1}^{N_B} {}^{\mathcal{G}[\mathcal{I}]}\underline{\boldsymbol{f}}_{\mathcal{L}_i}^{x} + \begin{bmatrix} m\,\boldsymbol{g} \\ \boldsymbol{0}_{3\times 1} \end{bmatrix} \tag{6.12}
$$

where, m is the mass of the model, $\boldsymbol{g} = \begin{bmatrix} 0 & 0 & -9.81 \end{bmatrix}^{\top}$ is the gravity vector expressed in the inertial frame, ${}^{\mathcal{G}[\mathcal{I}]}\dot{\boldsymbol{L}}_{ang}$ is the rate of change of angular momentum. Now, the above constraint when expressed in the base frame $\mathcal{B}$ of the system, it becomes

$$
{}^{\mathcal{B}}\boldsymbol{X}_{\mathcal{G},[\mathcal{I}]}^{*} \left({}^{\mathcal{G},[\mathcal{I}]}\dot{\boldsymbol{L}} - \boldsymbol{w} \right) = \sum_{i=1}^{n} {}^{\mathcal{B}}\boldsymbol{X}_{\mathcal{G},[\mathcal{I}]}^{*} {}^{\mathcal{G},[\mathcal{I}]}\boldsymbol{f}_{\mathcal{L}_i} \tag{6.13}
$$

The transformation matrix ${}^{\mathcal{B}}\boldsymbol{X}_{\mathcal{G},[\mathcal{I}]}$ is computed as,

$$
{}^{\mathcal{B}}\boldsymbol{X}_{\mathcal{G},[\mathcal{I}]} = {}^{\mathcal{B}}\boldsymbol{X}_{\mathcal{I}} \, {}^{\mathcal{I}}\boldsymbol{X}_{\mathcal{G},[\mathcal{I}]} \tag{6.14}
$$

where,

$$
{}^{\mathcal{I}}\boldsymbol{X}_{\mathcal{G},[\mathcal{I}]} = \begin{bmatrix} \boldsymbol{I}_{3\times 3} & \mathbf{S}({}^{\mathcal{I}}\boldsymbol{p}_{com}) \\ \boldsymbol{0}_{3\times 3} & \boldsymbol{I}_{3\times 3} \end{bmatrix} \tag{6.15}
$$

and, ${}^{\mathcal{I}}\boldsymbol{p}_{com}$ is the position of the system center of mass expressed with respect to the inertial frame.

Eq. (6.13) can be further represented in a compact form as,

$$
{}^{\mathcal{B}}\hat{\boldsymbol{L}} = \sum_{i=1}^{n} {}^{\mathcal{B}}\boldsymbol{X}_{\mathcal{G},[\mathcal{I}]}^{*} {}^{\mathcal{G},[\mathcal{I}]}\boldsymbol{f}_{\mathcal{L}_i} \tag{6.16}
$$

The acceleration of center of mass is computed using multibody kinematics library iDynTree (Nori et al., 2015b). However, the rate of change of angular momentum is set to $\mathbf{0}_3$ as at the moment this quantity cannot be measured or estimated directly from a human subject. Furthermore, rather than considering the wrench on all the links, we consider only the feet and hand links as of interest and the new constraint simplifies to,

$$
{}^{\mathcal{B}}\hat{\dot{\boldsymbol{L}}} = {}^{\mathcal{B}}\boldsymbol{X}^*_{\mathcal{LF}}\, {}^{\mathcal{LF}}\boldsymbol{f}_{LF} +{}^{\mathcal{B}}\boldsymbol{X}^*_{\mathcal{RF}}\, {}^{\mathcal{RF}}\boldsymbol{f}_{RF} +{}^{\mathcal{B}}\boldsymbol{X}^*_{\mathcal{LH}}\, {}^{\mathcal{LH}}\boldsymbol{f}_{LH} +{}^{\mathcal{B}}\boldsymbol{X}^*_{\mathcal{RH}}\, {}^{\mathcal{RH}}\boldsymbol{f}_{RH} \qquad (6.17)
$$

Where, *LF* correspond to *LeftFoot*, *RF* corresponds to *RightFoot*, *LH* corresponds to *LeftHand*, and *RH* corresponds to *RightHand*. Now, the estimation problem is separated as a two tasks problem.

First Task

The dynamic variables vector for the first task $\boldsymbol{d}'$ contains only the link external 6D force variables i.e.,

$$
\boldsymbol{d}' = \begin{bmatrix} \underline{\boldsymbol{f}}^x_0 & \underline{\boldsymbol{f}}^x_1 & \cdots & \underline{\boldsymbol{f}}^x_{N_B-1} \end{bmatrix}^\top \in \mathbb{R}^{6N_B} \qquad (6.18a)
$$

Similarly, the measurement vector of the first task $\boldsymbol{y}'$ is updated as,

$$
\boldsymbol{y}' = \begin{bmatrix} \underline{\boldsymbol{f}}^{x(mes)}_0 & \underline{\boldsymbol{f}}^{x(mes)}_1 & \cdots & \underline{\boldsymbol{f}}^{x(mes)}_{N_B-1} & {}^{\mathcal{B}}\hat{\dot{\boldsymbol{L}}} \end{bmatrix}^\top \in \mathbb{R}^{6N_B+6} \qquad (6.19a)
$$

Now, the new first set of measurements equation becomes,

$$
\boldsymbol{Y}'\boldsymbol{d}' + \boldsymbol{b}'_Y = \boldsymbol{y}' \qquad (6.20)
$$

The maximum a-posteriori solution of the first task contains the estimates of the external 6D forces on all the links denoted as,

$$\boldsymbol{d}'^{(\text{MAP})} \;=\; \begin{bmatrix} \underline{\boldsymbol{f}}_0^{x(est)} & \underline{\boldsymbol{f}}_1^{x(est)} & \cdots & \underline{\boldsymbol{f}}_{N_B-1}^{x(est)} \end{bmatrix}^\top \in \mathbb{R}^{6N_B} \tag{6.21a}$$

Coming to the choice of the covariances, the measurement covariance $\boldsymbol{\Sigma}_{y'}$ for the rate of change of momentum (6.13) is kept to a low value to place a high trust on these measurements. The measurement covariance $\boldsymbol{\Sigma}_{y'}$ for the feet is kept low and for hands it is kept high. For covariance related to the dynamics variables $\boldsymbol{\Sigma}_{d'}$, it is kept low for the feet and high for the hands. Such a choice guides the estimation algorithm to trust the forces and moments measured at the feet through the *ftShoes* and ensure the estimates to be close to the measurements. On the other hand, the forces and moments measurement of $\boldsymbol{0}_6$ at the hands is not given a high trust and the estimator is free to change the estimates guided by the new constraint of the net external forces in the measurement equation directed by Eq. (6.13). This task considers only the estimation of external link wrenches and it is decoupled from the internal wrench estimation.

Second Task

The second task equations are exactly as described in Section 6.1, except for the measurements vector. The measurements vector for the second task $\boldsymbol{y}''$ is updated with the external 6D force *estimates* Eq. (6.21a) from the first task instead of using the link external 6D force measurements directly.

A proper choice of covariances $\boldsymbol{\Sigma}_{y'}$ and $\boldsymbol{\Sigma}_{d'}$ in the first task results in the estimates that match the measurements of the forces and moments at the feet. On the other hand, the estimates of the forces and moments at the hands vary. Given these reasonable estimates from the first task will become the measurements for the second task, the measurement covariance $\boldsymbol{\Sigma}_y$ is kept low and the covariance $\boldsymbol{\Sigma}_{d'}$ is also kept to a low value. This choice translates to placing high trust on the force and moment values passed as measurements for the second task and the estimator does not change them further during the second task.

Optimization Problem

The new optimization problem for the stack of tasks stochastic estimation is formulated as,

$$
\begin{aligned}
\boldsymbol{d}^{\mathrm{MAP}} \quad = \quad & \arg\min_{d}(||\boldsymbol{D}\boldsymbol{d} + \boldsymbol{b}_D||^2_{\boldsymbol{\Sigma}_D^{-1}} + ||\boldsymbol{Y}\boldsymbol{d} + \boldsymbol{b}_Y - \boldsymbol{y}''||^2_{\boldsymbol{\Sigma}_y^{-1}} + ||\boldsymbol{d} - \boldsymbol{\mu}_d||^2_{\boldsymbol{\Sigma}_d^{-1}}) \\
s.t. \quad & \\
& \boldsymbol{d}'^{\mathrm{MAP}} = \arg\min_{d'}(||\boldsymbol{Y}'\boldsymbol{d}' + \boldsymbol{b}'_Y - \boldsymbol{y}'||^2_{\boldsymbol{\Sigma}_{y'}^{-1}} + ||\boldsymbol{d}' - \boldsymbol{\mu}_{d'}||^2_{\boldsymbol{\Sigma}_{d'}^{-1}}) \qquad (6.22a)
\end{aligned}
$$

6.2.2 Human Joint Torques Visualization

Human joint torques are computed through the joint 6D force estimates that are a part of the dynamic variables solution vector $\boldsymbol{d}^{\mathrm{MAP}}$. At the current stage of our research we do not validate the joint torque estimates through any sensory measurements. Each human joint in the model is composed of up to three one degrees of freedom joint. The effort of each joint is computed as,

$$
\tau_{effort} = \sqrt{\tau_x^2 + \tau_y^2 + \tau_z^2} \qquad (6.23)
$$

The human joint effort is visualized with the help of Rviz, a 3D visualizer popular in robotics community for displaying sensor data and state information using ROS (Robotic-Operating-System, 2018). A sphere that ranges from *green* color to *red* color indicates the joint efforts from [*minimum intensity, maximum intensity*] value. Depending on the task, the joint effort reflecting the articular stress changes for various joints of the human model.

6.2.3 Qualitative & Quantitative Results

We performed initial tests with human lifting a heavy object as shown in Fig. 6.1 after modifying the estimation problem with the new constraint on the net external forces in the measurement equation directed by Eq. (6.13). The feet external force estimation (shown in blue arrows) and the joint torques estimation while the human is in neutral N-pose without lifting any heavy object is shown in Fig. 6.4a. On the

other hand, under the influence of a heavy object lifted by the human, external force estimation at the hands is updated as shown in Fig. 6.4b and the associated joint torques are higher as expected. Also, note that in both the cases the feet force measurements (yellow arrows) and the estimates (blue arrows) are matching. This is a direct consequence of keeping a low value for the measurement covariance during the first task $\boldsymbol{\Sigma}_{y'}$ and the second task $\boldsymbol{\Sigma}_y$ related to the feet 6D force which translates to having higher trust in the measurements from the *ftShoes*.

An offline data analysis is carried with a reduced human model (48 DoF Chapter 3) is carried out and the Fig. 6.5 highlights the comparison between the measurements and the estimates of forces and moments at the feet and the hands. Under the influence of the weight of a heavy object, the ground reaction force measurements increases as seen on f_z^x plot changes of the feet and accordingly, the estimates at the hands increase as seen on on f_z^x plot changes of the hands. This change in the estimates at the hands is guided by choosing a high measurement covariance value $\boldsymbol{\Sigma}_{y'}$ and a high covariance for the dynamic variables $\boldsymbol{\Sigma}_{d'}$ during the first task. Such a choice of covariances translates to not trusting the measurement of $\mathbf{0}_6$ for the forces and moments at the hands and allowing the estimates to vary through the new constraint on the net external forces in the measurement equation directed by Eq. (6.13).

Although we do not validate the joint torque estimation through other sensory measurements for articular stress, the validation of the measurements with the estimates of the forces and moments at the feet presented in Fig. 6.5 is a guide to trust the joint torque estimates. Torque estimation changes at the joints of arms and legs as shown in Fig. 6.6 and Fig. 6.7 respectively.

(a)   (b)
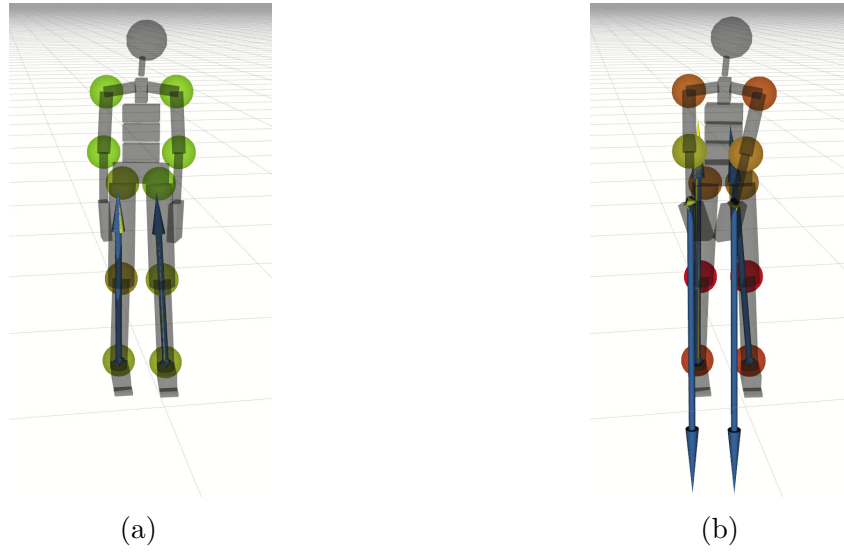
Fig. 6.4 Rviz visualization of external force and joint torques estimates in neutral N-pose (a) and while lifting a heavy object (b)
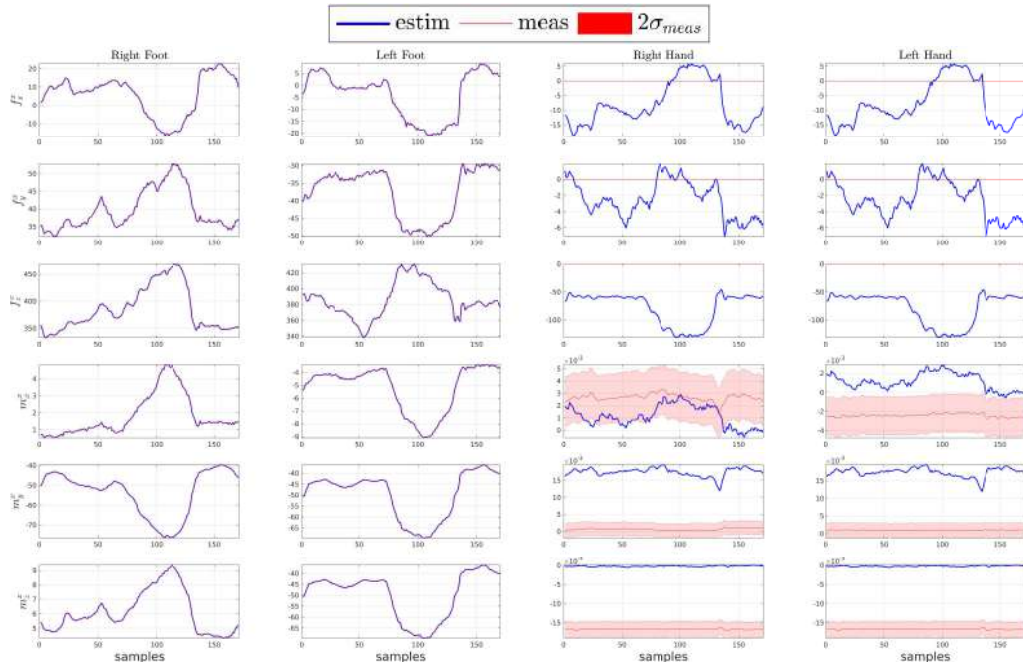
Fig. 6.5 Comparison of the 6D force measurements (mean and standard deviation, in red) and estimates (mean, in blue) at the feet and the hands; The measurements at the hand links are zero while the estimates increase under the influence of the weight of a heavy object held by the human

69
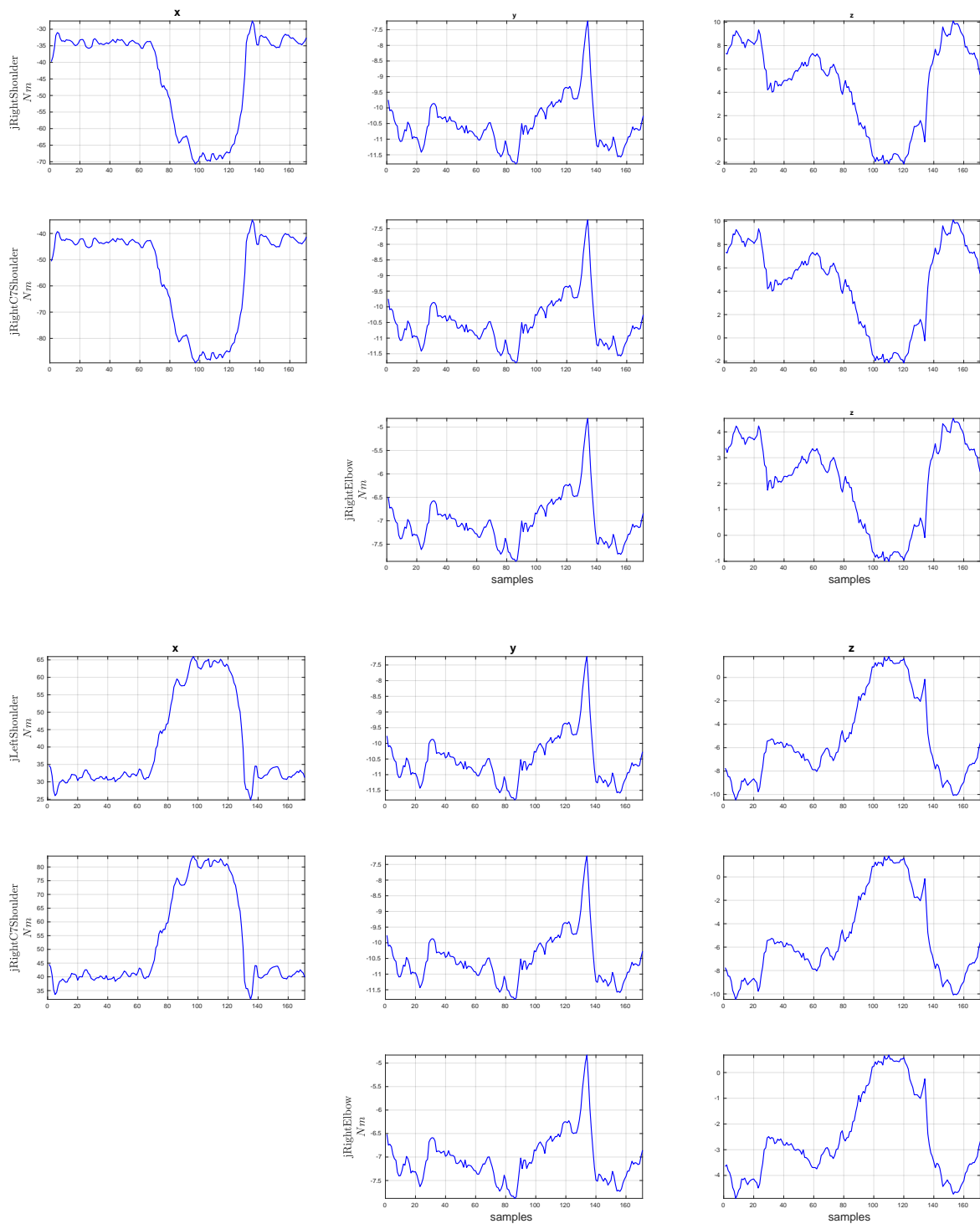
Fig. 6.6 Torques estimation changes of the arm joints under the influence of the weight of a heavy object
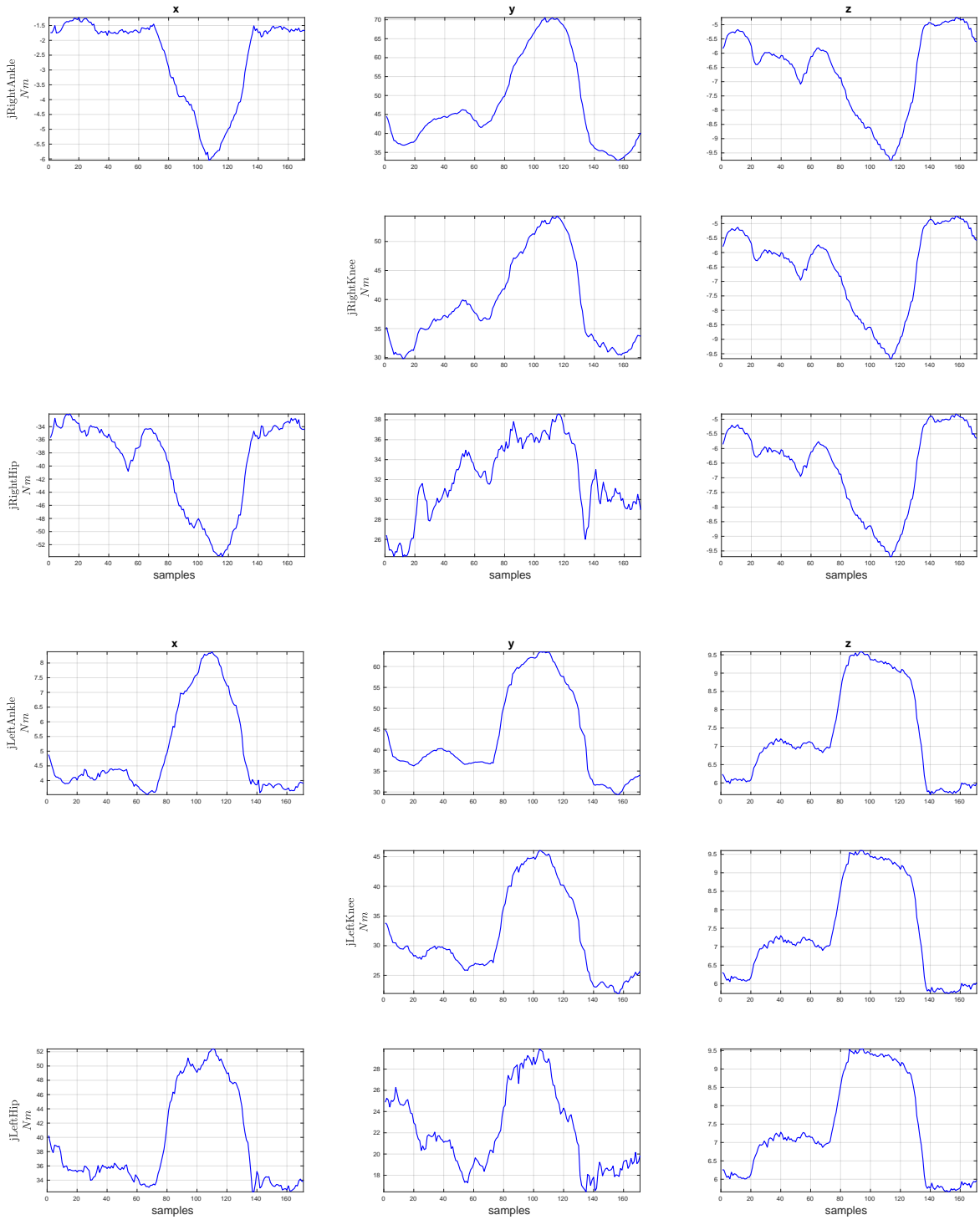
Fig. 6.7 Torques estimation changes of the leg joints under the influence of the weight of a heavy object

6.3 Floating Base Dynamics Estimation: Experimental Validation

The updated formulation for the floating base human dynamics estimation presented in section 6.1 is validated through various experiments where a healthy male subject was asked to perform a set of tasks as listed in Table 6.1.
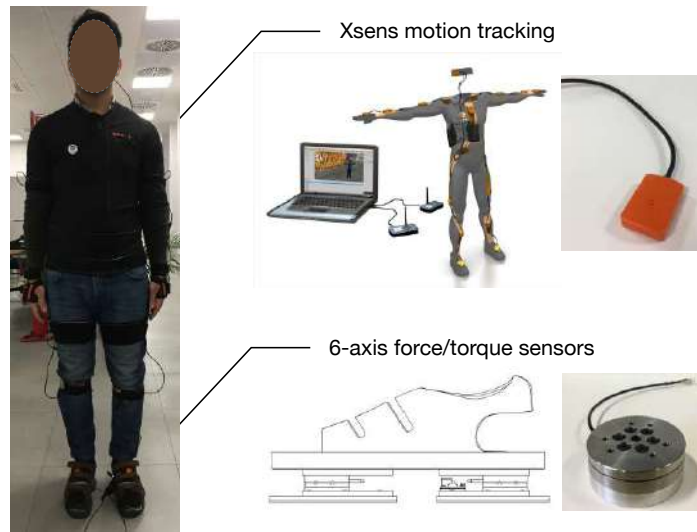
Fig. 6.8 Subject equipped with the Xsens wearable motion tracking system and six axis force-torque shoes (*ftShoes*)

Fig. 6.9 Task T4, Sequence 2: walking on a treadmill

Data from the *ftShoes* is analyzed to detect the feet contact switching based on a self-tuned threshold value for the force along the z-axis. The task representation with respect to the feet in contact with the ground is highlighted in Figure. 6.10.

The kinematic quantities are obtained through the human kinematics estimation presented in chapter 5.

Table 6.1 Tasks performed for validation of floating base dynamics estimation

Task	Type	Description
T1	Static double support	Neutral pose, standing still
T2	Static right single support	Sequence 1: static double support Sequence 2: weight balancing on the right foot
T3	Static left single support	Sequence 1: static double support Sequence 2: weight balancing on the left foot
T4	Static-walking-static	Sequence 1: static double support Sequence 2: walking on a treadmill (Figure 6.9) Sequence 3: static double support

6.3.1   Comparison between Measurements and Estimates

The dynamic variables vector $\boldsymbol{d}$ consists of link variables measurable via sensors i.e., linear proper sensor acceleration $\boldsymbol{\alpha}_{lin}^{g}[\mathrm{m/s^2}]$, external force $\boldsymbol{f}^{x}[\mathrm{N}]$ and external moment $\boldsymbol{m}^{x}[\mathrm{N\,m}]$; and joint variables that are not measured via sensors i.e. joint 6D force $\boldsymbol{f}$ and joint torques $\boldsymbol{\tau}$. The proposed stochastic approach of solving for the human dynamics can be evaluated by comparing the variables that have sensor measurements with respect to their estimates. The validation has been performed along with a Root Mean Square Error (RMSE) metric for linear accelerations and external 6D force presented in Table 6.2. The covariance matrix associated with the sensor measurements is an important factor for the estimation problem. In this experimental analysis, covariances are chosen in a range $[10^{-6}, 10^{-4}]$. A low covariance value reflects higher trust in the sensor measurements and the estimates are closer to the measurements leading to a lower RMSE for the associated variable.

The comparison between the base linear proper sensor acceleration $\boldsymbol{\alpha}_{lin}^{g}[\mathrm{m/s^2}]$ measurement (mean and standard deviation, in red) and the estimation (mean, in blue) for tasks considered tasks is shown in Figure. 6.11. Similarly, the same comparison for the external force $\boldsymbol{f}^{x}[\mathrm{N}]$ and external moment $\boldsymbol{m}^{x}[\mathrm{N\,m}]$ is shown in Figures. 6.12a and 6.12b for the left and right foot, respectively.
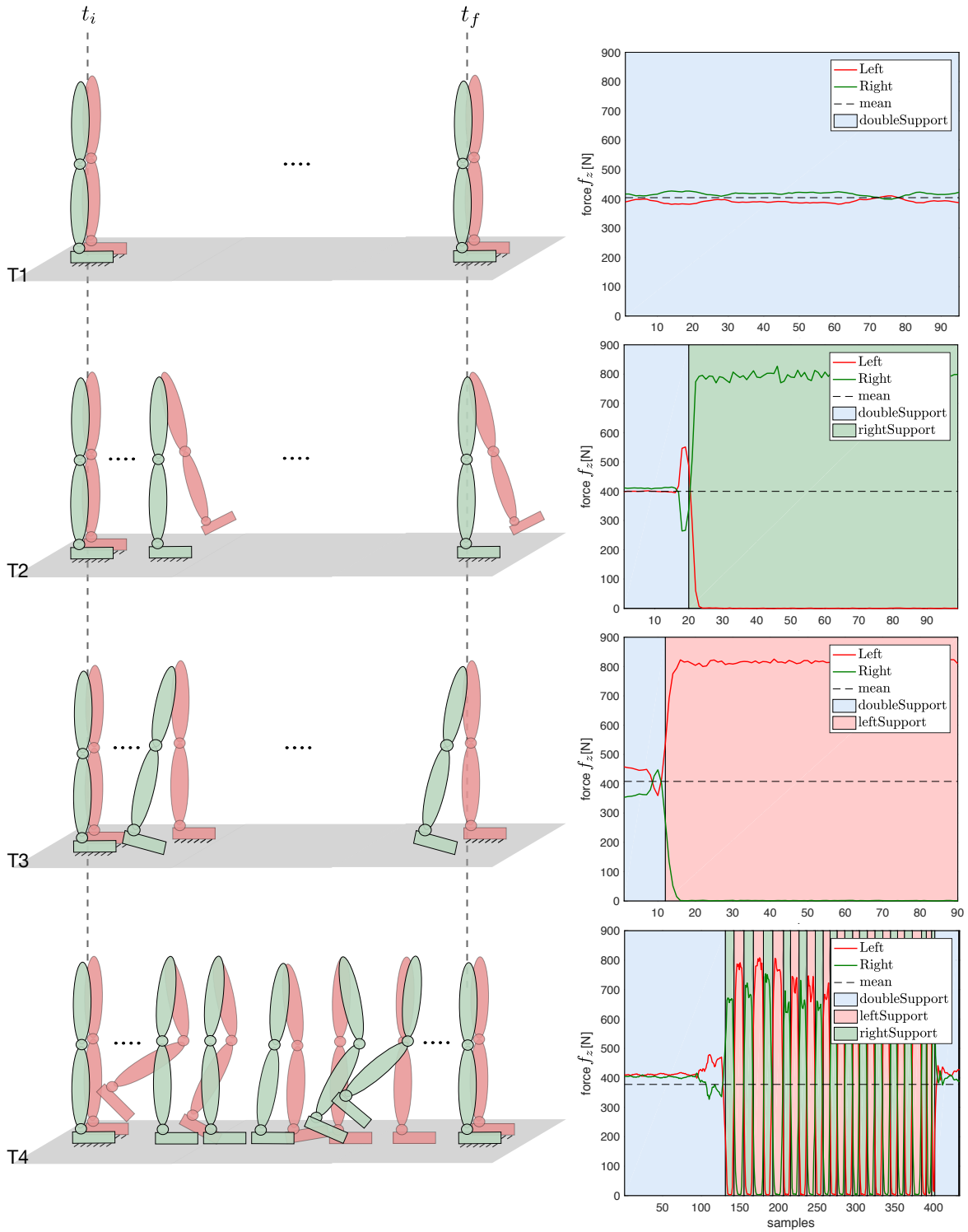
Fig. 6.10 Tasks representation from initial time t_i to final time t_f (*on the left*) and feet contact pattern classification (*on the right*)

Table 6.2 RMSE analysis of the base linear proper sensor acceleration $\boldsymbol{\alpha}^g_{lin}[\mathrm{m/s^2}]$, the external force $\boldsymbol{f}^x[\mathrm{N}]$ and moment $\boldsymbol{m}^x[\mathrm{N\,m}]$ floating-base algorithm estimations w.r.t. the measurements, for tasks T1, T2, T3 and T4.

Task	Link	$\alpha^g_{lin,x}[\mathrm{m/s^2}]$	$\alpha^g_{lin,y}[\mathrm{m/s^2}]$	$\alpha^g_{lin,z}[\mathrm{m/s^2}]$	$f^x_x[\mathrm{N}]$	$f^x_y[\mathrm{N}]$	$f^x_z[\mathrm{N}]$	$m^x_x[\mathrm{N\,m}]$	$m^x_y[\mathrm{N\,m}]$	$m^x_z[\mathrm{N\,m}]$
T1	Base (Pelvis)	0.008	0.014	0.002	-	-	-	-	-	-
	Left foot	-	-	-	0.050	0.030	$2.514e^{-4}$	$8.354e^{-4}$	0.002	$2.079e^{-5}$
	Right foot	-	-	-	0.031	0.048	0.004	0.0015	0.001	$1.038e^{-5}$
T2	Base (Pelvis)	0.003	0.027	0.018	-	-	-	-	-	-
	Left foot	-	-	-	0.153	0.071	0.009	0.002	$4.729e^{-4}$	$1.683e^{-4}$
	Right foot	-	-	-	0.013	0.074	0.005	0.002	$4.247e^{-4}$	$4.331e^{-5}$
T3	Base (Pelvis)	0.012	0.007	0.007	-	-	-	-	-	-
	Left foot	-	-	-	0.075	0.019	0.002	$5.968e^{-4}$	0.002	$5.218e^{-5}$
	Right foot	-	-	-	0.065	0.018	0.003	$6.096e^{-4}$	0.002	$1.196e^{-4}$
T4	Base (Pelvis)	0.011	0.018	0.033	-	-	-	-	-	-
	Left foot	-	-	-	0.089	0.056	0.012	0.002	0.003	$1.322e^{-4}$
	Right foot	-	-	-	0.084	0.056	0.019	0.002	0.003	$9.737e^{-5}$
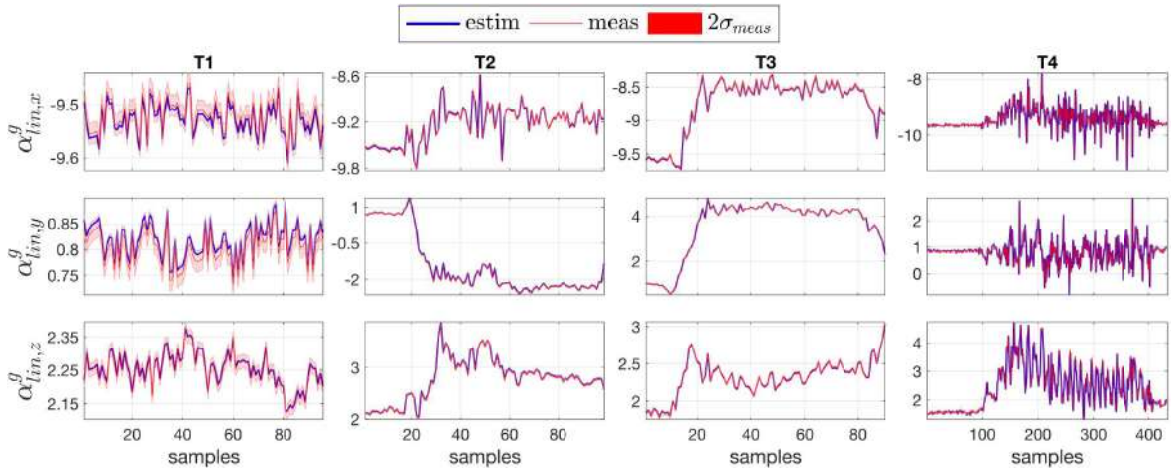
Fig. 6.11 The base linear proper sensor acceleration $\boldsymbol{\alpha}^g_{lin}[\mathrm{m/s^2}]$ comparison between measurement (mean and standard deviation, in red) and floating-base estimation (mean, in blue), for tasks T1, T2, T3 and T4
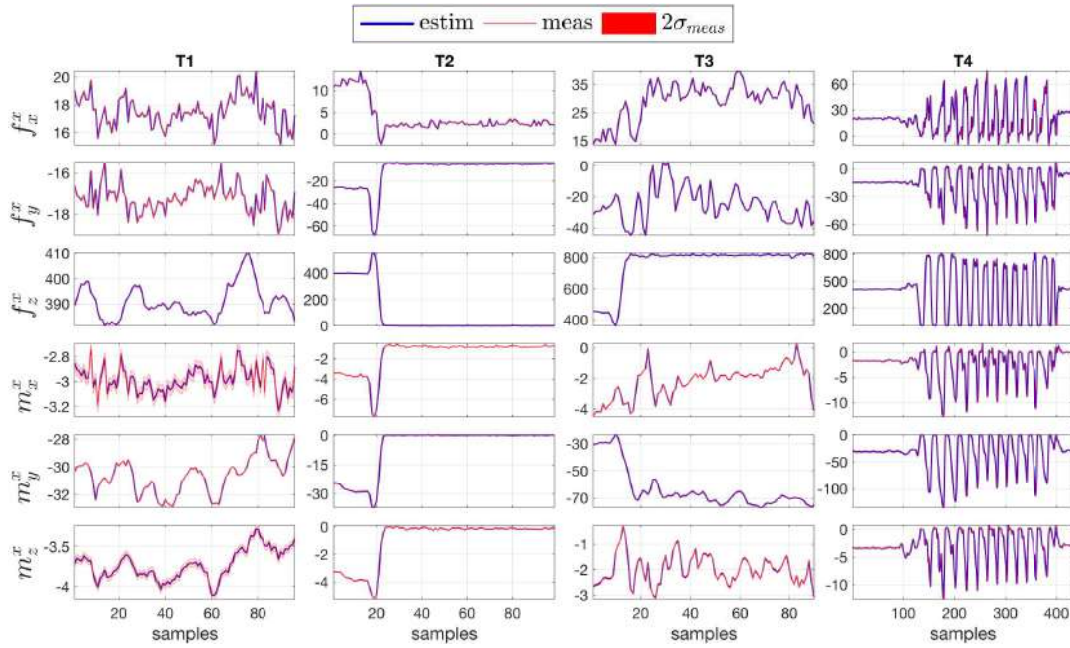
(a)
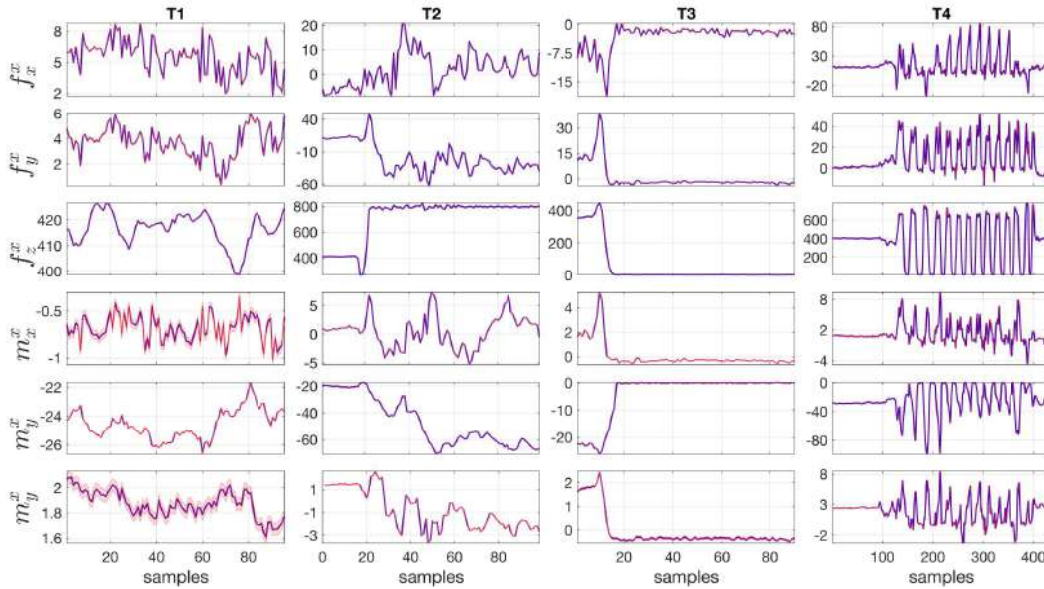
(b)

Fig. 6.12 The external force $\boldsymbol{f}^x$[N] and external moment $\boldsymbol{m}^x$[N m] comparison between measurement (mean and standard deviation, in red) and estimation via floating-base estimation (mean, in blue) for left (a) and right foot (b), for tasks T1, T2, T3 and T4

6.3.2 Human Joint Torques Estimation during Walking

Joint efforts during the balancing on single foot related to Tasks T2 (right foot 6.14a) and T3 (left foot 6.14b) are highlighted in Figure. 6.14. Similarly, joint effort changes during different phases of walking is shown in Figure. 6.15. Furthermore, the ground reaction force measurements (yellow arrows) and the estimates (blue arrows) are shown. The joint torque estimates along with the joint angles for right leg (b) and left leg (a) during the walking Task T4 is highlighted in Figure. 6.13.

(a)

(b)
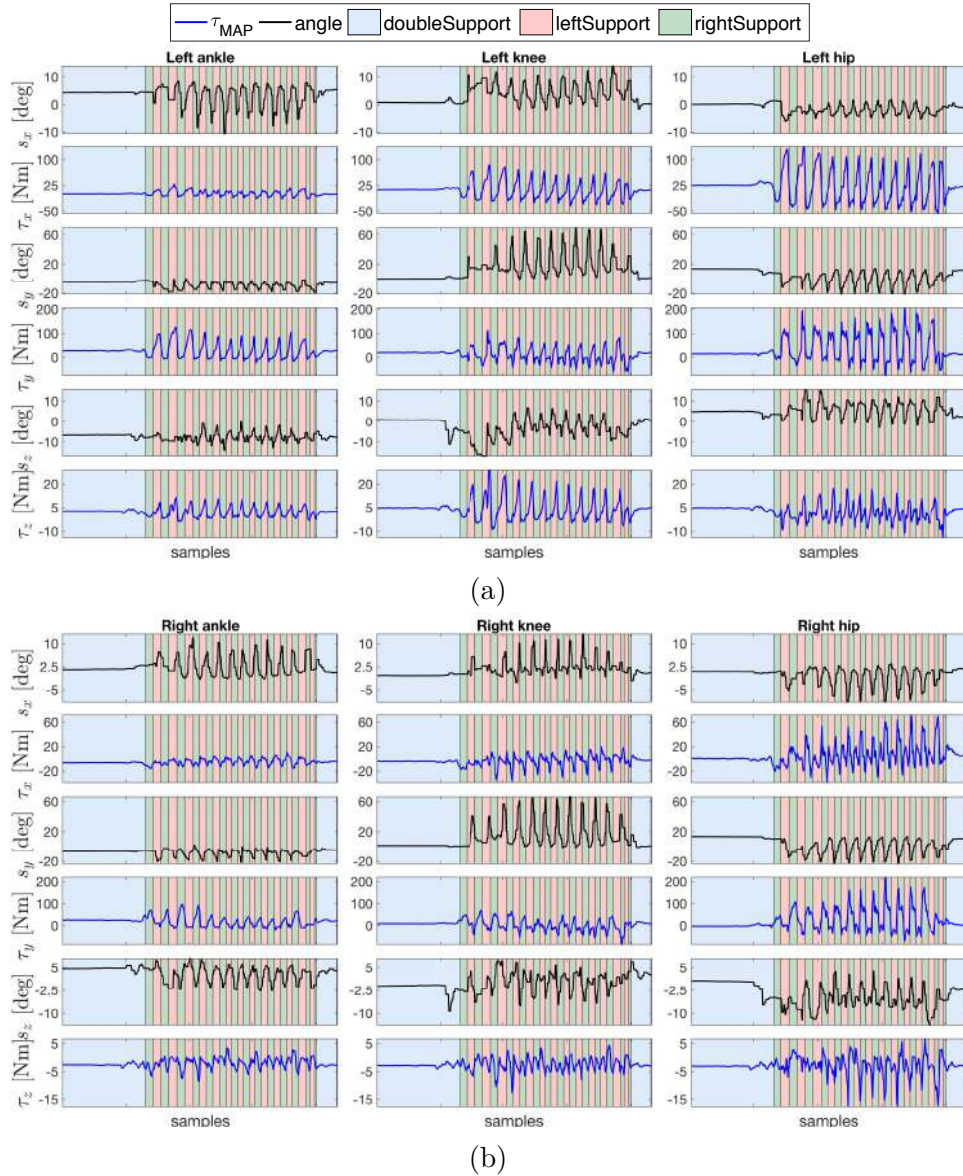
Fig. 6.13 Joint torques τ[N m] changes (in blue) during Task T4 walking along with the joint angles (in black) of right leg (a) and left leg (b).

Low fatigue   High fatigue
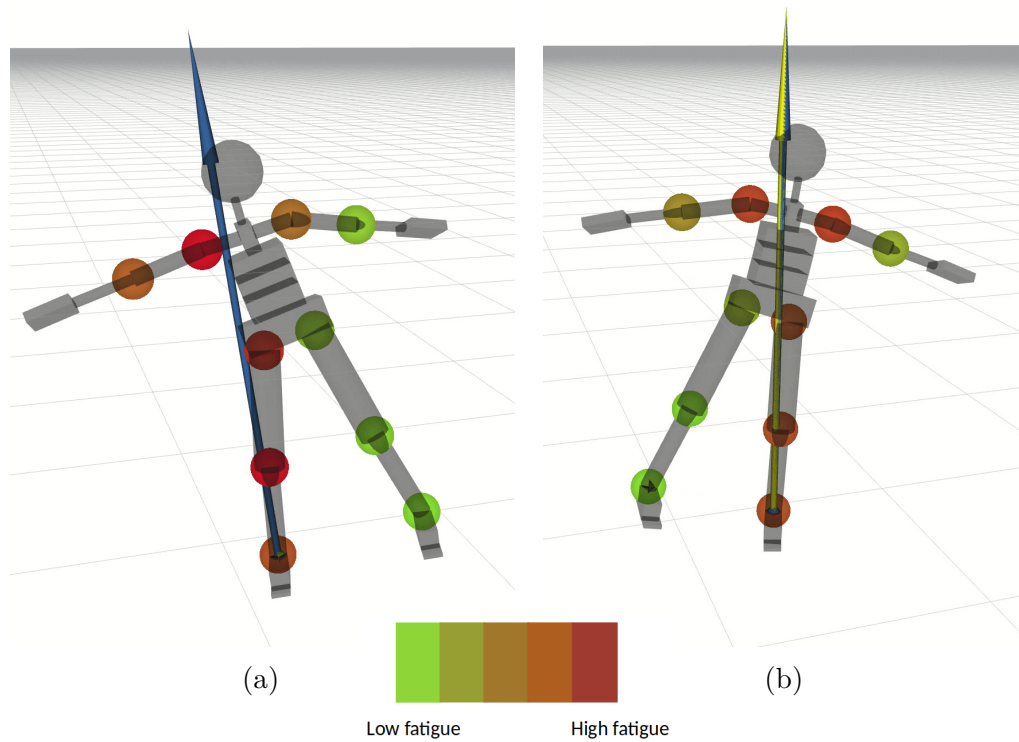
Fig. 6.14 Rviz visualization of joint efforts for Task T2 balancing on the right foot (a) and Task T3 balancing on the left foot (b)
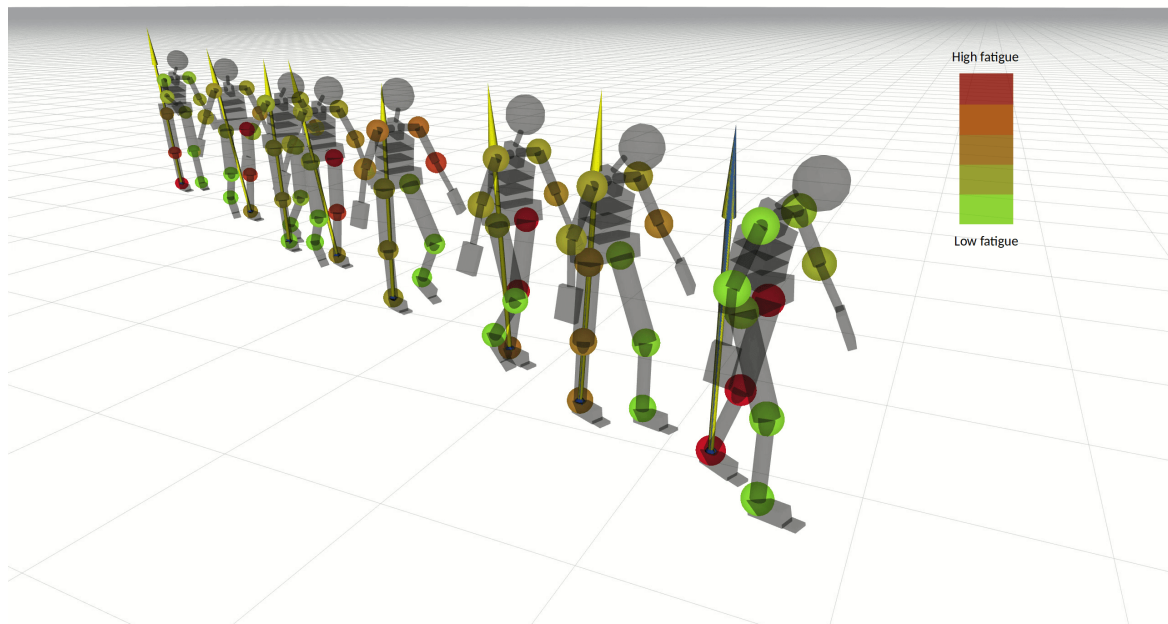
Fig. 6.15 Rviz visualization of joint efforts during different phases of Task 4, walking

<div align="right"><em style="font-size:3em">7</div>

Software Architecture

Our framework of holistic human perception is aimed to be a platform for future extensions that can incorporate different sensory modalities such as hear rate monitoring, muscular fatigue, emotional fatigue for various aspects of human perception. Given the anticipated complexity, a sound software architecture is of utmost importance for our system to be robust. This chapter presents our software architecture and explains the main components that are currently functional.

7.1 Motivation

Our ambition for holistic human perception is to develop a novel wearable whole body suit technology that integrates a lot of sensors and novel algorithms that processes the sensor inputs. Currently, at the hardware level we depend on commercial products like Xsens for human motion measurements and use our *ftShoes* technology for force monitoring. In the future, we aim at using commercial platforms of EMG sensors e.g., BTS engineering for monitoring human muscles activation. So, a set of sensors can be considered as a system that provides a particular set of measurements, not particularly homogeneous[1]. Synchronizing the sensory information coming from different hardware is often daunting task and an ideal solution for achieving system integration of heterogeneous hardware components is to focus on the software level synchronization (Samy et al., 2020). At the algorithmic level we currently perform real-time motion tracking and dynamics estimation. In the future, we aim at integrating algorithms for motion classification and prediction. Given the system complexity, it is crucial to have a modular software architecture that is extensible for future possibilities. This chapter lays the details of our novel software architecture to support our vision of

---

[1] One example is an exoskeleton worn by the human that provides its joint states and torques

holistic human perception and we hope this can serve as a platform to develop new wearable technologies, both in research and industry, to enable effective human-robot collaboration.

The software infrastructure is developed in `C++` using YARP middleware (Fitzpatrick et al., 2014). We opted for a device based architecture based on YARP devices[2]. The three core components in this architecture are 1) Device Drivers or Devices, 2) Interfaces, and 3) Network Wrappers. A device in YARP is a `C++` class and interfaces are abstract base classes. A device can be thought of as a computation unit that processes some inputs and implement the methods defined in the interface. Two devices exchange information through the interfaces by an "attach" method which enables one device to view the interface implemented in the other device to which it is attached. Furthermore, multiple devices can be launched as a single process[3]. Network wrappers are special cases of devices that provide network resources to send data to the network. A network wrapper attached to a device views the implemented interface and facilitates streaming of necessary data to the network.

At the highest level of abstraction we have two main layers: 1) Producer 2) Consumer

7.2 Producer Layer

The producer layer is composed of all the wearable technology that provides various measurements. The technologies involved at this layer are heterogeneous such as Xsens and *ftShoes*. An ideal design choice at the producer layer is to provide a unifying data structure down to the consumer layer. We developed an open source `C++` library called Wearables[4] to combine data coming from the heterogeneous sources and provide a standard data output. The core components of the wearbles library are 1) Sensor and Wearable Interfaces, 2) Wearable Devices, and 3) Wearable Wrapper. Fig. 7.1 provides the overview of the components in wearable library.

---

[2] https://www.yarp.it/note_devices.html
[3] All the devices need not be attached to one another
[4] https://github.com/robotology/wearables
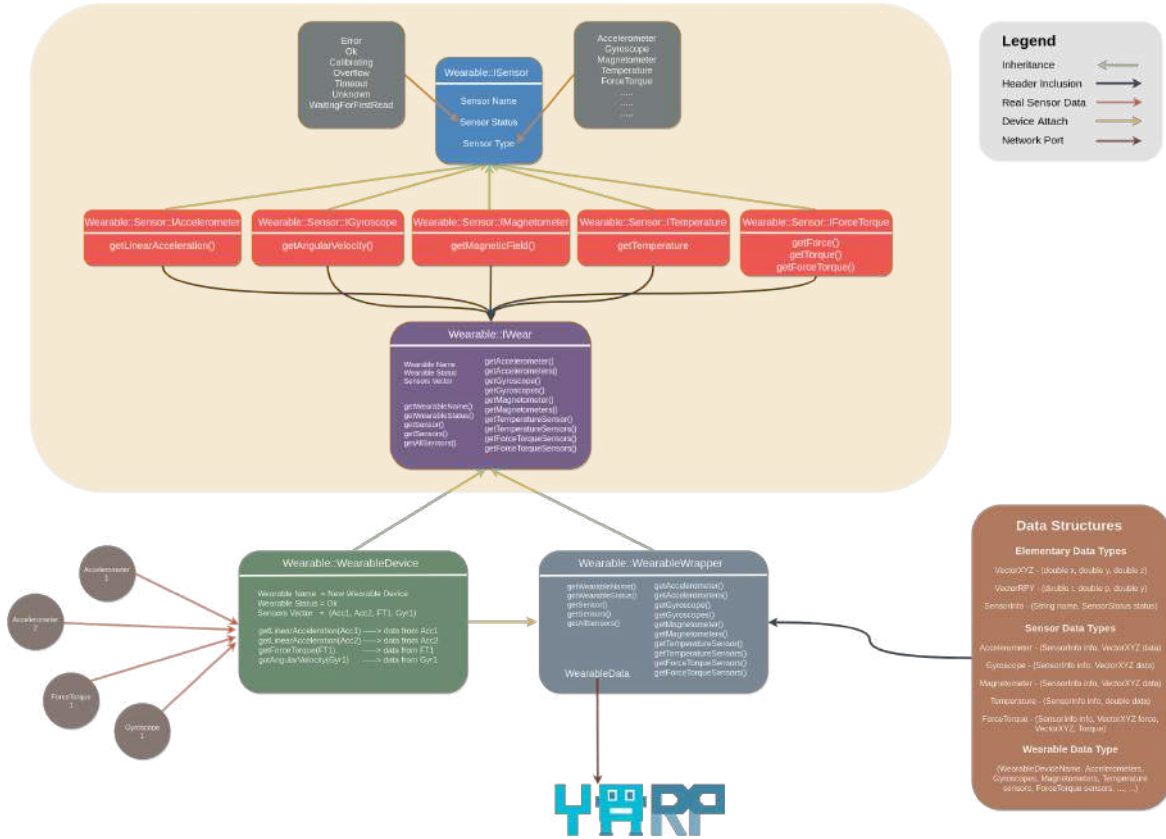
Fig. 7.1 Software architecture of wearables library

7.2.1 Sensor & Wearable Interfaces

Sensor Interfaces

The lowest level of abstract is sensor interface (`Wearable::ISensor`) containing the basic details that represent a sensor i.e., a) Sensor Name, b) Sensor Status, and c) Sensor Type. Sensor type represents the nature of the sensor and some of the examples are `accelerometers, gyroscopes, magnetometers, temperature sensors` and `force torque sensors`. Sensor status indicates the state of the sensor and some of the possible states are `Error, Ok, Calibrating, Overflow, Timeout, Unknown` and `WaitingForFirstRead`.

`Wearable::ISensor` is the base abstract class from which particular type of sensor interfaces are inherited. Depending on the type of the sensor, relevant methods are present for accessing the data generated by the particular sensor. Some of the examples of the sensor type interfaces are:

- `Wearable::Sensor::IAccelerometer` sensor interface that has `getLinearAcceleration()` method to access the linear acceleration data from accelerometers.

- `Wearable::Sensor::IGyroscope` sensor interface that has `getAngularVelocity()` method to access the angular velocity data from gyroscopes.

- `Wearable::Sensor::IMagnetometer` sensor interface that has `getMagneticField()` method to access the magnetic field data from magnetometers.

- `Wearable::Sensor::ITemperature` sensor interface that has `getTemperature()` method to access the temperature data from the sensor.

- `Wearable::Sensor::IForceTorque` sensor interface that has `getForce()`, `getTorque()`, and `getForceTorque()` methods to access the forces, moments, and forces and moments data from six axis force-torque sensors.

Wearable Interface

The next level of abstraction is the wearable interface that groups all the particular sensor type interfaces together. Similar to the `Wearable::ISensor` interface, some of the basic details are the name and the status along with the methods `getWearableName()` and `getWerableStatus()`. Additionally, there are methods like `getSensor()` to access a sensor based on the sensor name or `getSensors()` method to access all the sensors of a particular type. The `getAllSensors()` methods provides access to all the sensors. Furthermore, sensor specific methods like `getAccelerometer()` or `getAccelerometers()` provides access to sensors of a particular type. The `Wearable::IWear` interface is the base abstract class that forms the template for the rest of the components in the wearables library i.e., wearable device and wearable wrapper.

7.2.2   Wearable Device

A wearable device is a `C++` class that is inherited from the `Wearable::IWear` interface. The physical intuition of a wearable device is that it represents a system that constitute a set of sensors like Xsens or *ftShoes*. A specific name is assigned

to the wearable name parameter. The sensor measurements from real sensors of
a particular type are given as inputs to a wearable device e.g., `Accelerometer 1,`
`Accelerometer 2, Gyroscope 1, ForceTorque 1`. The status of a wearable device
is determined by the status of all the sensors it encapsulates. Furthermore, a wear-
able device has an implementation of all the methods related to the sensors it en-
capsulates e.g., `getLinearAcceleration(Acc1), getLinearAcceleration(Acc2),`
`getAngularVelocity(Gyr1), getForceTorque(FT1)`.

7.2.3 Wearable Wrapper

The wearable wrapper is a `C++` class that is inherited from the `Wearable::IWear`
interface.   As stated before, it is also a device that can be attached
to any wearable device.   It implements methods to get sensors present
in a particular wearable device to which the wearable wrapper is at-
tached to e.g., `getAccelerometer(), getAccelerometers(), getGyroscope(),`
`getGyroscopes(), getSensor(), getSensors(), getAllSensors()`. Furthermore,
the data from different sensors present in a particular wearable device are seri-
alized using a data structure template. The data structure template defines el-
ementary data types that are composed to represent sensor specific data.   Fur-
thermore, the "Wearable Data" type is defined which has the fixed structure as
(`WearableDeviceName::SensorName::SensorData`). As an example with respect to
Fig. 7.1 the wearable data output is composed as,

((`WearableDeviceName::Accelerometer1::SensorData`),
(`WearableDeviceName::Accelerometer2::SensorData`),
(`WearableDeviceName::Gyroscope1::SensorData`),
(),———> Mangetometer Data Placeholder
(),———> ForceTorque Data Placeholder
(`WearableDeviceName::ForceTorque1::SensorData`))

The wearable wrapper also implements a periodic thread[5] that calls sensor methods
like `getLinearAcceleration(Acc1), getLinearAcceleration(Acc2)` of a wearable
device to update the sensor readings from the associated sensors. Additionally. the
wearable wrapper has the functionality of streaming the composed wearable data from

---

[5] http://www.icub.org/doc/icub-main/icub_periodic_thread.html

a particular wearable device to the network through ports. Any other components can
access the wearable data through the network ports.
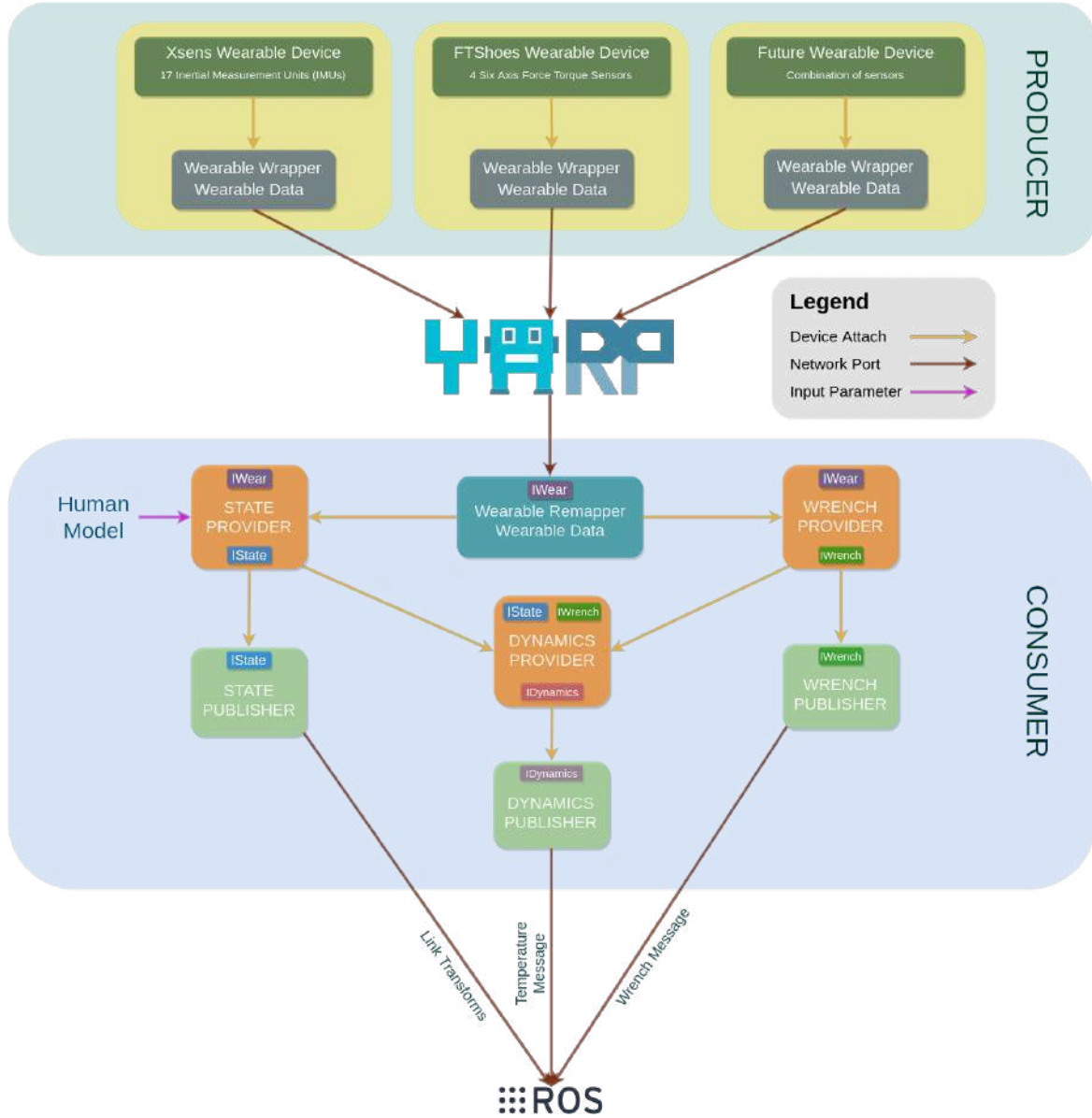
Fig. 7.2 Software architecture for holistic human perception

7.2.4 Current Wearable Devices

Our current architecture constitutes of two wearable devices that accesses data
coming from the Xsens motion tracking system and *ftShoes*. In the case of Xsens we rely
on the software development kit (SDK) provided to access various human measurements

such as sensor linear acceleration, link position and orientation. Similarly, in the case of *ftShoes* we rely on interfaces developed to access data from six axis force-torque sensors developed at IIT (Section 4.2). For the sake of clarity, these details are not captured in Fig. 7.2. Furthermore, a wearable device and an associated wearable wrapper are launched in a single process and they can be run on different machines connected to the same network.

7.2.5 Future Wearable Devices

Applications of human-robot collaboration may contain serial manipulators or humanoid robots or exoskeletons to augment humans. Under those scenarios, the robots or the exoskeleton can be considered as a new wearable device and implemented in the producer layer. So, various data like the joint quantities or interaction forces from these devices can be passed as wearable data for the consumer layer where new components can be implemented for different applications.

7.3 Consumer Layer

The consumer layer is composed of the components that utilizes the wearable data from the network. Fig. 7.2 presents a higher level of abstraction in the organization of producer and consumer layers. Similar to the components of the producer layer, the components of the consumer layer are also YARP device drivers. The main components present at the consumer layer are 1) Wearable Remapper, 2) Providers, and 3) Publishers. Another level of classification that can be done in the consumer layer is related to the aspect they deal with such as the state or the dynamics. Providers are devices that are attached to a wearable remapper device and access the wearable data through the wearable interface. Furthermore, providers are algorithmic units that implement a particular algorithm e.g. state estimation. On the other hand, publishers are devices that access output data from providers through relevant interfaces and publish the data to a ROS network (Robotic-Operating-System, 2018). This facilitates the use of ROS tools like Rviz visualizer for real-time visualization of human quantities. All the components of the consumer layer that deals with the human quantities are implemented as an open source `C++` library called human-dynamics-estimation[6]. The following section lays the details of the components at the consumer layer.

---

[6] https://github.com/robotology/human-dynamics-estimation

7.3.1 Wearable Remapper

A wearable remapper is a device that servers the dual purpose of a wearable wrapper. Analogous to the wearable wrapper, the remapper is also inherited from `Wearable::IWear` interface. It connects to the network ports to receive the wearable data sent to the network by the wearable wrapper. So, the wearable remapper can be thought as a component that deserializes the wearable data into respective sensor data coming from several wearable devices.

7.3.2 State Components

The state provider is a device that is inherited from `Wearable::IWear` interface. It is attached to the wearable remapper and accesses the data from Xsens wearable device through `Wearable::IWear` interface. One of the input parameters is the human model described in Chapter 3. The whole-body inverse kinematics problem for real-time human motion tracking described in Chapter 5 is implemented inside the state provider. The state provider is inherited from `IState` interface and the state estimation quantities i.e. human base pose, joint positions and velocities data is handled through the implementation of `IState` interface methods.

The state publisher is a device that is attached to the state provider. It accesses the estimated human state quantities through the `IState` interface and computes the link tranforms information through forward kinematics with the given human model. This information is essential for Rviz visualizer to animate the human model as shown in the case of human walking Fig 6.15. The computed link transforms are streamed to ROS network through topics[7].

7.3.3 Wrench Components

The wrench provider is a device that is inherited from `Wearable::IWear` interface. It is attached to the wearable remapper and accesses data from *ftShoes* wearable device through `Wearable::IWear` interface. This device transforms the wrench data and expresses them with respect to the human links that are given as an input parameter. Currently, the wrench information is available only on the human feet. However, this can be expanded in the future for applications of HRC, where a humanoid robot or an exoskeleton can be considered as a wearable device and the wrench information

---

[7] http://wiki.ros.org/Topics

from them is passed through the wearable data. The wrench provider is inherited from `IWrench` interface and the transformed wrench information coming various wearable devices is handled through the implementation of `IWrench` interface methods.

The wrench publisher is a device that is attached to the wrench provider. It accesses the transformed wrench through `IWrench` interface and expresses them as a wrench message data type to stream to ROS network through topics. This information is essential for Rviz visualizer to display the arrows corresponding to the external wrench as shown in Fig. 6.14.

7.3.4 Dynamics Components

The dynamics provider is a device that is inherited from `IState` and `IWrench` interfaces. It is attached to the state provider and the wrench provider. The human state quantities are received through `IState` interface and the wrench quantities are received through `IWrench` interface. The human dynamics estimation algorithm described in Chapter 6 is implemented in the dynamics provider. It is also inherited from `IDynamics` interface and implements the methods to handle the human joint torque estimates.

The dynamics publisher is a device that is attached to the dynamics provider. It accesses the human joint torque estimates through `IDynamics` interface and computes the joint effort that is described in section 6.2.2. The joint effort information is sent as a temperature message to ROS network through topics. This information is essential for Rviz visualizer to display the spheres at the human joints that show the joint effort as highlighted in Fig. 6.14. Furthermore, the dynamics publisher also expresses the external wrench estimates as a wrench message data type to stream to ROS network through topics. This is essential to display the arrows (blue color) shown in Fig. 6.14.

PART III:

REACTIVE ROBOT CONTROL

Partner-Aware Control

Classical feedback linearization approach has been proved to be very robust in controlling complex robots. However, there are some clear limitations to it for collaborative tasks. This chapter presents those limitations first, followed by a coupled dynamics formalism for collaboration scenarios. Furthermore, the interaction characterization in terms of joint torques from an external agent assisting the robot are considered towards partner-aware control techniques validated through experiments with two humanoid robots.

8.1 Recourse on Classical Robot Control

Typically, one may be interested in controlling a robot related quantity, say $\boldsymbol{x} \in \mathbb{R}^p$. In the case of joint space control, the interested task is to control a single robot joint or a group of robot joints along a pre-defined trajectory. Similarly, for the task space control, the interested task is to achieve a Cartesian trajectory tracking by a link of the robot where $\boldsymbol{x}_d(t), \dot{\boldsymbol{x}}_d(t), \ddot{\boldsymbol{x}}_d(t) \in \mathbb{R}^6$ denote the desired pose, velocity and acceleration in the Cartesian space, parametrized in time t. The velocity tracking error to be minimized is denoted by $\dot{\tilde{\boldsymbol{x}}} = \dot{\boldsymbol{x}}(t) - \dot{\boldsymbol{x}}_d(t)$.

The robot link's actual velocity has a linear map to the robot's velocity through the Jacobian matrix $\boldsymbol{J}(\boldsymbol{q})_x \in \mathbb{R}^{6 \times (n+6)}$, i.e.

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{J}_x(\boldsymbol{q}) \ \boldsymbol{\nu} \tag{8.1}$$

The task of Cartesian trajectory tracking is achieved through a simple control objective as defined below,

$$\ddot{\boldsymbol{x}} = \ddot{\boldsymbol{x}}^* := \ddot{\boldsymbol{x}}_d - \boldsymbol{K}_D \ \dot{\tilde{\boldsymbol{x}}} - \boldsymbol{K}_P \int_0^t \tilde{\boldsymbol{x}} \ du, \quad \boldsymbol{K}_D, \boldsymbol{K}_P > 0 \tag{8.2}$$

where $\boldsymbol{K}_P, \boldsymbol{K}_D \in \mathbb{R}^{6\times 6}$ are positive definite symmetric feedback matrices.

Classical feedback linearization approach (Khalil and Dombre, 2004) helps us in finding the robot joint torques $\boldsymbol{\tau}$ needed to realize the control objective (8.2) for Cartesian trajectory tracking and keeping the tracking error minimum. The robot control torques necessary for trajectory tracking with the desired dynamics directed by Eq. (8.2) are obtained using Eq. (8.1). On differentiating $\dot{\boldsymbol{x}}(t)$ we get the following relation,

$$\ddot{\boldsymbol{x}}(t) = \boldsymbol{J}_x\,\dot{\boldsymbol{\nu}} + \dot{\boldsymbol{J}}_x\,\boldsymbol{\nu} \tag{8.3}$$

The quantity $\dot{\boldsymbol{\nu}}$ in the above equation is the robot's acceleration that can be derived from the equations of motion (2.37) as $\dot{\boldsymbol{\nu}} = \boldsymbol{M}^{-1}[\boldsymbol{B}\boldsymbol{\tau} + \boldsymbol{J}_c^\top\boldsymbol{f} - \boldsymbol{h}]$ where, $\boldsymbol{h} = \boldsymbol{C}(\boldsymbol{q},\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{G}(\boldsymbol{q})$. Using this relation in (8.3), we get

$$\ddot{\boldsymbol{x}}(t) = \boldsymbol{J}_x\boldsymbol{M}^{-1}[\boldsymbol{B}\boldsymbol{\tau} + \boldsymbol{J}_c^\top\boldsymbol{f} - \boldsymbol{h}] + \dot{\boldsymbol{J}}_x\,\boldsymbol{\nu}$$

$$\ddot{\boldsymbol{x}}(t) = \boldsymbol{J}_x\boldsymbol{M}^{-1}\boldsymbol{B}\boldsymbol{\tau} + \boldsymbol{J}_x\boldsymbol{M}^{-1}\boldsymbol{J}_c^\top\boldsymbol{f} - \boldsymbol{J}_x\boldsymbol{M}^{-1}\boldsymbol{h} + \dot{\boldsymbol{J}}_x\,\boldsymbol{\nu}$$

$$\boldsymbol{J}_x\boldsymbol{M}^{-1}\boldsymbol{B}\boldsymbol{\tau} = \ddot{\boldsymbol{x}}(t) - \boldsymbol{J}_x\boldsymbol{M}^{-1}\boldsymbol{J}_c^\top\boldsymbol{f} + \boldsymbol{J}_x\boldsymbol{M}^{-1}\boldsymbol{h} - \dot{\boldsymbol{J}}_x\,\boldsymbol{\nu}$$

$$\boldsymbol{\tau} = [\boldsymbol{J}_x\boldsymbol{M}^{-1}\boldsymbol{B}]^\dagger[\ddot{\boldsymbol{x}}(t) - \boldsymbol{J}_x\boldsymbol{M}^{-1}\boldsymbol{J}_c^\top\boldsymbol{f} + \boldsymbol{J}_x\boldsymbol{M}^{-1}\boldsymbol{h} - \dot{\boldsymbol{J}}_x\,\boldsymbol{\nu}]$$

Now, using the desired dynamics from Eq. (8.2), we compute the control torques as

$$\boldsymbol{\tau} = [\boldsymbol{J}_x\boldsymbol{M}^{-1}\boldsymbol{B}]^\dagger[\ddot{\boldsymbol{x}}^* - \boldsymbol{J}_x\boldsymbol{M}^{-1}\boldsymbol{J}_c^\top\boldsymbol{f} + \boldsymbol{J}_x\boldsymbol{M}^{-1}\boldsymbol{h} - \dot{\boldsymbol{J}}_x\,\boldsymbol{\nu}] \tag{8.5}$$

On putting in a compact form, we have:

$$\boldsymbol{\tau} = \boldsymbol{\Delta}^\dagger[\ddot{\boldsymbol{x}}^* - \boldsymbol{\Omega}\boldsymbol{f} + \boldsymbol{\Lambda}] + \boldsymbol{N_\Delta}\boldsymbol{\tau}_0 \tag{8.6}$$

where

- $\boldsymbol{\Delta} = \boldsymbol{J}_x\boldsymbol{M}^{-1}\boldsymbol{B} \in \mathbb{R}^{6\times n}$

- $\boldsymbol{\Omega} = \boldsymbol{J}_x\boldsymbol{M}^{-1}\boldsymbol{J}_c^\top \in \mathbb{R}^{6\times 6n_c}$

- $\boldsymbol{\Lambda} = \boldsymbol{J}_x\,\boldsymbol{M}^{-1}\boldsymbol{h} - \dot{\boldsymbol{J}}_x\,\boldsymbol{\nu} \in \mathbb{R}^6$

- $\boldsymbol{N_\Delta} \in \mathbb{R}^{n\times n}$ is the nullspace projector of $\boldsymbol{\Delta}$

- $\boldsymbol{\tau}_0 \in \mathbb{R}^n$ represent torques required to satisfy lower priority tasks in case of redundancy in joint torques

The control torques generated using the relation (8.6) eventually facilitate good trajectory tracking based on the proper choice of the gains $\boldsymbol{K}_P, \boldsymbol{K}_D \in \mathbb{R}^{6 \times 6}$. Another key detail to observer here is that the control torques generated completely cancel out any external wrench applied to the robot. Although it proves to be robust to unwanted external perturbations, classical feedback linearization control approach is fundamentally limiting for human-robot collaboration applications that require active collaboration between an external agent and a robot through intentional physical interactions. Unlike unexpected collisions, intentional physical interactions during a collaborative task are often helpful to achieve the common goal. Under such situations, it makes sense not to be agnostic to the external interaction but rather exploit when it is helpful. However, two important questions to be addressed mathematically are: 1) How to classify the interaction as helpful, and 2) How to exploit the helpful interaction. The following subsections answer these two important questions.

8.1.1 Helpful Interaction

To explore the idea of mathematically characterizing an interaction to be helpful, let us consider a simple toy problem with an object of negligible mass m as shown in Fig. 8.1. Assume a control input u that is applied to the object and an external interaction force f^x acting on the object.
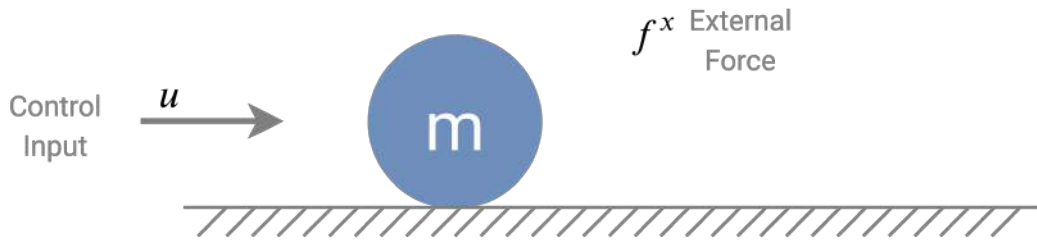
Fig. 8.1 A object of negligible mass

The equations of motion for the system relates the object acceleration $\ddot{x}$ to the control input u and the external force f^x, given by the following relation,

$$\ddot{x} = u + f^x \tag{8.7}$$

Let us consider a task of moving the object along a 1D trajectory. A simple trajectory tracking control objective can be defined as,

$$\ddot{x} = \ddot{x}^* := \ddot{x}_d - K_D \, \dot{\tilde{x}} - K_P \int_0^t \dot{\tilde{x}} \, du, \quad K_D, K_P > 0 \tag{8.8}$$

To understand the stability of the system, under the influence of the control input u and under the influence of the external force f^x, let us consider a simple Lyapunov function as below,

$$V = \frac{1}{2}\|\dot{x} - \dot{x}_d\|^2 + \frac{K_P}{2}\left\|\int_0^t (\dot{x} - \dot{x}_d) \, du\right\|^2 \tag{8.9}$$

On time differentiating the above function we get the relation,

$$\dot{V} = \dot{\tilde{x}} \, [\ddot{\tilde{x}} + K_P \int_0^t \dot{\tilde{x}} \, du]$$

Using the equation of acceleration $\ddot{x}$ in the above equation we get,

$$\dot{V} = \dot{\tilde{x}} \, [u + f^x - \ddot{x}_d + K_P \int_0^t \dot{\tilde{x}} \, du]$$

Given the task of trajectory tracking, we choose the control input u to be the desired dynamics i.e., $u = \ddot{x}^*$. Based on this choice for the control input, the derivative of the Lyapunov function becomes,

$$\dot{V} = \dot{\tilde{x}} f^x - \dot{\tilde{x}} K_D \dot{\tilde{x}}$$

The stability of the system is ensured as long as the derivative of the Lyapunov function is negative i.e., $\dot{V} \leq 0$. The term - $\dot{\tilde{x}} K_D \dot{\tilde{x}} \leq 0$. So, the stability of the system under the influence of the external force f^x is ensured as long as the term $\dot{\tilde{x}} f^x \leq 0$, and the interaction force can be characterized as a helpful interaction.

An intuitive understanding of characterizing an interaction force to be helpful can be better understood through the Fig. 8.2. Consider the Case I, when the current velocity is lagging behind the desired velocity. The velocity error $\dot{\tilde{x}} = \dot{x} - \dot{x}_d < 0$. Under such conditions, an external interaction force $f^x > 0$, applied in the positive direction, pushes the object towards the desired velocity. This interaction force f^x is considered to be helpful as the term $\dot{\tilde{x}} f^x < 0$. Similarly, in Case II, when the current velocity is leading the desired velocity. The velocity error $\dot{\tilde{x}} = \dot{x} - \dot{x}_d > 0$. Under such conditions,

an external interaction force $f^x < 0$, applied in the negative direction, pushes the object back towards the desired velocity. This interaction force f^x is considered to be helpful as the term $\dot{\tilde{x}} f^x < 0$.
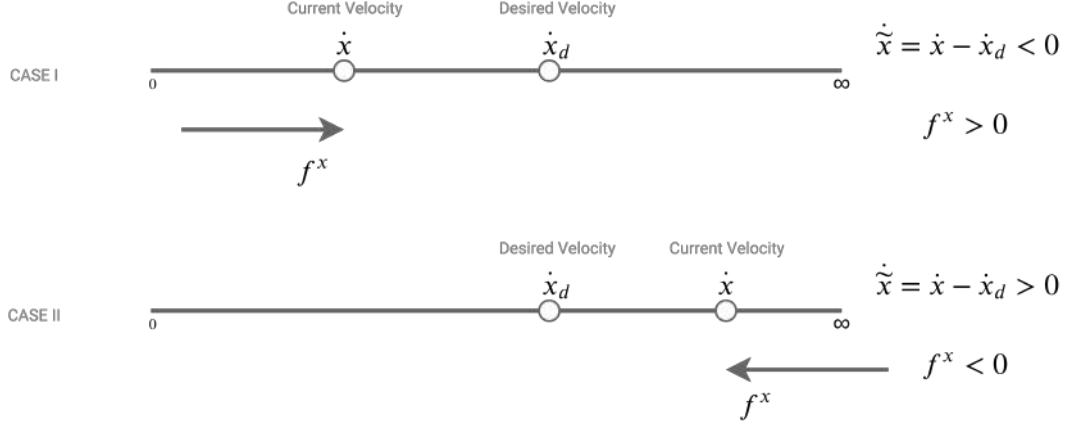
Fig. 8.2 Helpful interaction intuition

8.1.2 Interaction Exploitation

This section presents the idea and the intuition behind the exploitation of helpful interaction. Let us consider the Cartesian trajectory tracking task and consider the following control objective,

$$\ddot{\boldsymbol{x}} = \ddot{\boldsymbol{x}}^* := \ddot{\boldsymbol{x}}_d - \boldsymbol{K}_D \, \dot{\tilde{\boldsymbol{x}}} - \boldsymbol{K}_P \int_0^t \dot{\tilde{\boldsymbol{x}}} \, du + \boldsymbol{\Omega}\boldsymbol{f} \tag{8.10}$$

Similar to the Lyapunov stability analysis presented in subsection 8.1.1, let us consider a simple Lypunov function as below,

$$\mathrm{V} = \frac{1}{2}\|\dot{\boldsymbol{x}} - \dot{\boldsymbol{x}}_d\|^2 + \frac{\boldsymbol{K}_P}{2}\left\|\int_0^t (\dot{\boldsymbol{x}} - \dot{\boldsymbol{x}}_d) \, du\right\|^2 \tag{8.11}$$

On taking the time derivation of eq. (8.11), and using the desired dynamics following the control objective of eq. (8.10), we get

$$\dot{\mathrm{V}} = -\dot{\tilde{\boldsymbol{x}}} \, \boldsymbol{K}_D \, \dot{\tilde{\boldsymbol{x}}} + \dot{\tilde{\boldsymbol{x}}} \, \boldsymbol{\Omega}\boldsymbol{f} \tag{8.12}$$

Now, the stability of the system under the influence of the external interaction wrench $\boldsymbol{f}$ depends on the second term being negative, i.e., $\dot{\tilde{\boldsymbol{x}}} \, \boldsymbol{\Omega}\boldsymbol{f} \leq 0$. Let us decompose

the term $\boldsymbol{\Omega f}$ along the parallel and perpendicular directions of the velocity error as following,

$$\boldsymbol{\Omega f} = \alpha \ \dot{\tilde{\boldsymbol{x}}}^{\parallel} + \beta \ \dot{\tilde{\boldsymbol{x}}}^{\perp} \tag{8.13a}$$

$$\dot{\tilde{\boldsymbol{x}}}^{\parallel} = \frac{\dot{\tilde{\boldsymbol{x}}}}{\left\| \dot{\tilde{\boldsymbol{x}}} \right\|}, \ \ \alpha = \frac{\dot{\tilde{\boldsymbol{x}}}^{\top} \boldsymbol{\Omega} \ \boldsymbol{f}}{\left\| \dot{\tilde{\boldsymbol{x}}} \right\|} \tag{8.13b}$$

Following the helpful interaction characterization idea of section 8.1.1, the parallel wrench component α can be classified as a helpful interaction when $\alpha \leq 0$. Now, the idea of exploiting helpful interaction is achieved through considering the desired dynamics with the helpful interaction as following,

$$\ddot{\boldsymbol{x}}^* := \begin{cases} \ddot{\boldsymbol{x}}_d - \boldsymbol{K}_D \ \dot{\tilde{\boldsymbol{x}}} - \boldsymbol{K}_P \int_0^t \dot{\tilde{\boldsymbol{x}}} \ du & \text{if} \ \ \alpha > 0 \\ \ddot{\boldsymbol{x}}_d - \boldsymbol{K}_D \ \dot{\tilde{\boldsymbol{x}}} - \boldsymbol{K}_P \int_0^t \dot{\tilde{\boldsymbol{x}}} \ du + \alpha \left\| \dot{\tilde{\boldsymbol{x}}} \right\| & \text{if} \ \ \alpha \leq 0 \end{cases}$$

So, even when the feedback linearized robot torques generated through the relation from eq. (8.6) cancel any external interaction wrench, we can consider the helpful interaction ($\alpha \leq 0$) to be part of the desired dynamics and generate the robot torques by exploiting the helpful interaction. A preliminary investigation of the helpful interaction exploitation to generate robot control torques was carried in (Romano et al., 2018). However, the external interaction wrench considered is not directly measured through the force-torque sensors, but are estimated quantities through the robot control torques. This results in an algebraic loop, when the estimated interaction wrench is considered to generate the robot control torques by exploiting the helpful interaction. This chapter presents a more integrated formulation to achieve partner-aware control by leveraging the interaction in terms of the external agent joint torques.

Notational Nuances

The reader is advised to pay close attention to the notational nuances presented here in addition to the notion presented in the background notation (2.1). This notation is followed for the rest of this chapter and the associated Appendix B.

$$
\begin{array}{ll}
{}^{r}(.) & \text{Robot quantity (non-bold small letter)} \\
{}^{ea}(.) & \text{External Agent quantity (non-bold small letter)}
\end{array}
$$

Robot Agent Quantities

$$
\begin{array}{ll}
\mathbf{p} & \text{Vector (straight bold small letter)} \\
\mathbf{A} & \text{Matrix, tensor (straight bold capital letter)}
\end{array}
$$

External Agent Quantities

$$
\begin{array}{ll}
\mathbb{p} & \text{Vector (straight double-bold small letter)} \\
\mathbb{A} & \text{Matrix, tensor (straight double-bold capital letter)}
\end{array}
$$

Common/Coupled/Composite Quantities

$$
\begin{array}{ll}
\boldsymbol{p} & \text{vector (slanted-bold small letter)} \\
\boldsymbol{A} & \text{Matrix, tensor (slanted-bold capital letter)}
\end{array}
$$

8.2 Coupled Modeling

The external agent related terms are denoted with "$\mathtt{straight\ double-bold}$" notation, e.g., $\mathbb{M}$; robot related terms are denoted with "**straight bold**" notation, e.g., $\mathbf{M}$; and terms that are common/coupled/composite are denoted with ***slanted bold*** notation, e.g., $\boldsymbol{M}$.

A typical physical agent-robot interaction scenario where the external agent is a human is shown in Fig. 8.3. There are two agents: the external agent, and the robot. Both agents are physically interacting with the environment and, besides, are also engaged in physical interaction with each other.

In the first approximation, both the external agent (*ea*) and the robot agent (*r*) can be considered as multi rigid-body mechanical systems composed of ${}^{ea}n + 1$ and ${}^{r}n + 1$ rigid bodies respectively, called links, connected through ${}^{ea}n \in \mathbb{N}$ and ${}^{r}n \in \mathbb{N}$ joints with one degree of freedom. In case of the human being an external agent, even though the assumption of a human body being modeled as rigid bodies is far from reality, it serves as a rough approximation when formulating physical human-robot interaction dynamics and allows us to synthesize robot controllers optimizing both human and robot variables. Further, we consider both the external agent and the robot to be *free-floating* systems, i.e. none of the links have an *a priori* constant pose with respect to the inertial frame.
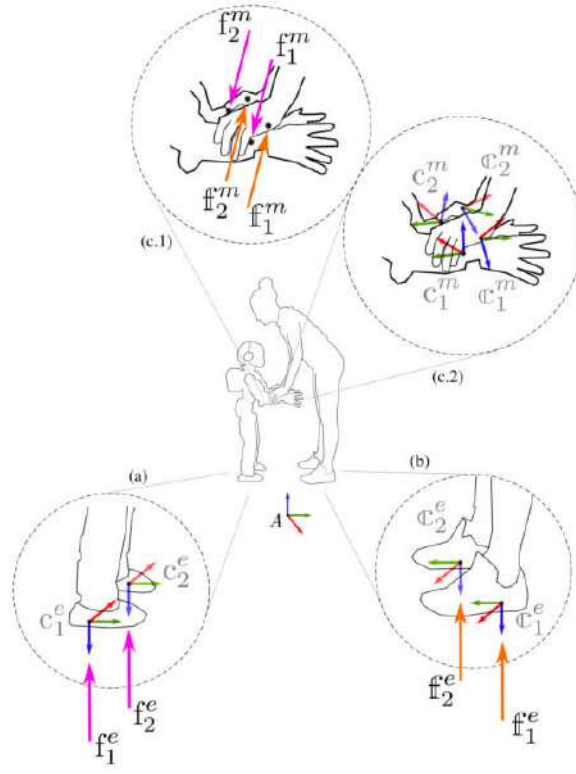
Fig. 8.3 A typical agent-robot dynamic interaction scenario

To the purpose of finding mathematical models for both the external agent and the robot, we apply the Euler-Poincaré formalism to both multi-body systems (Marsden and Ratiu, 2010). The equations of motion describing the dynamics of the external agent and the robotic agents respectively are:

$$\mathbb{M}(\mathbb{q})\dot{\nu} + \mathbb{C}(\mathbb{q}, \nu)\nu + \mathbb{G}(\mathbb{q}) = \mathbb{B}\bar{\tau} + \mathbb{J}_c^\top \mathbb{f} \tag{8.14a}$$

$$\mathbf{M}(\mathbf{q})\dot{\boldsymbol{\nu}} + \mathbf{C}(\mathbf{q}, \boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{G}(\mathbf{q}) = \mathbf{B}\boldsymbol{\tau} + \mathbf{J}_c^\top \mathbf{f} \tag{8.14b}$$

The number of degrees of freedom of all the joints for the external agent is denoted by $^{ea}n \in \mathbb{N}$ and $^r n \in \mathbb{N}$ for the robot agent. For the sake of clarity, the superscript is ignored when referring to a quantity that corresponds to both the agents. Another assumption is that each agent is subject to a total of $n_c = n_m + n_e \in \mathbb{N}$ distinct wrenches. These wrenches are composed of two subsets: the wrenches due to *mutual*

interaction, denoted with the subscript m and the wrenches exchanged between the *external agent* and the *environment*, denoted with the subscript e respectively, In either case, the contact wrenches are represented by:

$$\boldsymbol{f} = \begin{bmatrix} \boldsymbol{f}_1^m; & \boldsymbol{f}_2^m; & .... & \boldsymbol{f}_{n_m}^m; & \boldsymbol{f}_{n_m+1}^e; & \boldsymbol{f}_{n_m+2}^e; & .... & \boldsymbol{f}_{n_m}^e \end{bmatrix} \in \mathbb{R}^{6n_c}$$

Accordingly, $\mathbb{f} = \begin{bmatrix} \mathbb{f}_m & \mathbb{f}_e \end{bmatrix}^\top$ with $\mathbb{f}_m$ the external wrenches applied on the external agent by the robotic agent and $\mathbb{f}_e$ the external wrenches applied on the external agent by the environment. Similarly, $\mathbf{f} = \begin{bmatrix} \mathbf{f}_m & \mathbf{f}_e \end{bmatrix}^\top$ with $\mathbf{f}_m$ the external wrenches applied on the robotic agent by the external agent and $\mathbf{f}_e$ the external wrenches applied on the robotic agent by the environment.

A set of frames are defined: $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, ...\mathcal{C}_{n_m}, \mathcal{C}_{n_m+1}, \mathcal{C}_{n_m+2}, ..., \mathcal{C}_{n_e}\}$ and assume that the application point of the k-th external wrench on an agent is associated with a frame $\mathcal{C}_k \in \mathcal{C}$, attached to the agent's link on which the wrench acts, and has z-axis pointing in the direction normal to the contact plane. Furthermore, the external wrench $\boldsymbol{f}_k^{\mathrm{m/e}}$ is expressed in a frame whose orientation is that of the inertial frame $\mathcal{I}$, and whose origin is that of the frame $\mathcal{C}_k$.

Following the relation in Eq. (8.1), the jacobian $\boldsymbol{J}_{\mathcal{C}_k} = \boldsymbol{J}_{\mathcal{C}_k}(\boldsymbol{q})$ is the map between the agent's velocity $\nu = (^{\mathcal{I}}\boldsymbol{v}_{\mathcal{F}}, \dot{\boldsymbol{s}}) \in \mathbb{R}^{n+6}$ and the velocity of the frame $\mathcal{C}_k$ given by $^{\mathcal{I}}\boldsymbol{v}_{\mathcal{C}_k} = [^{\mathcal{I}}\dot{\boldsymbol{p}}_{\mathcal{C}_k}; \ ^{\mathcal{I}}\boldsymbol{\omega}_{\mathcal{C}_k}]$:

$$^{\mathcal{I}}\boldsymbol{v}_{\mathcal{C}_k} = \boldsymbol{J}_{\mathcal{C}_k} \ \nu \tag{8.15}$$

For a floating base system, the jacobian matrix has the following structure (Featherstone, 2007):

$$\boldsymbol{J}_{\mathcal{C}_k}(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{J}_{C_k}^b(\boldsymbol{q}) & \boldsymbol{J}_{\mathcal{C}_k}^j(\boldsymbol{q}) \end{bmatrix} \in \mathbb{R}^{6 \times n+6} \tag{8.16a}$$

$$\boldsymbol{J}_{\mathcal{C}_k}^b(\boldsymbol{q}) = \begin{bmatrix} \mathbf{1}_3 & -\mathbf{S}(^{\mathcal{I}}\boldsymbol{p}_{\mathcal{C}_k} - {}^{\mathcal{I}}\boldsymbol{p}_{\mathcal{F}}) \\ \mathbf{0}_{3\times 3} & \mathbf{1}_3 \end{bmatrix} \in \mathbb{R}^{6\times 6} \tag{8.16b}$$

8.2.1 Contact Constraints

We assume that holonomic constraints of the form $c(\boldsymbol{q}) = 0$ act on both the external agent and the robot during their interaction with the environment. The links that are in contact with the ground can be considered as end-effector links that are rigidly fixed to the ground for the duration of the contact and hence have zero velocity. Following equation (8.15), this can be represented as follows for the human and the robot respectively:

$$\mathbb{J}_{\mathbb{c}_k}\nu = 0 \tag{8.17a}$$

$$\mathbf{J}_{\mathbf{c}_k}\boldsymbol{\nu} = 0 \tag{8.17b}$$

Differentiating the above kinematic constraints yields:

$$\begin{bmatrix} \mathbb{J}^b_{\mathbb{c}_k} & \mathbb{J}^j_{\mathbb{c}_k} \end{bmatrix} \begin{bmatrix} \dot{\mathbb{V}}_B \\ \ddot{\mathbb{s}} \end{bmatrix} + \begin{bmatrix} \dot{\mathbb{J}}^b_{\mathbb{c}_k} & \dot{\mathbb{J}}^j_{\mathbb{c}_k} \end{bmatrix} \begin{bmatrix} \mathbb{V}_B \\ \dot{\mathbb{s}} \end{bmatrix} = 0 \tag{8.18a}$$

$$\begin{bmatrix} \mathbf{J}^b_{\mathbf{c}_k} & \mathbf{J}^j_{\mathbf{c}_k} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{v}}_B \\ \ddot{\mathbf{s}} \end{bmatrix} + \begin{bmatrix} \dot{\mathbf{J}}^b_{\mathbf{c}_k} & \dot{\mathbf{J}}^j_{\mathbf{c}_k} \end{bmatrix} \begin{bmatrix} \mathbf{v}_B \\ \dot{\mathbf{s}} \end{bmatrix} = 0 \tag{8.18b}$$

Now, during physical agent-robot interaction, there is a contact between the external agent and the robot. These contacts can be modeled as holonomic constraints of the form $c(\mathbb{q}, \mathbf{q}) = 0$. To this purpose, a frame $\mathbb{c}_m \in {}^{ea}\mathcal{C}$ attached to the external agent link which is in contact with the robot is considered. More precisely, let ${}^{\mathcal{I}}H_{\mathbb{c}_m}(\mathbb{q})$ denote the homogeneous transformation from $\mathbb{c}_m$ to the inertial frame. Similarly, another frame $\mathbf{c}_m \in {}^{r}\mathcal{C}$ attached to the robot link in contact with the external agent is considered. Let ${}^{\mathcal{I}}H_{\mathbf{c}_m}(\mathbf{q})$ denote the homogeneous transformation from $\mathbf{c}_m$ to the inertial frame. The relative transformation between the frames $\mathbb{c}_m$ and $\mathbf{c}_m$ is given by:

$$^{\mathbb{c}_m}H_{\mathbf{c}_m} = {}^{\mathcal{I}}H_{\mathbb{c}_m}^{-1}(\mathbb{q})\,{}^{\mathcal{I}}H_{\mathbf{c}_m}(\mathbf{q}) \tag{8.19}$$

When ${}^{\mathbb{c}_m}H_{\mathbf{c}_m}$ (or a part of it) is constant, it means that the external agent and the robot are in contact. By setting ${}^{\mathbb{c}_m}H_{\mathbf{c}_m}$ to a constant, the aforementioned holonomic constraint of the form $c(\mathbb{q}, \mathbf{q}) = 0$ is obtained. In general, there can be two kinds of contact between the external agent and the robot, either a point contact (Figure: 8.4a) or a plane contact (Figure: 8.4b).

(a) Point Contact   (b) Plane Contact
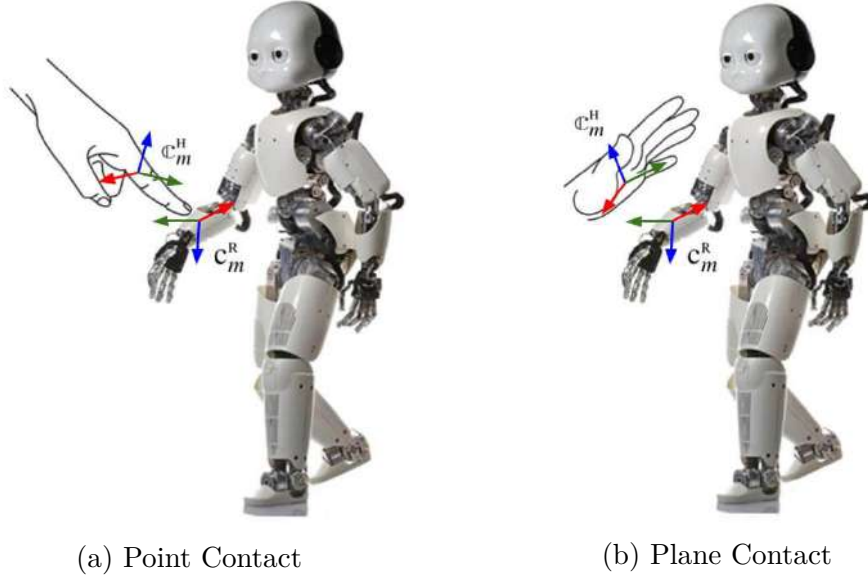
Fig. 8.4 Different contact types between agent-robot partners

A stable contact between the external agent and the robot during physical agent-robot interaction is considered, which leads to the condition that the relative velocity between the two frames $\mathbb{c}_m$ and $\mathbf{c}_m$ is zero, i.e., the two contact frames move with the same velocity with respect to the inertial frame as given by the following relation:

$$^{\mathcal{I}}\mathbb{v}_{\mathbb{c}_m} = \ ^{\mathbb{c}_m}X_{\mathbf{c}_m} \ ^{\mathcal{I}}\mathbf{v}_{\mathbf{c}_m} \tag{8.20}$$

where $^{\mathbb{c}_m}X_{\mathbf{c}_m}$ is an adjoint transformation matrix for 6D motion vectors. We consider the contact frames to be coinciding and hence, the transformation matrix is Identity i.e. $^{\mathbb{c}_m}X_{\mathbf{c}_m} = I_{6\times6}$

In light of the above, equation (8.20) can be written as:

$$^{\mathcal{I}}\mathbb{v}_{\mathbb{c}_m} = \ ^{\mathcal{I}}\mathbf{v}_{\mathbf{c}_m}, \tag{8.21}$$

which can be represented as follows

$$\mathbb{J}_{\mathbb{c}_m}\mathbb{v} = \mathbf{J}_{\mathbf{c}_m}\boldsymbol{\nu} \tag{8.22}$$

Differentiating the equation (8.22) we get,

$$
\begin{bmatrix} \mathbb{J}^b_{\mathbb{C}m} & \mathbb{J}^j_{\mathbb{C}m} \end{bmatrix} \begin{bmatrix} \dot{\mathbb{V}}_B \\ \dddot{\mathbb{s}} \end{bmatrix} + \begin{bmatrix} \dot{\mathbb{J}}^b_{\mathbb{C}m} & \dot{\mathbb{J}}^j_{\mathbb{C}m} \end{bmatrix} \begin{bmatrix} \mathbb{V}_B \\ \dot{\mathbb{s}} \end{bmatrix} =
$$

$$
\begin{bmatrix} \mathbf{J}^b_{\mathbf{c}m} & \mathbf{J}^j_{\mathbf{c}m} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{v}}_B \\ \ddot{\mathbf{s}} \end{bmatrix} + \begin{bmatrix} \dot{\mathbf{J}}^b_{\mathbf{c}m} & \dot{\mathbf{J}}^j_{\mathbf{c}m} \end{bmatrix} \begin{bmatrix} \mathbf{v}_B \\ \dot{\mathbf{s}} \end{bmatrix} \tag{8.23a}
$$

$$
\begin{bmatrix} \dot{\mathbb{J}}^b_{\mathbb{C}m} & \dot{\mathbb{J}}^j_{\mathbb{C}m} & -\dot{\mathbf{J}}^b_{\mathbf{c}m} & -\dot{\mathbf{J}}^j_{\mathbf{c}m} \end{bmatrix} \begin{bmatrix} \mathbb{V}_B \\ \dot{\mathbb{s}} \\ \mathbf{v}_B \\ \dot{\mathbf{s}} \end{bmatrix} +
$$

$$
\begin{bmatrix} \mathbb{J}^b_{\mathbb{C}m} & \mathbb{J}^j_{\mathbb{C}m} & -\mathbf{J}^b_{\mathbf{c}m} & -\mathbf{J}^j_{\mathbf{c}m} \end{bmatrix} \begin{bmatrix} \dot{\mathbb{V}}_B \\ \dddot{\mathbb{s}} \\ \dot{\mathbf{v}}_B \\ \ddot{\mathbf{s}} \end{bmatrix} = 0 \tag{8.23b}
$$

Furthermore, the constraint equations (8.18a) (8.18b) (8.23a) and (8.23b) can be represented in a compact form as follows:

$$
\boldsymbol{P}\dot{\boldsymbol{V}} + \dot{\boldsymbol{P}}\boldsymbol{V} = 0 \tag{8.24}
$$

where,

- $\boldsymbol{P} = \begin{bmatrix} \mathbb{J}^b_{\mathbb{C}k} & \mathbb{J}^j_{\mathbb{C}k} & 0 & 0 \\ 0 & 0 & \mathbf{J}^b_{\mathbf{c}k} & \mathbf{J}^j_{\mathbf{c}k} \\ \mathbb{J}^b_{\mathbb{C}m} & \mathbb{J}^j_{\mathbb{C}m} & -\mathbf{J}^b_{\mathbf{c}m} & -\mathbf{J}^j_{\mathbf{c}m} \end{bmatrix} \in \mathbb{R}^{6 \times (^{e_a}n + ^{r}n + 12)}$

- $\boldsymbol{V} = \begin{bmatrix} \mathbb{v} & \boldsymbol{\nu} \end{bmatrix}^{\top} \in \mathbb{R}^{^{e_a}n + ^{r}n + 12}$

8.2.2 Contact and Interaction Wrench

First, observe that Eq. (8.14a) and Eq. (8.14b) can be combined to obtain a single equation of motion for the composite system as presented in Eq. (8.25)

$$
\begin{bmatrix} \mathbb{M} & 0 \\ 0 & \mathbf{M} \end{bmatrix} \begin{bmatrix} \dot{\nu} \\ \dot{\boldsymbol{\nu}} \end{bmatrix} + \begin{bmatrix} \mathbb{h} \\ \mathbf{h} \end{bmatrix} = \begin{bmatrix} \mathbb{B} & 0 \\ 0 & \mathbf{B} \end{bmatrix} \begin{bmatrix} \mathbb{\tau} \\ \boldsymbol{\tau} \end{bmatrix} + \begin{bmatrix} \mathbb{J}_c^\top & 0 \\ 0 & \mathbf{J}_c^\top \end{bmatrix} \begin{bmatrix} \mathbb{f} \\ \mathbf{f} \end{bmatrix} \tag{8.25}
$$

where, $\mathbb{h} = \mathbb{C}(\mathbb{q}, \nu)\nu + \mathbb{G}(\mathbb{q})$, $\mathbf{h} = \mathbf{C}(\mathbf{q}, \boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{G}(\mathbf{q})$

According to Newtonian mechanics, in the case of rigid contacts, the perturbations exerted by the robot on the external agent is equal and opposite to the perturbation exerted by the external agent on the robot. As a consequence, when the external wrenches are expressed with respect to the inertial frame $\mathcal{I}$, the interaction wrenches $\boldsymbol{f}_m$ can be written as follows:

$$
\boldsymbol{f}_m \;=\; \mathbb{f}_m \;=\; -\mathbf{f}_m \tag{8.26}
$$

Furthermore, Eq. (8.25) can be written in a compact form as follows:

$$
\boldsymbol{M}\dot{\boldsymbol{V}} + \boldsymbol{h} = \boldsymbol{B}\tau + \boldsymbol{J}_c^\top \boldsymbol{f} \tag{8.27}
$$

where $\tau = \begin{bmatrix} \mathbb{\tau} & \boldsymbol{\tau} \end{bmatrix}^\top \in \mathbb{R}^{{}^{ea}n + {}^{r}n}$; $\boldsymbol{f} = \begin{bmatrix} \boldsymbol{f}_m & \mathbb{f}_e & \mathbf{f}_e \end{bmatrix}^\top \in \mathbb{R}^{6 \times (n_m + {}^{ea}n_e + {}^{r}n_e)}$ and $\boldsymbol{J}_c$ is a proper contact jacobian matrix. This equation implies that

$$
\dot{\boldsymbol{V}} = \boldsymbol{M}^{-1}[\boldsymbol{B}\tau + \boldsymbol{J}_c^\top \boldsymbol{f} - \boldsymbol{h}] \tag{8.28}
$$

Making use of Eq. (8.28) in the constraint equation Eq. (8.24)

$$
\dot{\boldsymbol{P}}\boldsymbol{V} + \boldsymbol{P}\boldsymbol{M}^{-1}[\boldsymbol{B}\tau + \boldsymbol{J}_c^\top \boldsymbol{f} - \boldsymbol{h}] = 0
$$

$$
\Rightarrow \boldsymbol{P}\boldsymbol{M}^{-1}\boldsymbol{J}_c^\top \boldsymbol{f} = -[\boldsymbol{P}\boldsymbol{M}^{-1}[\boldsymbol{B}\tau - \boldsymbol{h}] + \dot{\boldsymbol{P}}\boldsymbol{V}]
$$

$$
\Rightarrow \boldsymbol{f} = -\boldsymbol{\Gamma}^\dagger[\boldsymbol{P}\boldsymbol{M}^{-1}[\boldsymbol{B}\tau - \boldsymbol{h}] + \dot{\boldsymbol{P}}\boldsymbol{V}]
$$

where, $\boldsymbol{\Gamma} = \boldsymbol{P}\boldsymbol{M}^{-1}\boldsymbol{J}_c^\top$

Furthermore,

$$
\boldsymbol{f} = -\boldsymbol{\Gamma}^\dagger \boldsymbol{P}\boldsymbol{M}^{-1}\boldsymbol{B}\tau + \boldsymbol{\Gamma}^\dagger \boldsymbol{P}\boldsymbol{M}^{-1}\boldsymbol{h} - \boldsymbol{\Gamma}^\dagger \dot{\boldsymbol{P}}\boldsymbol{V} \tag{8.30}
$$

Through coupled-dynamics, Eq. (8.30) shows that the external wrenches are a function of the agent's configuration $\mathbb{q}$, $\mathbf{q}$, velocity ν, $\boldsymbol{\nu}$, and joint torques $\mathbb{\tau}$, $\boldsymbol{\tau}$. This

can be represented as a function $\boldsymbol{f} = g(\mathbb{q}, \mathbf{q}, \mathcal{v}, \boldsymbol{\nu}, \mathcal{\tau}, \boldsymbol{\tau})$. This relation can be further decomposed as,

$$\boldsymbol{f} = \boldsymbol{G}_1 \mathcal{\tau} + \boldsymbol{G}_2 \boldsymbol{\tau} + \boldsymbol{G}_3(\mathbb{q}, \mathbf{q}, \mathcal{v}, \boldsymbol{\nu}) \tag{8.31}$$

where,

- $\boldsymbol{G}_1 \in \mathbb{R}^{6(n_m + {}^{ea}n_e + {}^{r}n_e) \times {}^{ea}n}$

- $\boldsymbol{G}_2 \in \mathbb{R}^{6(n_m + {}^{ea}n_e + {}^{r}n_e) \times {}^{r}n}$

- $\boldsymbol{G}_3 \in \mathbb{R}^{6(n_m + {}^{ea}n_e + {}^{r}n_e)}$

8.3 Partner-Aware Robot Control

As presented in section 8.1, the robot joint torques from Eq. (8.6) are agnostic to the interaction wrench from the external agent which are eventually compensated by considering the system dynamics. Hence, the robot behavior is invariant with respect to the external agent interaction. Certainly, instead of completely canceling out any external interaction by the feedback control action, it is gainful and desirable to *exploit* external interaction to accomplish the robot's task. This poses, however, the question of characterizing and quantifying help from the external agent with respect to the robot's task. In a previous work (Romano et al., 2018), the interaction wrench *estimates* from the robot is used as intent information from the external agent who is a human. However, in a coupled system, wrench information introduces an algebraic loop in the control design as the wrench estimates from the robot are computed using the robot joint torques (Nori et al., 2015c). Instead, another beneficial approach is to leverage the joint torques of the external agent. In the case of human they are largely self-generated and self-regulated. Additionally, considering the joint torques of a human agent opens new possibilities for investigating and optimizing human ergonomics. Also, considering the joint torques of an external agent that is a robot will help formulate robust control techniques for the combined system to achieve a collaborative task.

The following lemma proposes a partner-aware control law that exploits the external agent contribution towards the achievement of the robot control objective, thus actively taking into account the physical agent-robot interaction. The associated analysis is based on the *energy* of the robot's control task.

Lemma 1. *Assume that the control objective is to asymptotically stabilize the following point*

$$\left(\widetilde{\boldsymbol{\chi}}, \int \widetilde{\boldsymbol{\chi}} \ ds \right) = (0, 0) \tag{8.32}$$

Apply the following robot torques to the robot system (8.14b)

$$\boldsymbol{\tau} = -\boldsymbol{\Delta}^{\dagger} \ [\ \boldsymbol{\Lambda} \ + \ \mathbf{K}_D \ \widetilde{\boldsymbol{\chi}} \ + \ max(0, \alpha) \ \widetilde{\boldsymbol{\chi}}^{\parallel} \ ] + \mathbf{N}_{\boldsymbol{\Delta}} \boldsymbol{\tau}_0 \tag{8.33}$$

with

- $\boldsymbol{\Delta} = \mathbf{K}_d \ \mathbf{J}_{\chi} \mathbf{M}^{-1} [\mathbf{B} + \mathbf{J}_c^{\top} \bar{\boldsymbol{G}}_2] \in \mathbb{R}^{6 \times n}$

- $\mathbf{N}_{\boldsymbol{\Delta}}$ *the null-space projector of the matrix* $\boldsymbol{\Delta}$

- $\boldsymbol{\tau}_0$ *a free* $^r n-$*dimensional vector*

- $\boldsymbol{\Lambda} = \mathbf{K}_d \ [\ [\ \mathbf{J}_{\chi} \mathbf{M}^{-1} \mathbf{J}_c^{\top} \ ] \bar{\boldsymbol{G}}_3(\mathbb{q}, \mathbf{q}, \mathbb{\nu}, \boldsymbol{\nu}) - \mathbf{J}_{\chi} \mathbf{M}^{-1} \mathbf{h} \ + \ \dot{\mathbf{J}}_{\chi} \ \boldsymbol{\nu} - \dot{\boldsymbol{\chi}}_d \ ] + \mathbf{K}_p \ \int_0^t (\boldsymbol{chi} - \boldsymbol{\chi}_d) \ ds \in \mathbb{R}^6$

- $\mathbf{K}_D \in \mathbb{R}^{6 \times 6}$ *is a symmetric, positive-definite matrix*

- $\alpha \in \mathbb{R}$ *is a component proportional to the external agent joint torques* $\overline{\tau}$ *projected along* $\widetilde{\boldsymbol{\chi}}^{\parallel}$ *i.e., the direction parallel to* $\widetilde{\boldsymbol{\chi}}$

Assume that the matrix $\boldsymbol{\Delta}$ *is full rank matrix* $\forall \ t \in \mathbb{R}^+,$

- *The trajectories* $(\widetilde{\boldsymbol{\chi}}, \int \widetilde{\boldsymbol{\chi}} \ ds)$ *are globally bounded*

- *The equilibrium point* (8.32) *is stable*

Proof. The stability of $\widetilde{\boldsymbol{\chi}}$ from Eq. (8.32) can be analyzed by considering the following Lyapunov function:

$$\mathrm{V} = \frac{\mathbf{K}_d}{2} \|\boldsymbol{\chi} - \boldsymbol{\chi}_d\|^2 + \frac{\mathbf{K}_p}{2} \left\| \int_0^t (\boldsymbol{\chi} - \boldsymbol{\chi}_d) \ ds \right\|^2 \tag{8.34}$$

where $\mathbf{K}_d, \mathbf{K}_p \in \mathbb{R}^{p \times p}$ are two symmetric, positive-definite matrices. Now, on differentiating V, we get:

$$\dot{\mathrm{V}} = \mathbf{K}_d \ (\boldsymbol{\chi} - \boldsymbol{\chi}_d)^{\top} \ (\dot{\boldsymbol{\chi}} - \dot{\boldsymbol{\chi}}_d) + \mathbf{K}_p \ \int_0^t (\boldsymbol{\chi} - \boldsymbol{\chi}_d)^{\top} ds \ (\boldsymbol{\chi} - \boldsymbol{\chi}_d)$$

$$\Rightarrow \dot{V} = \mathbf{K}_d \, \widetilde{\boldsymbol{\chi}}^\top \, (\dot{\boldsymbol{\chi}} - \dot{\boldsymbol{\chi}}_d) + \mathbf{K}_p \, (\boldsymbol{\chi} - \boldsymbol{\chi}_d)^\top \int_0^t (\boldsymbol{\chi} - \boldsymbol{\chi}_d) \, ds$$

$$\Rightarrow \dot{V} = \widetilde{\boldsymbol{\chi}}^\top \, [\mathbf{K}_d \, (\dot{\boldsymbol{\chi}} - \dot{\boldsymbol{\chi}}_d) + \mathbf{K}_p \int_0^t (\boldsymbol{\chi} - \boldsymbol{\chi}_d) \, ds]$$

$$\Rightarrow \dot{V} = \widetilde{\boldsymbol{\chi}}^\top \, [\mathbf{K}_d \, \dot{\boldsymbol{\chi}} + \mathbf{K}_p \int_0^t (\boldsymbol{\chi} - \boldsymbol{\chi}_d) \, ds - \mathbf{K}_d \, \dot{\boldsymbol{\chi}}_d] \tag{8.36}$$

Following the relation in Eq. (8.1), on differentiating the link velocity $\boldsymbol{\chi}$ we have the following relation:

$$\dot{\boldsymbol{\chi}} = \mathbf{J}_{\boldsymbol{\chi}} \, \dot{\boldsymbol{\nu}} + \dot{\mathbf{J}}_{\boldsymbol{\chi}} \, \boldsymbol{\nu} \tag{8.37}$$

Using equation (8.37) in equation (8.36) we can write:

$$\dot{V} = \widetilde{\boldsymbol{\chi}}^\top \, [\mathbf{K}_d \, [\mathbf{J}_{\boldsymbol{\chi}} \, \dot{\boldsymbol{\nu}} + \dot{\mathbf{J}}_{\boldsymbol{\chi}} \, \boldsymbol{\nu}] + \mathbf{K}_p \int_0^t (\boldsymbol{\chi} - \boldsymbol{\chi}_d) \, ds - \mathbf{K}_d \, \dot{\boldsymbol{\chi}}_d] \tag{8.38}$$

The quantity $\dot{\boldsymbol{\nu}}$ in equation (8.38) is the robot's acceleration. Following the equation (8.14b) $\dot{\boldsymbol{\nu}}$ can be written as follows:

$$\dot{\boldsymbol{\nu}} = \mathbf{M}^{-1}[\mathbf{B}\boldsymbol{\tau} + \mathbf{J}_c^\top \mathbf{f} - \mathbf{h}] \tag{8.39}$$

In equation (8.39) $\mathbf{f}$ can be extracted from the equation (8.31) by selecting the first and the last rows which can be represented as $\mathbf{f} = \bar{\boldsymbol{G}}_1 \bar{\tau} + \bar{\boldsymbol{G}}_2 \boldsymbol{\tau} + \bar{\boldsymbol{G}}_3(\mathbb{q}, \mathbf{q}, \mathbb{v}, \boldsymbol{\nu})$. Now, equation (8.39) can be written as,

$$\dot{\boldsymbol{\nu}} = \mathbf{M}^{-1}[\mathbf{B}\boldsymbol{\tau} + \mathbf{J}_c^\top [\bar{\boldsymbol{G}}_1 \bar{\tau} + \bar{\boldsymbol{G}}_2 \boldsymbol{\tau} + \bar{\boldsymbol{G}}_3(\mathbb{q}, \mathbf{q}, \mathbb{v}, \boldsymbol{\nu})] - \mathbf{h}] \tag{8.40}$$

$$\Rightarrow \dot{\boldsymbol{\nu}} = \mathbf{M}^{-1}[\mathbf{B}\boldsymbol{\tau} + \mathbf{J}_c^\top \bar{\boldsymbol{G}}_1 \bar{\tau} + \mathbf{J}_c^\top \bar{\boldsymbol{G}}_2 \boldsymbol{\tau} + \mathbf{J}_c^\top \bar{\boldsymbol{G}}_3(\mathbb{q}, \mathbf{q}, \bar{\tau}, \boldsymbol{\tau}) - \mathbf{h}]$$

$$\Rightarrow \dot{\boldsymbol{\nu}} = \mathbf{M}^{-1}[\mathbf{J}_c^\top \bar{\boldsymbol{G}}_1 \bar{\tau} + [\mathbf{B} + \mathbf{J}_c^\top \bar{\boldsymbol{G}}_2]\boldsymbol{\tau} + \mathbf{J}_c^\top \bar{\boldsymbol{G}}_3(\mathbb{q}, \mathbf{q}, \mathbb{v}, \boldsymbol{\nu}) - \mathbf{h}]$$

$$\Rightarrow \dot{\boldsymbol{\nu}} = [\mathbf{M}^{-1}\mathbf{J}_c^\top \bar{\boldsymbol{G}}_1]\bar{\tau} + [\mathbf{M}^{-1}[\mathbf{B} + \mathbf{J}_c^\top \bar{\boldsymbol{G}}_2]]\boldsymbol{\tau} + [\mathbf{M}^{-1}\mathbf{J}_c^\top]\bar{\boldsymbol{G}}_3(\mathbb{q}, \mathbf{q}, \mathbb{v}, \boldsymbol{\nu}) - \mathbf{M}^{-1}\mathbf{h} \tag{8.41}$$

We have $\dot{\boldsymbol{\nu}}$ in equation (8.38) which can be replaced with equation (8.41) leading to the following relation:

$$\dot{V} = \tilde{\boldsymbol{\chi}}^\top [\mathbf{K}_d\,\mathbf{J}_{\boldsymbol{\chi}}[\,[\mathbf{M}^{-1}\mathbf{J}_c^\top\bar{\boldsymbol{G}}_1]\tau + [\,\mathbf{M}^{-1}[\mathbf{B}+\mathbf{J}_c^\top\bar{\boldsymbol{G}}_2]\,]\boldsymbol{\tau}+$$
$$[\mathbf{M}^{-1}\mathbf{J}_c^\top]\bar{\boldsymbol{G}}_3(\mathbb{q},\mathbf{q},\mathbb{\nu},\boldsymbol{\nu})-\mathbf{M}^{-1}\mathbf{h}\,]$$
$$+\mathbf{K}_d\,\dot{\mathbf{J}}_{\boldsymbol{\chi}}\boldsymbol{\nu} + \mathbf{K}_p\,\int_0^t(\boldsymbol{\chi}-\boldsymbol{\chi}_d)\,ds - \mathbf{K}_d\,\dot{\boldsymbol{\chi}}_d]$$

$$\Rightarrow \dot{V} = \tilde{\boldsymbol{\chi}}^\top [\mathbf{K}_d\,[\mathbf{J}_{\boldsymbol{\chi}}\mathbf{M}^{-1}\mathbf{J}_c^\top\bar{\boldsymbol{G}}_1]\tau + [\mathbf{K}_d\,\mathbf{J}_{\boldsymbol{\chi}}\mathbf{M}^{-1}[\mathbf{B}+\mathbf{J}_c^\top\bar{\boldsymbol{G}}_2]\,]\boldsymbol{\tau}+$$
$$[\mathbf{K}_d\,\mathbf{J}_{\boldsymbol{\chi}}\mathbf{M}^{-1}\mathbf{J}_c^\top]\bar{\boldsymbol{G}}_3(\mathbb{q},\mathbf{q},\mathbb{\nu},\boldsymbol{\nu})-$$
$$\mathbf{K}_d\,\mathbf{J}_{\boldsymbol{\chi}}\mathbf{M}^{-1}\mathbf{h}+$$
$$\mathbf{K}_d\,\dot{\mathbf{J}}_{\boldsymbol{\chi}}\,\boldsymbol{\nu} + K_p\,\int_0^t(\boldsymbol{\chi}-\boldsymbol{\chi}_d)ds - \mathbf{K}_d\,\dot{\boldsymbol{\chi}}_d]$$

This can be written in a compact form as follows:

$$\Rightarrow \dot{V} = \tilde{\boldsymbol{\chi}}^\top\,\boldsymbol{\Omega}\,\tau + \tilde{\boldsymbol{\chi}}^\top\,[\,\boldsymbol{\Delta}\,\boldsymbol{\tau} + \boldsymbol{\Lambda}\,] \tag{8.42}$$

where,

- $\boldsymbol{\Omega} = \mathbf{K}_d\,\mathbf{J}_{\boldsymbol{\chi}}\mathbf{M}^{-1}\mathbf{J}_c^\top\bar{\boldsymbol{G}}_1 \in \mathbb{R}^{p\times^{ea}n}$

- $\boldsymbol{\Delta} = \mathbf{K}_d\,\mathbf{J}_{\boldsymbol{\chi}}\mathbf{M}^{-1}[\mathbf{B}+\mathbf{J}_c^\top\bar{\boldsymbol{G}}_2] \in \mathbb{R}^{p\times^r n}$

- $\boldsymbol{\Lambda} = \mathbf{K}_d\,[[\mathbf{J}_{\boldsymbol{\chi}}\mathbf{M}^{-1}\mathbf{J}_c^\top]\bar{\boldsymbol{G}}_3(\mathbb{q},\mathbf{q},\mathbb{\nu},\boldsymbol{\nu})-\mathbf{J}_{\boldsymbol{\chi}}\mathbf{M}^{-1}\mathbf{h}+\dot{\mathbf{J}}_{\boldsymbol{\chi}}\,\boldsymbol{\nu}-\dot{\boldsymbol{\chi}}_d]+\mathbf{K}_p\,\int_0^t(\boldsymbol{\chi}-\boldsymbol{\chi}_d)\,ds \in \mathbb{R}^p$

The stability of $\boldsymbol{\chi}$ is ensured when $\dot{V} \leq 0$, .i.e.,

$$\tilde{\boldsymbol{\chi}}^\top\,\boldsymbol{\Omega}\,\tau + \tilde{\boldsymbol{\chi}}^\top\,[\,\boldsymbol{\Delta}\,\boldsymbol{\tau} + \boldsymbol{\Lambda}\,] \leq 0 \tag{8.43}$$

Now, the quantity $\boldsymbol{\Omega}\,\tau^{\text{H}}$ comprising human joint torques can be decomposed into two orthogonal components according to the following relation:

$$\boldsymbol{\Omega}\,\tau = \alpha\,\tilde{\boldsymbol{\chi}}^{\parallel} + \beta\,\tilde{\boldsymbol{\chi}}^{\perp} \tag{8.44a}$$

$$\tilde{\boldsymbol{\chi}}^{\parallel} = \frac{\tilde{\boldsymbol{\chi}}}{\|\tilde{\boldsymbol{\chi}}\|}, \;\; \alpha = \frac{\tilde{\boldsymbol{\chi}}^\top\boldsymbol{\Omega}\,\tau}{\|\tilde{\boldsymbol{\chi}}\|} \tag{8.44b}$$

where α and β are the components of $\boldsymbol{\Omega} \, \tau$. On using the decomposition of equation (8.44), we can write equation (8.42) as:

$$\dot{V} = \widetilde{\boldsymbol{\chi}}^\top \, \alpha \, \widetilde{\boldsymbol{\chi}}^\| + \widetilde{\boldsymbol{\chi}}^\top \, [\, \boldsymbol{\Delta} \, \boldsymbol{\tau} + \, \boldsymbol{\Lambda} \,] \tag{8.45}$$

Note that the component β is projected in the perpendicular direction of $\widetilde{\boldsymbol{\chi}}$ and hence it does not contribute any value to $\dot{V}$. Using the above control law (8.33)) in equations (8.45) leads to the following relation:

$$\dot{V} = - \, \widetilde{\boldsymbol{\chi}}^\top \, \mathbf{K}_D \, \widetilde{\boldsymbol{\chi}} \, - \, \widetilde{\boldsymbol{\chi}}^\top \, max(0, \alpha) \, \widetilde{\boldsymbol{\chi}}^\| \tag{8.46}$$

The fact that the human joint torques help the robot to perform a control action is encompassed in the right hand side of above equation: a positive α makes the Lyapunov function decrease faster. Thus the control law (8.33) ensures that $\dot{V} \leq 0$ which proves that the trajectories are globally bounded. From lyapunov theory, as $\dot{V} \leq 0$ in the neighborhood of the point $(0, 0)$ the equilibrium point (8.32) is stable.

It is important to observe that Eq. (8.33) depends upon the external agent joint torques. In fact, the scalar α represents the projection of the external agent joint torques into the desired robot direction. As a consequence, all human joint torques that will make the energy function (8.34) decrease faster are not canceled out from the feedback control action.

Observe also that the results in Lemma 1 do not include the convergence to zero of the robot equilibrium point. This additional property of the control laws (8.33) is currently being investigated, and requires additional properties on the human control system for the application of the Barbalat's lemma. In fact, to ensure this convergence, the next step of the proof is to show that the Lyapunov function (8.34) has a bounded second order time derivative, i.e. $|\ddot{V}| < c$, which requires additional assumptions on the human motion currently being investigated.

Finally, observe that the control laws (8.33) include a degree of redundancy under the assumption that the matrix $\boldsymbol{\Delta}$ is fat, i.e. the dimension of the robot control task is lower than the robot actuation. As a consequence, the free vector $\boldsymbol{\tau}_0$ can be used for other control purposes.

$\square$

8.4 Whole-Body Standup Controller

The whole-body standup controller is based on momentum control that is briefly described in Appendix B. The main control variable is the momentum of the robot. So, Eq. (8.1) becomes,

$$\mathbf{L} = \mathbf{J}_{cmm}(\mathbf{q})\,\boldsymbol{\nu} \tag{8.47}$$

where, $\mathbf{J}_{cmm}$ is the centroidal momentum matrix. The primary task of the robot we considered is to perform a standup motion by moving its center of mass through the sit-to-stand transition with momentum control as the primary control objective. A Matlab Simulink controller using a whole-body toolbox[1] (Romano et al., 2017) is implemented with the external agent and robot as the main sub systems as shown in Fig. 8.5.

Fig. 8.5 Whole-body standup controller implemented in Matlab Simulink

---

[1] https://github.com/robotology/wb-toolbox

8.4.1 Robot Subsystem

The robot subsystem is composed of three main components as highlighted in Fig. 8.6.
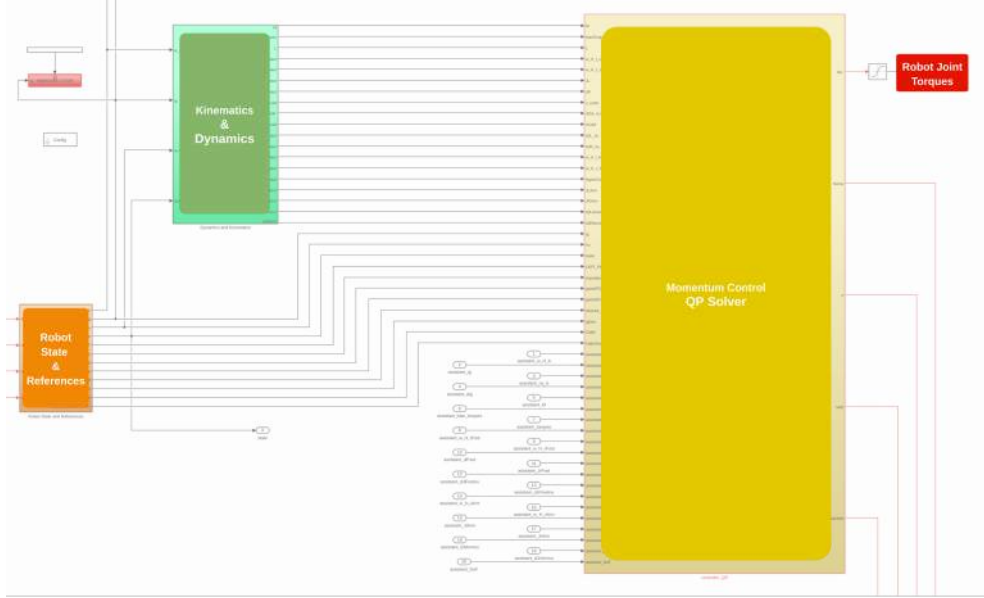

Fig. 8.6 Main components of whole-body standup controller robot subsystem implemented in Matlab Simulink

Robot State & References

This subsystem constitutes of a state machine that guides the robot behavior. There are four states for the robot as highlighted in Fig. 8.7. At each state a reference trajectory to the center of mass is generated through a minimum-jerk trajectory generator (Pattacini et al., 2010a). During the state 1, the robot balances on a chair and enters to state 2 when an interaction wrench of a set threshold is detected at the hands indicating the start of pull-up assistance from an external agent. During state 2, the robot moves its center of mass forwards and enters state 3 when the external wrench experienced at the feet of the robot is above a set threshold. During state 3, the robot moves its center of mass both in forward and upward directions and enters state 4 when the external wrench experienced at the feet of the robot are above another set threshold. Finally, during state 4 the robot moves its center of mass further upward to stand fully erect on both the feet. Accordingly, during the states 1 and 2 the contacts with the environment (chair), accounted in the controller, are at the *upper legs* of

the robot. Similarly, for the states 3 and 4 the *feet* contacts of the robot with the environment (ground) are accounted in the controller.
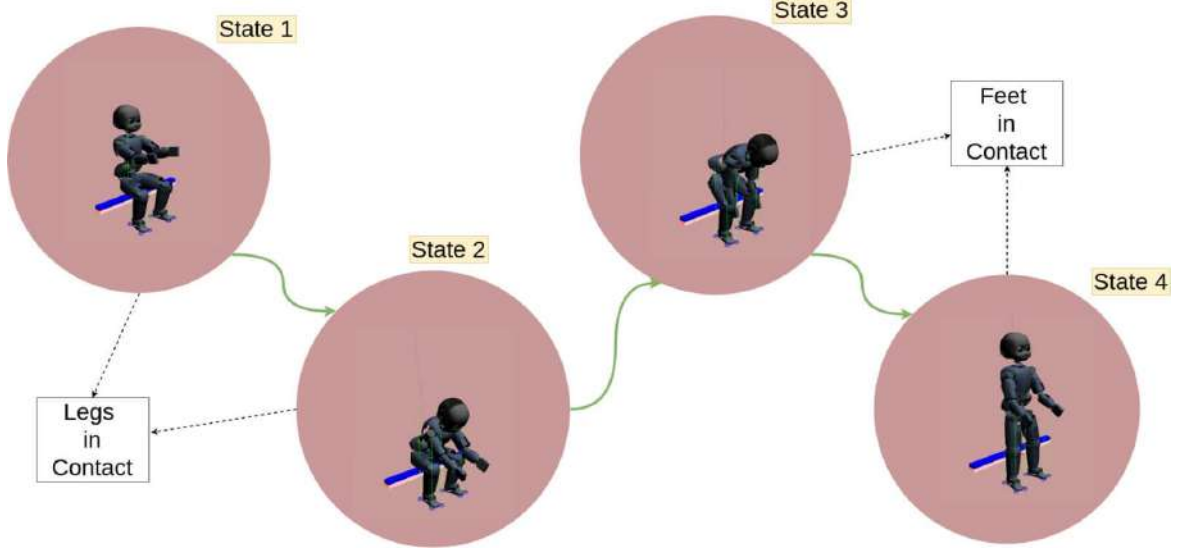

Fig. 8.7 State machine of whole-body standup controller

Kinematics & Dynamics

This subsystem computes the contact kinematics described by Eq. 8.24 and dynamic quantities like mass matrix $\mathbf{M}$ and bias forces $\mathbf{h}$ depending on the state of the robot.

Momentum Control QP Solver

This subsystem implements the momentum control objective as an optimization problem represented by the system of equations (8.48)-(8.49) that is solved through an open source quadratic programming solver qpOASES (Ferreau et al., 2014). The objective is to find the forces at the feet of the robot (8.48a) while subject to the constraints of the feet friction cones (8.48b) and the momentum control objective (8.48c). Having obtained the optimal contact forces $\mathbf{f}^*_{feet}$ at the feet of the robot, the robot control torques $\boldsymbol{\tau}^*$ are evaluated according to (8.49) subject to the constraints of the contact (8.49a), robot dynamics (8.49b) and the postural task (8.49c).

$$\mathbf{f}^*_{feet} = \underset{\mathbf{f}_{feet}}{\operatorname{argmin}} |\boldsymbol{\tau}^*(\mathbf{f})| \tag{8.48a}$$

$$s.t.$$

$$\mathbf{Af} < \mathbf{b} \tag{8.48b}$$

$$\dot{\mathbf{L}}(\mathbf{f}, \alpha) = \dot{\mathbf{L}}^* \tag{8.48c}$$

$$\boldsymbol{\tau}^*(\mathbf{f}) = \underset{\boldsymbol{\tau}}{\operatorname{argmin}} |\boldsymbol{\tau}(\mathbf{f}) - \boldsymbol{\tau}_0(\mathbf{f})| \tag{8.49}$$

$$s.t.$$

$$\boldsymbol{P}\dot{\boldsymbol{V}} + \dot{\boldsymbol{P}}\boldsymbol{V} = 0 \tag{8.49a}$$

$$\dot{\boldsymbol{\nu}} = \mathbf{M}^{-1}[\mathbf{B}\boldsymbol{\tau} + \mathbf{J}_c^\top \mathbf{f} - \mathbf{h}] \tag{8.49b}$$

$$\boldsymbol{\tau}_0 = \mathbf{h}_j - \mathbf{J}_j^\top \mathbf{f} - \mathbf{K}_p^j(\mathbf{s} - \mathbf{s}^d) - \mathbf{K}_d^j \dot{\mathbf{s}} \tag{8.49c}$$

8.5 Experiments

Considering the human model as a multi-body mechanical system of rigid links allows us to use another humanoid robot in place of a human as an external agent for the experiment without the loss of integrity of the experiment to validate our framework. So, we designed an experimental scenario with two iCub humanoid robots as shown in Fig. 8.8.

The *purple* iCub robot is run in torque control mode and receives torque inputs from the controller for the standup task. The *green* iCub robot is in position control mode. A predetermined trajectory generated using a minimum-jerk trajectory generator (Pattacini et al., 2010c) is given as a reference to the torso pitch, shoulder pitch and elbow joints of the *green* iCub robot. The resulting motion mimics the pull-up assistance to the *purple* iCub robot for performing the standup task. Hence, the *green* iCub robot is considered as an external interacting agent whose joint motion is indicated in Fig. 8.9 and the associated joint torques are shown in Fig. 8.10.
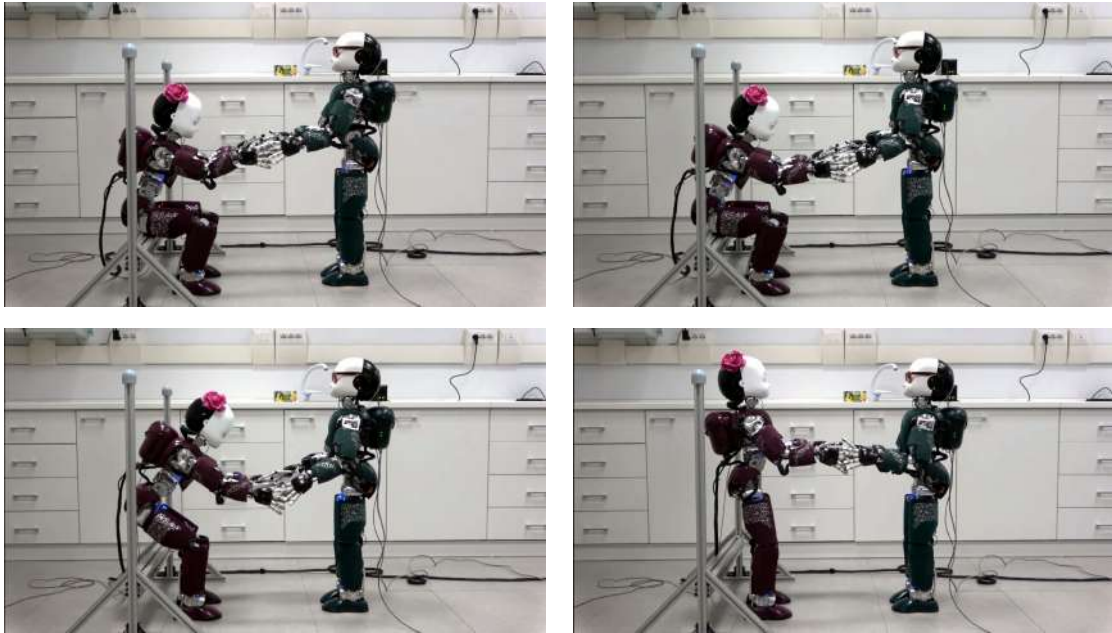
Fig. 8.8 Standup experimental scenario with two iCub robots involved in physical interaction
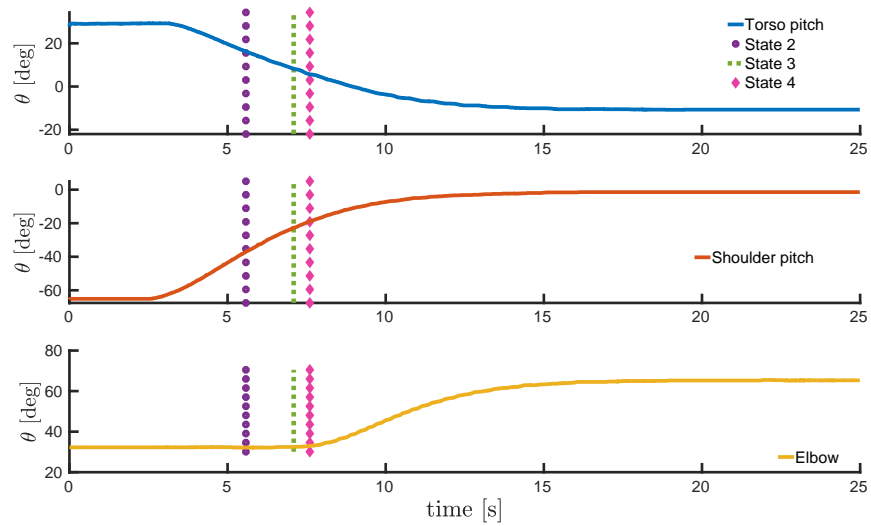
Fig. 8.9 Interacting agent joint trajectories
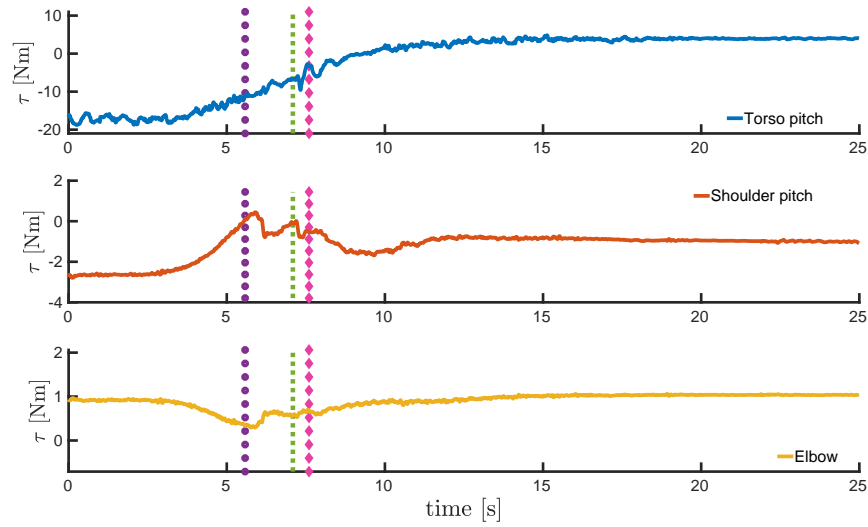
Fig. 8.10 Interacting agent joint torques

The hands of the iCub robot are quite fragile and are not designed to make *sustained* power grasps. This posed quite a challenge during our experiments. So, we designed a new mechanical part highlighted in Figure. 8.11 that is attached at the wrists of the iCub hands. Additionally, we designed new mechanical contraption shown in Figure. 8.12 that can be attached rigidly to the wrists of the robots ensuring rigid contacts between the hands during the entire duration of the experiment.
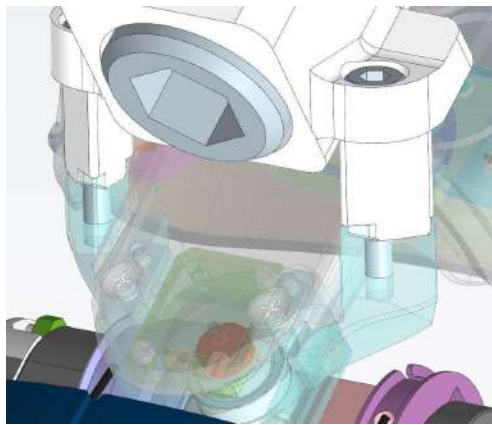
(a) CAD Design

(b) CNC Machined Part

Fig. 8.11 New mechanical component (cyan) added at the wrist of iCub hand
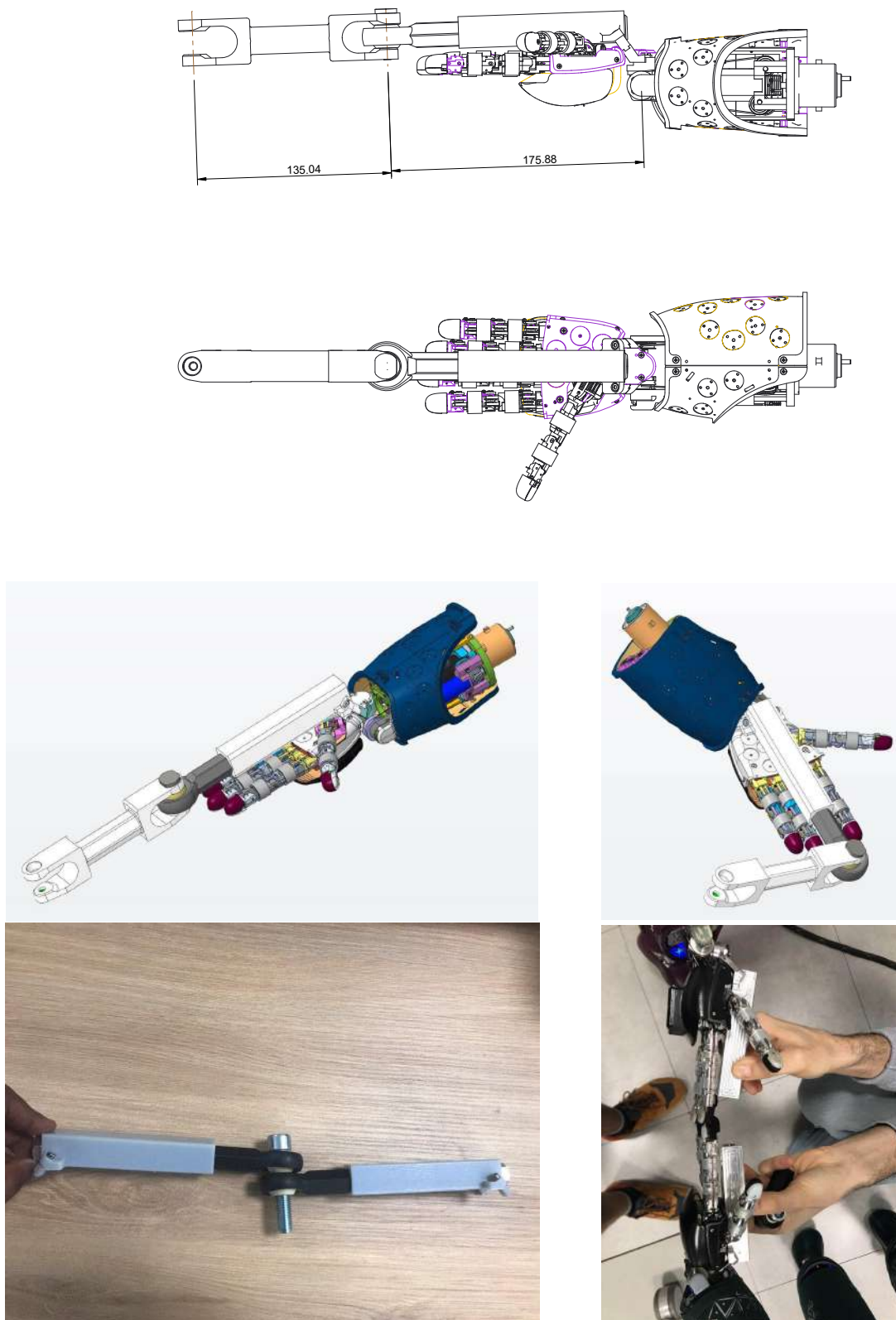
Fig. 8.12 Mechanical contraption to ensure rigid contacts between iCub hands

8.6 Results

A predetermined trajectory generated using a minimum-jerk trajectory generator (Pattacini et al., 2010c) is given as a reference to the center of mass of the *purple* iCub robot to perform the sit-to-stand transition. At the start of the Simulink controller the *purple* iCub robot is seated on the metallic structure that serves as a chair. Once the controller is started, it receives joint quantities as inputs from both the robots and actively generates joint torque inputs for the *purple* iCub robot to maintain its momentum and track the center of mass.

The time evolution of the center of mass tracking is shown in Fig. 8.13. The vertical lines indicate the time instance at which a new state begins. Between states 2 and 3, the *purple* iCub robot is seated on the chair with its upper leg as contact constraints. This seriously limits the robot motion along the x-direction and hence the tracking error of the center of mass along the x-direction is not negligible. Similarly, between states 3 and 4, the robot has to standup relative quickly as the trajectory smoothing is kept very low and the contact constraints change from the upper legs to the feet. This contact switching, along with unmodeled phenomena such as joint friction limits the robot motion along the z-direction and hence the tracking error of the center of mass along the z-direction is not negligible. Apart from these two instances, the overall center of mass tracking is good.
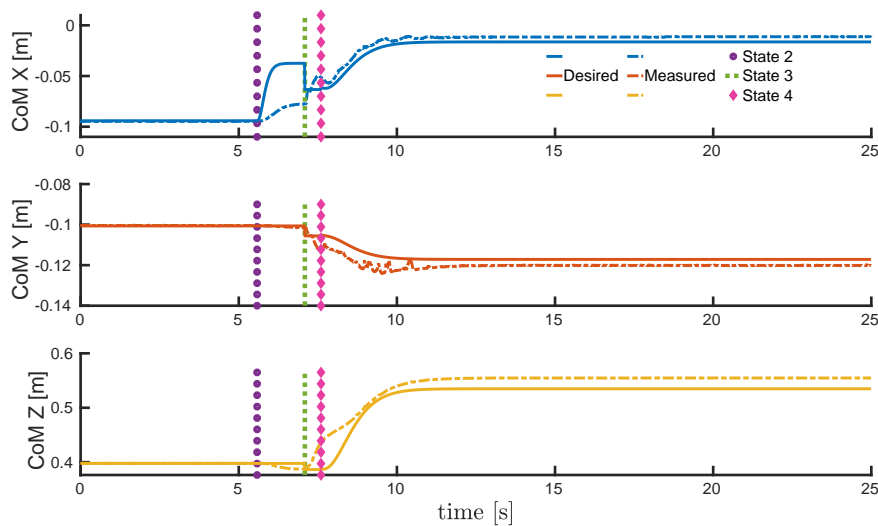

Fig. 8.13 Time evolution of the desired and measured CoM trajectory while performing standup motion on application of the control law (8.33)

The primary control objective of momentum control is also realized well as highlighted by the time evolution of the linear and angular momentum of the robot as shown in Fig. 8.14. Between states 3 and 4, both the linear and angular momentum error increased momentarily. Understandably this results from the impact at the contact switching that occurs at the beginning of state 3. However, the overall robot momentum is maintained close to zero and eventually, the momentum error converges to zero when the robot becomes stable after standing fully erect.
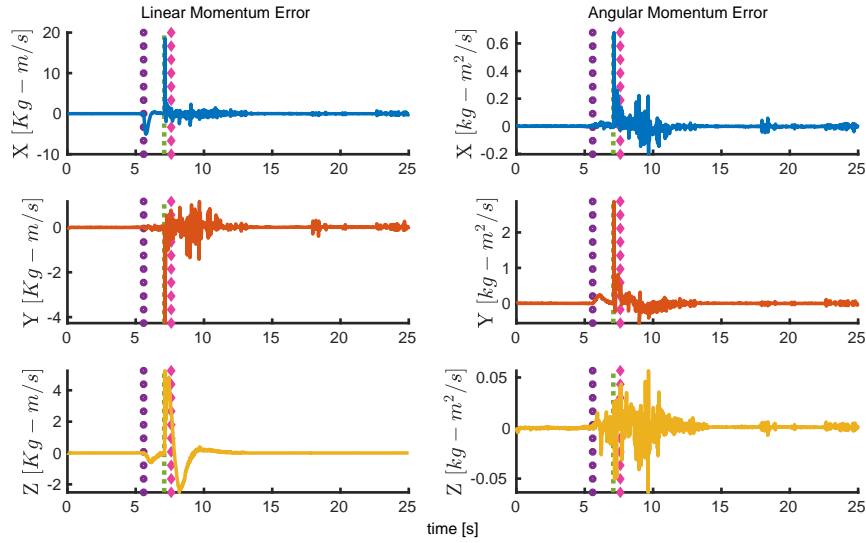
Fig. 8.14 Time evolution of the linear and angular momentum while performing standup motion on application of the control law (8.33)

The time evolution of α i.e. the component of the interaction agent joint torques projected in the direction parallel to the task is shown in Fig. 8.15. The instantaneous values of α change throughout the experiment according to the joint torque values of the *green* iCub robot. Between the states 2 and 3, the negative values of α contribute towards making the Lyapunov function decrease faster as indicated in Fig. 8.16. This highlights the fact that the physical interaction with an external agent is exploited (in terms of the joint torques) by the *purple* iCub robot to perform the standup task.

The time evolution of the Lyapunov function V from equation (8.34) is shown in Fig. 8.16. After the controller is started, during state 1 the system has small energy while the robot is seated on the chair. This is highlighted in the inset plot shown for the duration between 1-2 seconds. Starting from state 2 as the robot starts moving, the total energy of the systems starts to increase as shown between states 2 and 3. As the robot enters state 3 the energy quickly drops during the contact switching from
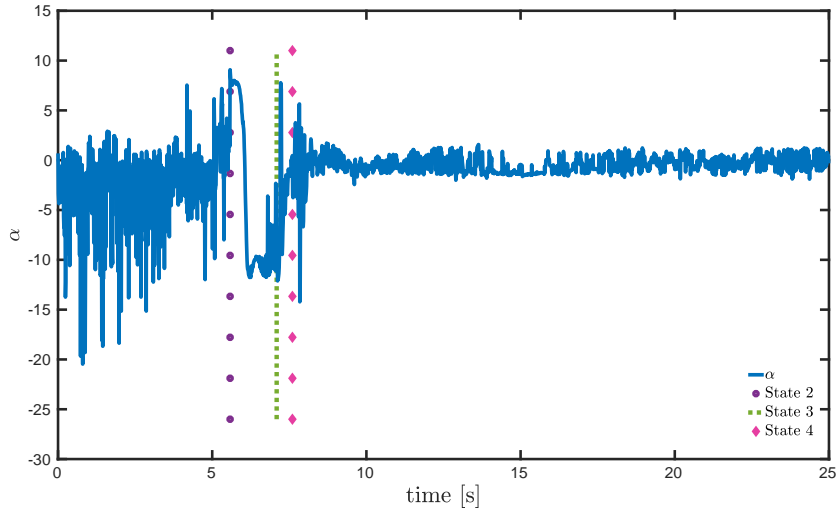
Fig. 8.15 Time evolution of α under the influence of physical interaction

upper legs to feet. This is a direct reflection of exploiting the physical interaction with the *green* iCub robot. Between states 3 and 4 while the robot is moving to a fully erect stance the energy rises slightly and eventually settles to a stable value. The inset plot during the duration between 16-17 seconds highlights the system energy when the robot is in a stable fully erect position.
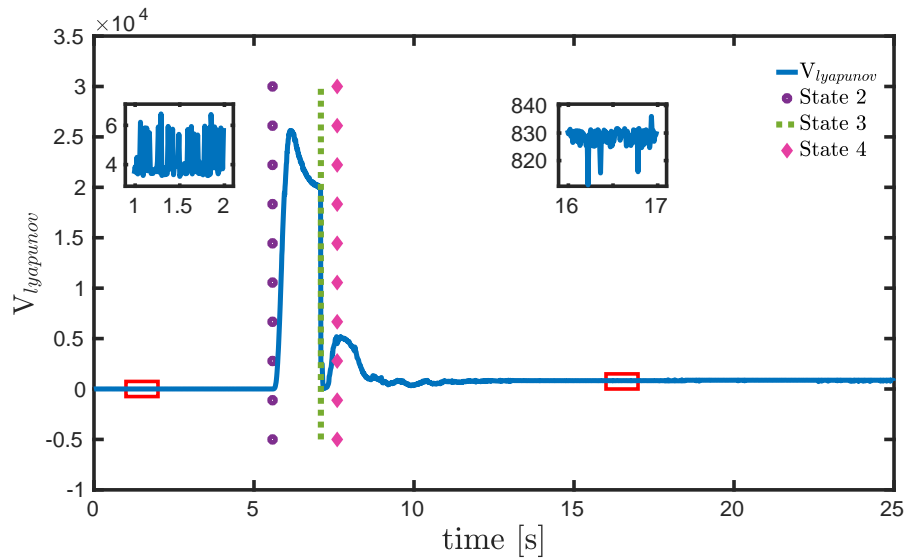
Fig. 8.16 Time evolution of lyapunov function considered (8.34) on application of the control law (8.33)

9

Trajectory Advancement

The previous chapter on partner-aware control proved to provide a stable robot behavior while following a given reference Cartesian trajectory for the robot's center of mass. Although the proposed partner-aware control law exploits help in terms of joint torques, the reference trajectory for the center of mass is not modified or updated under the influence of the help from an external agent. This chapter presents the concept of trajectory advancement through which the robot can advance along the reference trajectory leveraging assistance from physical interactions with a human partner.

Considering the particular experiment of robot standing up (Fig. 9.1a) using the assistance, an intuitive behavior one would expect for the robot is to stand up quicker by leveraging the assistance. Consider another example case of a manipulator robot moving along a given Cartesian reference trajectory performing a pick and place task (Fig. 9.1b). An intuitive interaction of a human with the intention to speed up the robot motion is to apply forces in the robot's desired direction. Under such circumstances, traditionally, the robot can either render a compliant behavior through impedance/admittance control or switch to gravity compensation mode that allows the human to move the robot freely (compromising task accuracy). Instead, a more intuitive behavior is to advance further along the reference trajectory and complete the task quicker.
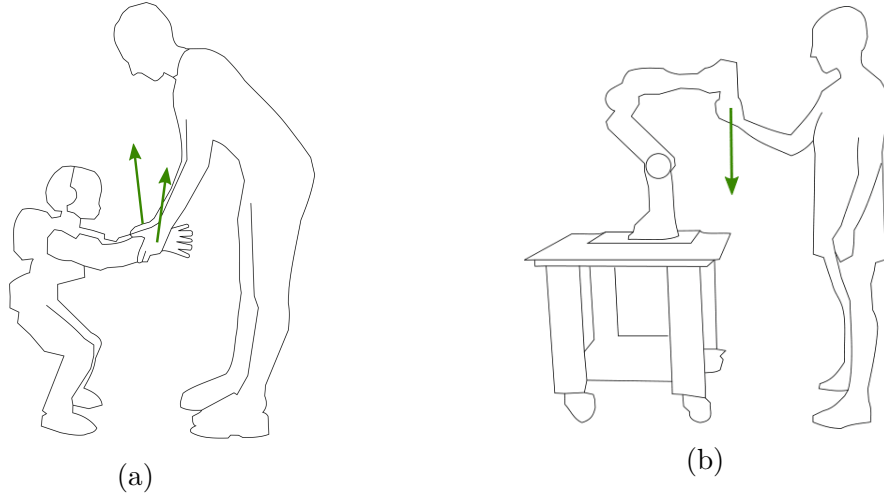
Fig. 9.1 Example scenarios of HRC with human and robot engaged in physical interactions

9.1 Parametrized Reference Trajectory

Traditionally, motion control problems involving tracking of a reference trajectory has both spatial dimension, encapsulated in a geometric path, and temporal dimension, encapsulated in the dynamic evolution of the geometric path (Aguiar et al., 2004). Accordingly, the reference trajectory is a time (t) parametrized curve and the control design drives the system to a specific point in space at a specific predefined time. In contrast, the path following the problem involves converging to and following a geometric path without any temporal constraints (Breivik and Fossen, 2005). In this work, we bank on the concepts of path following and design a parametric curve parametrized with a *free parameter* $\psi \in [0, \infty)$. The choice of ψ becomes clear in the subsequent sections. The resulting parametric curve $\boldsymbol{x}_d(\psi)$ is the desired geometric path to be followed spatially by a link of the robot. Assuming that the free parameter is time-dependent i.e., $\psi = \psi(t)$, the first and second time derivatives of the path are given as,

$$\dot{\boldsymbol{x}}_d(\psi, \dot{\psi}) = \partial_\psi \boldsymbol{x}_d(\psi) \; \dot{\psi} \tag{9.1a}$$

$$\ddot{\boldsymbol{x}}_d(\psi, \dot{\psi}, \ddot{\psi}) = \partial_\psi^2 \boldsymbol{x}_d(\psi) \; \dot{\psi}^2 + \partial_\psi \boldsymbol{x}_d(\psi) \; \ddot{\psi} \tag{9.1b}$$

9.2 Interaction Exploitation

Consider the control objective of trajectory tracking where at each time instant the reference position ($\boldsymbol{x}_d(\psi)$), velocity ($\dot{\boldsymbol{x}}_d(\psi, \dot{\psi})$), and acceleration ($\ddot{\boldsymbol{x}}_d(\psi, \dot{\psi}, \ddot{\psi})$) are taken from the reference trajectory parametrized in ψ. The term $\boldsymbol{\Omega f}$ in the robot control torques equation Eq. (8.6) represents the Cartesian resultant acceleration that results under the influence of external interaction wrench $\boldsymbol{f}$. Instead of completely canceling out the effects of external interaction wrench, it is desirable to exploit any *helpful* components to advance along the desired reference trajectory making an active collaboration possible during HRC. More specifically, let us define the *helpful* interaction by decomposing the resultant acceleration into *parallel* and *perpendicular* components along the desired velocity as,

$$\boldsymbol{\Omega f} = \alpha \; \dot{\boldsymbol{x}}_d^{\parallel} + \beta \; \dot{\boldsymbol{x}}_d^{\perp} \tag{9.2a}$$

$$\dot{\boldsymbol{x}}_d^{\parallel} = \frac{\dot{\boldsymbol{x}}_d}{\|\dot{\boldsymbol{x}}_d\|}, \quad \alpha = \frac{\dot{\boldsymbol{x}}_d^{T} \boldsymbol{\Omega f}}{\|\dot{\boldsymbol{x}}_d\|}$$

where $\dot{\boldsymbol{x}}_d^{\parallel} \in \mathbb{R}^6$ is the unit vector along the direction of the desired velocity, $\alpha \in \mathbb{R}$ is the resultant acceleration component projected along the direction parallel to the direction of the desired velocity. An intuitive choice for the component α is in the direction of the desired velocity i.e. $\alpha > 0$. Accordingly, we define a *correction wrench*[1] term given by $\alpha \; \dot{\boldsymbol{x}}_d^{\parallel} \in \mathbb{R}^6 \; \forall \; \alpha > 0$, which represents the *helpful* interaction mathematically.

9.3 Trajectory Advancement

Proposition. *The time evolution of the free parameter ψ for trajectory advancement leveraging assistance is given by the following update rule,*

$$\dot{\psi} = min \left\{ \dot{\psi}_{upper}, max \left\{ 1, \frac{\dot{\boldsymbol{x}}(t)^{\top} \; \partial_{\psi} \boldsymbol{x}_d(\psi)}{\left\| \partial_{\psi} \boldsymbol{x}_d(\psi) \right\|^2} \right\} \right\} \tag{9.3}$$

---

[1] The name *correction wrench* is an abuse of notation but has an intuitive meaning in conveying the notion of helpful interaction wrench. Also, the units of wrench [N, N m] are used.

The update rule in Eq. (9.3) reflects the time evolution of the free parameter ψ which helps in advancing along the desired reference trajectory exploiting the external interaction wrenches with the robot. The lower bound value 1 signifies that the new parametrization is exactly equal to the time parametrized trajectory i.e. $\psi = t$ until any external wrench $\boldsymbol{f}$ is applied such that it will help the robot's task.

Proof. Considering the task of Cartesian reference trajectory tracking, the desired dynamics for the control objective can be written as directed by Eq. (8.2). Given the correction wrench term Eq. (9.2a), the desired dynamics for the trajectory tracking task is updated as,

$$\ddot{\boldsymbol{x}} = \ddot{\boldsymbol{x}}^* := \ddot{\boldsymbol{x}}_d - \boldsymbol{K}_D \, \dot{\tilde{\boldsymbol{x}}} - \boldsymbol{K}_P \int_0^t \dot{\tilde{\boldsymbol{x}}} \, du + \alpha \, \dot{\boldsymbol{x}}_d^{\parallel} \ \ \forall \, \alpha > 0 \tag{9.4}$$

Using the above choice of the desired dynamics, the robot control torques defined in Eq. (8.6) will only compensate for external wrench that is not helpful. Now, consider the following Lyapunov function candidate,

$$\mathrm{V} = \frac{1}{2}\left\|\dot{\boldsymbol{x}}(t) - \dot{\boldsymbol{x}}_d(\psi,\dot{\psi})\right\|^2 + \frac{\boldsymbol{K}_P}{2}\left\|\int_0^t (\dot{\boldsymbol{x}}(t) - \dot{\boldsymbol{x}}_d(\psi,\dot{\psi}))du\right\|^2 \tag{9.5}$$

On differentiating V, we get:

$$\dot{\mathrm{V}} = \ \dot{\tilde{\boldsymbol{x}}}^{\top}\ddot{\tilde{\boldsymbol{x}}} + \int_0^t \dot{\tilde{\boldsymbol{x}}}^{\top} du \ \boldsymbol{K}_p\dot{\tilde{\boldsymbol{x}}}$$

$$\dot{\mathrm{V}} = \dot{\tilde{\boldsymbol{x}}}^{\top}\ddot{\tilde{\boldsymbol{x}}} + \dot{\tilde{\boldsymbol{x}}}^{\top} \ \boldsymbol{K}_p \int_0^t \dot{\tilde{\boldsymbol{x}}} \, du$$

$$\dot{\mathrm{V}} = \dot{\tilde{\boldsymbol{x}}}^{\top}[\ddot{\tilde{\boldsymbol{x}}} + \boldsymbol{K}_p \int_0^t \dot{\tilde{\boldsymbol{x}}} \, du]$$

Given the updated desired dynamics in Eq. (9.4) we rearrange it as $\ddot{\tilde{\boldsymbol{x}}} + \boldsymbol{K}_p \int_0^t \dot{\tilde{\boldsymbol{x}}} \, du = -\boldsymbol{K}_D \, \dot{\tilde{\boldsymbol{x}}} + \alpha \, \dot{\boldsymbol{x}}_d^{\parallel}$ and use it in the derivative of the Lyapunov function to obtain the following relation,

$$\dot{\mathrm{V}} = \dot{\tilde{\boldsymbol{x}}}^{\top}[-\boldsymbol{K}_D \, \dot{\tilde{\boldsymbol{x}}} + \alpha \, \dot{\boldsymbol{x}}_d^{\parallel}]$$

$$\dot{\mathrm{V}} = -\dot{\tilde{\boldsymbol{x}}}^{\top} \boldsymbol{K}_D \, \dot{\tilde{\boldsymbol{x}}} + \dot{\tilde{\boldsymbol{x}}}^{T} \alpha \, \dot{\boldsymbol{x}}_d^{\parallel}$$

According to Lyapunov theory, the stability of the system is ensured when $\dot{\mathrm{V}} \leq 0$. Given that $\boldsymbol{K}_D$ is a positive symmetric matrix, the term $-\dot{\tilde{\boldsymbol{x}}}^\top \boldsymbol{K}_D \, \dot{\tilde{\boldsymbol{x}}} \leq 0$. So, to ensure the stability of the system i.e. $\dot{\mathrm{V}} \leq 0$, the following condition has to be satisfied,

$$\dot{\tilde{\boldsymbol{x}}}^\top \alpha \, \dot{\boldsymbol{x}}_d^{\parallel} \leq 0$$

Considering that $\alpha > 0$ and $\|\dot{\boldsymbol{x}}_d\| > 0$, the above inequality is equivalent to $\dot{\tilde{\boldsymbol{x}}}^\top \dot{\boldsymbol{x}}_d(\psi, \dot{\psi}) \leq 0$,

$$(\dot{\boldsymbol{x}}(t) - \dot{\boldsymbol{x}}_d(\psi, \dot{\psi}))^\top \dot{\boldsymbol{x}}_d(\psi, \dot{\psi}) \leq 0$$

$$\dot{\boldsymbol{x}}(t)^\top \dot{\boldsymbol{x}}_d(\psi, \dot{\psi}) - \left\| \dot{\boldsymbol{x}}_d(\psi, \dot{\psi}) \right\|^2 \leq 0$$

$$\dot{\boldsymbol{x}}(t)^\top \partial_\psi \boldsymbol{x}_d(\psi) \dot{\psi} - \left\| \partial_\psi \boldsymbol{x}_d(\psi) \right\|^2 \dot{\psi}^2 \leq 0$$

$$(\dot{\boldsymbol{x}}(t)^\top \partial_\psi \boldsymbol{x}_d(\psi) - \left\| \partial_\psi \boldsymbol{x}_d(\psi) \right\|^2 \dot{\psi}) \dot{\psi} \leq 0$$

Assuming the lower bound $\dot{\psi} \geq 1$, we obtain

$$\dot{\psi} \geq \frac{\dot{\boldsymbol{x}}(t)^\top \partial_\psi \boldsymbol{x}_d(\psi)}{\left\| \partial_\psi \boldsymbol{x}_d(\psi) \right\|^2} \tag{9.9}$$

The condition in Eq. (9.9) reflects the time evolution of the free parameter ψ which helps in advancing along the desired reference trajectory exploiting the external interaction wrenches with the robot. The lower bound value 1 signifies that the new parametrization is exactly equal to the time parametrized trajectory i.e. $\psi = t$ until any external wrench $\boldsymbol{f}$ is applied such that it will help the robot's task. Under the influence of *helpful* external wrench, the value of $\dot{\psi}$ becomes greater than 1. On integrating/differentiating $\dot{\psi}$ we determine the advancement along the desired reference trajectory,

$$\psi^* = \int_{t_1}^{t_2} \dot{\psi} \; du, \quad \ddot{\psi}^* = \frac{d\dot{\psi}}{dt}$$

Now, the updated references for trajectory tracking becomes,

$$\boldsymbol{x}_d(\psi^*), \dot{\boldsymbol{x}}_d(\psi^*, \dot{\psi}), \ddot{\boldsymbol{x}}_d(\psi^*, \dot{\psi}, \ddot{\psi}^*)$$

Besides, an upper limit $\dot{\psi}_{upper}$ is set to bound the length of advancement along the reference trajectory ensuring safe physical interactions.

The updated equation of $\dot{\psi}$ becomes,

$$\dot{\psi} = min\left\{\dot{\psi}_{upper}, max\left\{1, \frac{\dot{\boldsymbol{x}}(t)^{\top} \, \partial_{\psi}\boldsymbol{x}_d(\psi)}{\left\|\partial_{\psi}\boldsymbol{x}_d(\psi)\right\|^2}\right\}\right\} \qquad (9.10)$$

Remark: Strictly speaking, the choice of $\dot{\psi}$ as stated in Eq. (9.10) induces an algebraic loop when applied with the control law Eq. (9.4). In fact, the updated reference acceleration $\ddot{\boldsymbol{x}}_d(\psi^*, \dot{\psi}, \ddot{\psi}^*)$ does depend on the Cartesian acceleration $\ddot{\boldsymbol{x}}$, and, consequently, on the joint torques $\boldsymbol{\tau}$. For this reason, no formal stability statement was claimed in Proposition 9.3. From the theoretical point of view, the algebraic loop can be avoided by designing an update rule for $\ddot{\psi}$ rather than $\dot{\psi}$, and by modifying the control law (9.4) so that the reference Cartesian acceleration is not compensated anymore. This choice, however, would imply the calculation of ψ^* through double numerical integration of $\ddot{\psi}$, which may lead to a fast divergence of the reference trajectory due to numerical drifts. For this reason, the proposed control solution (9.10)-(9.4), despite not being fully theoretically sound, resulted to be more robust when applied in practice. Furthermore, the algebraic loop can be resolved at the implementation level by computing the numerical derivative $\ddot{\psi}^* = \frac{d\dot{\psi}}{dt}$ with one time step of delay, and/or by low-pass filtering the signal to also attenuate the effect of numerical noise. Driven by these motivations we used Eq. (9.10)-(9.4) for controlling the robot and verified the closed-loop system stability numerically.

<div style="text-align: right">□</div>

9.4 End-Effector Trajectory Advancement Experiments

The robotic platform considered for the experiment is the iCub humanoid robot described in Section (4.4). The control objective is to move the *right foot* of the robot along the desired reference trajectory. The leg of the robot has 3 joints at the hip, 1 joint at the knee, and 2 joints at the ankle. For the sake of intuition, only one dimensional (1D) trajectory in the x-direction is considered. The reference trajectory is a sinusoidal function of amplitude 0.05m with frequency 0.1Hz and is designed to have a minimum jerk profile (Kyriakopoulos and Saridis, 1988). Concerning the task of trajectory tracking with the leg, the robot base is fixed on a pole as shown in Fig. 9.2a. The link frame associated with the right foot of the robot and the inertial frame of reference (shown under the base/pelvis of the robot) is highlighted in Fig. 9.2b.

<div align="center">(a)   (b)</div>
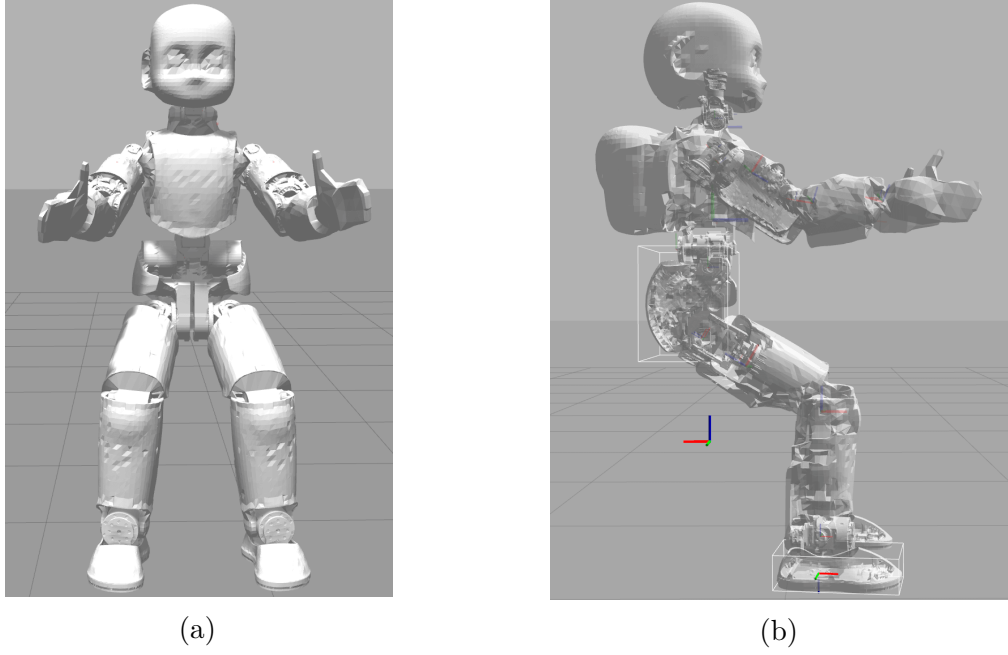
Fig. 9.2 iCub humanoid robot with base fixed on a pole in gazebo simulation

Experiments are carried out in both Gazebo simulation and on the real robot. The controller is implemented in Matlab Simulink, using whole-body toolbox[2] (Romano et al., 2017), as a stack-of-tasks controller with trajectory tracking as the primary objective. The controller gains are tuned to achieve good trajectory tracking both in simulation and on the real robot as highlighted in Fig. 9.3. The trajectory tracking error in the case of simulation is very small and can be attributed to numerical instability of the dynamics integration in Gazebo simulation and numerical noise in measurements. On the other hand, the trajectory tracking error on the real robot is certainly higher than in simulations owing to several unmodeled effects such as joint friction which are prominent on the real robot. Additionally, friction induces phase delays in following the desired trajectory resulting in higher tracking error. The upper limit $\dot{\psi}_{upper}$ is set to 10 for experiments both in simulation and on the real robot.

9.4.1 Wrench Classification

The iCub robot has a force-torque sensor embedded at the end-effector considered for the experiments i.e. the right foot. Instead of reading the sensor measurements directly in the sensor frame, the wrench measurements are expressed with a frame that

---

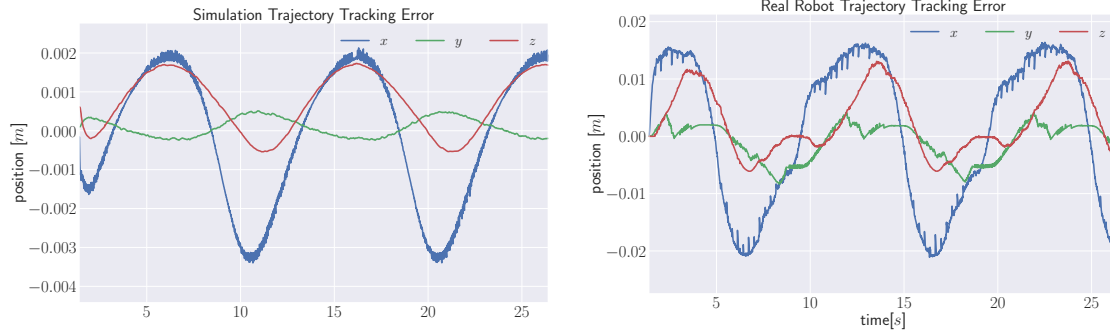[2] https://github.com/robotology/wb-toolbox

Fig. 9.3 Trajectory tracking error under no external wrenches

has the origin of the end-effector frame and the orientation of the inertial frame of reference (Nori et al., 2015b). An external wrench applied to the link of the robot is classified, in this work, in two ways:

- *Assistive wrench* if the external wrench has a vector component along the desired direction of motion

- *Agnostic wrench* if the external wrench does *not* have vector components along the desired direction of motion

Examples of external wrench classification are highlighted in Fig. 9.4. Considering that the desired direction of motion for the foot is in *positive x*-direction with respect to the inertial frame of reference, the external wrenches shown in Fig. 9.4a 9.4b 9.4c are assistive wrenches as they have a vector component along the positive x-direction. Similarly, the external wrenches shown in Fig. 9.4d 9.4e 9.4f are agnostic wrenches as they do *not* have any vector component along the positive x-direction.
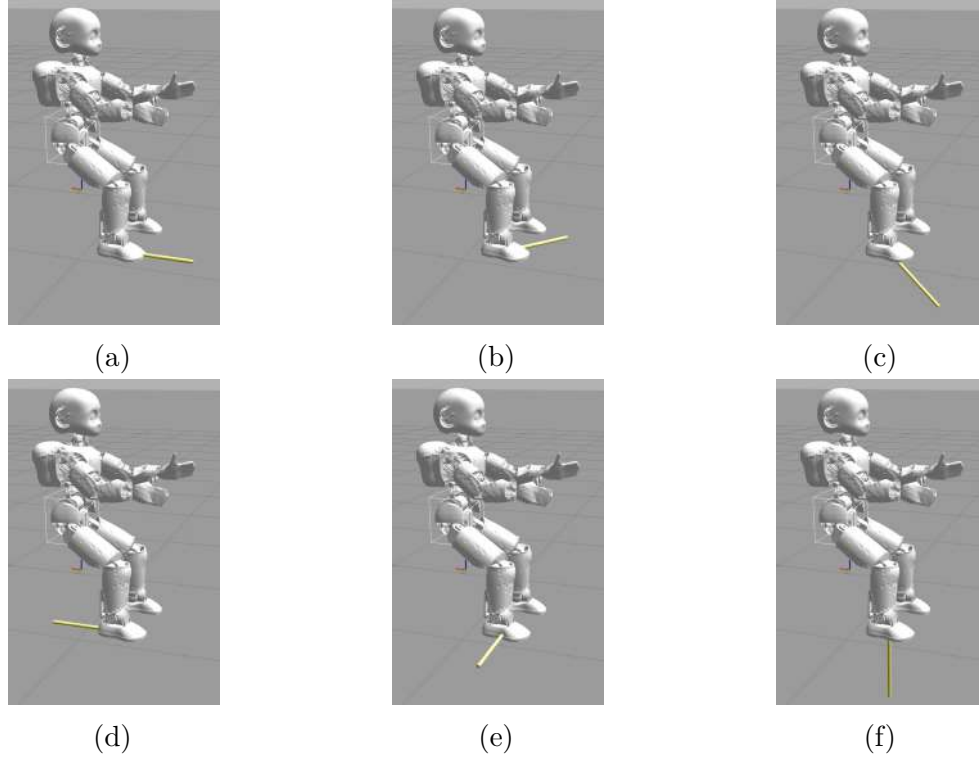
Fig. 9.4 External interaction wrench classification examples when the desired direction of motion in positive x-direction

Concerning the experiments conducted in Gazebo simulation environment, wrench is applied through a plugin[3] (Hoffman et al., 2014). Due to the limited operational space of the robot we chose a fixed duration of 0.75s for the wrench. Furthermore, the wrench applied has a smooth profile rather than an impulse profile. This is an experimental design choice made to mimic the intentional interaction wrench applied by a human on the real robot during HRC scenarios.

9.4.2 Simulation Results

The first set of experiments are performed in Gazebo simulation environment. A set of test wrenches listed in the table 9.1 are applied when the desired direction of motion is along the positive x-direction. These test wrench vectors are similar in *direction* to the wrench vectors highlighted in the wrench classification examples Fig. 9.4. The first three wrench vectors (a)(b)(c) are classified as assistive wrenches as they have a vector component (highlighted in *blue*) along the desired direction of motion i.e. positive

---
[3] https://github.com/robotology/gazebo-yarp-plugins

x-direction. The next three wrench vectors (d)(e)(f) are classified as agnostic wrenches as they do not have any vector component along the desired direction of motion.

	f_x	f_y	f_z	τ_x	τ_y	τ_z
(a)	10	0	0	0	0	0
(b)	5	10	0	0	0	0
(c)	5	0	10	0	0	0
(d)	−10	0	0	0	0	0
(e)	0	−10	0	0	0	0
(f)	0	0	10	0	0	0

Table 9.1 Test wrenches applied in gazebo simulation for end-effector trajectory advancement experiments

The results of experiments in Gazebo simulation under the application of the test wrenches listed in table 9.1 are highlighted in Fig. 9.5. The external interaction wrench experienced by the right foot link of the robot are shown in Fig. 9.5a and the *correction wrench* that is considered towards trajectory advancement is shown in Fig. 9.5b. In the case of agnostic wrenches, the correction wrench terms are insignificant and they are present due to the noise in the wrench estimation (Nori et al., 2015b). The reference trajectory is similar to a time parametrized trajectory i.e., $\psi = t$ until any helpful wrench is applied to the end-effector. Under the influence of assistive wrenches, the derivative of the trajectory free parameter $\dot{\psi}$ changes as highlighted in Fig. 9.5c and the corresponding trajectory advancement is reflected as an increase in ψ as seen in Fig. 9.5d. Accordingly, the reference is advanced further along the reference trajectory as shown in Fig. 9.5e.

The trajectory tracking error is slightly more when the reference trajectory is updated under the influence of the assistive wrenches however the error magnitude is of low order as highlighted in Fig. 9.5f proving that the task of trajectory tracking is achieved reliably by the controller. Another important observation is that the magnitude of change in $\dot{\psi}$ is related to the magnitude of the interaction wrench. The length of advancement under the influence of assistive wrench vector (a) is more than under the influence of assistive wrench vector (b) or (c) from table 9.1. The time evolution of the desired leg joint torques generated by the controller for the duration of the experiment is highlighted in Fig. 9.6.

9.4.3 Real Robot Results

The results of experiments on the real iCub robot with 1D reference trajectory along the x-axis are shown in Fig. 9.7. The external interaction wrenches experienced by the right foot of the robot are highlighted in Fig. 9.7a and the *correction wrench* that is considered towards trajectory advancement is shown in Fig. 9.7b. The reference trajectory is similar to a time parametrized trajectory i.e., $\psi = t$ until any helpful wrench is applied to the end-effector. Under the influence of assistive wrenches, the derivative of the trajectory free parameter $\dot{\psi}$ changes as shown in Fig. 9.7c and the corresponding trajectory advancement is reflected as an increase in ψ as seen in Fig. 9.7d. Accordingly, the reference is advanced further along the reference trajectory as shown in Fig. 9.7e. Furthermore, starting from $t = 30s$ wrench is applied in the positive x-direction continuously. While the reference trajectory is in the positive x-direction, this wrench is considered assistive but as the reference trajectory is changed to the negative x-direction the wrench becomes agnostic and the reference trajectory is unchanged. Although there are some noisy wrenches that are considered to be correction wrench, they are tuned out by a regularization parameter in computing $\dot{\psi}$ to not have any direct effect on trajectory advancement.

The trajectory tracking error on the real robot is highlighted in Fig. 9.7f. Although the tracking error is higher due to phase delays induced by joint friction, the desired amplitude of the reference trajectory is reached. The time evolution of the leg desired joint torques generated by the controller for the duration of the experiment is highlighted in Fig. 9.8.
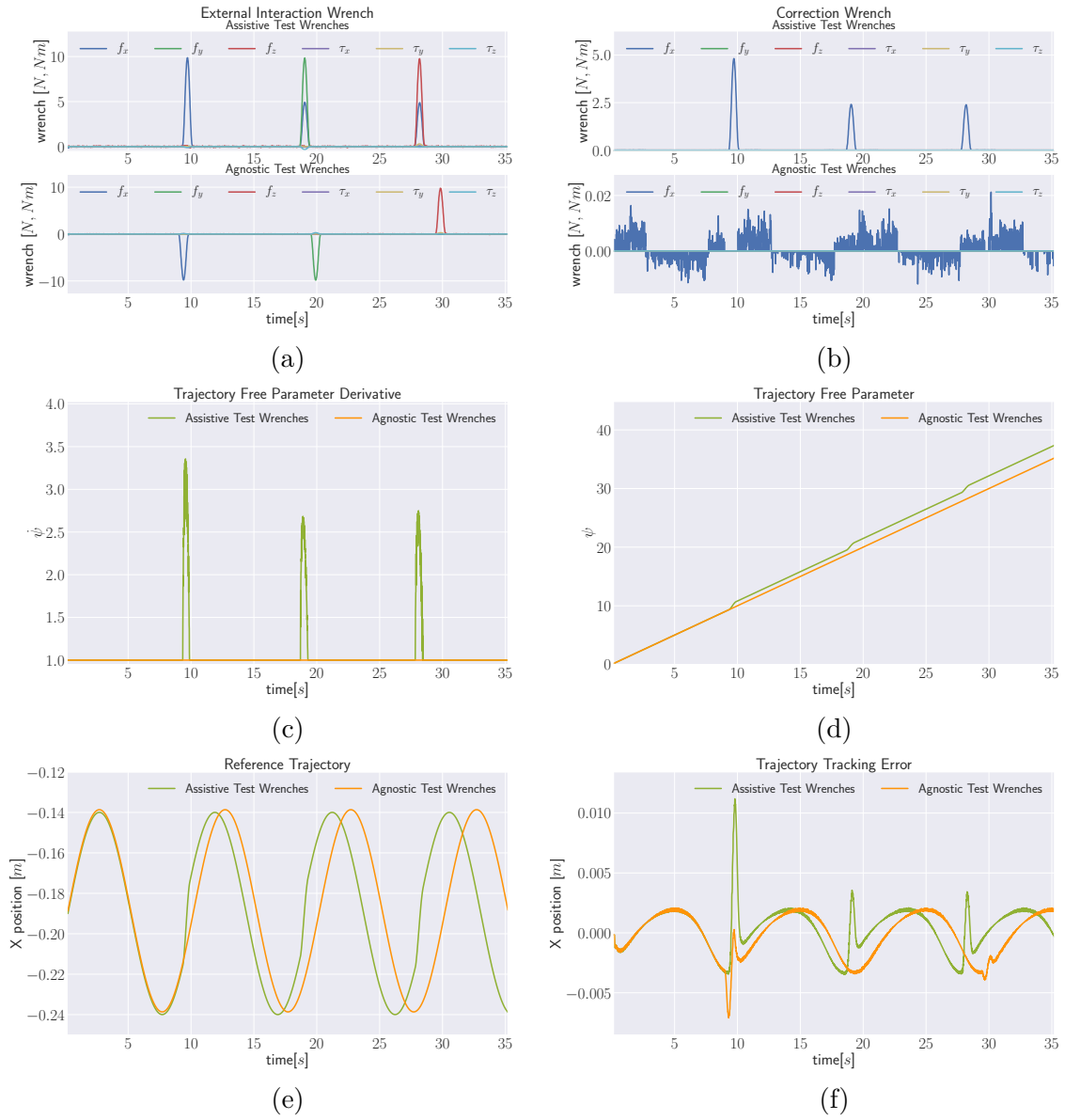
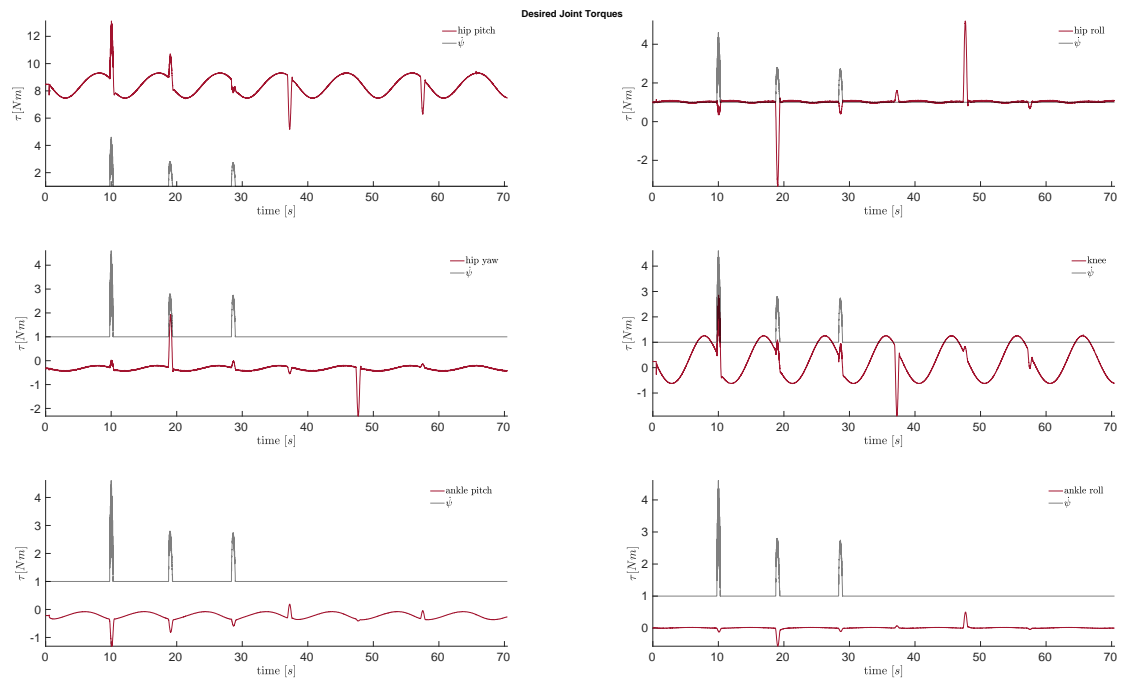Fig. 9.5 Gazebo simulation 1D trajectory advancement along X-direction

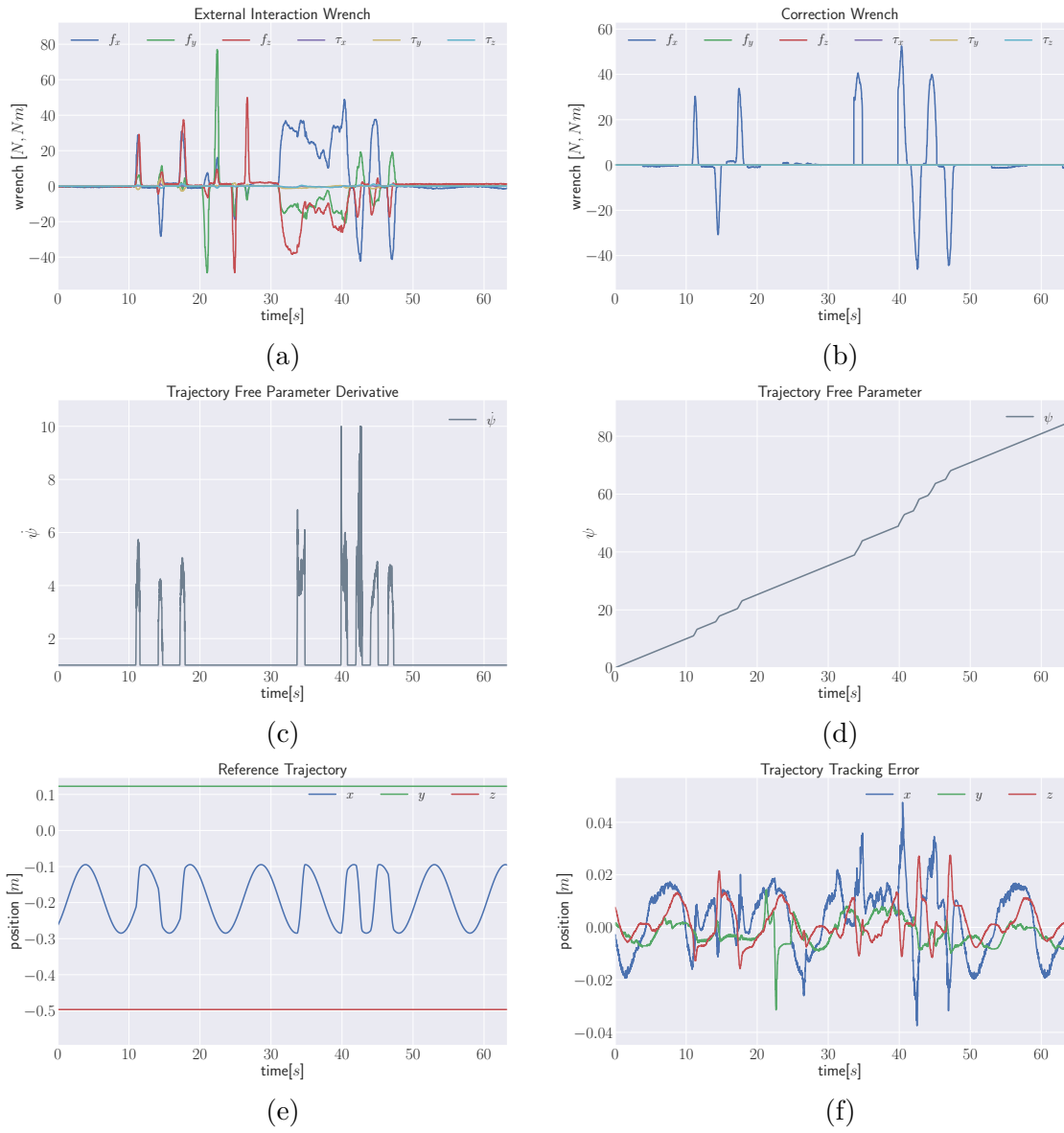Fig. 9.6 Gazebo simulation 1D trajectory advancement leg joint torques

Fig. 9.7 Real robot 1D trajectory advancement along X-direction
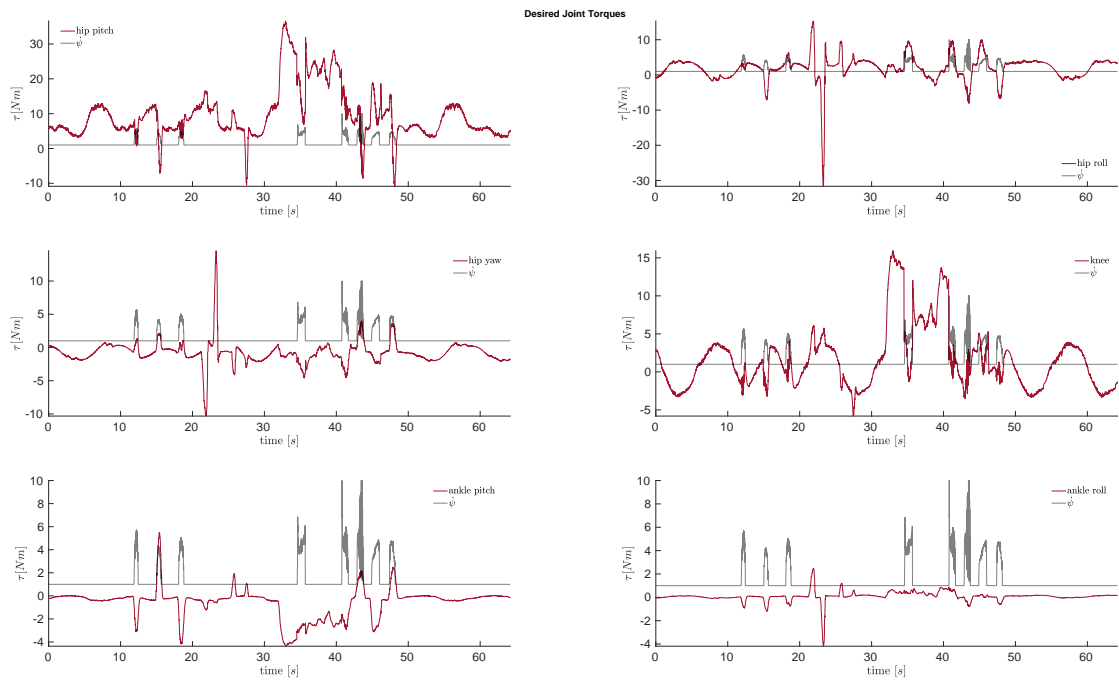
Fig. 9.8 Real robot 1D trajectory advancement leg joint torques

9.5 Whole-Body Standup Trajectory Advancement Experiments

Extended experiments of trajectory advancement are carried with the whole-body standup controller that is described in section 8.4. Unlike the use of external agent joint torques, the controller is updated to use the interaction wrench estimates from the hands of the robot to advance along the trajectory. The reference trajectory to be followed by the robot is given for the center of mass. Intuitively using the approach of trajectory advancement, when a human is helping the robot to standup, the interaction wrench is used to stand faster.

9.5.1 Simulation Results

Simulation experiments are carried using Gazebo, a popular 3D dynamics simulator. Different states of the standup controller are highlighted in Fig. 9.9. External interaction is mimicked through an application of wrench at the hands of the robot a plugin (Hoffman et al., 2014). The cylinders at the hands in Fig. 9.9 indicate the interaction wrench applied at the hands and the plots in Fig. 9.10a and Fig. 9.10b indicate the left and right hands wrench estimates respectively (Nori et al., 2015b).

| (a) | (b) | (c) | (d) |
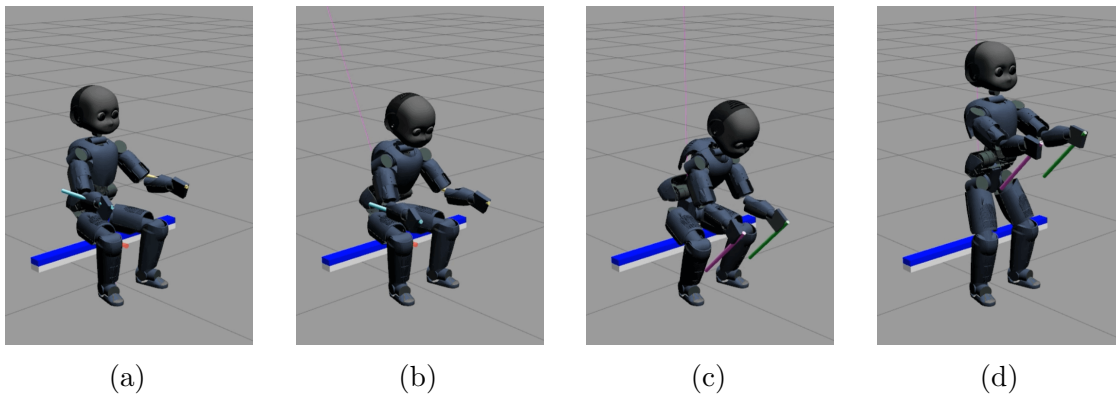
Fig. 9.9 iCub at different states during sit-to-stand transition with the whole-body standup controller (Sec 8.4)

The results of trajectory advancement in simulation are shown in Fig. 9.10. The wrench estimated at the hands of the robot under the influence of simulated assistance is highlighted in Fig. 9.10a 9.10b. The external wrench that is helpful to achieve the task is shown in Fig. 9.10c. The reference trajectory is similar to a time parametrized trajectory i.e., $\psi = t$ until any helpful wrench is applied at the hands of the robot. Under the influence of helpful wrenches, the derivative of the trajectory free parameter $\dot{\psi}$ changes as shown in Fig. 9.10d and the corresponding trajectory advancement is

reflected as an increase in ψ as seen in Fig. 9.10e. Accordingly, the reference is advanced further along the CoM reference trajectory as shown in Fig. 9.10f. The original CoM reference trajectory without trajectory advancement is shown in the same figure with reduced transparency. The time evolution of the leg desired joint torques generated by the controller for the duration of the experiment is highlighted in Fig. 9.11.

As highlighted in section 8.4, the robot behavior is guided through a state machine and the center of mass trajectory is generated between the states through a smoothing parameter. The experiments for trajectory advancement are conducted using the same state machine of the controller and as the time duration between the sates is small, the amount of trajectory advancement is also limited. As soon as the end of a trajectory between the states is reached, the trajectory advancement is disabled to keep the robot stabilized.
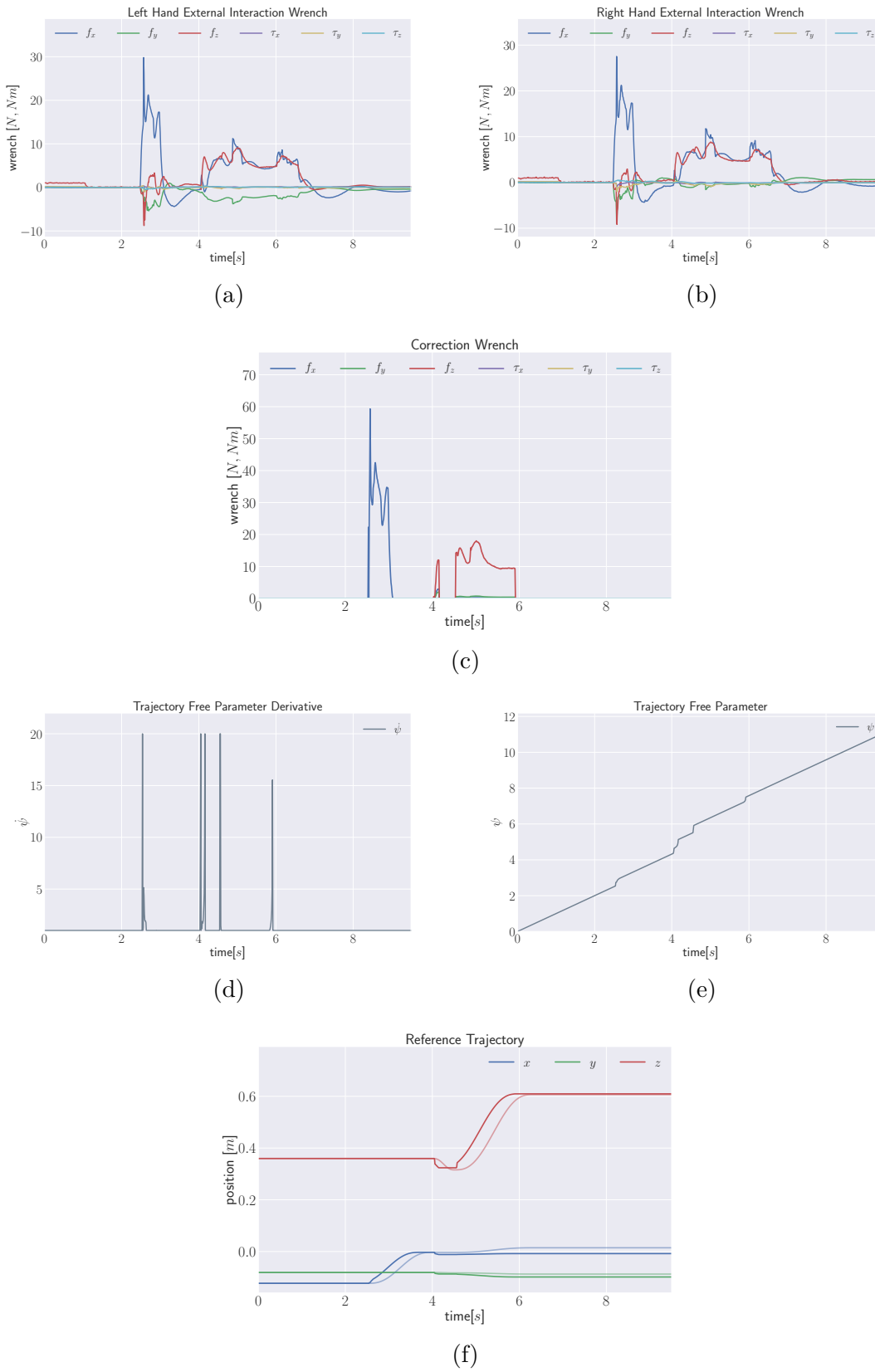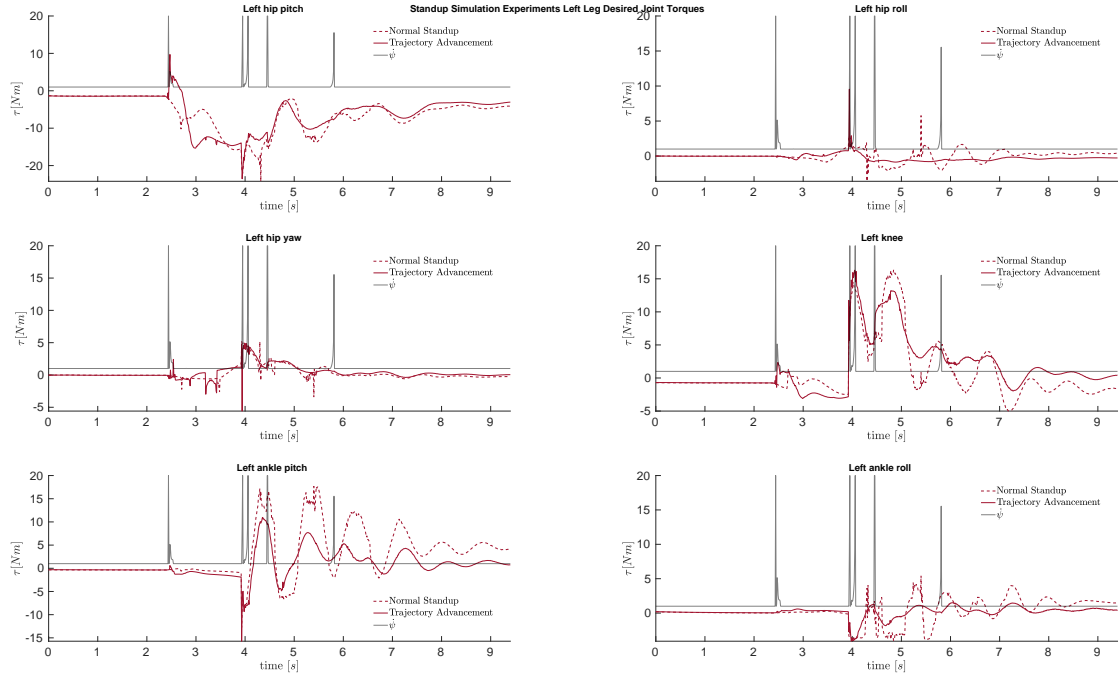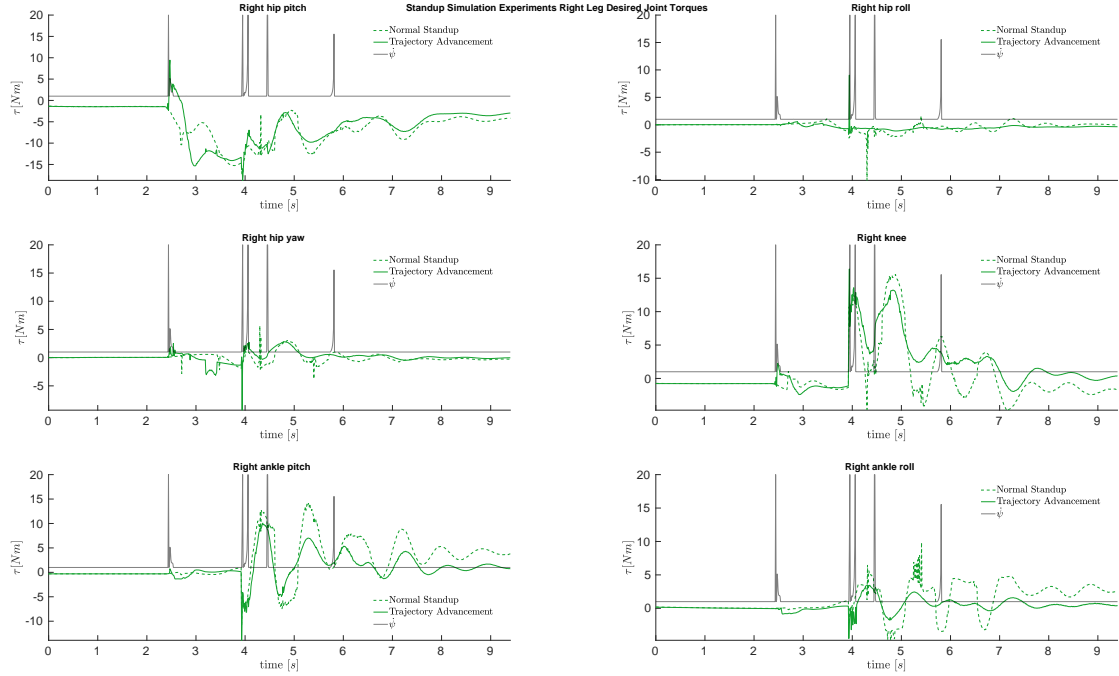
(a)

(b)

(c)

(d)

(e)

(f)

Fig. 9.10 Gazebo simulation trajectory advancement during sit-to-stand transition with whole-body standup controller   136

(a)

(b)

Fig. 9.11 Gazebo simulation standup experiment leg joint torques

9.5.2 Real Robot Results

The trajectory advancement experiments with the whole body standup controller are also carried with a real iCub humanoid robot as shown in Fig. 9.12. A human partner assists the robot by pulling it up and forward while the robot is performing the sit-to-stand transition.
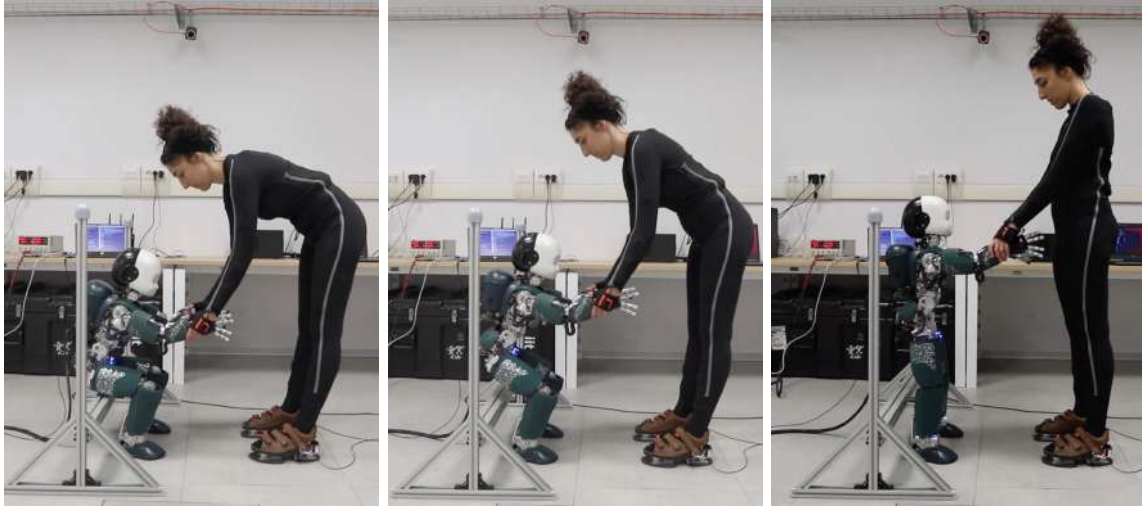
Fig. 9.12 iCub at different states during sit-to-stand transition with the whole-body standup controller (Sec 8.4) using assistance[1] from a human partner

The results of trajectory advancement on the real robot are shown in Fig. 9.13. Fig. 9.13a 9.13b highlights the wrench estimates at the hands of the robot under the influence of the external wrench applied while the human partner is assisting the robot. The external wrench that is helpful to achieve the task is shown in Fig. 9.13c. The reference trajectory is similar to a time parametrized trajectory i.e., $\psi = t$ until any helpful wrench is considered at the hands of the robot. Under the influence of helpful wrenches, the derivative of the trajectory free parameter $\dot{\psi}$ changes as shown in Fig. 9.13d and the corresponding trajectory advancement is reflected as an increase in ψ as seen in Fig. 9.13e. Accordingly, the reference is advanced further along the CoM reference trajectory as shown in Fig. 9.13f. The original CoM reference trajectory without trajectory advancement is shown in the same figure with reduced transparency. Furthermore, the center of mass trajectory tracking error is presented in Fig. 9.13g. The overall tracking error is kept low except during the moments of trajectory advancement, where there is a momentary increase in the error which decreases quickly.

---

[1] Human assistance is considered in terms of the interaction wrench

(a)

(b)

(c)

(d)

(e)

(f)

(g)
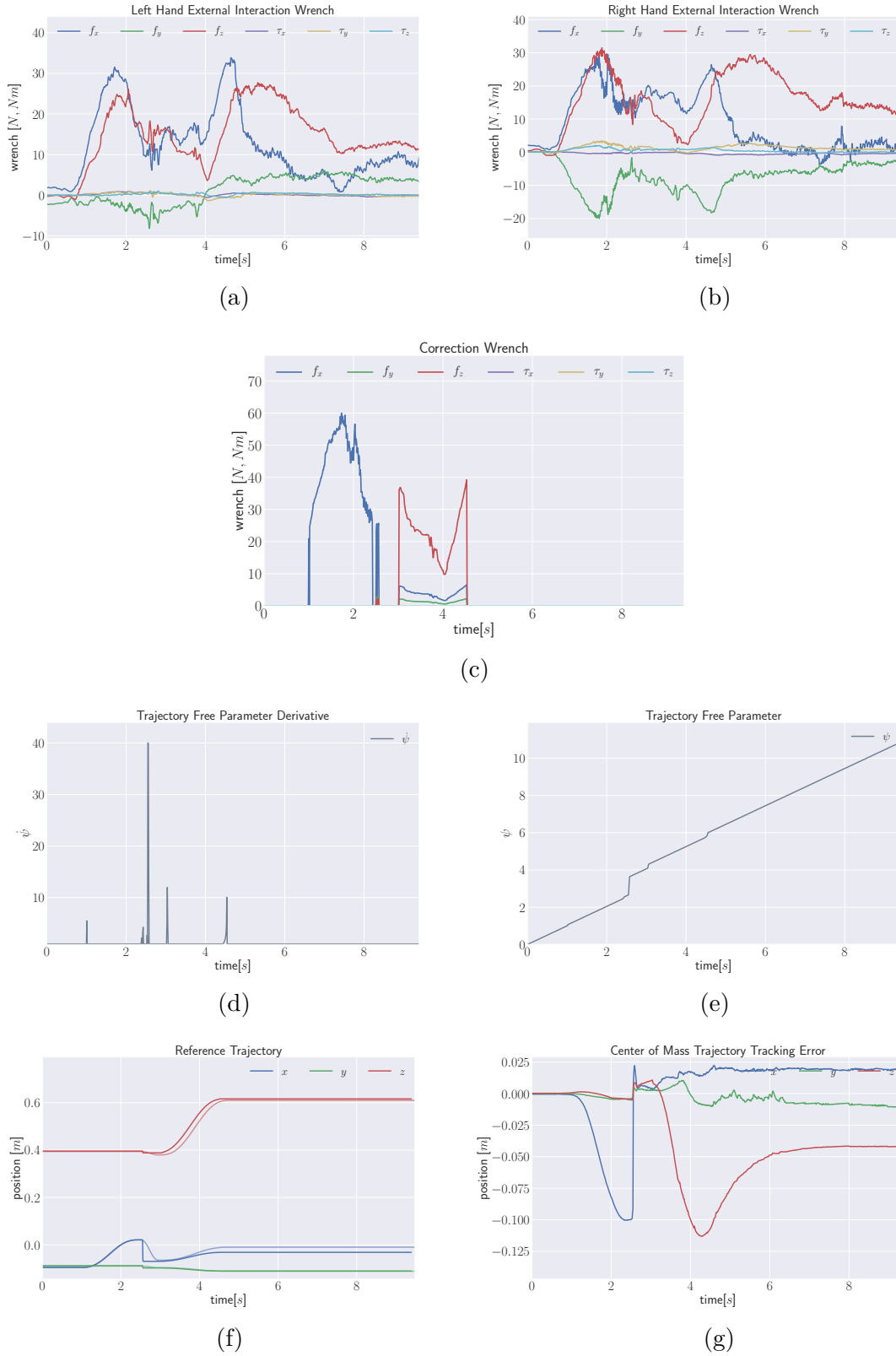
Fig. 9.13 Real robot trajectory advancement during sit-to-stand transition with whole-body standup controller

139

10

Whole-Body Retargeting & Teleoperation

The concept of telexistence of a human through a robotic avatar is a very challenging endeavor. It involves complex system integration of many technologies from research fields of perception, manipulation and control in robotics. This chapter presents the details of whole-body human motion retargeting to a humanoid avatar platform, and teleoperation experiments with state-of-the-art whole-body robot controllers for balancing and locomotion.

Some of the technologies involved on the human operator side and the robotic avatar side of a sophisticated telexistence setup are highlighted in Fig. 10.1. Potential applications of telexistence through a robotic avatar range from disaster response scenarios to providing health care and assistance remotely.
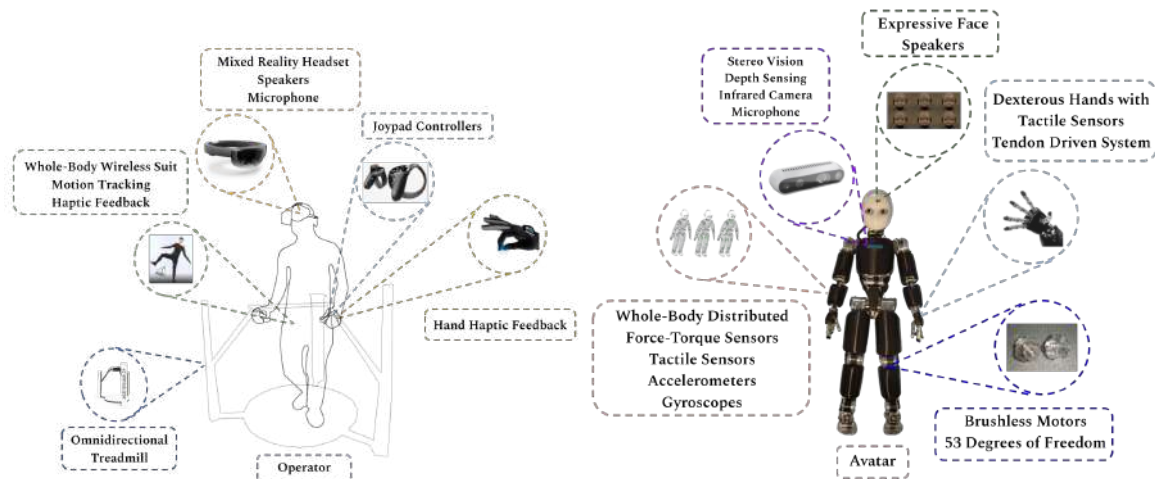
Fig. 10.1 Technologies involved on the human operator side and the robotic avatar side of a sophisticated telexistence setup

This chapter primarily focuses on the aspect of humanoid robot teleoperation using a novel framework of whole-body human motion retargeting to a humanoid robot avatar. Teleoperation is one of the core competencies for the successful realization of teleexistence of a human through a robotic avatar. Teleoperation stands for operating a robot from a distance that enables the extension of human capabilities to scenarios that are not reachable by humans due to the time and space constraints or the dangers posed by hazardous environments (Hokayem and Spong, 2006). The current robotic systems are still limited in terms of their perception and decision making capabilities to operate fully autonomous in real-world conditions. So, teleoperation facilitates an establishment of a human-robot team where the goal of the teleoperated robot is the same as the human operator and the chances of successful task completion are enhanced by bringing together the excellent cognitive capabilities of humans and the physical capabilities of robotic systems (Zucker et al., 2015). Teleoperation plays an important role in a wide range of real-world applications including manipulation in hazardous environments (Shimoga, 1993; Trevelyan et al., 2016), telepresence (Tachi et al., 1989), telesurgery (Burgner-Kahrs et al., 2015), and space exploration (Pedersen et al., 2003).

Concerning the robotic avatars, humanoid robots are an ideal platform as they are designed and built based on the principles of Anthropomorphism. Unlike other robotic systems, they often have higher maneuverability and manipulation capabilities that are similar to a human being (Ishiguro et al., 2018) which provides intuitive capabilities during telexistence. On the other hand, owing to their inherent complexity, humanoid robots are more challenging technologically for teleoperation in unstructured dynamic environments designed for humans. The level of autonomy, team organization and, the information exchange between the operator and the robot are some of the vital aspects in teleoperation performance to ensure successful task completion (Beer et al., 2014; Goodrich et al., 2008; Steinfeld et al., 2006). The level of autonomy ranges from being a semi-autonomous robot at the symbolic or the action level (high-level teleoperation) (Goodrich et al., 2008; Hokayem and Spong, 2006) to complete control of the robot at the kinematic and the dynamic level (low-level teleoperation), either in the robot's configuration space or task space. A core component of the low-level teleoperation system is the human motion retargeting to a robot.

Two of the most studied teleoperation paradigms in literature are: 1) master-slave; and 2) bilateral systems. Under master-slave teleoperation paradigm, the flow of information is unidirectional from the human to the robot, while under the bilateral teleoperation paradigm there is an exchange of information between the human and the

robot. In particular, haptic feedback to the human from the robot (Ishiguro et al., 2017; Wang et al., 2015). Teleoperation systems that involve humans in the control loop at the kinematic and dynamic level should have the prime objectives of situational awareness and transparency, i.e., the human operator experiencing the remote environment of the teleoperated robot as holistically as possible while maintaining the stability of the closed-loop system (Hokayem and Spong, 2006; Lichiardopol, 2007). Delays and information loss are some of the crucial problems with this approach that affect the transparency and stability of teleoperation greatly (Hokayem and Spong, 2006; Lichiardopol, 2007). Different approaches such as Lyapunov stability analysis (Chopra et al., 2003; Islam et al., 2015), passivity based control (Chopra et al., 2003) have been employed to address these limitations. However, these methods are studied extensively with manipulators and the stability measures for humanoid robot teleoperation are not well established (Ramos and Kim, 2018).

The research on teleoperation of humanoid robots can be broadly classified into three categories: upper body teleoperation, lower body teleoperation, and whole-body teleoperation. In upper body teleoperation, some of the research works consider mapping the human motion at the configuration space (Ayusawa and Yoshida, 2017; Liarokapis et al., 2013; Stanton et al., 2012) while some others consider the task space (Elobaid et al., 2019; Liarokapis et al., 2013). Furthermore, the effect of change in Center of Mass (COM) of the robot is also important to ensure the robot stability (Elobaid et al., 2019). So, concerning the lower body teleoperation of humanoid robots, the aspects of stability and locomotion have higher precedence over retargeting of all the lower limbs. A more detailed description of such approaches is discussed in (Feng et al., 2015; Romualdi et al., 2018). Coming to the topic of whole-body teleoperation of humanoid robots, the key challenge is to control the robot such that it does not fall while keeping its maneuverability and manipulability high, ensuring task completion by the human-robot team. Typically, the balance of the robot is achieved by either keeping the center of mass inside the support polygon or maintaining the net momentum about the Center of Pressure (COP) to zero (Ishiguro et al., 2018; Penco et al., 2018). Tasks involving multi-link dynamic contacts such as locomotion which involves careful monitoring and regulation of force exchange with the environment pose higher levels of complexity for teleoperation.

Whole-body human motion retargeting to a humanoid robot avatar is a critical component in the design of teleoperation technologies. One of the obvious shortcomings of the teleoperation systems proposed in the literature is the lack of ability to quickly and easily adapt the system for different human users and humanoid robots with

different geometries, kinematics, and dynamics. The system designer often spends time and effort in careful consideration of changes in the human models and the robot models to ensure successful teleoperation which limits *usability* and *scalability.*

This chapter aims at addressing this problem through a novel framework for whole-body human motion retargeting that requires minimal changes to use with different human operators or different robot avatars. Furthermore, experiments on whole-body teleoperation of humanoid robots with two state-of-the-art whole-body controllers for humanoid robots are presented.

Notation

This chapter contains new notations presented in the following table in addition to the details presented in the nomenclature (1) and background notation (2.1).

$$
\begin{array}{ll}
\mathbf{i} \quad (.)^R & \text{Robot quantity} \\
\quad (.)^H & \text{Human quantity}
\end{array}
$$

10.1 Human Motion Retargeting

Human motion measurements are acquired through the Xsens whole-body motion tracking suit with distributed inertial measurement units as described in Section 4.1. The two approaches of retargeting human motion can be categorized as configuration space retargeting and task space retargeting.

10.1.1 Configuration space retargeting

Given a human model and the measurements of various limbs in terms of position and orientation, an inverse kinematics algorithm is employed to retrieve the human joint positions and velocities. A configuration space mapping between the human operator model and the robotic avatar is used as a reference to compute the joint position references for controlling the robot motion.

The architecture shown in Fig. 10.2 represents a typical configuration space retargeting scheme (Ayusawa and Yoshida, 2017; Penco et al., 2018).
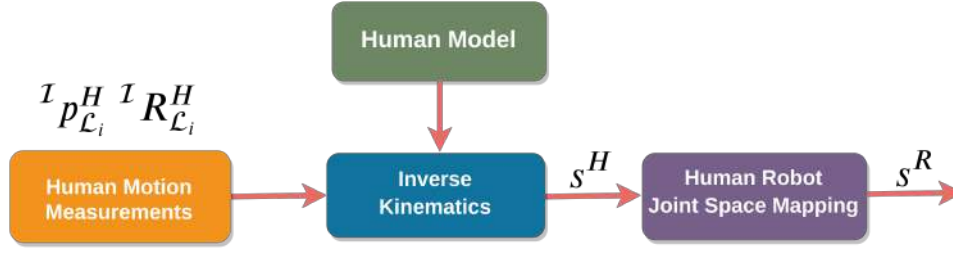
Fig. 10.2 Typical configuration space retargeting scheme

A key component in this approach is the customized mapping between the human operator and the robotic avatar in the configuration space. Oftentimes, the complexity and range of motion of certain human joints like the shoulder joint are very complex. So, the configuration space mapping must ensure these complexities and furthermore the limits of the robot joints to enable safe configuration space control references to the robot.

10.1.2 Task space retargeting

Task space retargeting approach depends on the task space mapping between the human operator limbs and the robotic avatar limbs. Leveraging the task space mapping, the human limb motion measurements are converted to appropriate robot limb motion references. Given a robot model and the computed robot limb motion references, an inverse kinematics algorithm is employed to compute the robot references in the joint space considering the robot joint limitations. The architecture shown in Fig. 10.3 represents a typical task space retargeting scheme (Elobaid et al., 2019; Ishiguro et al., 2018).

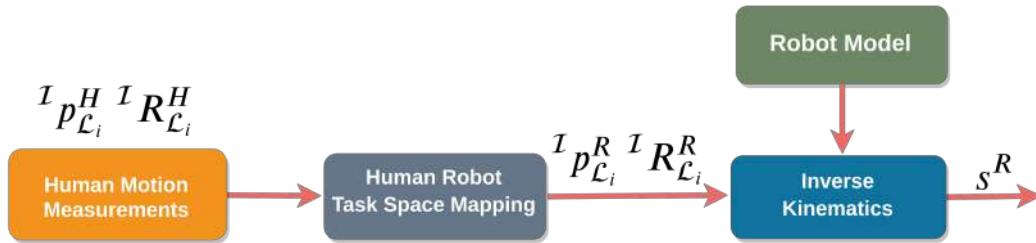Fig. 10.3 Typical task space retargeting scheme

Clearly, a key component in this approach is finding the task space mapping between the human operator limbs and the robotic avatar limbs. Oftentimes, the scale of the

human limbs are very different than that of the robot. Furthermore, considering the task space motion as a reference for the robot may lead to a robot joint configuration that may be dissimilar from the human joint configuration. This may often lead to psychological discomfort for the human operator or the people who are interacting with the robot as one cannot predict the robot's motions owing to non-anthropomorphic motions that depend on the inverse kinematics algorithm employed (Liarokapis et al., 2013). Moreover, the precise control of the robot's joint configuration becomes essential when the robot is deployed in cluttered environments where avoiding obstacles is crucial of mission success.

10.1.3 Our Approach

In our approach to whole-body kinematic motion retargeting, we first identify the task space mapping between the human links and the robot links in a neutral pose as shown in Figure. 10.4. The frame equivalence from the human to the robot links is indicated by the numbering. The task space mapping is identified in terms of the rotation between the human links and the robot links i.e. $^{H_{\mathcal{L}_i}}\boldsymbol{R}_{R_{\mathcal{L}_i}}$. The robot model is updated with new frames (attached to the robot links) that are identical to the human link frames. Given the rotation measurement from the human link frames to the inertial frame, $^{\mathcal{I}}\boldsymbol{R}_{H_{\mathcal{L}_i}}$, and the constant rotation from the robot link frames to human link frames, $^{H_{\mathcal{L}_i}}\boldsymbol{R}_{R_{\mathcal{L}_i}}$, the rotation measurement of the robot frame with respect to the inertial frame is computed as,

$$^{\mathcal{I}}\boldsymbol{R}^*_{R_{\mathcal{L}_i}} = {}^{\mathcal{I}}\boldsymbol{R}_{H_{\mathcal{L}_i}} {}^{H_{\mathcal{L}_i}}\boldsymbol{R}_{R_{\mathcal{L}_i}} \tag{10.1}$$
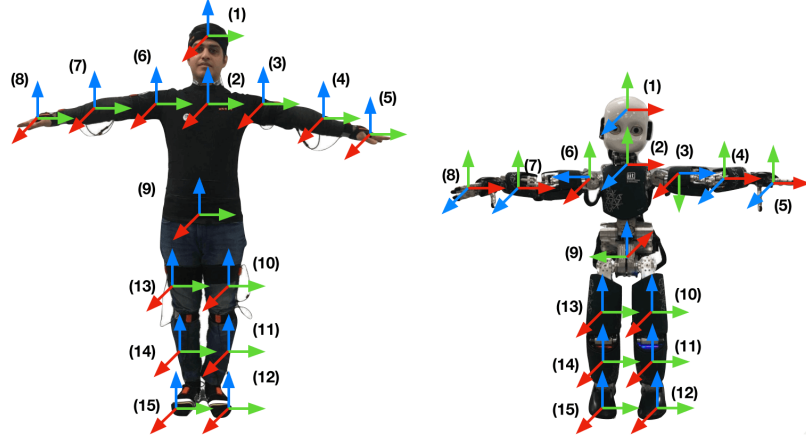
Fig. 10.4 Task space mapping between the human links and the robot links in neutral pose

Any measurements coming from the human are easily converted to the measurements of the robot links through the rotation mapping using Eq. 10.1. Once the desired robot link orientation is computed, thanks to our modular software architecture, we leverage the dynamical optimization inverse kinematics (Rapetti et al., 2019) outlined in Section 5.3 on the updated robot model to compute the robot joint positions s^R. Figure 10.5 highlights our whole-body kinematic human motion retargeting approach.
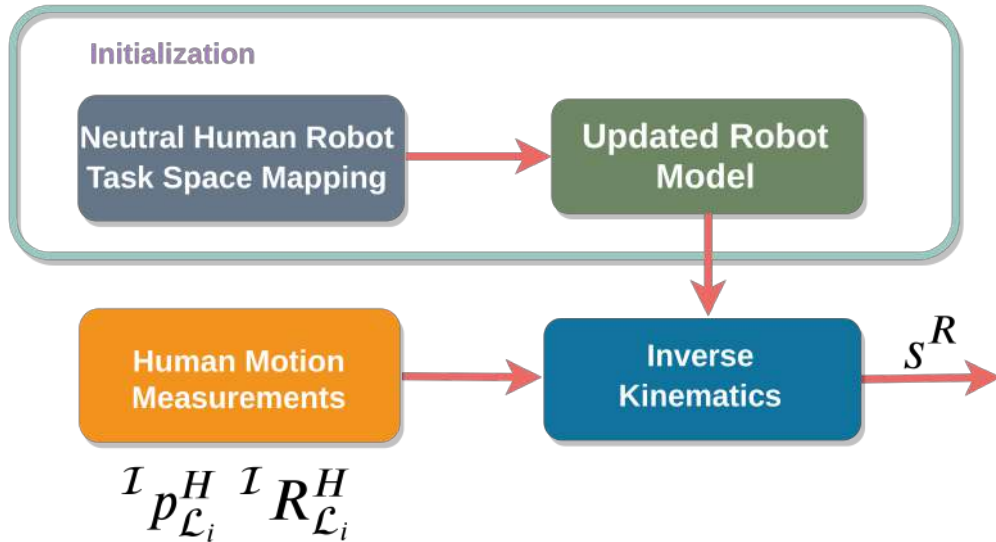
Fig. 10.5 Block diagram of our kinematic whole-body motion retargeting

10.2 Whole-Body Retargeting Experiments

The software architecture for performing whole-body retargeting experiments is highlighted in Fig. 10.6. Thanks to our modular infrastructure for holistic human perception presented in Chapter 7, only minimal changes are required to obtain the robot joint states for performing the retargeting. As highlighted in Fig. 10.6, the key modification is to use the updated robot model from section 10.1.3 instead of the human model.

The whole-body retargeting experiments are performed with motion data captured for two human subjects. To demonstrate the scalability and usability of our proposed method, we perform whole-body kinematic retargeting with multiple robots having different degrees of freedom (DoFs). The robot models we considered are a) iCub humanoid robot with 32 DoFs, b) NAO humanoid robot with 24 DoFs, c) Atlas humanoid robot with 30 DoFs. To show that our method is not limited to humanoid robots, we perform a retargeting scenario with Baxter dual arm 15 DoFs robot. Additionally, we show the retargeting with a human model that has 66 DoFs which is described in Chapter 3.

The Fig. 10.7 highlights our whole-body kinematic retargeting with different models and human subjects in Rviz, a 3D visualizer. The first row corresponds to the human motion of standing on the right foot from the first subject and the second row corresponds to the human motion of standing on the left foot by the second subject. Concerning the Baxter robot, the retargeting is done only for the arms and the head, as it is a fixed base robot.

The joint angles values of the right arm elbow joint for the human and the iCub robot model are highlighted in Fig. 10.8a. The overall retargeting of the joint position is good except for some configurations of the human where the robot is constrained by the joint limits e.g., time instant $t \sim 13\,\mathrm{s}$. The lower leg link orientation (in terms of *Euler angles*) of the human and the iCub robot model is highlighted in Fig. 10.8b.

10.3 Whole-Body Teleoperation Architecture

At the current state, our teleoperation architecture highlighted in Fig. 10.9 is composed of the following technologies:
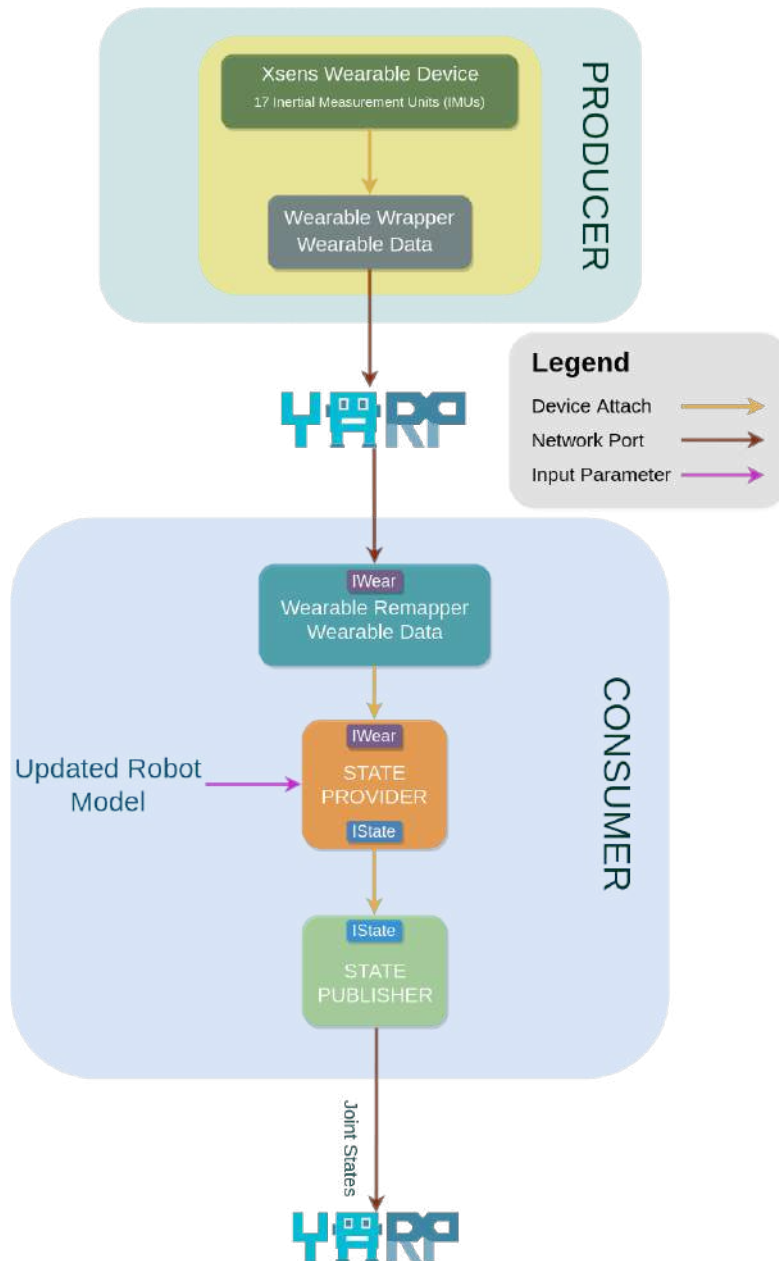
Fig. 10.6 Software architecture from Chapter 7 for whole-body retargeting using the updated robot model

- Oculus Virtual Reality Headset: Visual feedback from the robot environment by streaming the robot camera images to the human operator

- Joypads: The robot hands are controlled via the joypads for opening and closing the fingers during manipulation scenarios
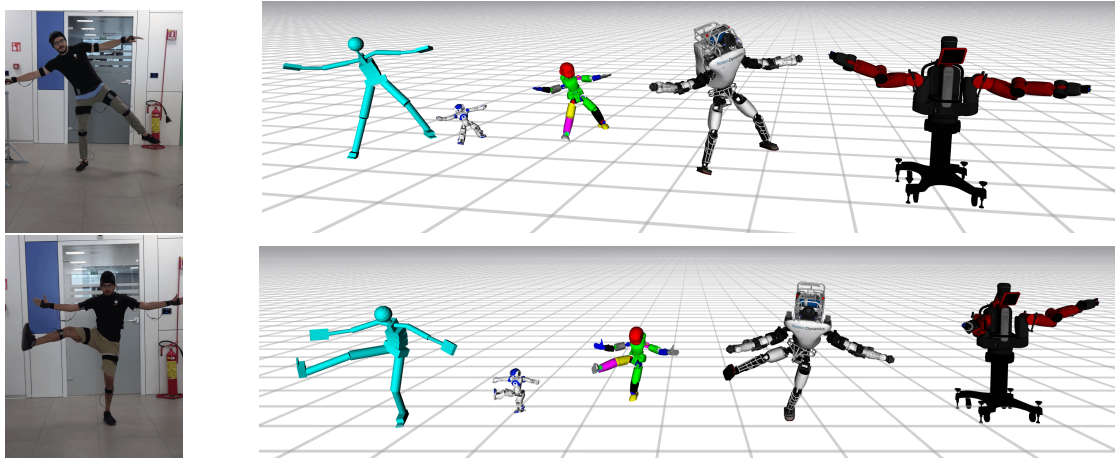
Fig. 10.7 Rviz visualization of whole-body retargeting of different human subjects motion to different models: a) Human Model b) Nao c) iCub d) Atlas e) Baxter; top: human subject stands on the right foot, bottom: human subject stands on the left foot

(a)

(b)
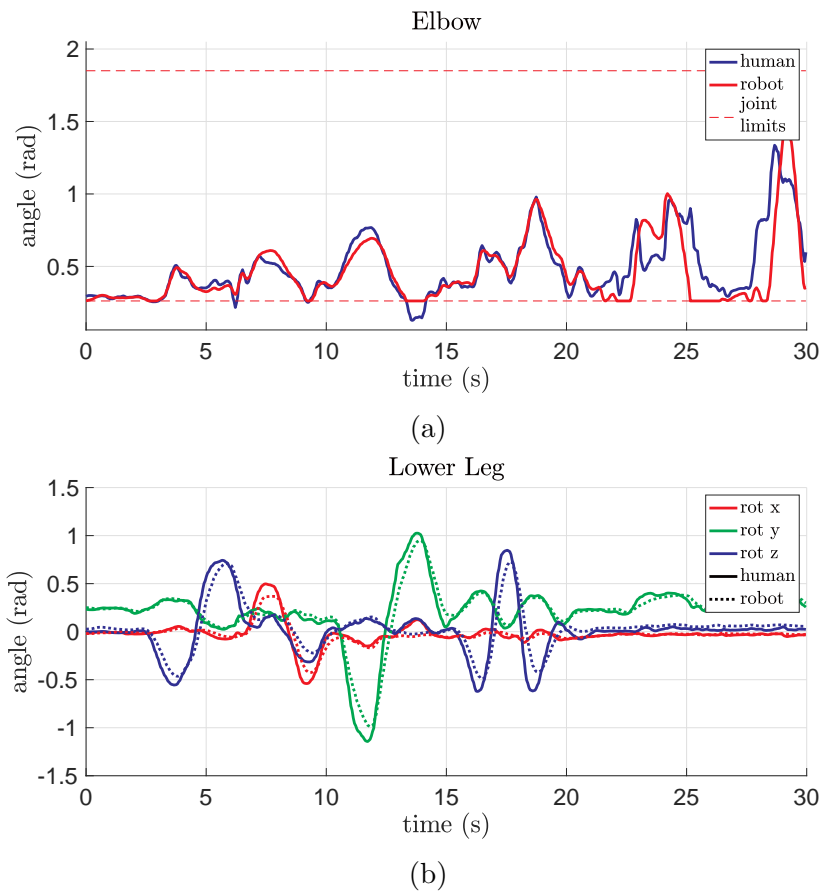
Fig. 10.8 Performance of whole-body retargeting of human motions to iCub

- Cyberith Virtualizer Virtual Reality Treadmill: The human locomotion information i.e., the linear velocity in x and y directions and the angular velocities about the z direction

- Whole-Body Motion Tracking Suit: Human motion perception through inertial tracking technology described in Section 4.1
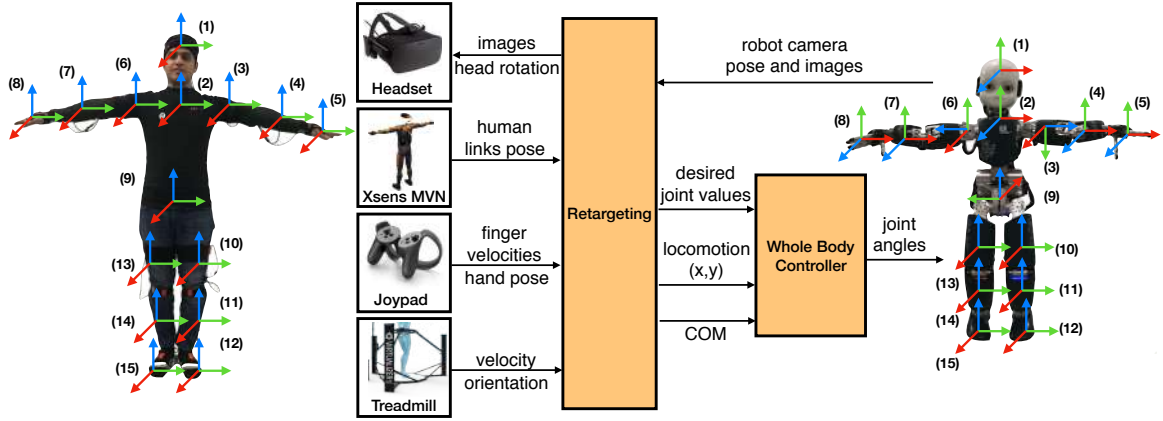
Fig. 10.9 Whole-body teleoperation architecture with real-time human motion retargeting

10.4   Whole-Body Teleoperation Experiments

Towards demonstrating the capabilities of our whole-body retargeting, we perform teleoperation experiments using two state-of-the-art *whole-body controllers* for humanoid robots. The whole-body teleoperation experiments are carried with the 53 degrees of freedom iCub robot that is 104 cm tall (Natale et al., 2017). The controllers run at 100 Hz while the retargeting application runs at 200 Hz. The average walking speed of the robot is $0.23 \, \mathrm{m \, s^{-1}}$. Both the applications are run on a machine of 4th generation Intel Core i7@1.7 GHz with 8GB of RAM.

10.4.1   Whole-Body Teleoperation with Balancing Controller

Momentum-based control (Herzog et al., 2014; Nava et al., 2016) proved to be effective for maintaining the robot's stability by controlling the robot's momentum as the primary objective. Additionally, a *postural task* projected into the nullspace of the primary task can be used for performing additional tasks like manipulation

while ensuring the stability of the robot. The control problem is formulated as an optimization problem to achieve the two tasks while carefully monitoring and regulating the contact wrenches, considering the associated feasible domains by resorting to quadratic programming (QP) solvers.

We considered one such momentum-based balancing controller (Nava et al., 2016) and extended the postural task by giving the joint references from whole-body retargeting. The technologies involved in the teleoperation with the whole-body balancing controller are highlighted in Fig. 10.10. The snapshots from the experiments of the whole-body retargeting with the balancing controller are shown in Fig. 10.11.
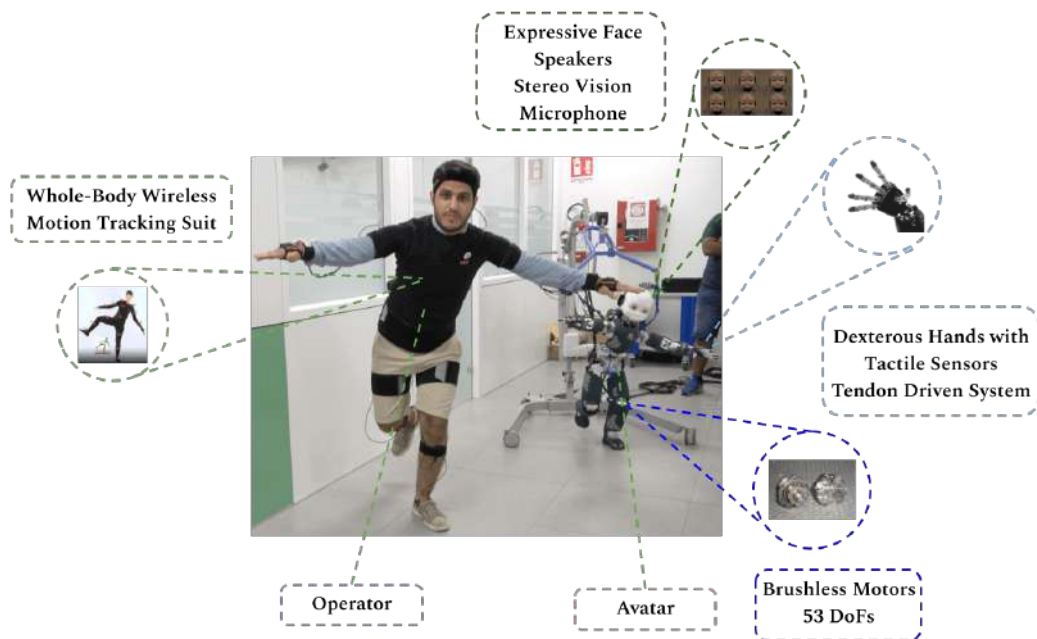
Fig. 10.10 Technologies involved in teleoperation with whole-body balancing controller

In this experiment, the robot is balancing on the left foot and maintaining the stability of its center of mass as shown in Fig. 10.12. Additionally, it tracks all the joints with the references coming from whole-body retargeting. The vertical dashed lines correspond to the experimental snapshots indicated in Fig. 10.11. The references to the x and y components of the CoM are close to zero to maintain the stability of the robot by keeping the CoM inside the support polygon and the gains are tuned to achieve good tracking. The CoM motion along the z-axis does not affect the stability of the robot and the gain value of the z components is kept lower in order to allow the vertical movements of the robot during retargeting. The input joint references from retargeting are smoothed through a minimum-jerk trajectory (Pattacini et al., 2010b).

(a)   (b)

(c)   (d)
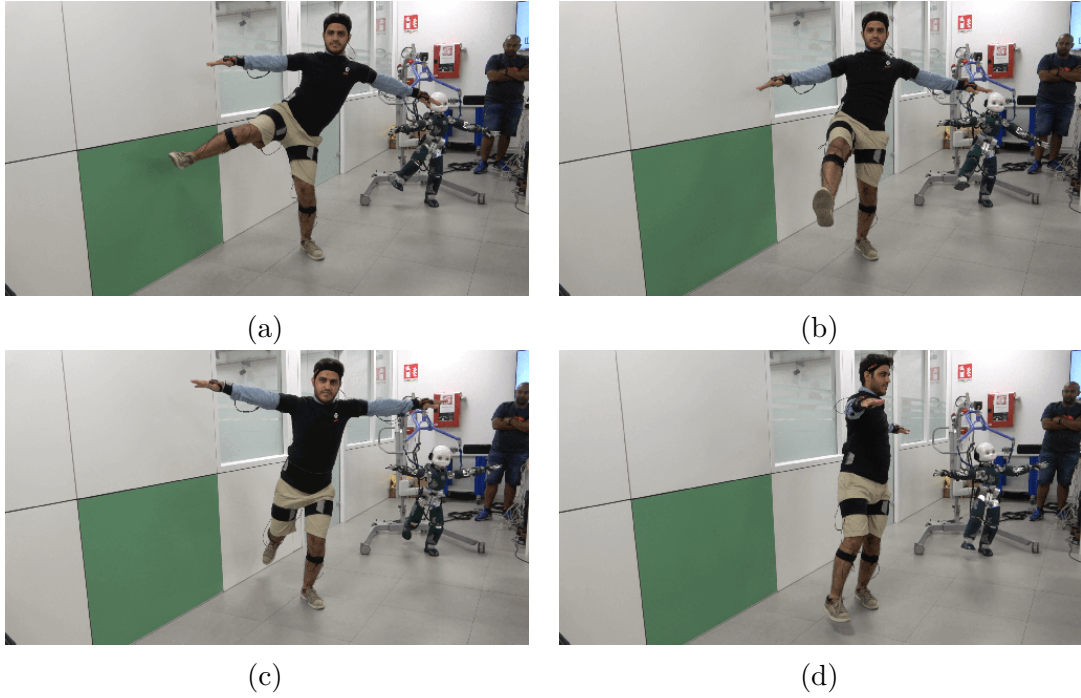
Fig. 10.11 Snapshots of real-time motion retargeting with whole-body balancing controller at different time instances

A smoothing time parameter is tuned in order to achieve a good balance between postural tracking and stability. Accordingly, the joints such as torso pitch, torso roll, and left knee for which the human does not move fast while balancing on left foot, achieve good tracking. On the other hand, the joints such as right shoulder pitch, right shoulder roll, and left ankle pitch are moved frequently while performing the retargeting and hence the tracking is not close owing to the delay from the smoothing time involved in producing minimum-jerk trajectory joint references for the robot joints. Ideally, the smoothing time can be kept lower considering that we receive continuous joint references from retargeting. At this point, we did not conduct exhaustive tests to find the lower threshold for the smoothing parameter that ensures fast and accurate retargeting of dynamic motions from the human while maintaining the robot's stability.

Fig. 10.12 Center of mass tracking and the time evolution of joint angles during real-time motion retargeting with whole-body balancing controller; blue line represents the desired quantity, orange line is the actual robot quantity. The vertical lines correspond to the time instances representing the snapshots from Fig. 10.11

10.4.2 Whole-Body Teleoperation with Walking Controller

Humanoid robot walking is another challenging control paradigm. Divergent-Component-of-Motion (DCM) based control architectures proved promising for humanoid robot locomotion (Englsberger et al., 2015; Romualdi et al., 2018). The architecture typically consists of three layers: 1) Trajectory generation and optimization layer that generates the desired footsteps and the DCM trajectories (Englsberger et al., 2015); 2) Simplified model control layer that implements an *instantaneous* control law with the objective of stabilizing the unstable DCM dynamics; and 3) Whole-body control layer that guarantees the tracking of the robot's set of tasks, including the Cartesian tasks and the postural tasks, using the stack-of-tasks paradigm implemented through a quadratic programming (QP) formalism.

We considered one such DCM based walking controller (Romualdi et al., 2018) and extended the postural task by giving the joint references from whole-body retargeting. The technologies involved in the teleoperation with the whole-body walking controller are highlighted in Fig. 10.13.
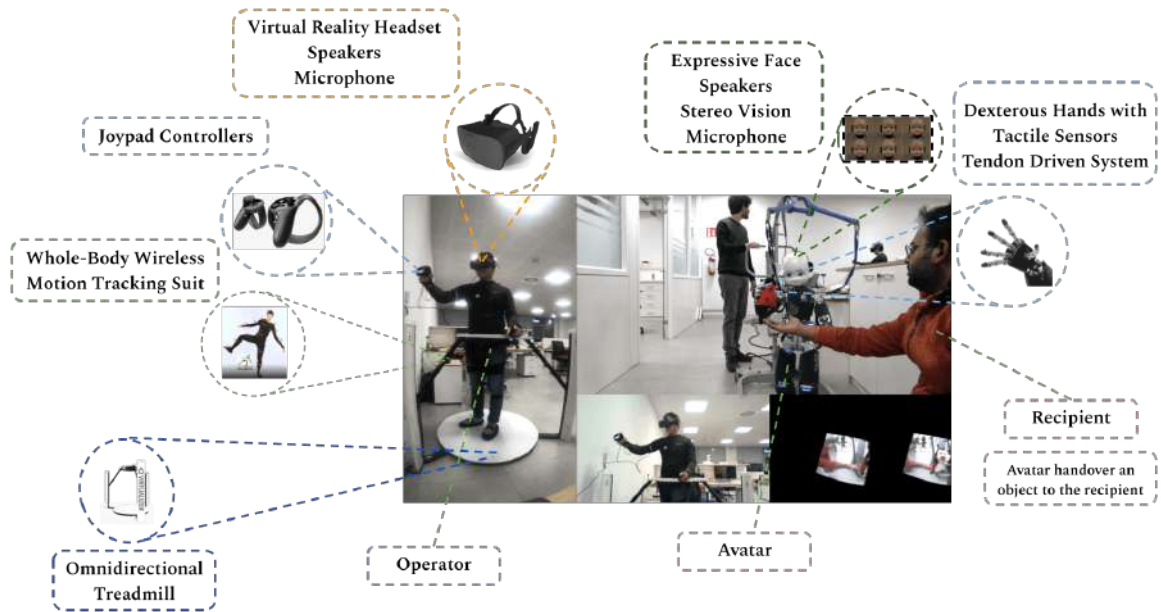
Fig. 10.13 Technologies involved in teleoperation with the whole-body walking controller for locomotion and manipulation

The three different experimental stages of whole-body retargeting with the walking controller are shown in Fig. 10.14. During the first and the third stages the robot is in

double support standstill phase while during the second stage the robot is in walking phase.

(a) first stage

(b) second stage

(c) third stage

Fig. 10.14 Experimental stages during real-time retargeting with whole-body walking controller

The walking controller's primary objective is to track the center of mass x and y components along the desired trajectory. The overall center of mass tracking of the x and y components is very good for the entire duration of the experiment as shown in Fig. 10.15.

Currently, we engage only the upper body retargeting, and the lower body is controlled by the walking controller. During our experiments, we observed that the weights for achieving satisfactory upper body retargeting of the postural task during the double support standstill phase and the walking phase are different. Having the same retargeting gains while walking leads to uncoordinated movements eventually compromising the robot's stability while walking. So, we choose higher retargeting gains during the double support standstill phase and the gain values are set to zero during the walking phase. The transition between the two sets of weights is achieved smoothly through minimum jerk trajectories (Pattacini et al., 2010b). Fig. 10.15 highlights tracking for some of the upper-body joints. The blue line represents the desired joint position provided by human motion retargeting and the orange line is the actual robot joint position. The purple vertical dashed line indicates the starting instance of the *second stage*, i.e., walking, and the green vertical dashed line indicates the stopping instance of the walking phase. During the *first stage*, human motion retargeting is good and the joint position error is low. Instead, during the *second stage*, as the robot starts walking the joint position error is higher as the retargeting gains are set to zero.
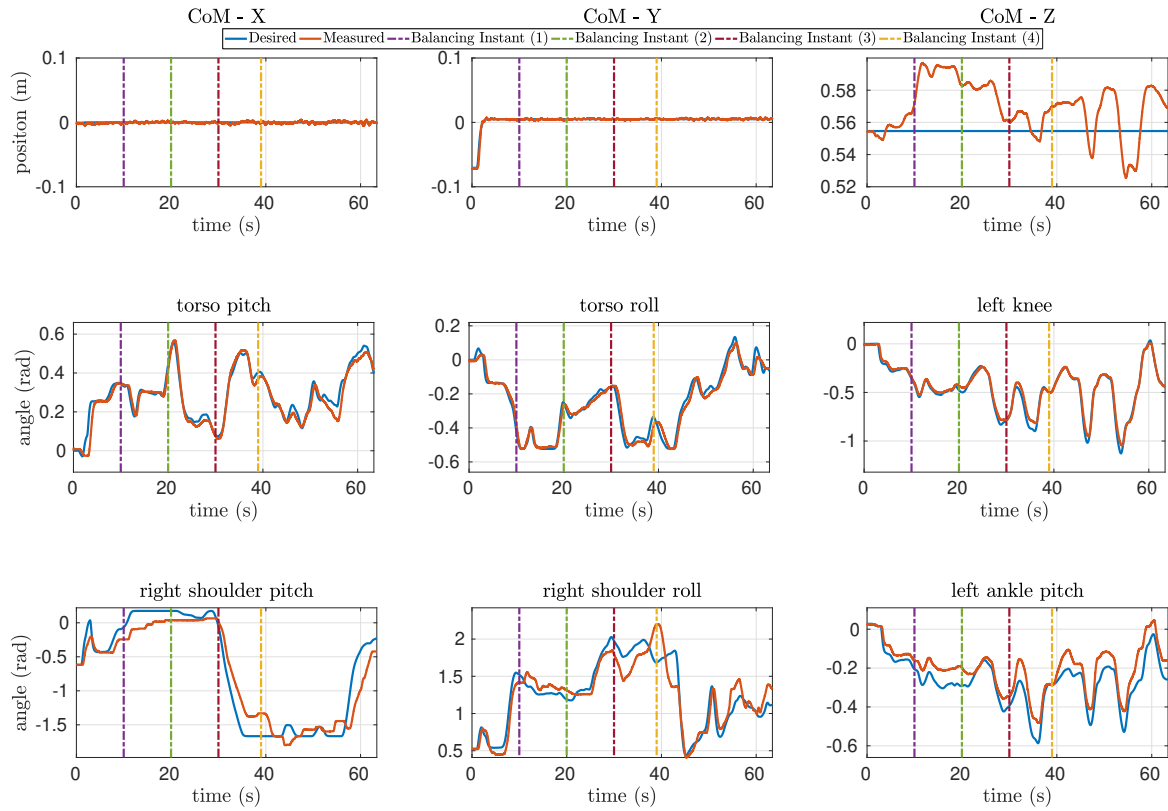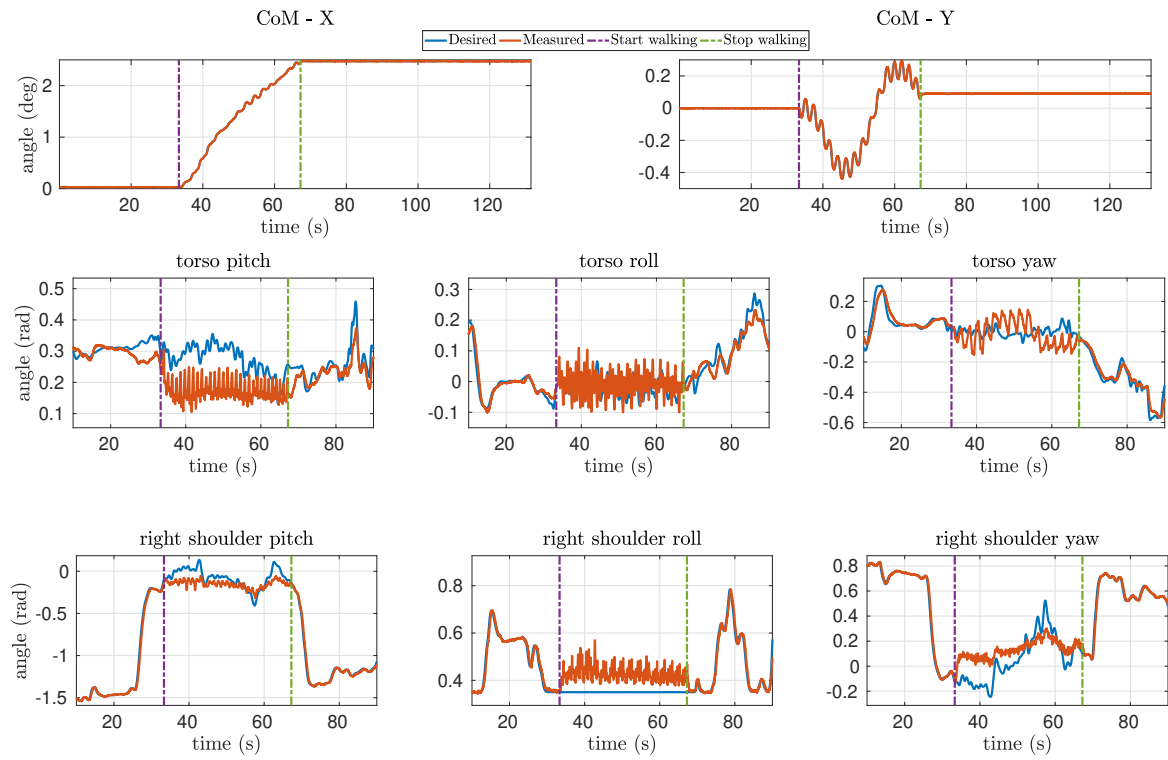
Fig. 10.15 Center of mass tracking and the time evolution of joint angles during whole-body retargeting with walking controller

Epilogue

This thesis attempts at addressing some of the challenges for enabling human-robot collaboration. It presents a holistic human perception framework for real-time monitoring of whole-body human motion and dynamics. Furthermore, it attempts at methodically defining what constitutes an assistance from a human partner and propose partner-aware robot control strategies to endow robots with the capacity to meaningfully engage in a collaborative task.

The manuscript is divided into three main parts:

- The first part describes the role of human and robot partners in future socio-technical systems, followed by the mathematical preliminaries and background on rigid body systems. Furthermore, it recalls the importance of the human modeling concerning human-robot collaboration and presents the different enabling technologies that are employed in carrying out the research presented in this thesis.

- The second part focuses on the challenge of holistic human perception and presents promising approaches to real-time human motion tracking of a highly articulated human model. A novel sensorless external force estimation on the human links through a stochastic human dynamics estimation for a floating base human model is presented with experimental validation. Lastly, a modular and extensible software architecture towards realizing a holistic human perception framework is explained in detail.

- The final part is dedicated to reactive robot control, where we attempt at methodically defining what constitutes an assistance from a human partner and propose partner-aware robot control strategies to endow robots with the capacity to meaningfully engage in a collaborative task. Furthermore, the concept of trajectory advancement for intuitive robot behavior leveraging interaction from a human partner is presented with experimental validation. Finally, our research towards sophisticated teleexistence setup is explained in detail followed by extensive experimental validation.

Before bringing down the curtains on this manuscript, we would like to reflect on some of the open questions and challenges that will guide future research aspirations. Firstly, inclusion of prediction and classification components in our human perception framework will facilitate new application scenarios. Concerning the aspect of human modeling, currently we precompute the inertial parameters of the human segments based on anthropometric tables and model it explicitly based on the measured parameters of the human subject. This approach however is not easily scalable and adaptable for different population (Matrangola et al., 2008). A promising research in this direction is the identification of body segment inertial parameters (Hansen et al., 2014; Venture et al., 2019). Automatic whole-body human model generation that is adaptable for multiple applications is still an open challenge (Venture et al., 2019) and there is a need for bridging the gaps between digital human models employed in different fields. Furthermore, higher level planning for task sharing between the human partner and the robot partner (Raessa et al., 2019) is an another interesting and potential research area that can facilitate new insights.

Concerning the robot control, the investigations presented in this thesis primarily focus on how the human partner assistance is leveraged by the robot partner for instantaneous control. On the other side of this question is how the robot partner can assist the human partner through holistic human perception and predictive control. Possible research endeavors in this direction are to build whole-body robot controllers that monitor the human joint torques and improve human ergonomics while performing complicated tasks such as collaborative transportation. Furthermore, the concept of trajectory advancement can be applied to exoskeleton control, where the physical interaction from the human guides the robotic exoskeleton to assist. Finally, adding a planning component will extend the applicability to collaborative scenarios that need deliberation between human and robot partners.

References

Aggarwal, J. K. and Cai, Q. (1999). Human motion analysis: A review. *Computer vision and image understanding*, 73(3):428–440.

Agravante, D. J., Cherubini, A., Bussy, A., Gergondet, P., and Kheddar, A. (2014). Collaborative human-humanoid carrying using vision and haptic sensing. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 607–612. IEEE.

Aguiar, A. P., Dačić, D. B., Hespanha, J. P., and Kokotović, P. (2004). Path-following or reference tracking?: An answer relaxing the limits to performance. *IFAC Proceedings Volumes*, 37(8):167–172.

Ajoudani, A., Zanchettin, A. M., Ivaldi, S., Albu-Schäffer, A., Kosuge, K., and Khatib, O. (2018). Progress and prospects of the human–robot collaboration. *Autonomous Robots*, 42(5):957–975.

Alami, R. (2013). On human models for collaborative robots. In *2013 International Conference on Collaboration Technologies and Systems (CTS)*, pages 191–194. IEEE.

Alami, R., Albu-Schäffer, A., Bicchi, A., Bischoff, R., Chatila, R., De Luca, A., De Santis, A., Giralt, G., Guiochet, J., Hirzinger, G., et al. (2006). Safe and dependable physical human-robot interaction in anthropic domains: State of the art and challenges. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1–16. IEEE.

Alexopoulos, K., Mavrikios, D., and Chryssolouris, G. (2013). Ergotoolkit: an ergonomic analysis tool in a virtual manufacturing environment. *Int. J. Computer Integrated Manufacturing*, 26:440–452.

Andrade Chavez, F. J., Traversaro, S., and Pucci, D. (2019). Six-axis force torque sensor model-based in situ calibration method and its impact in floating-based robot dynamic performance. *Sensors*, 19(24):5521.

An.Dy (2017). Advancing anticipatory behaviors in dyadic human-robot collaboration, h2020 project.

Aristidou, A., Lasenby, J., Chrysanthou, Y., and Shamir, A. (2018). Inverse kinematics techniques in computer graphics: A survey. In *Computer Graphics Forum*, volume 37, pages 35–58. Wiley Online Library.

Au, S., Berniker, M., and Herr, H. (2008). Powered ankle-foot prosthesis to assist level-ground and stair-descent gaits. *Neural Networks*, 21(4):654–666.

Ayusawa, K. and Yoshida, E. (2017). Motion retargeting for humanoid robots based on simultaneous morphing parameter identification and motion optimization. *IEEE Transactions on Robotics*, 33(6):1343–1357.

Bartneck, C., Kulić, D., Croft, E., and Zoghbi, S. (2009). Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots. *International journal of social robotics*, 1(1):71–81.

Beer, J. M., Fisk, A. D., and Rogers, W. A. (2014). Toward a framework for levels of robot autonomy in human-robot interaction. *Journal of Human-Robot Interaction*, 3(2):74–99.

Belhassein, K., Buisan, G., Clodic, A., and Alami, R. (2019). Towards methodological principles for user studies in human-robot interaction.

Bharatkumar, A. G., Daigle, K. E., Pandy, M. G., Qin Cai, and Aggarwal, J. K. (1994). Lower limb kinematics of human walking with the medial axis transformation. In *Proceedings of 1994 IEEE Workshop on Motion of Non-rigid and Articulated Objects*, pages 70–76.

Blender (2019). *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam.

Bragança, S., Costa, E., Castellucci, I., and Arezes, P. M. (2019). *A Brief Overview of the Use of Collaborative Robots in Industry 4.0: Human Role and Safety*, pages 641–650. Springer International Publishing, Cham.

Breivik, M. and Fossen, T. I. (2005). Principles of guidance-based path following in 2d and 3d. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 627–634. IEEE.

Burgner-Kahrs, J., Rucker, D. C., and Choset, H. (2015). Continuum robots for medical applications: A survey. *IEEE Transactions on Robotics*, 31(6):1261–1280.

Buss, S. R. (2004). Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods. *IEEE Journal of Robotics and Automation*, 17(1-19):16.

Bussy, A., Gergondet, P., Kheddar, A., Keith, F., and Crosnier, A. (2012a). Proactive behavior of a humanoid robot in a haptic transportation task with a human partner. In *RO-MAN, 2012 IEEE*, pages 962–967. IEEE.

Bussy, A., Kheddar, A., Crosnier, A., and Keith, F. (2012b). Human-humanoid haptic joint object transportation case study. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 3633–3638. IEEE.

Caux, S., Mateo, E., and Zapata, R. (1998). Balance of biped robots: special double-inverted pendulum. In *Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on*, volume 4, pages 3691–3696. IEEE.

Chavez, F. J. A., Traversaro, S., Pucci, D., and Nori, F. (2016). Model based in situ calibration of six axis force torque sensors. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 422–427. IEEE.

Cho, E., Chen, R., Merhi, L.-K., Xiao, Z., Pousett, B., and Menon, C. (2016). Force myography to control robotic upper extremity prostheses: a feasibility study. *Frontiers in bioengineering and biotechnology*, 4:18.

Chopra, N., Spong, M. W., Hirche, S., and Buss, M. (2003). Bilateral teleoperation over the internet: the time varying delay problem. In *Proceedings of the 2003 American Control Conference*, volume 1, pages 155–160, Denver, CO, USA.

Craig, J. J. (2009). *Introduction to robotics: mechanics and control, 3/E.* Pearson Education India.

Dafarra, S., Romano, F., and Nori, F. (2016). Torque-controlled stepping-strategy push recovery: Design and implementation on the icub humanoid robot. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 152–157. IEEE.

Damiano, L., Dumouchel, P., and Lehmann, H. (2015). Towards human–robot affective co-evolution overcoming oppositions in constructing emotions and empathy. *International Journal of Social Robotics*, 7(1):7–18.

De Santis, A., Lippiello, V., Siciliano, B., and Villani, L. (2007). Human-robot interaction control using force and vision. In *Advances in Control Theory and Applications*, pages 51–70. Springer.

Delp, S. L., Anderson, F. C., Arnold, A. S., Loan, P., Habib, A., John, C. T., Guendelman, E., and Thelen, D. G. (2007). Opensim: open-source software to create and analyze dynamic simulations of movement. *IEEE transactions on biomedical engineering*, 54(11):1940–1950.

Demirel, H. O. and Duffy, V. G. (2007). Applications of digital human modeling in industry. In *International Conference on Digital Human Modeling*, pages 824–832. Springer.

Denavit, J. and Hartenberg, R. S. (1955). A kinematic notation for lower-pair mechanisms based on matrices. *Trans. of the ASME. Journal of Applied Mechanics*, 22:215–221.

Donner, P. and Buss, M. (2016). Cooperative swinging of complex pendulum-like objects: Experimental evaluation. *IEEE Transactions on Robotics*, 32(3):744–753.

Duffy, V. G. (2016). *Handbook of digital human modeling: research for applied ergonomics and human factors engineering.* CRC press.

Elobaid, M., Hu, Y., Romualdi, G., Dafarra, S., Babic, J., and Pucci, D. (2019). Telexistence and teleoperation for walking humanoid robots. In *Proceedings of SAI Intelligent Systems Conference*, pages 1106–1121. Springer.

Endo, Y., Tada, M., and Mochimaru, M. (2014). Dhaiba: development of virtual ergonomic assessment system with human models. In *Proceedings of The 3rd International Digital Human Symposium*.

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.

Featherstone, R. (2007). *Rigid Body Dynamics Algorithms.* Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Feng, S., Xinjilefu, X., Atkeson, C., and Kim, J. (2015). Optimization based controller design and implementation for the atlas robot in the darpa robotics challenge finals. In *Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on*, pages 1028–1035.

Ferreau, H., Kirches, C., Potschka, A., Bock, H., and Diehl, M. (2014). qpOASES: A parametric active-set algorithm for quadratic programming. *Mathematical Programming Computation*, 6(4):327–363.

Fitzpatrick, P., Elena, C., Daniele, D., Ali, P., Giorgio, M., and Lorenzo, C. (2014). A middle way for robotics middleware.

Frey, C. B. and Osborne, M. A. (2017). The future of employment: How susceptible are jobs to computerisation? *Technological forecasting and social change*, 114:254–280.

Fritzsche, L. (2010). Ergonomics risk assessment with digital human models in car assembly: Simulation versus real life. *Human Factors and Ergonomics in Manufacturing & Service Industries*, 20(4):287–299.

Galin, R. and Meshcheryakov, R. (2019). Review on human–robot interaction during collaboration in a shared workspace. In *International Conference on Interactive Collaborative Robotics*, pages 63–74. Springer.

Gall, J., Stoll, C., De Aguiar, E., Theobalt, C., Rosenhahn, B., and Seidel, H.-P. (2009). Motion capture using joint skeleton tracking and surface estimation. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1746–1753. IEEE.

Gasparetto, A. and Scalera, L. (2019). From the unimate to the delta robot: the early decades of industrial robotics. In *Explorations in the History and Heritage of Machines and Mechanisms*, pages 284–295. Springer.

Geravand, M., Flacco, F., and De Luca, A. (2013). Human-robot physical interaction and collaboration using an industrial robot with a closed control architecture. In *2013 IEEE International Conference on Robotics and Automation*, pages 4000–4007. IEEE.

Goldenberg, A., Benhabib, B., and Fenton, R. (1985). A complete generalized solution to the inverse kinematics of robots. *IEEE Journal on Robotics and Automation*, 1(1):14–20.

Goodrich, M. A., Schultz, A. C., et al. (2008). Human–robot interaction: a survey. *Foundations and Trends® in Human–Computer Interaction*, 1(3):203–275.

Gopinathan, S., Otting, S., and Steil, J. (2017). A user study on personalized adaptive stiffness control modes for human-robot interaction. In *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 831–837. IEEE.

Gribovskaya, E., Kheddar, A., and Billard, A. (2011). Motion learning and adaptive impedance for robot control during physical interaction with humans. In *2011 IEEE International Conference on Robotics and Automation*, pages 4326–4332. IEEE.

Guo, A., Ma, J., Sun, G., and Tan, S. (2020). A personal character model of affect, behavior and cognition for individual-like research. *Computers & Electrical Engineering*, 81:106544.

H-Anim (2006). Human animation working group. *http://www.h-anim.org/*.

Haddadin, S. (2015). *Physical Safety in Robotics*, pages 249–271. Springer Fachmedien Wiesbaden, Wiesbaden.

Haddadin, S., De Luca, A., and Albu-Schäffer, A. (2017). Robot collisions: A survey on detection, isolation, and identification. *IEEE Transactions on Robotics*, 33(6):1292–1312.

Hanavan, E. P. (1964). A mathematical model of human body. Technical report, Air force aerospace medical research lab Wright-Patterson AFB OH.

Hansen, C., Venture, G., Rezzoug, N., Gorce, P., and Isableu, B. (2014). An individual and dynamic body segment inertial parameter validation method using ground reaction forces. *Journal of biomechanics*, 47(7):1577–1581.

Herzog, A., Righetti, L., Grimminger, F., Pastor, P., and Schaal, S. (2014). Balancing experiments on a torque-controlled humanoid with hierarchical inverse dynamics. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 981–988. IEEE.

Hirai, K., Hirose, M., Haikawa, Y., and Takenaka, T. (1998). The development of honda humanoid robot. In *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, volume 2, pages 1321–1326. IEEE.

Hoffman, E. M., Traversaro, S., Rocchi, A., Ferrati, M., Settimi, A., Romano, F., Natale, L., Bicchi, A., Nori, F., and Tsagarakis, N. G. (2014). Yarp based plugins for gazebo simulator. In *International Workshop on Modelling and Simulation for Autonomous Systems*, pages 333–346. Springer.

Hofmann, A., Popovic, M., and Herr, H. (2009). Exploiting angular momentum to enhance bipedal center-of-mass control. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 4423–4429. IEEE.

Hogan, N. (1984). Impedance control: An approach to manipulation. In *1984 American control conference*, pages 304–313. IEEE.

Hokayem, P. F. and Spong, M. W. (2006). Bilateral teleoperation: An historical survey. *Automatica*, 42(12):2035–2057.

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

Ikemoto, S., Amor, H. B., Minato, T., Jung, B., and Ishiguro, H. (2012). Physical human-robot interaction: Mutual learning and adaptation. *IEEE robotics & automation magazine*, 19(4):24–35.

Imamura, Y., Ayusawa, K., Yoshida, E., and Tanaka, T. (2018). Evaluation framework for passive assistive device based on humanoid experiments. *International Journal of Humanoid Robotics*, 15(03):1750026.

Irshad, L., Ahmed, S., Demirel, O., and Tumer, I. Y. (2019). Coupling digital human modeling with early design stage human error analysis to assess ergonomic vulnerabilities. In *AIAA Scitech 2019 Forum*, page 2349.

Ishiguro, Y., Kojima, K., Sugai, F., Nozawa, S., Kakiuchi, Y., Okada, K., and Inaba, M. (2017). Bipedal oriented whole body master-slave system for dynamic secured locomotion with lip safety constraints. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 376–382.

Ishiguro, Y., Kojima, K., Sugai, F., Nozawa, S., Kakiuchi, Y., Okada, K., and Inaba, M. (2018). High speed whole body dynamic motion experiment with real time master-slave humanoid robot system. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–7.

Islam, S., Liu, P. X., Saddik, A. E., and Yang, Y. B. (2015). Bilateral control of teleoperation systems with time delay. *IEEE/ASME Transactions on Mechatronics*, 20(1):1–12.

Ito, T., Ayusawa, K., Yoshida, E., and Kobayashi, H. (2018). Evaluation of active wearable assistive devices with human posture reproduction using a humanoid robot. *Advanced Robotics*, 32(12):635–645.

Karatsidis, A., Bellusci, G., Schepers, H., de Zee, M., Andersen, M., and Veltink, P. (2017). Estimation of ground reaction forces and moments during gait using only inertial motion capture. *Sensors*, 17(1):75.

Khalil, W. and Dombre, E. (2004). *Modeling, identification and control of robots*. Butterworth-Heinemann.

Koolen, T., Bertrand, S., Thomas, G., De Boer, T., Wu, T., Smith, J., Englsberger, J., and Pratt, J. (2016). Design of a momentum-based control framework and application to the humanoid robot atlas. *International Journal of Humanoid Robotics*, 13(01):1650007.

Krotkov, E., Hackett, D., Jackel, L., Perschbacher, M., Pippine, J., Strauss, J., Pratt, G., and Orlowski, C. (2017). The darpa robotics challenge finals: Results and perspectives. *J. Field Robot.*, 34(2):229–240.

Kyriakopoulos, K. J. and Saridis, G. N. (1988). Minimum jerk path generation. In *Proceedings. 1988 IEEE International Conference on Robotics and Automation*, pages 364–369. IEEE.

Kyrkjebø, E. (2018). Inertial human motion estimation for physical human-robot interaction using an interaction velocity update to reduce drift. In *Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, pages 163–164. ACM.

Latella, C. (2018). *Human Whole-Body Dynamics Estimation for Enhancing Physical Human-Robot Interaction*. PhD thesis, University of Genoa.

Latella, C., Lorenzini, M., Lazzaroni, M., Romano, F., Traversaro, S., Akhras, M. A., Pucci, D., and Nori, F. (2019a). Towards real-time whole-body human dynamics estimation through probabilistic sensor fusion algorithms. *Autonomous Robots*, 43(6):1591–1603.

Latella, C., Traversaro, S., Ferigo, D., Tirupachuri, Y., Rapetti, L., Andrade Chavez, F. J., Nori, F., and Pucci, D. (2019b). Simultaneous floating-base estimation of human kinematics and joint torques. *Sensors*, 19(12):2794.

Lecours, A., Mayer-St-Onge, B., and Gosselin, C. (2012). Variable admittance control of a four-degree-of-freedom intelligent assist device. In *2012 IEEE International Conference on Robotics and Automation*, pages 3903–3908. IEEE.

Leung, M. K. and Yee-Hong Yang (1995). First sight: A human body outline labeling system. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(4):359–377.

Li, Y. and Ge, S. S. (2013). Impedance learning for robots interacting with unknown environments. *IEEE Transactions on Control Systems Technology*, 22(4):1422–1432.

Liarokapis, M. V., Artemiadis, P., Bechlioulis, C., and Kyriakopoulos, K. (2013). Directions, methods and metrics for mapping human to robot motion with functional anthropomorphism: A review. *School of Mechanical Engineering, National Technical University of Athens, Tech. Rep.*

Lichiardopol, S. (2007). A survey on teleoperation. *Technische Universitat Eindhoven, DCT report.*

Losey, D. P., McDonald, C. G., Battaglia, E., and O'Malley, M. K. (2018). A review of intent detection, arbitration, and communication aspects of shared control for physical human–robot interaction. *Applied Mechanics Reviews*, 70(1):010804.

Magrini, E., Flacco, F., and De Luca, A. (2015). Control of generalized contact motion and force in physical human-robot interaction. In *2015 IEEE international conference on robotics and automation (ICRA)*, pages 2298–2304. IEEE.

Marsden, J. E. and Ratiu, T. S. (2010). *Introduction to Mechanics and Symmetry: A Basic Exposition of Classical Mechanical Systems*. Springer Publishing Company, Incorporated.

Marsden, J. E. and Scheurle, J. (1993). The reduced euler-lagrange equations. *Fields Institute Comm*, 1:139–164.

Matrangola, S. L., Madigan, M. L., Nussbaum, M. A., Ross, R., and Davy, K. P. (2008). Changes in body segment inertial parameters of obese individuals with weight loss. *Journal of biomechanics*, 41(15):3278–3281.

Mattar, E. A., Al-Junaid, H. J., and Al-Seddiqi, H. H. (2018). Biomimetic based eeg learning for robotics complex grasping and dexterous manipulation. In *Biomimetic Prosthetics*. InTech.

McMullen, D. P., Hotson, G., Katyal, K. D., Wester, B. A., Fifer, M. S., McGee, T. G., Harris, A., Johannes, M. S., Vogelstein, R. J., Ravitz, A. D., et al. (2014). Demonstration of a semi-autonomous hybrid brain–machine interface using human intracranial eeg, eye tracking, and computer vision to control a robotic upper limb prosthetic. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22(4):784–796.

Metta, G., Fitzpatrick, P., and Natale, L. (2006). Yarp: yet another robot platform. *International Journal of Advanced Robotic Systems*, 3(1):8.

Moes, N. (2010). Digital human models: an overview of development and applications in product and workplace design. In *Proceedings of TMCE (Tools and Methods of Competitive Engineering) International Conference, Ancona*.

Mourtzis, D. (2019). Simulation in the design and operation of manufacturing systems: state of the art and new trends. *International Journal of Production Research*, 0(0):1–23.

Natale, L., Bartolozzi, C., Pucci, D., Wykowska, A., and Metta, G. (2017). icub: The not-yet-finished story of building a robot child. *Science Robotics*, 2(13):eaaq1026.

Nava, G., Romano, F., Nori, F., and Pucci, D. (2016). Stability analysis and design of momentum-based controllers for humanoid robots. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 680–687. IEEE.

Niyogi and Adelson (1994). Analyzing and recognizing walking figures in xyt. In *1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 469–474.

Nori, F., Kuppuswamy, N., and Traversaro, S. (2015a). Simultaneous state and dynamics estimation in articulated structures. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3380–3386. IEEE.

Nori, F., Traversaro, S., Eljaik, J., Romano, F., Del Prete, A., and Pucci, D. (2015b). iCub Whole-Body Control through Force Regulation on Rigid Non-Coplanar Contacts. *Frontiers in Robotics and AI*, 2:6.

Nori, F., Traversaro, S., Eljaik, J., Romano, F., Del Prete, A., and Pucci, D. (2015c). iCub whole-body control through force regulation on rigid noncoplanar contacts. *Frontiers in Robotics and AI*, 2(6).

Norton, A., Ober, W., Baraniecki, L., McCann, E., Scholtz, J., Shane, D., Skinner, A., Watson, R., and Yanco, H. (2017). Analysis of human–robot interaction at the darpa robotics challenge finals. *The International Journal of Robotics Research*, 36(5-7):483–513.

Ogenyi, U. E., Liu, J., Yang, C., Ju, Z., and Liu, H. (2019). Physical human-robot collaboration: Robotic systems, learning methods, collaborative strategies, sensors, and actuators. *IEEE transactions on cybernetics.*

Ott, C., Roa, M. A., and Hirzinger, G. (2011). Posture and balance control for biped robots based on contact force optimization. In *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, pages 26–33. IEEE.

Parmiggiani, A., Maggiali, M., Natale, L., Nori, F., Schmitz, A., Tsagarakis, N. G., Victor, J. S., Becchi, F., Sandini, G., and Metta, G. (2012). The Design of the iCub Humanoid Robot. *International Journal of Humanoid Robotics*, 09(04).

Parviainen, J. and Pirhonen, J. (2017). Vulnerable bodies in human–robot interactions: embodiment as ethical issue in robot care for the elderly.

Pattacini, U., Nori, F., Natale, L., Metta, G., and Sandini, G. (2010a). An experimental evaluation of a novel minimum-jerk cartesian controller for humanoid robots. In *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, pages 1668–1674. IEEE.

Pattacini, U., Nori, F., Natale, L., Metta, G., and Sandini, G. (2010b). An experimental evaluation of a novel minimum-jerk Cartesian controller for humanoid robots. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).*

Pattacini, U., Nori, F., Natale, L., Metta, G., and Sandini, G. (2010c). An experimental evaluation of a novel minimum-jerk cartesian controller for humanoid robots. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1668–1674.

Pedersen, L., Kortenkamp, D., Wettergreen, D., Nourbakhsh, I., and Korsmeyer, D. (2003). A survey of space robotics. In *Proceedings of 7th International Symposium on Artificial Intelligence, Robotics and Automation in space (i-SAIRAS-03).*

Penco, L., Clement, B., Modugno, V., Mingo Hoffman, E., Nava, G., Pucci, D., Tsagarakis, N. G., Mouret, J. ., and Ivaldi, S. (2018). Robust real-time whole-body motion retargeting from human to humanoid. In *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids).*

Peternel, L. and Babič, J. (2013). Learning of compliant human–robot interaction using full-body haptic interface. *Advanced Robotics*, 27(13):1003–1012.

Peternel, L., Tsagarakis, N., Caldwell, D., and Ajoudani, A. (2018). Robot adaptation to human physical fatigue in human–robot co-manipulation. *Autonomous Robots*, pages 1–11.

Pfeiffer, S. (2016). Robots, industry 4.0 and humans, or why assembly work is more than routine work. *Societies*, 6(2):16.

Pons-Moll, G., Baak, A., Gall, J., Leal-Taixe, L., Mueller, M., Seidel, H.-P., and Rosenhahn, B. (2011). Outdoor human motion capture using inverse kinematics and von mises-fisher sampling. In *2011 International Conference on Computer Vision*, pages 1243–1250. IEEE.

Pucci, D., Romano, F., Traversaro, S., and Nori, F. (2016a). Highly dynamic balancing via force control. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 141–141. IEEE.

Pucci, D., Romano, F., Traversaro, S., and Nori, F. (2016b). Highly dynamic balancing via force control. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 141–141.

Radmand, A., Scheme, E., and Englehart, K. (2014). A characterization of the effect of limb position on emg features to guide the development of effective prosthetic control schemes. In *Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE*, pages 662–667. IEEE.

Raessa, M., Chen, J. C. Y., Wan, W., and Harada, K. (2019). Human-in-the-loop robotic manipulation planning for collaborative assembly. *arXiv preprint arXiv:1909.11280*.

Rajesh, R. and Srinath, R. (2016). Review of recent developments in ergonomic design and digital human models. *Ind Eng Manage*, 5(186):2169–0316.

Ramos, J. and Kim, S. (2018). Humanoid dynamic synchronization through whole-body bilateral feedback teleoperation. *IEEE Transactions on Robotics*, 34(4):953–965.

Ranatunga, I., Lewis, F. L., Popa, D. O., and Tousif, S. M. (2016). Adaptive admittance control for human–robot interaction using model reference design and adaptive inverse filtering. *IEEE Transactions on Control Systems Technology*, 25(1):278–285.

Rapetti, L., Tirupachuri, Y., Darvish, K., Latella, C., and Pucci, D. (2019). Model-Based Real-Time Motion Tracking using Dynamical Inverse Kinematics. *arXiv e-prints*, page arXiv:1909.07669.

Rasouli, M., Chellamuthu, K., Cabibihan, J.-J., and Kukreja, S. L. (2016). Towards enhanced control of upper prosthetic limbs: A force-myographic approach. In *Biomedical Robotics and Biomechatronics (BioRob), 2016 6th IEEE International Conference on*, pages 232–236. IEEE.

Rautaray, S. S. and Agrawal, A. (2015). Vision based hand gesture recognition for human computer interaction: a survey. *Artificial Intelligence Review*, 43(1):1–54.

Reily, B., Han, F., Parker, L. E., and Zhang, H. (2018). Skeleton-based bio-inspired human activity prediction for real-time human–robot interaction. *Autonomous Robots*, 42(6):1281–1298.

Riemer, R. and Hsiao-Wecksler, E. T. (2008). Improving joint torque calculations: Optimization-based inverse dynamics to reduce the effect of motion errors. *Journal of biomechanics*, 41(7):1503–1509.

Robotic-Operating-System (2018). Robotic operating system(ros).

Roetenberg, D., Luinge, H., and Slycke, P. (2009). Xsens mvn: full 6dof human motion tracking using miniature inertial sensors. Technical report, Xsens Motion Technologies BV.

Romano, F., Nava, G., Azad, M., Čamernik, J., Dafarra, S., Dermy, O., Latella, C., Lazzaroni, M., Lober, R., Lorenzini, M., Pucci, D., Sigaud, O., Traversaro, S., Babič, J., Ivaldi, S., Mistry, M., Padois, V., and Nori, F. (2018). The codyco project achievements and beyond: Toward human aware whole-body controllers for physical human robot interaction. *IEEE Robotics and Automation Letters*, 3(1):516–523.

Romano, F., Traversaro, S., Pucci, D., and Nori, F. (2017). A whole-body software abstraction layer for control design of free-floating mechanical systems. *Journal of Software Engineering for Robotics*.

Romualdi, G., Dafarra, S., Hu, Y., and Pucci, D. (2018). A Benchmarking of DCM Based Architectures for Position and Velocity Controlled Walking of Humanoid Robots. In *2018 IEEE 18th International Conference on Humanoid Robots (Humanoids)*, pages 1–9.

Saccon, A., Traversaro, S., Nori, F., and Nijmeijer, H. (2017). On centroidal dynamics and integrability of average angular velocity. *IEEE Robotics and Automation Letters*, 2(2):943–950.

Saenz, J., Elkmann, N., Gibaru, O., and Neto, P. (2018). Survey of methods for design of collaborative robotics applications-why safety is a barrier to more widespread robotics uptake. In *Proceedings of the 2018 4th International Conference on Mechatronics and Robotics Engineering*, pages 95–101. ACM.

Saikia, A. and Hazarika, S. M. (2017). cbdi: Towards an architecture for human–machine collaboration. *International Journal of Social Robotics*, 9(2):211–230.

Samy, V., Ayusawa, K., Yoshiyasu, Y., Sagawa, R., and Yoshida, E. (2020). Fusion of multiple motion capture systems for musculoskeletal analysis. In *2020 IEEE/SICE International Symposium on System Integration*.

Sandini, G., Metta, G., and Vernon, D. (2004). RobotCub: an open framework for research in embodied cognition. In *4th IEEE/RAS International Conference on Humanoid Robots, 2004.*, volume 1, pages 13–32. IEEE.

Sarac, M., Koyas, E., Erdogan, A., Cetin, M., and Patoglu, V. (2013). Brain computer interface based robotic rehabilitation with online modification of task speed. In *Rehabilitation Robotics (ICORR), 2013 IEEE International Conference on*, pages 1–7. IEEE.

Sciavicco, L. and Siciliano, B. (1988). A solution algorithm to the inverse kinematic problem for redundant manipulators. *IEEE Journal on Robotics and Automation*, 4(4):403–410.

Shahrokhi, M. and Bernard, A. (2009). A framework to develop an analysis agent for evaluating human performance in manufacturing systems. *CIRP Journal of Manufacturing Science and Technology*, 2(1):55 – 60.

Shimoga, K. B. (1993). A survey of perceptual feedback issues in dexterous telemanipulation. ii. finger touch feedback. In *Proceedings of IEEE Virtual Reality Annual International Symposium*, Seattle, WA, USA.

Shio, A. and Sklansky, J. (1991). Segmentation of people in motion. In *Proceedings of the IEEE Workshop on Visual Motion*, pages 325–332.

Siciliano, B. and Khatib, O. (2007). *Springer Handbook of Robotics*. Springer-Verlag New York, Inc.

Siciliano, B., Sciavicco, L., Villani, L., and Oriolo, G. (2008). *Robotics: Modelling, Planning and Control*. Springer Publishing Company, Incorporated, 1st edition.

Siciliano, B. and Villani, L. (2012). *Robot force control*, volume 540. Springer Science & Business Media.

Skals, S., Jung, M. K., Damsgaard, M., and Andersen, M. S. (2017). Prediction of ground reaction forces and moments during sports-related movements. *Multibody system dynamics*, 39(3):175–195.

Song, R., Tong, K.-y., Hu, X., Li, L., et al. (2008). Assistive control system using continuous myoelectric signal in robot-aided arm training for patients after stroke. *IEEE transactions on neural systems and rehabilitation engineering*, 16(4):371–379.

Stanton, C., Bogdanovych, A., and Ratanasena, E. (2012). Teleoperation of a humanoid robot using full-body motion capture, example movements, and machine learning. In *Proc. Australasian Conference on Robotics and Automation*.

Steinfeld, A., Fong, T., Kaber, D., Lewis, M., Scholtz, J., Schultz, A., and Goodrich, M. (2006). Common metrics for human-robot interaction. In *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*, pages 33–40. ACM.

Stephens, B. J. and Atkeson, C. G. (2010). Dynamic balance force control for compliant humanoid robots. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 1248–1255. IEEE.

Tachi, S., Arai, H., and Maeda, T. (1989). Development of an anthropomorphic tele-existence slave robot. In *Proceedings of the International Conference on Advanced Mechatronics (ICAM)*, volume 385, page 390.

Traversaro, S. (2017). *Modelling, Estimation and Identification of Humanoid Robots Dynamics*. PhD thesis, University of Genoa.

Traversaro, S., Pucci, D., and Nori, F. (2015). In situ calibration of six-axis force-torque sensors using accelerometer measurements. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2111–2116. IEEE.

Traversaro, S., Pucci, D., and Nori, F. (2016). On the base frame choice in free-floating mechanical systems and its connection to "centroidal" dynamics. *Submitted to Humanoid Robots (Humanoids), 2016 IEEE-RAS International Conference on*.

Traversaro, S. and Saccon, A. (2019). *Multibody dynamics notation (version 2)*. Technische Universiteit Eindhoven. Dept. of Mechanical Engineering. Report locator DC 2019.100.

Trevelyan, J., Hamel, W. R., and Kang, S.-C. (2016). Robotics in hazardous applications. In *Springer handbook of robotics*, pages 1521–1548. Springer.

Venture, G., Bonnet, V., and Kulic, D. (2019). *Creating Personalized Dynamic Models*, pages 91–104. Springer International Publishing, Cham.

Wächter, A. and Biegler, L. T. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57.

Wachter, S. and Nagel, H.-H. (1999). Tracking persons in monocular image sequences. *Computer Vision and Image Understanding*, 74(3):174–192.

Wagh, K. and Kanade, S. S. (2019). Robust human tracking using harmonious polling tracker. *SN Applied Sciences*, 1(10):1227.

Wang, A., Ramos, J., Mayo, J., Ubellacker, W., Cheung, J., and Kim, S. (2015). The hermes humanoid system: A platform for full-body teleoperation with balance feedback. In *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*.

Wensing, P. M. and Orin, D. E. (2013). Generation of dynamic humanoid behaviors through task-space control with conic optimization. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 3103–3109. IEEE.

Winter, D. A. (1990). *Biomechanics and motor control of human movement*, chapter Anthropometry. Wiley.

Xiang, Y., Chung, H.-J., Kim, J. H., Bhatt, R., Rahmatalla, S., Yang, J., Marler, T., Arora, J. S., and Abdel-Malek, K. (2010). Predictive dynamics: an optimization-based novel approach for human motion simulation. *Structural and Multidisciplinary Optimization*, 41(3):465–479.

Yap, H. K., Mao, A., Goh, J. C., and Yeow, C.-H. (2016). Design of a wearable fmg sensing system for user intent detection during hand rehabilitation with a soft robotic glove. In *Biomedical Robotics and Biomechatronics (BioRob), 2016 6th IEEE International Conference on*, pages 781–786. IEEE.

Yoshida, E., Ayusawa, K., Imamura, Y., and Tanaka, T. (2017). Towards new humanoid applications: wearable device evaluation through human motion reproduction.

Zamalloa, I., Kojcev, R., Hernández, A., Muguruza, I., Usategui, L., Bilbao, A., and Mayoral, V. (2017). Dissecting robotics-historical overview and future perspectives. *arXiv preprint arXiv:1704.08617*.

Zhou, Y., Fang, Y., Zeng, J., Li, K., and Liu, H. (2018). A multi-channel emg-driven fes solution for stroke rehabilitation. In *International Conference on Intelligent Robotics and Applications*, pages 235–243. Springer.

Zhu, R. and Zhou, Z. (2004). A real-time articulated human motion tracking using tri-axis inertial/magnetic sensors package. *IEEE Transactions on Neural systems and rehabilitation engineering*, 12(2):295–302.

Zucker, M., Joo, S., Grey, M. X., Rasmussen, C., Huang, E., Stilman, M., and Bobick, A. (2015). A general-purpose system for teleoperation of the drc-hubo humanoid robot. *Journal of Field Robotics*, 32(3):336–351.

Human Dynamics Estimation Maximum A-Posteriori Solution

The Maximum A-Posteriori (MAP) Solution for the human dynamics estimation is originally presented in detail in (Latella, 2018). The following mathematical details are directly borrowed from (Latella, 2018) and presented in this appendix for the sake of minimal clarity to the reader.

Since the normal distributions $\boldsymbol{d}$ and $\boldsymbol{y}$ are jointly Gaussian, the conditional probability distribution $p(\boldsymbol{d}|\boldsymbol{y})$ is such that

$$p(\boldsymbol{d}|\boldsymbol{y}) = \frac{p(\boldsymbol{d}, \boldsymbol{y})}{p(\boldsymbol{y})} = \frac{p(\boldsymbol{d})p(\boldsymbol{y}|\boldsymbol{d})}{p(\boldsymbol{y})} \tag{A.1}$$

In the following computation, the term $p(\boldsymbol{y})$ is negligible since it does not depend on $\boldsymbol{d}$. This is the reason of the proportionality between the conditional probability distribution and the joint distribution in (6.9). Hereafter each term in (A.1) is computed separately to obtain the final analytical solution. For the sake of simplicity, $(\boldsymbol{q}, \dot{\boldsymbol{q}})$ dependencies are omitted in the computations.

Computation of $p(\boldsymbol{y}|\boldsymbol{d})$

Let us first give an expression for the conditional probability density function (PDF) $p(\boldsymbol{y}|\boldsymbol{d})$:

$$
\begin{aligned}
p(\boldsymbol{y}|\boldsymbol{d}) \quad &\propto \quad \exp{-\frac{1}{2}\left\{ \left(\boldsymbol{y} - \boldsymbol{\mu}_{y|d}\right)^{\top} \boldsymbol{\Sigma}_{y|d}^{-1} \left(\boldsymbol{y} - \boldsymbol{\mu}_{y|d}\right) \right\}} \\
&= \quad \exp{-\frac{1}{2}\left\{ \left[\boldsymbol{y} - (\boldsymbol{Y}\boldsymbol{d} + \boldsymbol{b}_Y)\right]^{\top} \boldsymbol{\Sigma}_{y|d}^{-1} \left[\boldsymbol{y} - (\boldsymbol{Y}\boldsymbol{d} + \boldsymbol{b}_Y)\right] \right\}}
\end{aligned}
\tag{A.2}
$$

which implicitly makes the assumption that the measurements equation (6.4) is affected by a Gaussian noise with zero mean and covariance $\boldsymbol{\Sigma}_{y|d}$.

Computation of $p(\boldsymbol{d})$

Define now a PDF for the normal distribution $\boldsymbol{d}$. By pursuing the same methodology, we would like to write its distribution in the following form

$$\boldsymbol{d} \sim \mathcal{N}(\boldsymbol{\mu}_D, \boldsymbol{\Sigma}_D) \tag{A.3}$$

such that the PDF

$$p(\boldsymbol{d}) \propto \exp -\frac{1}{2}\big\{\boldsymbol{e}(\boldsymbol{d})^\top \boldsymbol{\Sigma}_D^{-1} \boldsymbol{e}(\boldsymbol{d})\big\} \tag{A.4}$$

taking into account constraints of equation (6.3) with $\boldsymbol{e}(\boldsymbol{d}) = \boldsymbol{D}\boldsymbol{d} + \boldsymbol{b}_D$.

However, this intuitive choice leads to a degenerate normal distribution and a regularization term is needed. For example, if we have a Gaussian prior knowledge on $\boldsymbol{d}$ in the form of $\boldsymbol{d} \sim \mathcal{N}(\boldsymbol{\mu}_d, \boldsymbol{\Sigma}_d)$ distribution, we can reformulate Equation (A.3) as follows:

$$\boldsymbol{d} \sim \mathcal{N}(\bar{\boldsymbol{\mu}}_D, \bar{\boldsymbol{\Sigma}}_D) \tag{A.5}$$

such that (A.4) becomes

$$
\begin{aligned}
p(\boldsymbol{d}) \quad &\propto \quad \exp -\frac{1}{2}\big\{\boldsymbol{e}(\boldsymbol{d})^\top \boldsymbol{\Sigma}_D^{-1}\boldsymbol{e}(\boldsymbol{d}) + (\boldsymbol{d}-\boldsymbol{\mu}_d)^\top \boldsymbol{\Sigma}_d^{-1}(\boldsymbol{d}-\boldsymbol{\mu}_d)\big\} \\
&= \quad \exp -\frac{1}{2}\big\{(\boldsymbol{D}\boldsymbol{d}+\boldsymbol{b}_D)^\top \boldsymbol{\Sigma}_D^{-1}(\boldsymbol{D}\boldsymbol{d}+\boldsymbol{b}_D) + (\boldsymbol{d}-\boldsymbol{\mu}_d)^\top \boldsymbol{\Sigma}_d^{-1}(\boldsymbol{d}-\boldsymbol{\mu}_d)\big\} \\
&= \quad \exp -\frac{1}{2}\big\{\big(\boldsymbol{d}-\bar{\boldsymbol{\mu}}_D\big)^\top \bar{\boldsymbol{\Sigma}}_D^{-1}\big(\boldsymbol{d}-\bar{\boldsymbol{\mu}}_D\big)\big\}
\end{aligned} \tag{A.6}
$$

where the covariance and the mean are, respectively,

$$\bar{\boldsymbol{\Sigma}}_D = \big(\boldsymbol{D}^\top \boldsymbol{\Sigma}_D^{-1}\boldsymbol{D} + \boldsymbol{\Sigma}_d^{-1}\big)^{-1} \tag{A.7a}$$

$$\bar{\boldsymbol{\mu}}_D = \bar{\boldsymbol{\Sigma}}_D \big(\boldsymbol{\Sigma}_d^{-1}\boldsymbol{\mu}_d - \boldsymbol{D}^\top \boldsymbol{\Sigma}_D^{-1}\boldsymbol{b}_D\big) \tag{A.7b}$$

The role of $\boldsymbol{\Sigma}_D$ is to establish how much the dynamic model (6.3) should be considered correct. The quantities $\boldsymbol{\mu}_d$ and $\boldsymbol{\Sigma}_d$, instead, define the Gaussian prior distribution on $\boldsymbol{d}$ (namely, the regularization term).

Computation of $p(\boldsymbol{d}|\boldsymbol{y})$

By combining Equations (A.2) and (A.6) we are now ready to give a new formulation of the estimation problem for the conditional PDF of $\boldsymbol{d}$ given $\boldsymbol{y}$, i.e.,

$$
\begin{aligned}
p(\boldsymbol{d}|\boldsymbol{y}) \quad \propto \quad & \exp -\frac{1}{2}\left\{\left(\boldsymbol{d}-\bar{\boldsymbol{\mu}}_D\right)^\top \bar{\boldsymbol{\Sigma}}_D^{-1}\left(\boldsymbol{d}-\bar{\boldsymbol{\mu}}_D\right) + \right. \\
& \left. + \left[\boldsymbol{y}-(\boldsymbol{Y}\boldsymbol{d}+\boldsymbol{b}_Y)\right]^\top \boldsymbol{\Sigma}_{y|d}^{-1}\left[\boldsymbol{y}-(\boldsymbol{Y}\boldsymbol{d}+\boldsymbol{b}_Y)\right]\right\}
\end{aligned} \tag{A.8}
$$

with a covariance matrix and a mean as follows:

$$
\boldsymbol{\Sigma}_{d|y} \quad = \quad \left(\bar{\boldsymbol{\Sigma}}_D^{-1} + \boldsymbol{Y}^\top \boldsymbol{\Sigma}_{y|d}^{-1}\boldsymbol{Y}\right)^{-1} \tag{A.9a}
$$

$$
\boldsymbol{\mu}_{d|y} \quad = \quad \boldsymbol{\Sigma}_{d|y}\left[\boldsymbol{Y}^\top \boldsymbol{\Sigma}_{y|d}^{-1}(\boldsymbol{y}-\boldsymbol{b}_Y) + \bar{\boldsymbol{\Sigma}}_D^{-1}\bar{\boldsymbol{\mu}}_D\right] \tag{A.9b}
$$

Moreover, in the Gaussian case the MAP solution coincides with the mean of the PDF $p(\boldsymbol{d}|\boldsymbol{y})$ yielding to:

$$
\boldsymbol{d}^{\text{MAP}} = \boldsymbol{\mu}_{d|y} \tag{A.10}
$$

Whole-Body Momentum Control

In the case of complex humanoids robots, state-of-the-art whole-body controllers often consider controlling the robot momentum (Nava et al., 2016; Pucci et al., 2016b). For the sake of minimal clarity, this appendix presents whole-body momentum control for humanoid robots originally presented in (Nava et al., 2016).

Recall the equations of motion of a robotic system from Eq.(8.14b),

$$\mathbf{M}(\mathbf{q})\dot{\boldsymbol{\nu}} + \mathbf{C}(\mathbf{q},\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{G}(\mathbf{q}) = \mathbf{B}\boldsymbol{\tau} + \sum_{i=1}^{n_c} \mathbf{J}_{\mathcal{C}_i}^{\top}\mathbf{f}_i, \tag{B.1}$$

The coordinates of the state space are $(\mathbf{q}, \boldsymbol{\nu})$. Consider the mass matrix partitioned as following,

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_b & \mathbf{M}_{bj} \\ \mathbf{M}_{bj}^{\top} & \mathbf{M}_j \end{bmatrix} \tag{B.2}$$

with $\mathbf{M}_b \in \mathbb{R}^{6\times 6}$, $\mathbf{M}_{bj} \in \mathbb{R}^{6\times n}$ and $\mathbf{M}_j \in \mathbb{R}^{n\times n}$. Now, consider the following change of state variables:

$$\mathbf{q} := \mathbf{q}, \quad \bar{\boldsymbol{\nu}} := T(\mathbf{q})\boldsymbol{\nu}, \tag{B.3}$$

with

$$T := \begin{bmatrix} {}^c\boldsymbol{X}_{\mathcal{F}} & {}^c\boldsymbol{X}_{\mathcal{F}}\mathbf{M}_b^{-1}\mathbf{M}_{bj} \\ \mathbf{0}_{n\times 6} & \mathbf{1}_n \end{bmatrix}, \tag{B.3a}$$

$$ {}^c\boldsymbol{X}_{\mathcal{F}} := \begin{bmatrix} \mathbf{1}_3 & -\mathbf{S}({}^{\mathcal{I}}\mathbf{p}_c - {}^{\mathcal{I}}\mathbf{p}_{\mathcal{F}}) \\ \mathbf{0}_{3\times 3} & \mathbf{1}_3 \end{bmatrix} \tag{B.3b}$$

where the superscript c denotes the frame with the origin located at the center of mass, and with the orientation of $\mathcal{I}$. Then, the equations of motion with new state variables $(\mathbf{q}, \bar{\boldsymbol{\nu}})$ becomes,

$$\overline{\mathbf{M}}(\mathbf{q})\dot{\bar{\boldsymbol{\nu}}} + \overline{\mathbf{C}}(\mathbf{q}, \bar{\boldsymbol{\nu}})\bar{\boldsymbol{\nu}} + \overline{\mathbf{G}} = \mathbf{B}\boldsymbol{\tau} + \sum_{i=1}^{n_c} \bar{\mathbf{J}}_{\mathcal{C}_i}^{\top} \mathbf{f}_i, \tag{B.4}$$

with

$$\overline{\mathbf{M}}(\mathbf{q}) = T^{-\top}\mathbf{M}T^{-1} = \begin{bmatrix} \overline{\mathbf{M}}_b(\mathbf{q}) & \mathbf{0}_{6\times n} \\ \mathbf{0}_{n\times 6} & \overline{\mathbf{M}}_j(\mathbf{q}_j) \end{bmatrix}, \tag{B.4a}$$

$$\overline{\mathbf{C}}(\mathbf{q}, \bar{\boldsymbol{\nu}}) = T^{-\top}(\mathbf{M}\dot{T}^{-1} + \mathbf{C}T^{-1}), \tag{B.4b}$$

$$\overline{\mathbf{G}} = T^{-\top}\mathbf{G} = mg\boldsymbol{e}_3, \tag{B.4c}$$

$$\bar{\mathbf{J}}_{\mathcal{C}_i}(\mathbf{q}) = \mathbf{J}_{\mathcal{C}_i}(\mathbf{q})T^{-1} = \begin{bmatrix} \bar{\mathbf{J}}_{\mathcal{C}_i}^b(\mathbf{q}) & \bar{\mathbf{J}}_{\mathcal{C}_i}^j(\mathbf{q}_j) \end{bmatrix}, \tag{B.4d}$$

$$\overline{\mathbf{M}}_b(\mathbf{q}) = \begin{bmatrix} m\mathbf{1}_3 & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{I}(\mathbf{q}) \end{bmatrix}, \bar{\mathbf{J}}_{\mathcal{C}_i}^b(\mathbf{q}) = \begin{bmatrix} \mathbf{1}_3 & -\mathbf{S}(\mathbf{p}_{\mathcal{C}_i} - {}^{\mathcal{I}}\mathbf{p}_c) \\ \mathbf{0}_{3\times 3} & \mathbf{1}_3 \end{bmatrix}$$

where m is the mass of the robot and $\mathbf{I}$ is the inertia matrix computed with respect to the center of mass, with the orientation of the inertial frame $\mathcal{I}$.

The mass matrix of the transformed system (B.4) is a block matrix that decouples the transformed base acceleration and the joint acceleration (Traversaro et al., 2016). More precisely, the transformed robot's velocity $\bar{\boldsymbol{\nu}}$ is given by $\bar{\boldsymbol{\nu}} = \begin{pmatrix} {}^{\mathcal{I}}\dot{\mathbf{p}}_c^{\top} & {}^{\mathcal{I}}\omega_c^{\top} & \dot{\mathbf{s}}^{\top} \end{pmatrix}^{\top}$ where ${}^{\mathcal{I}}\dot{\mathbf{p}}_c$ is the velocity of the center-of-mass of the robot, and ${}^{\mathcal{I}}\omega_c$ is the so-called *average angular velocity*[1].

Assuming that the only contact constraint is between the environment and the robot's foot, we can write:

$$\sum_{k=1}^{n_c} \bar{\mathbf{J}}_{\mathcal{C}_k}^{\top} \mathbf{f}_i = \bar{\mathbf{J}}_c^{\top}(\mathbf{q})\mathbf{f} \tag{B.5}$$

where $\bar{\mathbf{J}}_c(\mathbf{q}) \in \mathbb{R}^{6\times n+6}$ is the Jacobian of a frame attached to the foot's sole in contact with the environment, and $\mathbf{f} \in \mathbb{R}^6$ the contact wrench. Differentiating the

---

[1] The term ${}^{\mathcal{I}}\omega_c$ is also known as the *locked angular velocity* (Marsden and Scheurle, 1993).

kinematic constraint

$$\bar{\mathbf{J}}_c(\mathbf{q})\bar{\boldsymbol{\nu}} = \begin{bmatrix} \bar{\mathbf{J}}_b & \bar{\mathbf{J}}_j \end{bmatrix} \bar{\boldsymbol{\nu}} = 0 \tag{B.6}$$

associated with the contact, yield

$$\bar{\mathbf{J}}_c(\mathbf{q})\dot{\bar{\boldsymbol{\nu}}} + \dot{\bar{\mathbf{J}}}_c(\mathbf{q})\bar{\boldsymbol{\nu}} = 0 \tag{B.7}$$

$$\begin{bmatrix} \bar{\mathbf{J}}_b & \bar{\mathbf{J}}_j \end{bmatrix} \begin{bmatrix} \dot{\mathbf{v}}_{\mathcal{F}} \\ \ddot{\mathbf{s}} \end{bmatrix} + \begin{bmatrix} \dot{\bar{\mathbf{J}}}_b & \dot{\bar{\mathbf{J}}}_j \end{bmatrix} \begin{bmatrix} \mathbf{v}_{\mathcal{F}} \\ \dot{\mathbf{s}} \end{bmatrix} = 0 \tag{B.8}$$

The term $\dot{\bar{\boldsymbol{\nu}}}$ is the robot's acceleration that can be obtained from the decoupled equations of motion Eq. (B.4) as,

$$\dot{\bar{\boldsymbol{\nu}}} = \bar{\mathbf{M}}^{-1}[\mathbf{B}\boldsymbol{\tau} + \bar{\mathbf{J}}_c^\top \mathbf{f} - \bar{\mathbf{h}}] \tag{B.9}$$

where, $\bar{\mathbf{h}} = \bar{\mathbf{C}}(\mathbf{q}, \bar{\boldsymbol{\nu}})\bar{\boldsymbol{\nu}} + \bar{\mathbf{G}} \in \mathbb{R}^{n+6}$. Using the relation from Eq. (B.9) in the constraints equation Eq. (B.7) we obtain,

$$\bar{\mathbf{J}}_c\bar{\mathbf{M}}^{-1}[\mathbf{B}\boldsymbol{\tau} + \bar{\mathbf{J}}_c^\top \mathbf{f} - \bar{\mathbf{h}}] + \dot{\bar{\mathbf{J}}}_c\bar{\boldsymbol{\nu}} = 0 \tag{B.10}$$

Now, the robot's momentum $\mathbf{L} \in \mathbb{R}^6$ is given by $\mathbf{L} = \mathbf{M}_b\mathbf{v}_{\mathcal{F}}$. The rate-of-change of the robot momentum equals the net external wrench acting on the robot which in the present case reduces to the contact wrench $\mathbf{f}$ plus the gravity wrench. The contact wrench $\mathbf{f}$ is assumed to be a virtual control input through which the robot momentum is controlled. Note that given the particular form of (B.4), the first six rows correspond to the dynamics of the robot's momentum, i.e.

$$\frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{M}_b\mathbf{v}_{\mathcal{F}}) = \bar{\mathbf{J}}_b^\top \mathbf{f} - mg\mathbf{e}_3 = \dot{\mathbf{L}}(\mathbf{f}^*) \tag{B.11}$$

where $\mathbf{L} := (\mathbf{L}_L, \mathbf{L}_\omega)$, with $H_L, H_\omega \in \mathbb{R}^3$ linear and angular momentum, respectively. The control objective can then be defined as the stabilization of a desired robot momentum $\mathbf{L}^d$. Let us define the momentum error as follows $\tilde{\mathbf{L}} = \mathbf{L} - \mathbf{L}^d$. So, the virtual control input $\mathbf{f}$ in Eq. (B.11) is chosen such that,

$$\dot{\mathbf{L}}(\mathbf{f}) = \dot{\mathbf{L}}^* := \dot{\mathbf{L}}^d - \mathbf{K}_p \tilde{\mathbf{L}} - \mathbf{K}_i \int \tilde{\mathbf{L}} \tag{B.11a}$$

where $\mathbf{K}_p, \mathbf{K}_i \in \mathbb{R}^{6\times6}$ are two symmetric, positive definite matrices.

Given the assumption that the contact wrench $\mathbf{f}$ is a virtual control input, it is chosen as to satisfy the following relation,

$$\mathbf{f}^* = \bar{\mathbf{J}}_b^{-\top} \left(\dot{\mathbf{L}}^* + mg\mathbf{e}_3 \right) \tag{B.12}$$

Now, to determine the control torques that instantaneously realize the contact force given by (B.12), we use the relation from Eq. (B.10) that is derived from the decoupled dynamic equations (B.4) along with the constraints (B.8), which yield

$$\boldsymbol{\tau} = \boldsymbol{\Lambda}^\dagger (\bar{\mathbf{J}}_c \bar{\mathbf{M}}^{-1} (\bar{\mathbf{h}} - \bar{\mathbf{J}}_c^\top \mathbf{f}^*) - \dot{\bar{\mathbf{J}}}_c \bar{\boldsymbol{\nu}}) + N_{\boldsymbol{\Lambda}} \tau_0 \tag{B.13}$$

where $\boldsymbol{\Lambda} = \bar{\mathbf{J}}_j \bar{\mathbf{M}}_j^{-1} \in \mathbb{R}^{6\times n}$, $N_{\boldsymbol{\Lambda}} \in \mathbb{R}^{n\times n}$ is the nullspace projector of $\boldsymbol{\Lambda}$, and $\tau_0 \in \mathbb{R}^n$ is a free variable that can be used for additional tasks. The free variable $\boldsymbol{\tau}_0$ is exploited to achieve the postural task. A classical state-of-the-art choice for this postural task consists in: $\boldsymbol{\tau}_0 = \bar{\mathbf{h}}_j - \bar{\mathbf{J}}_j^\top \mathbf{f} - \mathbf{K}_p^j (\mathbf{s} - \mathbf{s}^d) - \mathbf{K}_d^j \dot{\mathbf{s}}$, where $\bar{\mathbf{h}}_j - \bar{\mathbf{J}}_j^\top \mathbf{f}$ compensates for the nonlinear effect and the external wrenches acting on the joint space of the system.

The reader is advised to refer (Nava et al., 2016) for a detailed discussion on whole-body momentum control of a humanoid robot for the balancing task.